BIOINFORMATICS APPLICATIONS NOTE Vol. 21 no. 20 2005, pages 3940–3941
doi:10.1093/bioinformatics/bti623

Data and text mining

ROCR: visualizing classifier performance in R

Tobias Sing[1,*], Oliver Sander[1], Niko Beerenwinkel[2] and Thomas Lengauer[1]

[1]Department of Computational Biology and Applied Algorithmics, Max-Planck-Institute for Informatics, Stuhlsatzenhausweg 85, 66123 Saarbrücken, Germany and [2]Department of Mathematics, University of California, Berkeley, CA 94720-3840, USA

Received on March 10, 2005; revised on June 1, 2005; accepted on August 9, 2005
Advance Access publication August 11, 2005

ABSTRACT

Summary: ROCR is a package for evaluating and visualizing the performance of scoring classifiers in the statistical language R. It features over 25 performance measures that can be freely combined to create two-dimensional performance curves. Standard methods for investigating trade-offs between specific performance measures are available within a uniform framework, including receiver operating characteristic (ROC) graphs, precision/recall plots, lift charts and cost curves. ROCR integrates tightly with R's powerful graphics capabilities, thus allowing for highly adjustable plots. Being equipped with only three commands and reasonable default values for optional parameters, ROCR combines flexibility with ease of usage.

Availability: http://rocr.bioinf.mpi-sb.mpg.de. ROCR can be used under the terms of the GNU General Public License. Running within R, it is platform-independent.

Contact: tobias.sing@mpi-sb.mpg.de

Pattern classification has become a central tool in bioinformatics, offering rapid insights into large data sets (Baldi and Brunak, 2001). While one area of our work involves predicting phenotypic properties of HIV-1 from genotypic information (Beerenwinkel *et al*., 2002, 2003; Sing *et al*., 2004), scoring or ranking predictors are also vital in a wide range of other biological problems. Examples include microarray analysis (e.g. prediction of tissue condition based on gene expression), protein structural and functional characterization (remote homology detection, prediction of post-translational modifications and molecular function annotation based on sequence or structural motifs), genome annotation (gene finding and splice site identification), protein–ligand interactions (virtual screening and molecular docking) and structure–activity relationships (predicting bioavailability or toxicity of drug compounds). In many of these cases, considerable class skew, class-specific misclassification costs, and extensive noise due to variability in experimental assays complicate predictive modelling. Thus, careful predictor validation is compulsory.

The real-valued output of scoring classifiers is turned into a binary class decision by choosing a cutoff. As no cutoff is optimal according to all possible performance criteria, cutoff choice involves a trade-off among different measures. Typically, a trade-off between a pair of criteria (e.g. sensitivity versus specificity) is visualized as a cutoff-parametrized curve in the plane spanned by the two measures. Popular examples of such trade-off visualizations include receiver operating characteristic (ROC) graphs, sensitivity/specificity curves, lift charts and precision/recall plots. Fawcett (2004) provides a general introduction into evaluating scoring classifiers with a focus on ROC graphs.

Although functions for drawing ROC graphs are provided by the Bioconductor project (http://www.bioconductor.org) or by the machine learning package Weka (http://www.cs.waikato.ac.nz/~ml), for example, no comprehensive evaluation suite is available to date. ROCR is a flexible evaluation package for R (http://www.r-project.org), a statistical language that is widely used in biomedical data analysis. Our tool allows for creating cutoff-parametrized performance curves by freely combining two out of more than 25 performance measures (Table 1). Curves from different cross-validation or bootstrapping runs can be averaged by various methods. Standard deviations, standard errors and box plots are available to summarize the variability across the runs. The parametrization can be visualized by printing cutoff values at the corresponding curve positions, or by coloring the curve according to the cutoff. All components of a performance plot are adjustable using a flexible mechanism for dispatching optional arguments. Despite this flexibility, ROCR is easy to use, with only three commands and reasonable default values for all optional parameters.

In the example below, we will briefly introduce ROCR's three commands—`prediction`, `performance` and `plot`—applied to a 10-fold cross-validation set of predictions and corresponding class labels from a study on predicting HIV coreceptor usage from the sequence of the viral envelope protein. After loading the dataset, a prediction object is created from the raw predictions and class labels.

```
data(ROCR.hiv)
pred <- prediction(
            ROCR.hiv$hiv.svm$predictions,
            ROCR.hiv$hiv.svm$labels)
```

Performance measures or combinations thereof are computed by invoking the `performance` method on this prediction object. The resulting performance object can be visualized using the method `plot`. For example, an ROC curve that trades off the rate of true positives against the rate of false positives is obtained as follows:

```
perf <- performance(pred, "tpr", "fpr")
plot(perf, avg="threshold",
     spread.estimate="boxplot")
```

*To whom correspondence should be addressed.

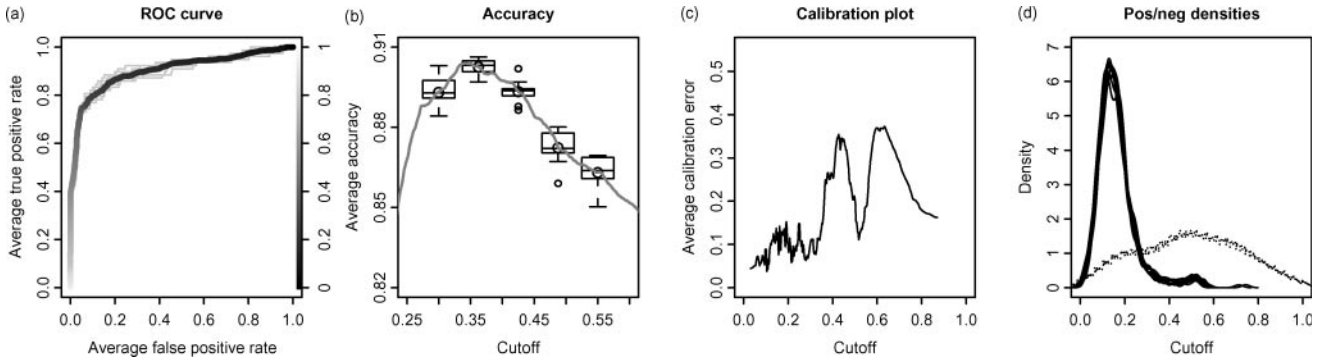

Fig. 1. Visualizations of classifier performance (HIV coreceptor usage data): (**a**) receiver operating characteristic (ROC) curve; (**b**) peak accuracy across a range of cutoffs; (**c**) absolute difference between empirical and predicted rate of positives for windowed cutoff ranges, in order to evaluate how well the scores are calibrated as probability estimates. Owing to the probabilistic interpretation, cutoffs need to be in the interval [0,1], in contrast to other performance plots. (**d**) Score density estimates for the negative (solid) and positive (dotted) class.

Table 1. Performance measures in the ROCR package

Contingency ratios	error rate, accuracy, sensitivity, specificity, true/false positive rate, fallout, miss, precision, recall, negative predictive value, prediction-conditioned fallout/miss.
Discrete covariation measures	Phi/Matthews correlation coefficient, mutual information, χ^2 test statistic, odds ratio
Information retrieval measures	F-measure, lift, precision-recall break-even point
Performance in ROC space	ROC convex hull, area under the ROC curve
Absolute scoring performance	calibration error, mean cross-entropy, root mean-squared error
Cost measures	expected cost, explicit cost

The optional parameter `avg` selects a particular form of performance curve averaging across the validation runs; the visualization of curve variability is determined with the parameter `spread.estimate`.

Issuing `demo(ROCR)` starts a demonstration of further graphical capabilities of ROCR. The command `help(package=ROCR)` points to the available help pages. In particular, a complete list of available performance measures can be obtained via `help(performance)`. A reference manual can be downloaded from the ROCR website.

In conclusion, ROCR is a comprehensive tool for evaluating scoring classifiers and producing publication-quality figures. It allows for studying the intricacies inherent to many biological datasets and their implications on classifier performance.

ACKNOWLEDGEMENT

Work at MPI supported by EU NoE BioSapiens (LSHG-CT-2003-503265).

Conflict of Interest: none declared.

REFERENCES

Baldi,P. and Brunak,S. (2001) *Bioinformatics: The Machine Learning Approach*. MIT Press, Cambridge, MA.

Beerenwinkel,N. *et al*. (2003) Geno2pheno: estimating phenotypic drug resistance from HIV-1 genotypes. *Nucleic Acids Res*., **31**, 3850–3855.

Beerenwinkel,N. *et al*. (2002) Diversity and complexity of HIV-1 drug resistance: a bioinformatics approach to predicting phenotype from genotype. *Proc. Natl Acad. Sci. USA*, **99**, 8271–8276.

Fawcett,T. (2004) ROC graphs: notes and practical considerations for researchers. *Technical Report HPL-2003-4*. HP Labs, Palo Alto, CA.

Sing,T., Beerenwinkel,N. and Lengauer,T. (2004) Learning mixtures of localized rules by maximizing the area under the ROC curve. Valencia, Spain. In *Proceedings of the 1st International Workshop on ROC Analysis in Artificial Intelligence*, 89–96.