BIOINFORMATICS APPLICATIONS NOTE
Vol. 28 no. 8 2012, pages 1182–1183
doi:10.1093/bioinformatics/bts096

Data and text mining

Advance Access publication February 24, 2012

R/DWD: distance-weighted discrimination for classification, visualization and batch adjustment

Hanwen Huang[1,3,4,*], Xiaosun Lu[1,3], Yufeng Liu[1,2], Perry Haaland[3] and J.S. Marron[1]

[1]Department of Statistics and Operations Research, [2]Carolina Center for Genome Sciences, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599, [3]BD Technologies, 21 Davis Drive, RTP, NC 27709 and [4]Center for Clinical and Translational Sciences, University of Texas Health Science Center, Houston, TX 77030, USA

Associate Editor: Martin Bishop

ABSTRACT

Summary: R/DWD is an extensible package for classification. It is built based on a recently developed powerful classification method called distance weighted discrimination (DWD). DWD is related to, and has been shown to be superior to, the support vector machine in situations that are fundamental to bioinformatics, such as very high dimensional data. DWD has proven to be very useful for several fundamental bioinformatics tasks, including classification, data visualization and removal of biases, such as batch effects. Earlier DWD implementations, however, relied on Matlab, which is not free and requires a license. The major contribution of the R/DWD package is an implementation that is completely in R and thus can be used without any requirements for licensing or software purchase. In addition, R/DWD also provides efficient solvers for second-order-cone-programming and quadratic programming.

Availability and implementation: The package is freely available from cran.r-project.org.

Contact: hanwen.huang@uth.tmc.edu; Perry_Haaland@bd.com

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on January 3, 2012; revised on February 15, 2012; accepted on February 17, 2012

1 INTRODUCTION

Classification plays an important role in the analysis of bioinformatics data and, as a result, has a significant impact on a broad array of applications. The new freely available package R/DWD (http://cran.r-project.org/web/packages/DWD/index.html) provides a powerful classification tool for analyzing high dimensional data. It has been shown (both through intuitive explanation and improved classification error rates) to perform better than the popular support vector machine (SVM) method in high dimensional situations (Marron *et al.*, 2007). SVM is a well-known classification technique and has achieved great success in bioinformatics applications (see Byvatov and Schmeider 2003 for a review). However, as shown in Figure 1(a), SVM suffers from the data piling problem which leads to suboptimal performance in high dimension, low sample size (HDLSS) contexts. This is increasingly relevant as HDLSS problems continue to become more and more common in bioinformatics contexts. Examples include many types of genetic data (such as microarrays, next gen sequencing and many more), medical image analysis (Macenko *et al.*, 2009) and chemometrics. HDLSS data pose a great challenge to many classical statistical multivariate analysis methods. Distance weighted discrimination (DWD) is a recently developed classification method (Marron *et al.*, 2007), which was originally motivated by HDLSS problems but can be applied in many other cases as well. One of the big advantages of DWD over SVM is that it can overcome the data piling problem in high dimensional situations as illustrated in Figure 1b. More discussion about data piling can be found in the Supplementary Material which also includes the classification performance of DWD in comparison with SVM for various real data sets.

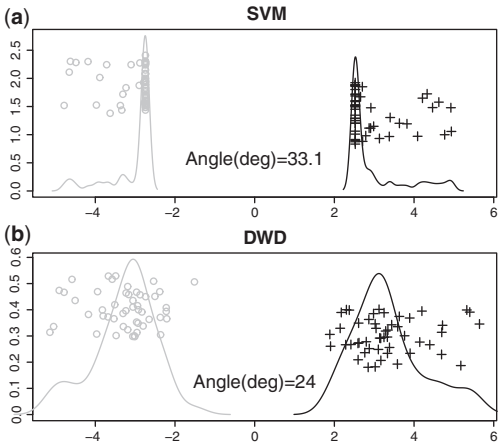

Fig. 1. Superior performance of DWD over SVM in a HDLSS setting. Horizontal axis shows projections, vertical axes are jitter heights, and density. Data piling appears in SVM (**a**) but not in DWD (**b**). The angles from the signal direction also indicate that DWD performs better than SVM.

A simulated toy example is given in Figure 1 with dimension $d = 200$, $n_+ = 50$ data vectors from Class $+1$ (black plus signs), and $n_- = 50$ data vectors from Class -1 (grey circles). The data were drawn from two distributions that are each standard normal but the mean in the first dimension is shifted to $+3$ (-3) for Class $+1$ (-1). The horizontal coordinates represent the projection of the data points onto the SVM and DWD directions. To avoid overplotting of the data points, we use Tukey's jitter plot approach, where one adds a random vertical component to each point. Also included is a smooth histogram for each class, where the vertical axis shows density per unit length. The plot in Figure 1a shows clear piling up of the data

*To whom correspondence should be addressed.

at the margin (the interior points where data from both classes tend to accumulate) which indicates that the SVM is affected by spurious properties of this particular realization of the training data. The plot in Figure 1b shows no 'data piling'.

One of the important applications of DWD in bioinformatics is the adjustment of systematic biases that are present within many types of data. For example, the primary goal of a microarray study is to extract useful information about gene expression and provide insight into biological effects. However, non-biological experimental variation such as batch effects are commonly observed in microarray experiments due to different experimental conditions. Therefore, it is important to identify and adjust for batch effects prior to microarray data analysis. The DWD classification method has been shown to provide effective batch adjustment for microarray data by (Benito *et al.*, 2004). A web implementation for systematic bias adjustment using DWD can be found at the caBIG website https://cabig.nci.nih.gov/tools/DWD. Deeper insights into the effectiveness of DWD relative to competitors, in terms of unknown, unbalanced subtypes was provided by (Liu *et al.*, 2009).

An interesting example, showing both the impact of DWD as a bias adjustment tool, and also as a visualization method is shown in Figure SD in the Supplementary Material. That studies the well known NCI 60 Affymetrics and cDNA microarray data, which were once considered to be impossible to normalize (Kuo *et al.*, 2002). The figure shows that DWD did indeed solve that problem. Furthermore, DWD provides a special viewpoint allowing very clear visual separation of all eight important cancer classes in the data.

The original DWD package was written in Matlab which is not a free software package. To make it much more accessible to bioinformatics researchers, we have now developed an R version of DWD. R/DWD is based on the existing Matlab version (http://www.unc.edu/~marron/marron_software.html), but includes some additional features such as a multiclass version. For the convenience of users who are familiar with using SVM in R, the main classification functions and arguments in `R/DWD` are formatted in a similar way to the ones used by the corresponding SVM functions in the `kernlab` package. To help users of this software, some examples to illustrate the coding are provided.

2 FEATURES

Both SVM and DWD are margin-based classification methods in the sense that they build the classifier through finding a decision boundary to separate the classes (Liu *et al.*, 2011). DWD uses a different criterion from SVM. It seeks to achieve the goal by minimizing the average inverse distance rather than maximizing the minimum distance among the classes. More detailed description about SVM and DWD can be found in the Supplementary Material. Similar to the `ksvm` function from `kernlab`, which has been widely used in SVM analysis, we developed a function called `kdwd` in this package for doing DWD analysis.

The implementation of DWD is more challenging than the implementation of SVM because it requires solving an optimization problem called second-order-cone-programming (SOCP), see (Marron *et al.*, 2007) for a detailed description of the SOCP implementation of DWD, and see (Alizadeh and Goldfarb, 2003) for an introduction to SOCP. The DWD Matlab version employed a very efficient SOCP solver from the `SDPT3` (semidefinite-quadratic-linear programming) package (Toh *et al.*, 1999). Central to R/DWD is the SOCP solver which was implemented using exactly the same algorithm as the one used by the corresponding Matlab version. The optimization problem that underlies SVM is called quadratic programming (QP). An efficient R/QP solver based on the SOCP is also included in this package which provides a useful tool for those users who want to make their own modifications of the SVM method.

Every DWD analysis requires two elements from a dataset: (i) a matrix of predictors, which should be in the form of an $n \times d$ matrix, where n represents the number of samples and d represents the dimension; and (ii) a response vector of length n with each element corresponding to one sample. The basic output from `kdwd` is an object of class *kdwd*, which is very similar to the object of class *ksvm*. Showing objects of class *kdwd* will print details on the results for all classifiers included in the model.

To illustrate the use of R/DWD, an example with the famous `iris` data is shown. First create the training and test sets

```
> data(iris); n=nrow(iris); index=sample(1:n)
> train=iris[index[1:floor(2*n/3)],]
> test=iris[index[((2*ceiling(n/3))+1):n],]
```

Then train the classifier using the training set

```
> irisfit=kdwd(Species~.,data=train)
```

The predicted classes for the test data result from

```
> predict(irisfit,test)
```

3 DISCUSSION

The main purposes of R/DWD are classification, data adjustment and visualization, which are all very important in bioinformatics fields. A variety of additional features are planned for future R/DWD releases. Only the linear discrimination method is considered in the current version. Future plans include incorporating some kernel tricks into the DWD method such that it can be used to solve more general non-linear problems.

Funding: Grant number NIH/NCI R01 CA-149569.

Conflict of Interest: none declared.

REFERENCES

Alizadeh,F. and Goldfarb,D. (2003) Second-order cone programming. *Math. Program. B*, **95**, 3–51.

Benito,M. *et al.* (2004) Adjustment of systematic microarray data biases. *Bioinformatics*, **20**, 105–114.

Byvatov,E. and Schneider,G. (2003) Support vector machine applications in bioinformatics. *Appl. Bioinformatics*, **2**, 67–77.

Kuo,W. *et al.* (2002) Analysis of matched mRNA measurements from two different microarray technologies. *Bioinformatics*, **18**, 405–412.

Liu,X. *et al.* (2009) Visualization of cross-platform microarray normalization. In Scherer,A. (eds) *Batch Effects and Noise in Microarray Experiments: Sources and Solutions*. Wiley, New York, pp. 167–181.

Liu,Y. *et al.* (2011) Soft or hard classification? Large margin unified machines. *J. Am. Stat. Assoc.*, **106**, 166–177.

Macenko,M. *et al.* (2009) A method for normalizing histology slides for quantitative analysis. *Sixth IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Proceedings*, pp. 1107–1110.

Marron,J. *et al.* (2007) Distance-weighted discrimination. *J. Am. Stat. Assoc.*, **102**, 1267–1271.

Toh,K *et al.* (1999) SDPT3 — a Matlab software package for semidefinite programming. *Optim. Meth. Softw.*, **11**, 545–581.