

SOFTWARE

Open Access



# PROPER: Performance visualization for optimizing and comparing ranking classifiers in MATLAB

Samad Jahandideh<sup>1\*</sup>, Fatemeh Sharifi<sup>2</sup>, Lukasz Jaroszewski<sup>1</sup> and Adam Godzik<sup>1\*</sup>

## Abstract

**Background:** One of the recent challenges of computational biology is development of new algorithms, tools and software to facilitate predictive modeling of big data generated by high-throughput technologies in biomedical research.

**Results:** To meet these demands we developed PROPER - a package for visual evaluation of ranking classifiers for biological big data mining studies in the MATLAB environment.

**Conclusion:** PROPER is an efficient tool for optimization and comparison of ranking classifiers, providing over 20 different two- and three-dimensional performance curves.

**Keywords:** Predictive modeling, Scoring classifier, Big data, Structural genomics

## Background

One of the main challenges of computational biology is developing new algorithms, tools and software to facilitate analysis of Big Data generated by biomedical research. Ranking or scoring predictors play central roles in a very wide range of biological data analysis problems, e.g. protein structure and function annotation, sequence alignment, genome annotation and many others. Most of biological datasets contain extensive noise that complicates predictive modeling. In different applications optimal predictors may differ depending on the purpose of specific study and characteristics of specific datasets. Thus, optimization and comparison of different prediction methods, selection and evaluation of importance of different features, and simple and efficient validation of predictors' performance is crucial for successful application of machine learning algorithms.

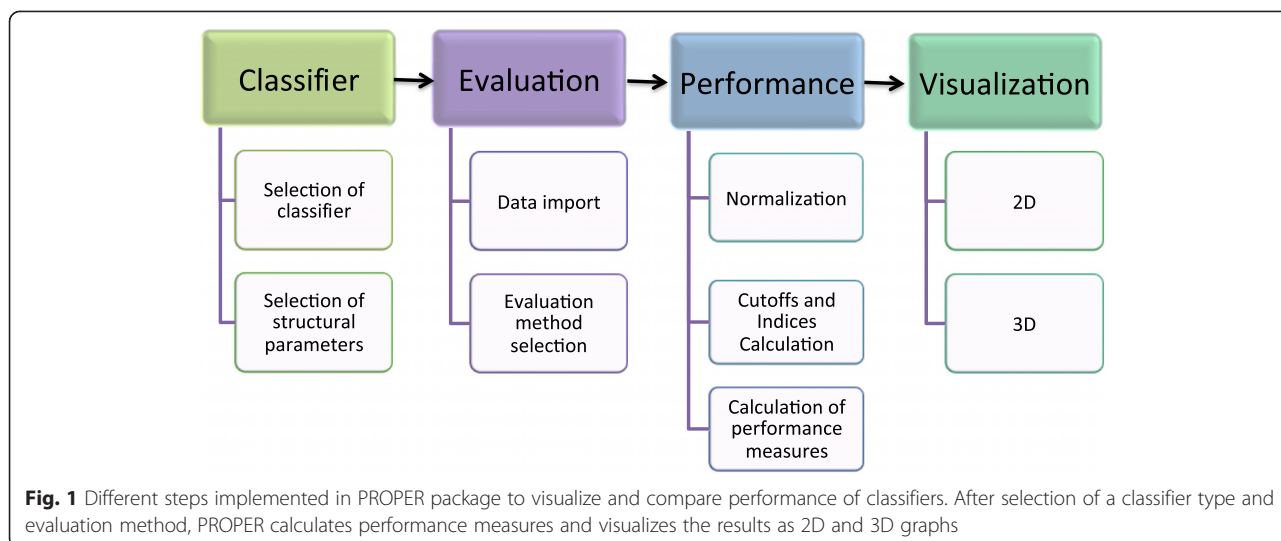
To assist in this task, PROPER provides visual monitoring of optimization and comparison of ranking classifiers. PROPER also allows feature selection and evaluation of features' importance using l1-regularized logistic regression [1] and Random Forest [2], respectively. At the same

time PROPER allows semi-automated optimization of complex methods, such as Artificial Neural Network (ANN). Moreover, output of scoring classifiers currently not implemented in PROPER can be uploaded and used for performance visualization and comparison with available methods.

## Implementation

PROPER is a flexible classifier evaluation package implemented in MATLAB (<http://www.mathworks.com>), a statistical language that is widely used in biomedical data analysis. PROPER imports datasets, applies selected classifiers, evaluates them using different methods, calculates performance measures, and visualizes the results (Fig. 1). Several classifiers and evaluation methods are implemented in PROPER's prediction function, and calculation of performance measures is implemented in performance function. Also, visualization of outputs is implemented in visualization2D function and visualization3D function. All optional parameters of PROPER can be modified by a user when these functions are executed. Graphs plotted using PROPER can be modified and annotated using the MATLAB Plot Editor, which provides user-friendly graphical interface and allows producing high quality figures for publication

\* Correspondence: [sjahandideh@sbspdiscovery.org](mailto:sjahandideh@sbspdiscovery.org); [adam@godziklab.org](mailto:adam@godziklab.org)  
<sup>1</sup>Bioinformatics and Systems Biology Program, Sanford Burnham Prebys Medical Discovery Institute, 10901 N Torrey Pines Rd, La Jolla, CA 92037, USA  
Full list of author information is available at the end of the article



(for details see PROPER manual (Additional file 1) available at <https://sourceforge.net/projects/PROPER-Package>).

### Results and discussion

In PROPER the scoring output of ranking classifiers is translated into a binary class decision by applying a spectrum of cutoffs. Usually no specific cutoff can optimally satisfy all possible performance criteria, hence cutoff choice involves a trade-off between different measures. Typically, a trade-off between a pair of measures (e.g. precision versus recall) is visualized as a cutoff-parametrized curve in the plane spanned by the two measures. Many machine learning and statistical learning packages, e.g. Weka [3] and SLEP [1] are available, but none of them offers standardized comprehensive optimizing, comparison, and performance evaluation of biological classifiers. As no cutoff is optimal according to all possible performance criteria, PROPER allows plotting cutoff-parameterized performance curves for any pair of more than 13 predictors' performance measures and may also plot three-dimensional performance curves by combining three different performance measures in a 3D graph where each facet represents a standard performance curve (Fig. 2).

Calculated performance measures used to comprehensively evaluate the performance of classifiers include:

$$T = \text{True Predictions} = (\text{True Positives} + \text{True Negatives})$$

$$F = \text{False Predictions} = (\text{False Positives} + \text{False Negatives})$$

$$MCC = \frac{(\text{True Positives} * \text{True Negatives}) - (\text{False Positives} * \text{False Negatives})}{\sqrt{(\text{True Positives} + \text{False Positives})(\text{True Positives} + \text{False Negatives})(\text{True Negatives} + \text{False Positives})(\text{True Negatives} + \text{False Negatives})}}$$

$$TPR = \text{Sensitivity} = \text{Recall} = \frac{\text{True Positives}}{(\text{True Positives} + \text{False Negatives})}$$

$$FNR = \text{False Negative Rate} = \frac{\text{False Negatives}}{(\text{True Positives} + \text{False Negatives})}$$

$$FPR = \text{False Positive Rate} = \text{Fallout} = \frac{\text{False Positives}}{(\text{True Negatives} + \text{False Positives})}$$

$$TNR = \text{Negative Negative Rate} = \text{Specificity} = \frac{\text{True Negatives}}{(\text{True Negatives} + \text{False Positives})}$$

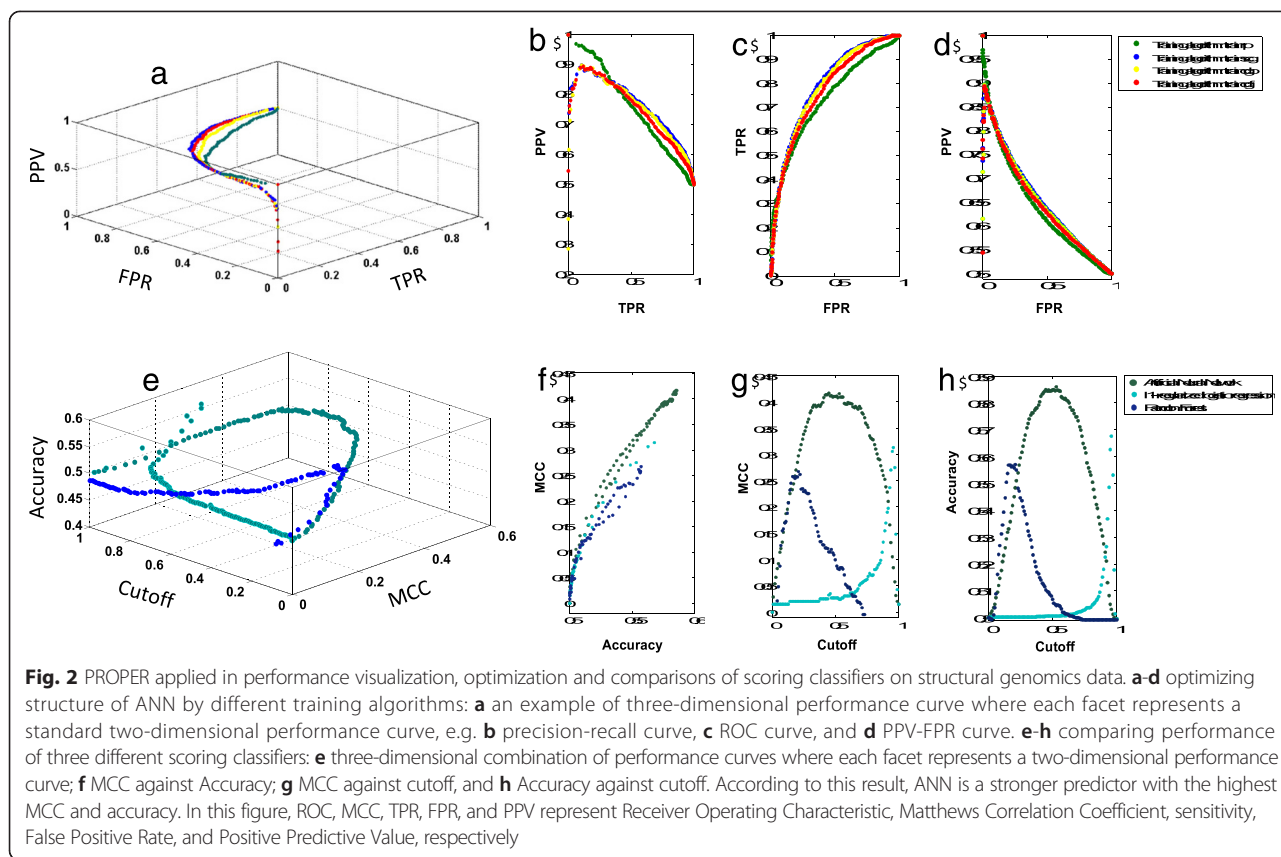
$$PPV = \text{Positive Predictive Value} = \frac{\text{True Positives}}{(\text{True Positives} + \text{False Positives})}$$

$$NPV = \text{Negative Predictive Value} = \frac{\text{True Negatives}}{(\text{True Negatives} + \text{False Negatives})}$$

$$RPP = \text{Rate of Positive Prediction} = \frac{\text{True Positives} + \text{False Positives}}{(\text{True} + \text{False})}$$

$$RNP = \text{Rate of Negative Prediction} = \frac{\text{True Negatives} + \text{False Negatives}}{(\text{True Predictions} + \text{False Predictions})}$$

$$ACC = \text{Accuracy of Classifier} = \frac{\text{True Positives} + \text{True Negatives}}{(\text{True Predictions} + \text{False Predictions})}$$



$$FMeasure = F\_score = 2 \times \frac{(Precision * Recall)}{(Precision + Recall)}$$

Several illustrative examples below demonstrate different features of PROPER. An example presented in Fig. 2 illustrates PROPERs functions, i.e. optimization, comparison, and visualization, applied to independent training and testing sets of data from a study on prediction of protein sequence crystallizability [4]. In this study, we have used a dataset of 5691 protein sequences in negative set and 4924 protein sequences in positive set. For each protein sequence 48 different features were calculated and fed into machine learning methods. This data is available at <http://ffas.burnham.org/XtalPred/help.html>. After loading the data, optimization of model's structure, e.g. selection of ANN learning algorithm, is performed by generating two- and three-dimensional performance curves and then similar curves are generated to compare performance of different optimized models. ANN training begins with initial random weights for each feature and, after each iteration, a learning algorithm changes these weights to reach the highest level of accuracy. Figure 2(a-d) shows differences in performance of four standard learning algorithms applied to training of ANN on this database. More examples and detailed

information about installing PROPER is available from user manual that could be downloaded from the distribution directory at sourceforge.

### Conclusions

In summary, PROPER is a freely available package for performance visualization, comparison and optimization of scoring classifiers in MATLAB. Performance visualization can be applied to output of any scoring classifier available or not available in PROPER. PROPER will be helpful in improving reproducibility and standardization of research in the field of biological Big Data outcome prediction.

### Availability and requirements

- Project name: ROPER
- <https://sourceforge.net/projects/PROPER-Package>
- Operating system(s): Platform independent
- Other requirements: It requires the MATLAB Statistics Toolbox
- Any restrictions to use by non-academics: None

### Additional file

**Additional file 1:** Proper user manual. (DOCX 932 kb)

**Competing interests**

The authors declare that they have no competing interests.

**Authors' contributions**

Conceived and designed the experiments: SJ AG LJ. Analyzed the data: SJ LJ. Contributed re-agents/materials/analysis tools: SA FS LJ. Wrote the paper: SJ AG LJ. All authors read and approved the final manuscript.

**Conflict of interest**

None declared.

**Acknowledgments**

This work is supported by the National Institute of General Medical Sciences of the National Institutes of Health (NIH) under Award No. R01 GM095847 (SER/XtalPred). The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

**Author details**

<sup>1</sup>Bioinformatics and Systems Biology Program, Sanford Burnham Prebys Medical Discovery Institute, 10901 N Torrey Pines Rd, La Jolla, CA 92037, USA.

<sup>2</sup>Division of Computer Science, School of Informatics and Computing, Indiana University, Bloomington, IN 47405, USA.

Received: 13 July 2015 Accepted: 30 November 2015

Published online: 03 December 2015

**References**

1. Liu J, Ji S, Ye J. SLEP: Sparse Learning with Efficient Projections. Arizona State University. 2009. <http://www.public.asu.edu/~jye02/Software/SLEP/citation.htm>.
2. Breiman L. Random Forests. *Mach Learn.* 2001;45:5–32.
3. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA Data Mining Software: An Update; SIGKDD Explorations. 2009; Volume 11, Issue 1. <http://dl.acm.org/citation.cfm?id=1656278>.
4. Jahandideh S, Jaroszewski L, Godzik A. Improving the chances of successful protein structure determination with a random forest classifier. *Acta Cryst D.* 2014;70:627–35.

Submit your next manuscript to BioMed Central and we will help you at every step:

- We accept pre-submission inquiries
- Our selector tool helps you to find the most relevant journal
- We provide round the clock customer support
- Convenient online submission
- Thorough peer review
- Inclusion in PubMed and all major indexing services
- Maximum visibility for your research

Submit your manuscript at  
[www.biomedcentral.com/submit](http://www.biomedcentral.com/submit)

