Improving Classification Performance with Discretization on Biomedical Datasets

Jonathan L. Lustgarten, MS[1], Vanathi Gopalakrishnan, PhD[1], Himanshu Grover, MS[1], Shyam Visweswaran, MD, PhD[1]
[1]Department of Biomedical Informatics, University of Pittsburgh, Pittsburgh, PA

Abstract

Discretization acts as a variable selection method in addition to transforming the continuous values of the variable to discrete ones. Machine learning algorithms such as Support Vector Machines and Random Forests have been used for classification in high-dimensional genomic and proteomic data due to their robustness to the dimensionality of the data. We show that discretization can help improve significantly the classification performance of these algorithms as well as algorithms like Naïve Bayes that are sensitive to the dimensionality of the data.

Introduction

Discretization is typically used as a pre-processing step for machine learning algorithms that handle only discrete data. In addition, discretization also acts as a variable (feature) selection method that can significantly impact the performance of classification algorithms used in the analysis of high-dimensional biomedical data. This has important implications for the analysis of high dimensional genomic and proteomic data derived from microarray and mass spectroscopy experiments.

Discretization is the process of transforming a continuous-valued variable into a discrete one by creating a set of contiguous intervals (or equivalently a set of cutpoints) that spans the range of the variable's values. Discretization methods fall into two distinct categories: unsupervised, which do not use any information in the target variable (e.g., disease state), and supervised methods, which do. It has been shown that supervised discretization is more beneficial to classification than unsupervised discretization; hence we focus on the former category[1]. Typically, supervised discretization methods will discretize a variable to a single interval if the variable has little or no correlation with the target variable. This effectively removes the variable as an input to the classification algorithm. Liu et al. showed that this variable selection feature of discretization is beneficial for classification[2].

We show that machine learning classification algorithms such as Support Vector Machines (SVM) and Random Forests (RF) that are favored for their ability to handle high-dimensional data, benefit from discretization in the analysis of genomic and proteomic biomedical data. In addition, Naïve Bayes (NB), which is a simple probabilistic classification algorithm that often performs well in many domains, also benefits from discretization when applied to biomedical data.

Methods and Materials

Biomedical Datasets. The 24 biomedical datasets that we used are described in Table 1. All 21 genomic datasets and 2 proteomic datasets are from the domain of cancer, while a third proteomic dataset is from the domain of Amyotrophic Lateral Sclerosis (ALS). Of the genomic datasets, 14 are diagnostic while 7 are prognostic. Out of the 24 datasets, 10 are multi-categorical where the target variable has 3 to 11 classes, while 14 are binary. All datasets except Ranganathan et al. were obtained from the sources given in[3-7]. Ranganathan et al. was acquired from the Bowser lab at the University of Pittsburgh[8]. Table 1 also gives the proportion of the dataset that has the commonest target value (M) and the number of variables (#V).

Discretization Method. We used a new discretization method called the Efficient Bayesian Discretization that we have developed.

Boullé has developed a supervised discretization method called the Minimum Optimal Description Length (MODL) algorithm based on the minimal description length (MDL) principle[9]. The MODL algorithm scores all possible discretization models and selects the one with the best score. This algorithm is optimal in that it examines all possible discretizations of a variable given a dataset of values for the variable and the corresponding target variable values. The optimal MODL algorithm as described by Boullé runs in $O(n^3)$ time where n is the number of instances in the dataset. We have developed a new supervised discretization method called the Efficient Bayesian Discretization (EBD) that uses a Bayesian score to evaluate a discretization model[10]. The Bayesian score is a generalization of the score used in the MODL algorithm. EBD, like MODL, is also an

optimal algorithm but runs faster: in $O(n^2)$ time where n is the number of instances in the dataset. We have shown that EBD has better performance than the commonly used Fayyad and Irani's MDLPC discretization algorithm.

Application of Discretization. We applied EBD in two ways: 1) selecting those variables that had one or more cut points without transforming their continuous values, and 2) selecting those variables that had one or more cut points and transforming the continuous values into the discrete values generated by discretization. This led to the creation of three datasets for every biomedical dataset analyzed: the first was the same as the original dataset, the second consisted of variables selected by discretization but no transformation, and the third consisted of variables selected by discretization with the variables taking on discrete values.

Machine Learning. We applied three machine learning algorithms that can handle both discrete continuous-valued variables, namely, Support Vector Machines (SVM)[11] , Random Forests (RF)[12], and Naïve Bayes (NB). For each biomedical dataset, we performed two runs of 10-fold stratified cross-validation for a total of 20 folds. In each fold, we generated three versions of the dataset as mentioned in the previous section: no variable selection, variables selected by discretization but no transformation, and variables selected by discretization with the continuous values discretized. In each run, the discretization cutpoints were learned only from the training fold and then applied to the training and the corresponding test folds. We averaged the results over the 20 runs to calculate the performance statistics.

For our experiments, we used the implementations of SVM, RF and NB in the Waikato Environment for Knowledge Acquisition (WEKA) version 3.5.6. For SVM, we used the linear kernel and the polynomial kernel of degree 2 with WEKA's default settings. For RF, we used the settings as described in Statnikov and Aliferis[3]. Thus, we selected three different RF parameters: (500, 1), (1000, 2), and (2000, 2) where the first number is the number of trees to be built and the second number is the multiplicative factor of the default value denoting the number of variables to be randomly selected for each tree. For NB with continuous variables, we used a kernel method for

Dataset	Dataset name	Type	P/D	# Classes	# Samples	#V	M
1	Alon et al	Genomic	Diagnostic	2	61	6584	0.651
2	Armstrong et al	G	D	3	72	12582	0.387
3	Beer et al	G	Prognostic	2	86	5372	0.795
4	Bhattacharjee et al	G	D	7	203	12600	0.657
5	Bhattacharjee et al	G	P	2	69	5372	0.746
6	Golub et al	G	D	4	72	7129	0.513
7	HedeNAalk et al	G	D	2	36	7464	0.500
8	Iizuka et al	G	P	2	60	7129	0.661
9	Khan et al	G	D	4	83	2308	0.345
10	Nutt et al	G	D	4	50	12625	0.296
11	Pomeroy et al	G	D	5	90	7129	0.642
12	Pomeroy et al	G	P	2	60	7129	0.645
13	Rosenwald et al	G	P	2	240	7399	0.574
14	Staunton et al	G	D	9	60	7129	0.145
15	Shipp et al	G	D	2	77	7129	0.506
16	Singh et al	G	D	2	102	12599	0.746
17	Su et al	G	D	11	174	12533	0.150
18	Staunton et al	G	D	9	60	5726	0.150
19	Veer et al	G	P	2	78	24481	0.562
20	Welsch et al	G	D	2	39	7039	0.878
21	Yeoh et al	G	P	2	249	12625	0.805
22	Petricoin et al	Proteomic	D	2	322	11003	0.784
23	Pusztai et al	P	D	3	159	11170	0.364
24	Ranganathan et al	P	D	2	52	36778	0.556

Table 1. Datasets used in the discretization experiments. In the Type column G stands for genomic and P for proteomic. In the P/D column P signifies prognostic and D diagnostic. #V is the number of variables. M is the proportion of the dataset that has the commonest target value.

the estimation of the distribution which has been shown to be superior to Gaussian estimation[13].

The abbreviations for the various classification algorithms are as follows: SVM-1 is SVM with a linear kernel, SVM-2 is SVM with a polynomial kernel of degree 2, RF-X-Y is RF with 100*X for the number of trees to be built and Y is the multiplicative factor. NB is Naive Bayes.

Classification Performance Measure. We evaluated classification performance with Relative Classifier Information (RCI). RCI is an entropy-based performance measure that quantifies the amount of uncertainty of a decision problem that is reduced by a classifier relative to classifying using only the prior probabilities of each class[14]. RCI's minimum value is 0% denoting the worst performance while the best performance is 100%, which signifies perfect discrimination. It is similar to the area under the ROC curve (though not equivalent) in that it measures the discrimination power of the classifier while minimizing the effect of the distribution of the classes. Both RCI and the area under the ROC curve (AUC) are better discriminative measures than accuracy; hence we did not use accuracy as an evaluation measure. We did not use AUC since there are several interpretations and methods to compute the AUC when the target variable has more than two values.

Statistical Tests. To compare RCI values, we used the Wilcoxon paired samples signed rank test and the paired samples t-test. The Wilcoxon paired samples signed rank test is a non-parametric procedure used to test whether there is sufficient evidence that the median of two probability distributions differ in location. Being a non-parametric test, it does not make any assumptions about the form of the underlying probability distribution of the sampled population.

The paired samples t-test is a parametric procedure used to determine whether there is a significant difference between the average values of the same performance measure for two different algorithms. The test assumes that the paired differences are independent and identically normally distributed. Although the measurements themselves may not be normally distributed, the pair wise differences often are.

All statistical tests were two-sided and performed at the 0.05 significance level. For each machine learning algorithm we performed the following comparisons: (1) No Variable Selection (NVS) versus Discretization Variable Selection and Transformation (DVST), (2) Discretization Variable Selection (DVS) versus Discretization Variable Selection and Transformation (DVST). To adjust for multiple testing, we utilized the Holm-Bonferroni method[15] which is done as follows. Let there be k hypotheses to be tested and let the overall type 1 error rate be α. The p-values are ordered and the smallest p-value is compared to α/k. If the smallest p-value is less than α/k, the null hypothesis is rejected and the process is repeated with the same α and the remaining k-1 hypotheses. This is continued until the hypothesis with the smallest p-value cannot be rejected. At that point, all null hypotheses that have not been rejected at previous steps are accepted. This method is less conservative than the Bonferroni method and limits the family-wise error rate to the specified α.

Results

Application of EBD resulted in a substantial decrease in the number of selected variables (Table 2). The largest reduction in the number of variables was 98% while the average reduction in the number of variables over all datasets was 61%.

The RCI performance of the machine learning methods under the conditions of NVS, DVS and DVST are given in Table 3. Table 4 gives the results of the paired t-test and the Wilcoxon paired samples signed rank test that compares the RCI performance of DVST with NVS. All the algorithms (for both the t-test and the Wilcoxon test) except SVM-2 retain the significant improvement of RCI with DVST over NVS when corrected for multiple hypothesis testing with the Holm-Bonferroni method.

Table 5 gives the results of the paired t-test and the Wilcoxon paired samples signed rank test that compares the RCI performance of DVST with DVS. All the algorithms (for both the t-test and the Wilcoxon test) except the SVMs (both linear and polynomial kernels) retain the significant improvement of RCI with DVST over DVS when corrected for multiple hypothesis testing with the Holm-Bonferroni method.

Discussion

Overall, discretization with EBD with variable selection and transformation to discrete values, improved the performance of all the algorithms we tested: SVM, RF and NB. In addition, using the discrete values over continuous values for selected variables statistically significantly improved the performance of RF and NB but not the performance of SVM. Transformation of continuous values to

Dataset	# V	Fraction Removed	Remaining #V
1	6584	0.67	2173
2	12582	0.31	8682
3	5372	**0.85**	806
4	12600	0.19	10206
5	5372	**0.93**	376
6	7129	0.60	2852
7	7464	0.69	2314
8	7129	**0.90**	713
9	2308	0.35	1500
10	12625	0.22	9848
11	7129	0.01	7058
12	7129	0.38	4420
13	7399	**0.91**	666
14	7129	**0.93**	499
15	7129	0.39	4349
16	12599	**0.71**	3654
17	12533	0.05	11906
18	5726	**0.78**	1260
19	24481	**0.81**	4651
20	7039	**0.72**	1971
21	12625	**0.98**	253
22	11003	**0.71**	3191
23	11170	**0.85**	1676
24	36778	**0.80**	7356
Average	10376	0.61	4003

Table 2. Effect of discretization by EBD on variable selection. #V refers to the total number of variables, Fraction Removed is the fraction of variables removed by variable selection and Remaining #V is the average number of variables left after discretization. The results were obtained by averaging over a total of 20 folds. Greater than 70% reduction is in bold font.

discrete values provided a 2-8% performance gain in RCI.

The largest gain in performance was seen with NB. This is supported by the observations of Yang and Webb who found that NB benefits from the smoothing of the parameters that discretization provides[16]. With SVM, there was no improvement in performance with discrete values over the use of continuous values for the selected variables. One possible explanation is that each discrete variable is converted to a set of binary variables in WEKA before being presented to the SVM learner. In the setting of DVST, this results in a large increase in the number of variables that may have degraded the performance of SVM.

Algorithm	NVS	DVS	DVST
SVM-1	57.66	**60.59**	**60.95**
SVM-2	58.29	**61.70**	**60.29**
RF-10-2	53.40	**55.46**	**56.07**
RF-20-2	52.98	**54.41**	**55.36**
RF-5-2	52.98	**55.44**	**56.52**
NB	54.37	**56.48**	**57.71**
Average	54.95	**57.35**	**57.81**

Table 3. Averaged RCI across all the datasets. NVS refers to no variable selection, DVS refers to variable selection based on EBD but no transformation to discrete values, and DVST refers to variable selection based on EBD with transformation to discrete values. RCI values for DVS or DVST that are significantly different from NVS on both statistical tests are shown in bold font.

Algorithm	Diff	t-test	Wilcoxon
SVM-1	3.49	**0.014**	**0.006**
SVM-2	2.14	**0.048**	**0.036**
RF-10-2	2.67	**0.015**	**0.020**
RF-20-2	3.53	**0.007**	**0.005**
RF-5-2	3.34	**0.001**	**0.002**
NB	8.42	**0.003**	**< 0.001**

Table 4. Results of the paired t-test and the Wilcoxon paired samples signed rank test on comparing the RIC performance of DVST with NVS. A positive Diff value indicates better performance by DVST. All statistically significant results at the 0.05 significance level are in bold font.

Algorithm	Diff	t-test	Wilcoxon
SVM-1	0.41	0.724	0.546
SVM-2	-1.41	0.091	0.054
RF-10-2	0.61	**0.007**	**0.008**
RF-20-2	1.29	**0.003**	**0.001**
RF-5-2	1.23	**0.013**	**0.006**
NB	2.41	**0.007**	**0.001**

Table 5. Results of the paired t-test and the Wilcoxon paired samples signed rank test on comparing the RCI performance of DVST with DVS. A positive Diff value indicates better performance by DVS. All statistically significant results at the 0.05 significance level are in bold font.

Due to redundancy and noise in biomedical data, variable selection often improves classification performance[2, 17]. The use of discretization in a pre-processing step thus improves classification performance by performing variable selection. In addition, discretization converts continuous values to

discrete ones, which has the potential to further improve classification performance.

In future work, we plan to compare other discretization methods with EBD. We also plan to compare other variable selection methods with discretization.

Conclusion

Discretization is an essential pre-processing step for machine learning algorithms that can handle only discrete data. However, discretization can also be useful for machine learning algorithms that directly handle continuous variables. Our results indicate that the improvement in classification performance from discretization accrues to a large extent from variable selection and to a smaller extent from the transformation of the variable from continuous to discrete.

References

1. Kohavi R, Sahami M. Error-based and entropy-based discretization of continuous features. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining; 1996; Portland, Oregon: AAAI Press; 1996. p. 114-119.
2. Liu H, Setiono R. Feature selection via discretization. Knowledge and Data Engineering 1997;9(4):642-645.
3. Statnikov A, Aliferis CF. Are random forests better than support vector machines for microarray-based cancer classification. In: American Medical Informatics Association Symposium; 2007; Chicago, IL; 2007. p. 686-690.
4. Michiels S, Koscielny S, Hill C. Prediction of cancer outcome with microarrays: A multiple random validation strategy. Lancet 2005;365(9458):488-92.
5. Patel S, Lyons-Weiler J. A web application for the integrated analysis of global gene expression patterns in cancer. Applied Bioinformatics 2004;3(1):49-62.
6. Petricoin EF, III, Ornstein DK, Paweletz CP, Ardekani A, Hackett PS, Hitt BA, et al. Serum proteomic patterns for detection of prostate cancer. Journal of National Cancer Institute 2002;94(20):1576-1578.
7. Pusztai L, Gregory BW, Baggerly KA, Esteva FJ, Laronga C, Gabriel HN, et al. Pharmacoproteomic analysis of pre-and post-chemotherapy plasma samples from patients receiving neoadjuvant or adjuvant chemotherapy for breast cancer. J Clin Oncol (Meeting Abstracts) 2004;22(14_suppl):2109.
8. Ranganathan S. Proteomic profiling of cerebrospinal fluid identifies diagnostic biomarkers for amyotrophic lateral sclerosis. Pittsburgh, PA: University of Pittsburgh; 2003.
9. Boullé M. Modl: A bayes optimal discretization method for continuous attributes. Machine Learning 2006;65(1):131-165.
10. Lustgarten JL, Visweswaran S, Gopalakrishnan V, Cooper GF. Efficient bayesian discretization. Submitted to BMC Bioinformatics 2008.
11. Burges CJC. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 1998;2(2):121-167.
12. Breiman L. Random forests. Machine Learning 2001;45(1):5-32.
13. Witten IH, Frank E. Data mining: Practical machine learning tools and techniques. 2nd Edition ed. San Francisco: Morgan Kaufmann; 2005.
14. Sindhwani V, Bhattacharya P, Rakshit S. Information theoretic feature crediting in multiclass support vector machines. In: Proceedings of the First SIAM International Conference on Data Mining; 2001 April 5-7th, 2001; Chicago, IL; 2001.
15. Holm S. A simple sequentially rejective multiple test procedure. Scandinavian Journal of Statistics 1979;6:65070.
16. Yang Y, Webb G. On why discretization works for Naive-Bayes classifiers. Lecture Notes in Computer Science 2003;2903:440-452.
17. Hauskrecht M, Pelikan R, Malehorn DE, Bigbee WL, Lotze MT, III HJZ, et al. Feature selection for classification of SELDI-TOF-MS proteomic profiles. Applied Bioinformatics 2005;4(4):227-246.