A New Over-Sampling Approach: Random-SMOTE for Learning from Imbalanced Data Sets

Yanjie Dong and Xuehua Wang

Institution of Information and Decision-making Technology Dalian University of Technology
Dalian, China
wangxh@dlut.edu.cn

Abstract. For imbalanced data sets, examples of minority class are sparsely distributed in sample space compared with the overwhelming amount of majority class. This presents a great challenge for learning from the minority class. Enlightened by SMOTE, a new over-sampling method, Random-SMOTE, which generates examples randomly in the sample space of minority class is proposed. According to the experiments on real data sets, Random-SMOTE is more effective compared with other random sampling approaches.

Keywords: Imbalanced Data sets, Over-sampling Approach, Random-SMOTE.

1 Introduction

Imbalanced data sets refer to data sets whose numbers of examples in each class are not even. For a two-class problem, a data set is imbalanced when the data for one class greatly outnumbers the other class. Examples include the detection of possible churners before they effectively leave the company in service industries [1], identifying fraudulent credit card transactions [2], detecting oil spills from satellite images [3] and the diagnosis of rare diseases [4], etc. In most cases, the minority class is the class of interest and the errors coming from minority class have higher penalty errors. However, traditional algorithms, such as neural networks, support vector machines and decision trees, are commonly built to achieve overall accuracy to which the minority class contributes very little. As a result, minority class can't be well identified.

Various approaches have been proposed to address the class imbalance problem. In certain cases, approaches at algorithm level may be quite effective, but they have the disadvantage of being algorithm-specific [5]. This is a problem since data sets presenting different characteristics are better classified by different algorithms [6], and it might be quite difficult to transplant the modification proposed for the class imbalance problem from one classifier to others. On the other hand, approaches at data level can be viewed as data preprocessing methods. After re-sampling, the drawbacks of class imbalance diminish and thus they can be learned better by standard classifiers. So, approaches at data level are independent and more flexible. This is why we choose re-sampling technique to reduce the sparseness of minority class.

H. Xiong and W.B. Lee (Eds.): KSEM 2011, LNAI 7091, pp. 343–352, 2011.

2 Sampling Techniques for Learning from Imbalanced Data Sets

As mentioned above, this paper will focus on methods at data level, i.e. sampling techniques. In this section, some developed sampling techniques for learning from imbalanced data sets are introduced.

According to the intelligence of sampling techniques, there are non-heuristic sampling and heuristic sampling. Random under-sampling and random over-sampling, which are two basic methods for reducing class imbalance, are non-heuristic methods. Random under-sampling may discard potentially useful information and may keep useless information which can do harm to the classification. Random over-sampling increases the size of the training set and overfitting is likely to occur [7]. Recent research has focused on improving these basic methods and many new heuristic sampling techniques have been developed.

Estabrooks and Japkowicz[8] show that a mixture-of-experts approach can produce consistently good results. Weiss[8] proposes a heuristic "budget-sensitive" progressive sampling algorithm for selecting training examples.

Kubat and Matwin[9] employ an under-sampling strategy, named as one-sided selection, by removing "redundant" examples which are considered as useless for classification and "borderline" examples that are close to the boundary between classes as well as noise examples. Chan et al. [10] take a different approach. They first run preliminary experiments to determine the best class distribution for learning and generate multiple training sets with this distribution, then apply a classification algorithm to each training set. This approach ensures all available training data are used.

Visa et al.[11] propose over-sampling approaches based on aggregation of class information such as spread, imbalance factor and the distance between the classes. New samples of minority class are generated to balance the training set. It sounds good that the idea of generating new samples taking the characteristics of the data set into account. However, it may be difficult to consider the complexity, diversity and unknown distribution in the real data sets. Nickerson et al.[12] propose a guided re-sampling study.

SMOTE (Synthetic Minority Over-sampling Technique) is a novel approach to counter the effect of having few instances of the minority class in a data set[13]. SMOTE creates synthetic minority class examples by interpolating between minority class examples that lie close together. By synthetically generating more minority class examples, the inductive learners, such as decision trees or rule-learners, are able to broaden their decision regions for the minority class. Hui Han et al. [14] propose a new minority over-sampling method, borderline-SMOTE, in which only the minority examples near the borderline are over-sampled.

3 A New Over-Sampling Approach: Random-SMOTE

To better illustrate Random-SMOTE, we give SMOTE a deeper view. The main idea of SMOTE is to form new minority class examples by interpolating between several minority class examples that lie together. To be specific, for each minority example x,

its k (which is usually set to 5 in SMOTE) nearest neighbors of minority class are spotted firstly. Then, depending upon the over-sampling rate N required, N neighbors from the k nearest neighbors are randomly chosen. Finally, synthetic examples Pj are generated in the following way:

$$P_j = x + \text{rand}(0, 1) * (y_j - x),\ j = 1, 2, \ldots, N, \tag{1}$$

Where y_j (j=1, 2,…, N) is one of the nearest neighbors randomly selected in 5 nearest neighbors of x, and rand (0,1) generates a random number between 0 and 1.

It can be seen that SMOTE just generates new examples along the line between the minority example and its selected nearest neighbors. After SMOTE, the data sets maintain its intensive or sparse characteristic**.** So SMOTE can't predict well for unknown examples which fall in the sparse area of sample space, and hence there's still some room for the improvement of SMOTE.

3.1 An Introduction to Random-SMOTE Approach

To solve the problem, a new over-sampling approach—Random-SMOTE is proposed. Its main idea is to improve the sparseness of minority class by generating synthetic examples randomly in the existing minority-class space.

In Random-SMOTE, for each example x of minority class, two examples y_1, y_2 are randomly selected from minority class. As a result, a triangle is formed by the minority example x and its selected two minority class examples y_1 and y_2. Then, according to over-sampling rate N, a number of N new minority examples are generated randomly in the triangle area. The procedures can be illustrated in Fig. 1.

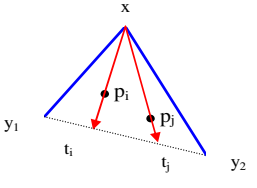

Fig. 1. The illustration of how Random-SMOTE generates new examples

The detailed procedure of generating synthetic examples is shown as follows.

(1) Generate temporary examples t_j (j=1,2,…,N) along the line between the two selected minority examples y_1 and y_2;

$$t_j = y_1 + \text{rand}(0,1) * (y_2 - y_1),\quad j = 1,2,\ldots,N \tag{2}$$

(2) Generate synthetic minority class examples pj (j=1,2,…,N) along the line between each temporary example tj (j=1,2,…,N) and example x.

$$pj = x + \text{rand}(0,1) * (tj - x),\ j = 1,2,\ldots, N, \tag{3}$$

Four cases may exist for the location among x and its selected two temporary examples y_1 and y_2.

Case 1: Three points coincide. In this case, an exact copy of x is made. So case 1 is actually a special case of non-heuristic sampling: random over-sampling.

Case 2: y_1 and y_2 coincide while x is a different point or Only one example, either y_1 or y_2, coincides with x. In this case, a new point is generated along the line between the line of x and y_1 or y_2 which is different with x. This is the same way that SMOTE generates synthetic examples.

Case 3: They are there different points without coincidence. In this case, synthetic examples are generated in the triangle area formed by x and y_1, y_2. This is the most usual case.

To sum up, Random-SMOTE is a more general method, random over-sampling or SMOTE is just a special case of Random-SMOTE.

3.2 The Description of Random-SMOTE Algorithm

It's easy to interpolate between samples so as to generate synthetic examples for datasets including only numerical attributes.

Supposing a data set has n attributes. Taking an attribute a as an example, if a is a numerical attribute, the new attribute value is generated as follows in Random-SMOTE. The name of the sample with the name of the attribute as subscript represents the value of the sample on this attribute. We use the same representation in the context below.

(1) Generate temporary values t_{ja} for N temporary examples t_j $(j=1,2,\ldots,N)$ randomly along the line between the attribute values, y_{1a} and y_{2a};

$$t_{ja} = y_{1a} + \text{rand}(0,1) * (y_{2a} - y_{1a}), \quad j=1,2,\ldots,N \tag{4}$$

(2) Generate attribute value p_{ja} for the synthetic minority class example p_j $(j=1,2,\ldots,N)$ randomly along the line between the attribute values, x_a and t_{ja}, for each temporary example t_j $(j=1,2,\ldots,N)$ and example x.

$$p_{ja} = x_a + \text{rand}(0,1) * (t_{ja} - x_a), \quad j=1,2,\ldots,N \tag{5}$$

Non-numerical attributes fall in ordinal attribute and nominal attribute categories. First, we code the non-numerical attribute. For ordinal attributes, the values are mapped as integers according to the sequences. For nominal attributes, as there are no sequences among different values, we only need to map them into different integers. After that, we generate new values for non-numerical attributes as follows.

- Ordinal attributes

Interpolate in the same way as the numerical values and round the value.

(1) Generate N temporary values t_{ja} for N temporary examples t_j $(j=1,2,\ldots,N)$ randomly along the line between the attribute values, y_{1a} and y_{2a};

$$t_{ja} = y_{1a} + \text{rand}(0,1) * (y_{2a} - y_{1a}), \quad j=1,2,\ldots,N \tag{6}$$

(2) Generate the attribute value p_{ja} for the synthetic minority class example p_j $(j=1,2,\ldots,N)$ randomly along the line between the attribute values, x_a and t_{ja}, for each temporary example t_j $(j=1,2,\ldots,N)$ and example x. Round the attribute value p_{ja}.

$$p_{ja}= \text{round } (xa+ \text{rand}(0,1) * (tja-xa)), \quad j=1,2,\ldots, N, \tag{7}$$

Where round(x) generates integer value after x is rounded.

- Nominal attributes

Unlike numerical and ordinal attributes, it's meaningless to interpolate between values of nominal attributes. Let the values p_{ja} for synthetic example p_j (j=1, 2, …, N) equal the value x_a of the original example x. Synthetic attribute value p_{ja} is generated.

$$p_{ja}= x_a \ , \quad j=1,2,\ldots,N \tag{8}$$

4 Experiments

In this section, we conducted experiments on 10 UCI datasets [15] using the Random-SMOTE and three other popular sampling techniques to evaluate the performance of Random-SMOTE. The classifier used here is *k*-NN algorithm. The ratio of the amount of examples in majority class to the amount of examples in minority class is denoted by imbalance level (*IL*).

4.1 Data Sets

The data sets come from UCI machine learning repository. Among these sets, Pima, German, Hagerman and Transfusion are two-class data sets; the other six are multiclass data sets. We defined our task as learning to distinguish one selected class (minority class) from all other classes (majority class). The selection of minority class here refers to settings in other papers where the class with least or nearly least number of examples is chosen as minority class. Table 1 shows the characteristics of the 10 data sets sorted by *IL* in ascending order, including the class label selected as minority class, number of attributes (# Attr), number of examples in minority class (# Minor), number of examples in majority class (# Major) and imbalance level *(IL)*.

Among sets, 2th and 5th data sets include both numerical and non-numerical attributes.

The data sets used here can be divided into absolute rarity and relative rarity of minority class. Absolute rarity means that the number of examples associated with minority class is small in an absolute sense. Relative rarity means that the number of examples is not rare in absolute sense, but is rare relative to other classes, which makes it hard for greedy search heuristics [16]. It's always the case that the absolute rarity

Table 1. Data sets used in the experiments

ID	Data set	Label	# Attr	# Minor	# Major	*IL*
1	Pima	1	8	268	500	1.87
2	German	2	20	300	700	2.33
3	Haberman	2	3	81	225	2.78
4	Transfusion	1	4	178	570	3.20
5	Cmc	2	9	333	1140	3.42
6	Segmentation	1	19	330	1980	6
7	Glass	7	10	29	185	6.38
8	Satimage	4	36	626	5810	9.28
9	Vowel	0	10	90	900	10
10	Yeast	ME2	8	51	1433	28.1

makes it harder for the classifier to learn from minority class. Among the ten data sets, Haberman, Vowel, Glass and Yeast can roughly fall into the category of absolute rarity, and the other six data sets fall into the category of relative rarity.

4.2 Experimental Design

K-NN algorithm is used here as classification algorithm, where k is set to be 3. *K*-NN uses Euclidean distance to compute the similarity of examples. But it's not applicable for non-numerical attribute. To deal with heterogeneous attributes, we use HEOM (Heterogeneous Euclidean-Overlap Metric) to compute the similarity.

HEOM is an integrated metric, which uses different metrics for different types of attributes. For nominal attribute, overlap metric is used, while for numerical attribute, Euclidean metric is used.

For a given attribute a, we define the distance between two examples x and y on attribute a as $d_a(x,y)$. According to HEOM,

$$d_a(x,y) = \begin{cases} 1, & \text{if x or y is unknown;} \\ overlap(x,y), & \text{if a is non-numerical;} \\ rn\_diff_a(x,y), & \text{if a is numerical} \end{cases} \tag{9}$$

Where

$$overlap(x,y) = \begin{cases} 0, & \text{if x = y} \\ 1, & \text{else} \end{cases} \tag{10}$$

$$rn\_diff_a(x, y) = \frac{|x - y|}{range_a}, \quad range_a = \max_a - \min_a \tag{11}$$

Supposing the examples have m attributes, then the distance between 2 examples x and y HEOM (x, y) is shown below:

$$HEOM(x, y) = \sqrt{\sum_{a=1}^{m} d_a(x_a, y_a)^2} \tag{12}$$

5-fold cross validation with stratified sampling is used in the experiment. More specifically, examples of both minority class and majority class are equally divided into 5 non-overlapping sets. Then each subset of minority class is randomly combined with a different subset of majority class. This setting ensures that the imbalance level of each combined subset is equal to the original data set. After that, one combined subset is used as test set, with the others as training set to train the classifier. This validation is repeated five times, with each combined subset used once as the test set. The experimental results are the average value of five tests.

In random under-sampling, examples of majority class are eliminated randomly until the number of examples in majority class is equal to that of minority class. For the three over-sampling techniques, the same over-sampling rate *N* is set for the ease of comparability. Without loss of generality, the over-sampling rate m is set as follows:

$$N=\text{round}(IL) - 1 \tag{13}$$

where x is rounded in function round(x). Taking glass data set for example, N is 5 as its imbalance level is 6.28.

In scenarios when the data set is very imbalanced, a trivial classifier that labels everything with the majority class can achieve high accuracy. Here, we use *g-mean* and classification accuracy for minority class and majority class as evaluation metrics. *g-mean* is a popular metric for the evaluation of classifiers' performance on imbalanced data sets, and is widely used[3, 9, 17,18].

$$g - mean = \sqrt{acc^+ \times acc^-}, \tag{14}$$

where acc^+ and acc^- are the classification accuracy of majority and minority class respectively.

4.3 Experimental Results

We will compare Random-SMOTE with SMOTE, random under-sampling (RUS) as well as random over-sampling (ROS). The classification results are shown in Fig.3.

We can draw the conclusion that Random-SMOTE is more suitable for the classification of imbalanced data sets compared with the other sampling techniques used here. This can be explained in the following aspects: (1) Random-SMOTE behaves stable on data sets with different characteristics. Although the performances of

different approaches fluctuate on the above ten data sets, Random-SMOTE is always superior to other approaches, and always maintain the leading position. (2) Random-SMOTE has outstanding advantage on data sets of absolute rarity. For data sets of absolute rarity, such as Haberman, Glass and Yeast, Random-SMOTE behaves much better than other approaches. (3) Random-SMOTE behaves better than SMOTE on imbalanced data sets. (4) Random-SMOTE is applicable to data sets with heterogeneous attributes.

In general, Random-SMOTE, as an over-sampling approach, is more suitable for imbalanced datasets. Random-SMOTE can deal with the case of absolute rarity, and can be applied to non-numerical attribute. So, Random-SMOTE is very robust, scalable and applicable.

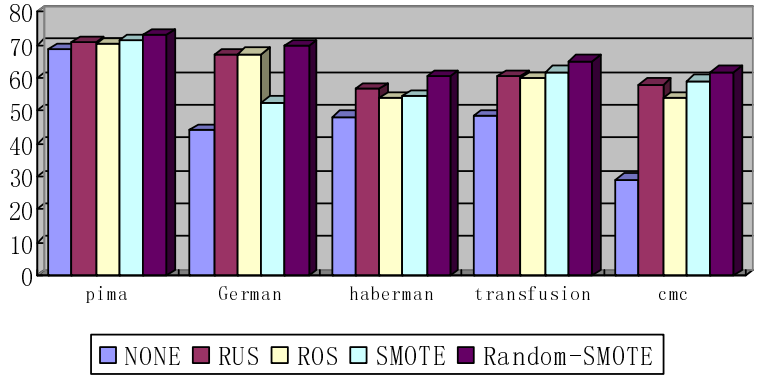

Fig. 3a. g-mean of the first five data sets

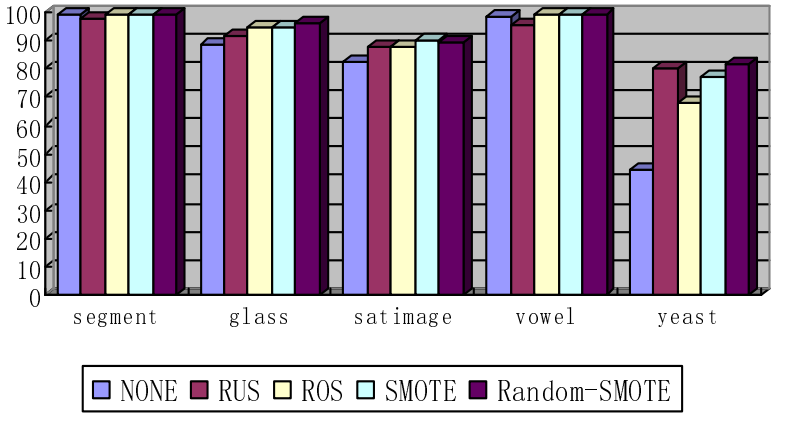

Fig. 3b. g-mean of the last five data sets

5 Conclusion and Future Research

In imbalanced data sets, minority class is inferior to majority class in quantity, and minority class examples distribute sparsely in the sample space, which poses a great challenge to standard classifiers. To overcome this problem, a new over-sampling approach, Random-SMOTE, is proposed in this paper, which generate synthetic examples randomly in sample space of minority class to improve the situation of the sparseness. Although our approach is very simple, the experiments show that as a whole, it performed better according to *g-mean* than other popular sampling techniques, such as random under-sampling, random over-sampling as well as SMOTE. For the case of absolute rarity, Random-SMOTE behaves even better compared with other approaches. This shows that Random-SMOTE is an effective approach for learning from imbalanced data sets.

Because of the diversity and complexity of real-life data sets, the distribution of data is also various. If the underlying distribution of minority class can be estimated, we can generate new examples according to this distribution and better performance will surely be obtained. Other factors, such as concept complexity, class overlapping, noise, within class imbalance may also contribute to the difficulty of classification. Future work will thus include improving the way of generating synthetic examples to make the new data sets generated agree more with the real distribution.

References

1. Burez, J., Van den Poel, D.: Handling class imbalance in customer churn prediction. Expert Systems with Applications 36(3 PART 1), 4626–4636 (2009)
2. Chan, P.K., Stolfo, S.J.: Toward scalable learning with non-uniform class and cost distributions: a case study in credit card fraud detection. In: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, pp. 164–168 (2001)
3. Kubat, M., Holte, R.C., Matwin, S.: Machine learning for the detection of oil spills in satellite radar images. Machine Learning 30(2), 195–215 (1998)
4. Woods, K., Doss, C., Bowyer, K.W., Solka, J., Priebe, C., Kegelmeyer, W.P.: Comparative evaluation of pattern recognition techniques for detection of microcalcifications in mammography. Pattern Recognition and Artificial Intelligence 7, 1417–1436 (1993)
5. Estabrooks, A., Jo, T., Japkowicz, N.: A Multiple Resampling Method for Learning from Imbalanced Data Sets. Computational Intelligence 20(1), 18–36 (2004)
6. Weiss, S., Kapouleas, I.: An empirical comparison of pattern recognition, neural nets and machine learning methods. Readings in Machine Learning (1990)
7. Weiss, G.M., Provost, F.: Learning When Training Data are Costly: The Effect of Class Distribution on Tree Induction. JAIR 19, 315–354 (2003)
8. Estabrooks, A., Japkowicz, N.: A Mixture-of-Experts Framework for Learning from Imbalanced Data Sets. In: Hoffmann, F., Adams, N., Fisher, D., Guimarães, G., Hand, D.J. (eds.) IDA 2001. LNCS, vol. 2189, p. 34. Springer, Heidelberg (2001)
9. Kubat, M., Matwin, S.: Addressing the curse of imbalanced training sets: One-sided selection. In: Proceedings of the Fourteenth International Conference on Machine Learning, pp. 179–186. Morgan Kaufmann, San Francisco (1997)

10. Chan, P., Stolfo, S.: Toward scalable learning with non-uniform class and cost distributions: a case study in credit card fraud detection. In: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, Menlo Park, pp. 164–168 (1998)
11. Visa, S., Ralescu, A.: Experiments in guided class rebalance based on class structure. In: Proc. of the MAICS Conference, pp. 8–14 (2004a)
12. Nickerson, A.S., Japkowicz, N., Milios, E.: Using unsupervised learning to guide re-sampling in imbalanced data sets, pp. 261–265 (2001)
13. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic Minority Over-Sampling Technique. Journal of Artificial Intelligence Research 16, 321–357 (2002)
14. Han, H., Wang, W.-Y., Mao, B.-H.: Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. In: Huang, D.-S., Zhang, X.-P., Huang, G.-B. (eds.) ICIC 2005. LNCS, vol. 3644, pp. 878–887. Springer, Heidelberg (2005)
15. Blake, C., Merz, C.: UCI Repository of Machine Learning Databases (1998), http://www.ics.uci.edu/~mlearn/~MLRepository.html
16. Weiss, G.M.: Mining with Rarity: A Unifying Framework. SIGKDD Explorations 6(1), 7–19 (2004)
17. Wu, G., Chang, E.Y.: Class-Boundary Alignment for Imbalanced Dataset Learning. In: Workshop on Learning from Imbalanced Datasets II, ICML, Washington DC (2003)
18. Guo, H., Viktor, H.L.: Learning from Imbalanced Data Sets with Boosting and Data Generation: The DataBoost-IM Approach. Sigkdd Explorations 6(1), 30–39 (2004)