Feature Selection in Imbalance data sets

Ilnaz Jamali [1], Mohammad Bazmara[2] and Shahram Jafari[3]

[1] *School of Electrical and Computer Engineering, Shiraz university, Shiraz, 71348-51154,Iran*

[2] *School of Electrical and Computer Engineering, Shiraz university, Shiraz, 71348-51154,Iran*

[3] *School of Electrical and Computer Engineering, Shiraz university, Shiraz, 71348-51154,Iran*

Abstract

Feature selection methods have been used these days in the various fields. Like information retrieval and filtering, text classification, risk management, web categorization, medical diagnosis and the detection of credit card fraud. In this paper we focus on feature selection for imbalanced problems. One of the greatest challenges in machine learning and data mining research is the class imbalance problems. Imbalance problems can appear in two different types of data sets: binary problems, where one of the two classes comprises considerably more samples than the other, and multiclass problems, where each class only contains a tiny fraction of the samples. In this paper we want to explain a prior knowledge for an expert system which can tell us which feature selection metrics perform best based on our data characteristics and regardless of the classifier used.

Keywords: feature selection, imbalance data set, Expert system

1. Introduction

Due to the various type of feature selection, and their different results on different data sets, we decided to compare these feature selection metrics on different datasets to show which feature selection metrics performs best on our data. Today we don't have any expert system which can help us to reduce our training and testing time. Researchers should spend a lot of time to find the best feature selection method which can work on their especial data sets. In this paper all methods are implemented in matlab codes and we use a weka package to evaluate them. According to achieve the various knowledge for designing an expert system we evaluate all different methods on different imbalance datasets. The rest of the paper is organized as follows in section 2 we will explain our feature selection metrics. In section 3 we will explain our implementation and our results.

2. Feature Selection Methods

2.1 Correlation coefficient

The correlation coefficient is a statistical test that measures the strength and quality of the relationship between two variables. Correlation coefficients can range from -1 to 1. The absolute value of the coefficient gives the strength of the relationship; absolute values closer to 1 indicate a stronger relationship. The sign of the coefficient gives the direction of the relationship: a positive sign indicates then the two variables increase or decrease with each other and a negative sign shows that one variable increases as the other decreases.
In machine learning problems, the correlation coefficient is used to evaluate how accurately a feature predicts the target independent of the context of other features. The features are then ranked based on the correlation score [11]. For problems where the covariance cov(Xi , Y) between a feature (Xi) and the target (Y) and the variances of the feature (var(Xi)) and target
(var(Y)) are known, the correlation can be directly calculated [2].

2.2 Chi-square

Chi is a statistical test measuring the independence of a feature from the class labels. It is a two-sided metric. Forman noted that this test can behave erratically when there are small expected counts of features; this is fairly common with imbalanced data sets [12]. While the chi-square test generalizes well for nominal data, it breaks down when testing on continuous data [1].

2.3 Odds Ratio

OR looks at the odds of a feature occurring in the positive class normalized by the odds of the feature occurring in the negative class. The standard odds ratio is a one-sided metric. If we have a zero count in the denominator, we replace the denominator with 1. This is consistent with how Forman computed his two-sided odds ratio [14]. A pilot study found the one-sided algorithm performed better on

our data, but other researchers such as Forman [14] have used the two-sided algorithm. This metric is designed to operate solely on binary data sets [1].

2.4 Signal-to-noise Correlation Coefficient

S2N measures the ratio of some desired signal (i.e. the class labels) to the background noise in a feature. While this ratio is originally an electrical engineering concept, in the machine learning community, it has been applied to leukemia classification with strong results [13]. It is a one-sided metric [1].

2.5 Information Gain

IG measures the difference between the entropy of the class labels and the conditional entropy of the class labels given a feature. This measure is two-sided. Like the chi-square test, it generalizes for nominal data but cannot handle continuous data well for similar reasons [1].

2.6 RELIEF

RELIEF is a feature selection metric based on the nearest neighbor rule designed by Kira and Rendell [15]. It evaluates a feature based on how well its values differentiate themselves from nearby points. When RELIEF selects any specific instance, it searches for two nearest neighbors: one from the same class (the nearest hit), and one from the other class (the nearest miss).
This is justified by the thinking that instances of different classes should have vastly different values, while instances of the same class should have very similar values. Because the true probabilities cannot be calculated, we must estimate the difference in equation 3. This is done by calculating the distance between random instances and their nearest hits and misses. For discrete variables, the distance is 0 if the same and 1 if different; for continuous variables, we use the standard Euclidean distance. We may select any number of instances up to the number in the set, and more selections indicate a better approximation [16],[2].

2.7 FAST

Most single feature classifiers set the decision boundary at the mid-point between the mean of the two classes [11]. This may not be the best choice for the decision boundary. By sliding the decision boundary, we can increase the number of true positives we find at the expense of classifying more false positives. Alternately, we could slide the threshold to decrease the number of true positives found in order to avoid misclassifying negatives. Thus, no single choice for the decision boundary may be ideal for quantifying the separation between two classes.
We can avoid this problem by classifying the samples on multiple thresholds and gathering statistics about the performance at each boundary. If we calculate the true positive rate and false positive rate at each threshold, we can build an ROC curve and calculate the area under the curve. Because the area under the ROC curve is a strong predictor of performance, especially for imbalanced data classification problems, we can use this score as our feature ranking: we choose those features with the highest areas under the curve because they have the best predictive power for the dataset. By using a ROC curve as the means to rank features, we have introduced another problem: deciding where to place the thresholds. If there are a large number of samples clustered together in one region, we would like to place more thresholds between these points to find how separated the two classes are in this cluster. Likewise, if there is a region where samples are sparse and spread out, we want to avoid placing multiple thresholds between these points so as to avoid placing redundant thresholds between two points. One possible solution is to use a histogram to determine where to place the thresholds. A histogram fixes the bin width and varies the number of points in
each bin. This method does not accomplish the goals detailed above. It may be the case that a particular histogram has multiple neighboring bins that have very few points. We would prefer that these bins be joined together so that the points would be placed into the same bin. Likewise, a histogram may also have a bin that has a significant proportion of the points. We would rather have this bin be split into multiple different bins so that we could better differentiate inside this cluster of points.
We use a modified histogram, or an even-bin distribution, to correct both of these problems. Instead of fixing the bin width and varying the number of points in each bin, we fix the number of points to fall in each bin and vary the bin width. This even-bin distribution accomplishes both of the above goals: areas in the feature space that have fewer samples will be covered by wider
bins, and areas that have many samples will be covered by narrower bins. We then take the mean of each sample in each bin as our threshold and classify each sample according to this threshold [2].

2.8 FAIR: Feature Assessment by sliding Threshold:

FAIR uses a modification of the FAST algorithm that instead finds the P-R curve associated with a feature's predictions for the class labels. Those features with the greatest area under the P-R curve are selected. This is a two-sided metric [1].

3 implementation and evaluation:

All methods for feature selection which are mentioned in part 2 are implemented in matlab codes and then we use a weka package to evaluate them. According to achieve a reliable results we use different data sets which are mentioned below.

Table 1: data sets

name	*Number of features*	*Number of data in each class*	*Number of all data*
CNS1	7129	30-10	40
CNS2	7129	60-30	90
LEUKAEMIA	7129	48-25	73
LYMPHOMA_1	7129	45-32	77
LYMPHOMA_2	7129	51-26	77
PROSTATE	15154	63-26	89
LUNG	12533	150-30	180
OVARIAN_1	15154	100-16	116
NIPS_1	9344	301-90	391
NIPS_2	9344	301-95	396
NIPS_3	9344	301-144	445
NIPS_4	9344	301-144	445
NIPS_5	9344	301-140	441
NIPS_6	9344	301-152	453
NIPS_7	9344	301-151	453
NIPS_8	9344	301-151	452
IONOSPHERE	34	225-126	351
SONAR	60	111-97	208

As we shown below we can say which feature selection metric performs best according to the number of features we have and the characteristics of the data we have, so we can implement an expert system according to these results.
The performance of a feature selection metric depends on the number of data of each class and the number of features we want to select. To that end, we compared each of the eight feature selection metrics using AUC and PRC. These feature selection metrics were tested by the SVM using different number of features. We evaluated each metric using 10-fold cross validation.
Some feature selection metrics work very well with specific learning methods. The odds ratio metric helps the Naive Bayes classifier achieve the best result possible [17]. RELIEF was designed based on a nearest neighbor philosophy [18], [19] and gives the 1-NN more improvement than simple correlation coefficients [2]. C4.5 and other decision tree algorithms intrinsically use information gain as their node-splitting statistic.
We evaluated the feature selection metrics on the different classifiers. We then took the mean of the performance of each classifier on each evaluation statistic to compare between feature selection metrics.
We consider different issues to find the suitable feature selection method for our data. First we will consider the difference of the number of data in each class of the data set.
Our research show that when the data set is extreme skew and have two classes and for each data of the minority class we have about 5 data in our majority class we should use FAST feature selection method. If for each data of the minority class we have about 3 data IG Is the best method we can use. And if for each data of the minority class we have 2 or less data in our majority class OR Is the best method we can use.

The other thing that we consider here is the number of features we want to select. Because most of the times we need to have extremely limitation for selecting features. When selecting between 10 and 50 features, FAST is the best performer and S2N is the second best. When selecting 100 or more features, S2N is the most effective; FAST is the second most effective until we select 1000 features. Thus, depending on the number of features desired, FAST or S2N would be a good feature selection metric to use regardless of the classifier choice [1].

4. Conclusions

In this paper we show that which feature selection method is better for special data set with special characteristics. This finding will help researchers to reduce their time for finding the suitable feature selection method for their data. This knowledge can be used for designing an expert system for feature selection in imbalance data sets.

References

[1] M. Wasikowski and X. Chen, “Combating the small sample class imbalance problem using feature selection”, IEEE Transactions on knowledge and data engineering, 2009.

[2]X. Chen and M. Wasikowski. “FAST: a roc-based feature selection metric for small samples and imbalanced data classification problems”. In Proceeding of the 14th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining (Las Vegas, Nevada, USA, August 24 - 27, 2008). KDD '08. ACM, New York, NY, 2008, pp. 124-132.

[3] Tian-Yu Liu “Easy Ensemble and Feature Selection for Imbalance Data Sets” International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing,2009.

[4] , D. Casasent and X.-W Chen “ Feature reduction and morphological processing for hyperspectral image data”. Applied Optics, 2004,vol. 43,PP. 1-10.

[5] , N. Japkowicz editor. Proceedings of the AAAI’2000 Workshop on Learning from Imbalanced Data Sets. AAAI Tech Report WS-00-05,2000.

[6] N. Chawla, , N. Japkowicz, , and A.Kolcz, editors 2003. Proceedings of the ICML’2003 Workshop on Learning from Imbalanced Data Sets.

[7] G. Weiss. “Mining with rarity” : A unifying framework. SIGKDD Explorations, 2004, Vol. 6, PP. 7-19.

[8] , M. Kubat and S. Matwin“Addressing the curse of imbalanced data set: One sided sampling”. 1997,pp. 179-186.

[9] X., Chen, B. Gerlach, and, D. Casasent "Pruning support vectors for imbalanced data classification". In Proc. Of International Joint Conference on Neural Networks, 2005, 1883-88.

[10] M. Kubat, and S.Matwin" Learning when negative examples abound". In Proceedings of the Ninth European Conference on Machine Learning ECML97, 1997, PP. 146-153.

[11] I. Guyon, an A Elisseeff " An introduction to variable and feature selection" JMRL special Issue on variable and Feature Selection vol. 3,2003,PP 1157-1182.

[12] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *Journal of Machine Learning Research*, 2003 , vol. 3, pp. 1289–1305.

[13] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, pp. 531–537, 1999.

[14] G. Weiss and F. Provost, "Learning when training data are costly: The effect of class distribution on tree induction," *Journal of Artificial Intelligence Research*, vol. 19, pp. 315–354, 2003.

[15] K. Kira, and, L.Rendell. The feature selection problem: Traditional methods and new algorithms. In Proc. of the 9th International Conference on Machine Learning, 1992,PP. 249-256.

[16]I. Kononenko,. " Estimating attributes: Analysis and extension of RELIEF". In Proc. of the 7th European Conference on Machine Learning,1994, PP.171-182.

[17] D. Mladeni´c and M. Grobelnik, "Feature selection for unbalanced class distribution and naive bayes," in *Proceedings of the 16th International Conference on Machine Learning*, 1999, pp. 258–267.

[18] K. Kira and L. Rendell, "The feature selection problem: Traditional methods and new algorithm," in *Proceedings of the 9th International Conference on Machine Learning*,1992, pp. 249–256.

[19] I. Kononenko, "Estimating attributes: Analysis and extensions of RELIEF," in *Proceedings of the 7th European Conference on Machine Learning*, 1994,pp. 171–182.