

Information Measure Toolbox for Classifier Evaluation on Open Source Software Scilab

Bao-Gang Hu^{1,2}

¹ NLPR/LIAMA, Institute of Automation

² Beijing Graduate School, Chinese Academy of Sciences

P.O. Box 2728, Beijing, 100080 China

hubg@nlpr.ia.ac.cn

Abstract—This paper describes an information measure toolbox for classifier evaluations based on an open-source software platform Scilab. Twenty four normalized information measures are given in the toolbox, which are derived from three types of information definitions, namely, mutual information, information divergence, and cross entropy. Different from the conventional performance measures which apply the heuristic or empirical formulas, the information measures are more theoretically sound with a higher degree of applicability, say, classifications including a reject option. The specific attention is paid to the singularity aspect in the implementation of the toolbox. With the toolbox, users are able to test the included numerical examples on binary and three-class classifications easily. The toolbox developed in this work provides users a useful tool of assessing classifiers from an information theoretic basis. The complete source code of the toolbox is available at website “OpenPR”: <http://www.openpr.org.cn/> with a file name of “confmatrix2ni.zip”.

Keywords – abstaining classifier, entropy, mutual information, divergence, evaluation, open source, toolbox

I. INTRODUCTION

Classifier evaluation has been a wide and long time concern in the studies of pattern recognition [1] and data processing [2]. In recent years, it becomes a hot topic in machine learning, data mining, and web search [3-5]. A selection of proper evaluation measures is still an open problem in machine learning. This is particularly true for classifications. If considering binary classifications, one can find out over thirty measures available for uses [5]. However, most of these performance measures are unable to handle the generic classifications including a rejected class.

In engineering and medical practices, a reject option is a common scheme which often improves the classification accuracy [6-9]. In this case, the terms of “abstaining classifiers” are often used to describe the classifiers [8]. This scheme is well adopted in the tests of medical diagnosis because abstaining from making a prediction may be more desirable than making a wrong classification. When applying this scheme, one may employ an “Error-Reject” curve technique for objective evaluations of abstaining classifiers [10]. The “area under the curve (termed *AUC*)” is calculated

as a quantitative measure for a comparison between two classifiers. The larger value of the *AUC* measure, the better the classifier. If two classifiers, denoted by C_A and C_B , have only a single-point observation, say,

$$A(C_A) = 0.8 \text{ when } Rej(C_A) = 0.2,$$

$$A(C_B) = 0.9 \text{ when } Rej(C_B) = 0.4,$$

where *A* is for “Accuracy” and *Rej* for “Rejection” (also called “Abstention”), a measure, *Eff* for “Efficiency”, is used for the classifier assessment [11]:

$$Eff = (A - Rej + 1) / 2.$$

Then, one is able to rank the two classifiers from the calculations of

$$Eff(C_A) = (0.8 - 0.2 + 1) = 0.80,$$

$$Eff(C_B) = (0.9 - 0.4 + 1) = 0.75.$$

and conclude that the classifier C_A is better than C_B due to its higher value on the *Eff* measure. In principle, one can derive that both *Eff* and *AUC* measures are equivalent for a single-point observation.

The classifier evaluation becomes more complicated if considering both error types and rejection types in classifications. Mackay [10, page 533] showed two classifiers for discussion about evaluation measures. The two classifiers, denoted here by C_D and C_E respectively, were given by the following corresponding confusion matrices [10]:

$$C_D = \begin{bmatrix} 74 & 6 & 10 \\ 0 & 9 & 1 \end{bmatrix}, \quad C_E = \begin{bmatrix} 78 & 6 & 6 \\ 0 & 5 & 5 \end{bmatrix},$$

where

$$C = \begin{bmatrix} TP & FN & U_1 \\ FP & TN & U_2 \end{bmatrix},$$

and within the binary classifications, *TP* is denoted for “True Positive”, *FP* for “False Positive”, *FN* for “False Negative”, *TN* for “True Negative”, U_1 and U_2 for the unknown class from Class 1 and Class 2 samples, respectively. The conventional performance measures will show the following results for the two classifiers:

$$A(C_D) = 0.830, \quad Rej(C_D) = 0.110, \quad Eff(C_D) = 0.860,$$

$$P(C_D) = 1.000, \quad R(C_D) = 0.822, \quad FI(C_D) = 0.902,$$

$$A(C_E) = 0.830, \quad Rej(C_E) = 0.110, \quad Eff(C_E) = 0.860,$$

$$P(C_E) = 1.000, \quad R(C_E) = 0.867, \quad FI(C_E) = 0.929.$$

where P is for “Precision”, R for “Recall” and $F1$ for “F1 measure”, respectively. The two classifiers share the same values on accuracy and rejection rates (also for efficiency and precision measures), implying an equivalence of the two classifiers if using an “Error-Reject” curve technique. Further checking on the results from the recall and F1 measures indicates that C_E is a “better” classifier. However, through comparing the confusion matrices carefully, we may mostly consider C_D to be “better” than C_E since it has a higher accuracy on a smaller class. The implication behind this conclusion is attributed to the considerations about the error types and class distributions. That is, a misclassification for a smaller class will suffer more loss than that for a larger class.

In classifier evaluations, if considering error types and rejection types, most investigators may resort to the subjective approach. Taking a binary classifier in abstaining classification for example [6,9,11], one needs to provide a complete information about the cost matrix in a form:

$$\Lambda = \begin{bmatrix} \lambda_{TP} & \lambda_{FN} & \lambda_{UI} \\ \lambda_{FP} & \lambda_{TN} & \lambda_{U2} \end{bmatrix},$$

where λ_{ij} is the term of the cost in associated with the confusion matrix. For example, for the Bayesian classifiers [6], the given λ_{ij} will directly control the decision boundary for the classifications. Without this information, one will be unable to minimize the expected (or Bayesian) risk.

When the different values to the different terms of λ_{ij} are known, one is able to distinguish classifiers with respect to the error types and rejection types. If a measure is defined by free parameters (like λ_{ij}), it is called subjective measure [12]. The main difficulties for using subjective measures in evaluations may come from:

- a) the issue between “subjectivity” and “consistency”, or
- b) the exact knowledge about the free parameters.

Different from the subjective evaluation approach, information-theoretic measures provide a promising objective solution to distinguishing classifiers with respect to error types and rejection types [12]. The pioneer works of introducing information measures for classifier evaluations may be attributed to the studies by Kvålseth [13] and Wickens [14], respectively. More studies appeared in recent years, particularly in the area of remote sensing [15] (although a contingency table is used).

This work is an extension of our early studies [16-18,12] on developing the information measures for classifier evaluations, particularly on the programming implementation. From the author's knowledge, there does not exist any toolbox/toolkit in the public on the related subject. Therefore, this work, under the soon-coming OpenPR Project [19], aims to build up a novel toolbox of information measures for classifier evaluations on open-source software Scilab [20]. We select Scilab platform due to its powerfulness and flexibility in applications from both research and educational purposes. It is also the author's belief that open-source culture and practice will greatly foster scientific progress, which should be much encouraged in the research community [21,22].

This paper is organized as follows. In Sections II, the basic formulas used in the toolbox are given. All information measures appeared in [12] are included into the toolbox. The general descriptions and implementation of the toolbox on the open-source Scilab are presented in Section III. The results on the two numerical examples in this section are shown in Section IV. Closing remarks are given in Section V.

II. FORMULAS

The toolbox in this paper provides all information measures in [12]. For readers to follow up the formulas used in the toolbox, the basic equations are given in this section. Readers can refer [12] for more detailed descriptions about the formulas.

We presume that the basic data available for classifier evaluations is only a confusion matrix. For a generic classification problem, an augmented confusion matrix, C , is considered below in which one column for a rejected class is added on a conventional confusion matrix [17]:

$$C = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1m} & c_{1(m+1)} \\ c_{21} & c_{22} & \dots & c_{2m} & c_{2(m+1)} \\ & & \dots & & \\ c_{m1} & c_{m2} & \dots & c_{mm} & c_{m(m+1)} \end{bmatrix}, \quad (1)$$

where c_{ij} represents the sample number of the i th class that is classified as the j th class. The total class number is m . The row data corresponds to the exact classes, and the column data corresponds to the prediction classes. The last column represents a rejected, or unknown, class.

An empirical joint distribution $p_e(t,y)$ between the output (or prediction) data set y and target data set t for a classifier can be obtained from the confusion matrix in the following form [17]:

$$p_e(t,y) = \frac{1}{n} C, \quad (2)$$

where n is the total number of samples in classifications. Then the three basic definitions are used for forming information measures. The first definition is mutual information [23]:

$$I(T,Y) = \sum_t \sum_y p(t,y) \log_2 \frac{p(t,y)}{p(t)p(y)}, \quad (3)$$

where $p(t)$ and $p(y)$ are the marginal distribution. The second definition is from information divergences, we tried to include mostly-commonly used information divergence measures, such as KL divergence [23]:

$$KL(T,Y) = \sum_z p_t(z) \log_2 \frac{p_t(z)}{p_y(z)}, \quad (4)$$

where we apply the following notations for defining the marginal distributions:

$$p(t) = p_t(z=t) = p_t(z). \quad (5)$$

The third definition is the cross-entropy:

$$H(T;Y) = - \sum_z p_t(z) \log_2 p_y(z). \quad (6)$$

For realizing an objective evaluation, we apply Shannon entropy [24] for the definitions above. Due to the different normalization schemes, twenty four normalized information measures, denoted by NI_k ($k=1,2, \dots, 24$), are formed for a systematic comparison. Tables I-III list all measures used in the toolbox, and readers can see the original reference sources in [12] for some measures.

III. DESCRIPTION AND IMPLEMENTATION OF THE TOOLBOX

In this section, we will describe the general information of toolbox first and then discuss its implementation.

A. General Description

The toolbox is available on the website of OpenPR [19], which stands for “Open Pattern Recognition Project”. The toolbox is given by a file called “confmatrix2ni.zip”. Within this zip file, three individual function files are included (Table IV). The total number of 24 normalized information measures, NI_k ($k=1,\dots,24$), are calculated in the respective function files. All programs are coded by Scilab language and released under the BSD open-source license. The toolbox, still in a preliminary stage, can be extended easily from using Scilab language for including more information measures.

For a convenient use of the toolbox, each function file is designed by including two parts. The first one is the function itself, and the second one is the numerical examples for running the included function. Scilab software enables users to apply “copy” and “paste” operations to the file for running the examples directly. Among the total fifteen examples given in the function files, the first six examples are for binary classifications and the remaining nine examples for three-class classifications. Users are able to test their own examples easily by giving the confusion matrix. If for a formal calling of the function, the second part on numerical examples should be removed.

Because of its feature of being a high-level language, Scilab provides a simple and flexible platform for extensions. One can easily add new information measures and other conventional performance measures for a wider comparison. The present toolbox using Scilab can be easily transferred into other computer languages.

B. Implementation

In the implementation of the toolbox, the specific attention is made on the singularity aspect. When calculating entropy, one may encounter two types of singularities, namely

- a) removable singularity (say, $0 \cdot \log 0 = 0$), and
- b) absolute singularity (say, $a \cdot \log 0 = -\infty$, $a > 0$).

Removable singularity is a specific feature for some function at the singularity point where the function is undefined, but its limit exists at this point. We call the “absolute singularity” if it has no limit.

Scilab program applies command: `ieee(k)` for handling singularity calculations by three modes ($k=0, 1$, or 2). The default mode is `ieee(0)` for warning and stopping when encountering singularity. In the program of the toolbox, a mode `ieee(2)` is suggested in which the program will run until the normal ending even it encounters the calculation for singularity. If users want to know the occurrence of removable singularity for some measure(s), they can select the mode `ieee(0)` with the specific code on singularity examination during the calculations. The present toolbox includes such specific code, but users may need to add code for outputting the singularity (or warning) message to the specific measure(s).

Running the program of the toolbox, one can find that the difficulty from removable singularity is overcome by using the strategy above. All normalized information measures should be fall within the normal range from 0 to 1. However, specific cases do occur to some measures. When one measure shows

TABLE I. NI MEASURES WITHIN THE MUTUAL-INFORMATION BASED GROUP [12].

No.	Name	Formulas on NI_k
1	NI based on mutual information	$NI_1(T, Y) = \frac{I(T, Y)}{H(T)}$
2	NI based on mutual information	$NI_2(T, Y) = \frac{I_{Y T}(T, Y)}{H(T)}$
3	NI based on mutual information	$NI_3(T, Y) = \frac{I(T, Y)}{H(Y)}$
4	NI based on mutual information	$NI_4(T, Y) = \frac{1}{2} \left[\frac{I(T, Y)}{H(T)} + \frac{I(T, Y)}{H(Y)} \right]$
5	NI based on mutual information	$NI_5(T, Y) = \frac{2I(T, Y)}{H(T) + H(Y)}$
6	NI based on mutual information	$NI_6(T, Y) = \frac{I(T, Y)}{\sqrt{H(T)H(Y)}}$
7	NI based on mutual information	$NI_7(T, Y) = \frac{I(T, Y)}{H(T, Y)}$
8	NI based on mutual information	$NI_8(T, Y) = \frac{I(T, Y)}{\max(H(T), H(Y))}$
9	NI based on mutual information	$NI_9(T, Y) = \frac{I(T, Y)}{\min(H(T), H(Y))}$

TABLE II. INFORMATION MEASURES WITHIN THE DIVERGENCE BASED GROUP [12].

No.	Name of D_k	Formulas on D_k , $NI_k = \exp(-D_k)$
10	ED-Quadratic Divergence	$D_{10} = QD_{ED}(T, Y) = \sum_z (p_t(z) - p_y(z))^2$
11	CS-Quadratic Divergence	$D_{11} = QD_{CS}(T, Y) = \log_2 \frac{\sum_z p_t(z)^2 \sum_z p_y(z)^2}{[\sum_z (p_t(z) p_y(z))]^2}$
12	KL Divergence	$D_{12} = KL(T, Y) = \sum_z p_t(z) \log_2 \frac{p_t(z)}{p_y(z)}$
13	Bhattacharyya Distance	$D_{13} = D_B(T, Y) = -\log_2 \sum_z \sqrt{p_t(z) p_y(z)}$
14	χ^2 (Pearson) Divergence	$D_{14} = \chi^2(T, Y) = \sum_z \frac{(p_t(z) - p_y(z))^2}{p_y(z)}$
15	Hellinger Distance	$D_{15} = H^2(T, Y) = \sum_z (\sqrt{p_t(z)} - \sqrt{p_y(z)})^2$
16	Variation Distance	$D_{16} = V(T, Y) = \sum_z (p_t(z) - p_y(z)) $
17	J divergence	$D_{17} = J(T, Y) = \sum_z p_t(z) \log_2 \frac{p_t(z)}{p_y(z)} + \sum_z p_y(z) \log_2 \frac{p_y(z)}{p_t(z)}$
18	L (or JS) divergence	$D_{18} = L(T, Y) = KL(T, M) + KL(Y, M)$, $M = \frac{(p_t(z) + p_y(z))}{2}$
19	Symmetric χ^2 Divergence	$D_{19} = \chi^2_s(T, Y) = \sum_z \frac{(p_t(z) - p_y(z))^2}{p_y(z)} + \sum_z \frac{(p_y(z) - p_t(z))^2}{p_t(z)}$
20	Resistor Average Distance	$D_{20} = D_{RA}(T, Y) = \frac{KL(T, Y)KL(Y, T)}{KL(T, Y) + KL(Y, T)}$

the calculation result like “Inf” after running the program, it means that the current measure encounters the absolute singularity from the given confusion matrix. In this case, that measure becomes inappropriate for achieving the meaningful evaluation to the classifier investigated.

For keeping the program code simple and understandable, the present toolbox does not include the code for input data checking on a confusion matrix. One needs to make sure that the given matrix should satisfy the following constrains for a proper evaluation of a classifier [17]:

$$C_i = \sum_{j=1}^{m+1} c_{ij}, C_i > 0, c_{ij} \geq 0, i = 1, \dots, m, j = 1, \dots, m+1. \quad (7)$$

The constraints require that the matrix be given in a size of m by $(m+1)$, each term of the matrix be nonnegative, and each class number be bigger than zero.

IV. NUMERICAL EXAMPLES

Users are able to test all fifteen examples attached in the individual function file of the toolbox. The confusion matrices in the examples are specifically designed so that users can examine the effects of error types and rejection types, respectively. Although we engage an objective evaluation, our intuitions do exist for error types and rejection types for unbalanced classifications. Numerical solutions on the given examples present a direct observation on which information measure satisfies our intuitions. The detailed discussions about the fifteen examples are given in [12].

In this paper, we only discuss the two classifiers given in the introduction, C_D and C_E . The two examples were originally given by Mackay in [10]. Although the two examples are artificially designed, they do demonstrate the meaningful and important issue behind the classifications with a reject option.

TABLE III. NI MEASURES WITHIN THE CROSS-ENTROPY BASED GROUP [12].

No.	Name of NI_k	Formulas on NI_k
21	NI based on cross-entropy	$NI_{21} = \frac{H(T)}{H(T; Y)}$, $H(T; Y) = -\sum_z p_t(z) \log_2 p_y(z)$
22	NI based on cross-entropy	$NI_{22} = \frac{H(Y)}{H(Y; T)}$, $H(Y; T) = -\sum_z p_y(z) \log_2 p_t(z)$
23	NI based on cross-entropy	$NI_{23} = \frac{1}{2} \left(\frac{H(T)}{H(T; Y)} + \frac{H(Y)}{H(Y; T)} \right)$
24	NI based on cross-entropy	$NI_{24} = \frac{H(T) + H(Y)}{H(T; Y) + H(Y; T)}$

TABLE IV. FUNCTION FILE LIST IN THE TOOLBOX.

Function File Name	NI_k	Description of NI_k Applied
confmatrix2ni_mi.sci	$NI_1 - NI_9$	Defined from Mutual Information
confmatrix2ni_id.sci	$NI_{10} - NI_{20}$	Defined from Information Divergence
confmatrix2ni_ce.sci	$NI_{21} - NI_{24}$	Defined from Cross Entropy

TABLE V. RESULTS OF INFORMATION MEASURES FOR THE EXAMPLES SHOWN IN [10]. S = ABSOLUTE SINGULARITY .

Model	NI_1	NI_2	NI_3	NI_4	NI_5	NI_6	NI_7	NI_8	NI_9	NI_{10}	NI_{11}	NI_{12}
C _D	0.586	0.586	0.254	0.420	0.355	0.386	0.215	0.254	0.586	0.961	0.959	0.822
C _E	0.534	0.393	0.255	0.395	0.345	0.369	0.209	0.255	0.534	0.974	0.971	0.842
Model	NI_{13}	NI_{14}	NI_{15}	NI_{16}	NI_{17}	NI_{18}	NI_{19}	NI_{20}	NI_{21}	NI_{22}	NI_{23}	NI_{24}
C _D	0.913	0.851	0.884	0.726	S	0.879	S	S	0.706	0.0	0.353	0.0
C _E	0.918	0.879	0.892	0.787	S	0.890	S	S	0.732	0.0	0.366	0.0

Mackay [10] suggested to apply mutual information measures for evaluation, but did not present the numerical results. Table V shows the results of all 24 information measures on the two classifiers. One can see that some measures may fail to fulfill the evaluations since the absolute singularity occurs in the calculations (Symbol ‘‘S’’ for singularity result in Table V). At the same time, the measures showing zero values also indicate (by singularity checking with `ieee(0)` mode) the occurrence of removable singularity during the calculations for the given examples. However, most information measures are applicable and effective for realizing the evaluations without such difficulties.

One still can try to apply the conventional (or performance) measures listed in [5]. Then, one may observe that the most conventional measures will fail to conduct such evaluation task on the given examples. The examples shown in [10] do support the need of resorting information measures for an objective evaluation of abstaining classifiers.

In comparing the two classifiers with NI_k , a higher value indicates a better classifier. It is interesting to see that the conclusion may change due to using different measures. Within the mutual information group, the normalized information measures support Classifier C_D to be better than C_E. However, the measures from other two groups will prefer the selection of Classifier C_E, instead of C_D. In [12], we proposed and discussed the ‘‘meta-measures’’ (or ‘‘measures about measures’’) from a high level of evaluation knowledge.

The numerical examples here demonstrate that the toolbox will provide users a useful tool to examine and understand the differences among the information measures. Readers can test more complicated examples, say, multi-class problems. Numerical investigations will be helpful for reaching empirical guidelines in the selection of measures.

In [18], we derived for a first time the nonlinear relations between NI_1 and conventional performance measures. From using the toolbox, one will be easy to see that the nonlinearity exist for all other information measures.

V. CONCLUSIONS

This work introduces an open-source toolbox of information measures for classifier evaluations. To the author's knowledge, this is the first toolkit on the subjects in the public. The toolbox includes 24 normalized information measures for objective evaluations. Applying the toolbox, users are able to obtain the empirical findings about cons and pros of each measure from the numerical examples. The toolbox was developed on the open-source software Scilab so that it provides users a better degree of flexibility in their applications. Users are fully free in open-source senses for testing their own examples, or modifying the toolbox from their needs.

This work does not mean that information-based measures will replace the conventional measures for classifier evaluations. Information-based measures do open a wider perspective for our deeper understanding of the mechanism of learning machines. However, a more systematic comparison with other heuristic measures (say, similarity measures [25, 26] such as Jaccard coefficient, Dice coefficient, and Rand Index) will be necessary. As discussed in the introduction of this paper, classifier evaluations are still an open problem in machine learning. We need not only theoretical study but also numerical tools. In the cases of multi-class classifications, a toolbox will be more proper and important for numerical investigations. It is the author's hope that the information measure toolbox developed in this work will promote more systematic studies on the comparisons among both information-based measures and performance-based measures.

ACKNOWLEDGMENT

This work was supported in part by NSF of China (No. 60275025) and Institute of Automation, Chinese Academy of Sciences. Thanks to Dr. Serge Steer and Dr. Claude Gomez for helps on using Scilab. Special thanks to the anonymous referees whose valuable comments have improved the presentation of this paper.

REFERENCES

- [1] R.O. Duda, P.E. Hart, and D. Stork, *Pattern Classification* (2nd eds.). John Wiley, NY, 2001.
- [2] C.M. Bishop, *Neural Networks for Pattern Recognition*. Clarendon Press, London, 1995.
- [3] N. Japkowicz, "Why question machine learning evaluation methods", AAAI Workshop on Evaluation Methods for Machine Learning, 2006.
- [4] J. Huang and C.X. Ling, "Constructing new and better evaluation measures for machine learning". in Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI), pp. 859-864, 2007.
- [5] N. Lavesson, and P. Davidsson, "Analysis of multi criteria methods for classifier and algorithm evaluation". in Proc. 24th Annual Workshop of the Swedish, 2007.
- [6] C. Chow, "On optimum recognition error and reject tradeoff", IEEE Transactions on Information Theory, vol. 16, pp. 41-46, 1970.
- [7] F. Tortorella, "An optimal reject rule for binary classifiers", in Advances in Pattern Recognition, Joint IAPR International Workshops SSPR 2000 and SPR 2000, volume 1876 of Lecture Notes in Computer Science, pp. 611-620, Alicante, Spain, 2000. Springer-Verlag.
- [8] T. Pietraszek, "On the use of ROC analysis for the optimization of abstaining classifiers", Machine Learning, vol. 68, pp. 137-169, 2007.
- [9] S. Vanderlooy, I.G. Sprinkhuizen-Kuyper, E. N. Smirnov, and H.J. van den Herik, "The ROC isometrics approach to construct reliable classifiers", Intelligent Data Analysis, vol. 13, pp. 3-27, 2009.
- [10] D.J.C. Mackay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 2003.
- [11] C. Ferri and J. Hernandez-Orallo, "Cautious classifiers", in Proceedings of first International Workshop on the ROC Analysis in Artificial Intelligence, pp. 27-36, 2004.
- [12] B.-G. Hu, R. He, and X.-T. Yuan, "Information-theoretic measures for objective evaluation of classifiers", submitted to a journal, (2009)
- [13] T.O. Kväseth, "Entropy and correlation: Some comments", IEEE Transactions on Systems, Man, and Cybernetics, vol. 17, pp. 517-519, 1987.
- [14] T.D. Wickens, *Multiway Contingency Tables Analysis for the Social Sciences*. Lawrence Erlbaum, Hillsdale, New Jersey, 1989.
- [15] C. Liu, P. Frazier and L. Kumar, "Comparative assessment of the measures of thematic classification accuracy", Remote Sensing of Environment, vol. 107, pp. 606-616, 2007.
- [16] Y. Wang and B.-G. Hu, "A study on integrated evaluating kernel classification performance using statistical methods", Chinese Journal of Computers, vol. 31, pp. 64-74, 2008 (in Chinese)
- [17] B.-G. Hu, and Y. Wang, "Evaluation criteria based on mutual information for classifications including rejected class", Acta Automatica Sinica, vol. 34, pp. 1396-1403, 2008.
- [18] Y. Wang, and B.-G. Hu, "Derivations of Normalized Mutual Information in Binary Classifications", in Proceedings of the 6th International Conference on Fuzzy Systems and Knowledge Discovery, IEEE Press, pp. 155-163, 2009.
- [19] OpenPR Project website, <http://www.openpr.org.cn/>
- [20] Scilab website, <http://www.scilab.org/>
- [21] G. von Krogh and E. von Hippel, "The Promise of Research on Open Source Software", Management Science, vol. 52, pp. 975-983, 2006.
- [22] S. Sonnenburg, M. L. Braun, C. S. Ong, S. Bengio, L. Bottou, G. Holmes, Y. LeCun, K.-R. Müller, F. Pereira, C. E. Rasmussen, G. Rätsch, B. Schölkopf, A. Smola, P. Vincent, J. Weston and R. Williamson, "The need for open source software in machine learning", Journal of Machine Learning Research, vol. 8, pp. 2443-2466, 2007.
- [23] T.M. Cover and J.A. Thomas, *Elements of Information Theory*. John Wiley, NY, 2006.
- [24] C.E. Shannon, "A mathematical theory of communication", Bell System Technical Journal, vol. 27, pp. 379-423 and pp. 623-656, 1948.
- [25] D. Pfützner, R. Leibbrandt, and D. Powers, "Characterization and evaluation of similarity measures for pairs of clusterings", Knowledge and Information Systems, vol. 19, pp. 361-394, 2009.
- [26] F. R. Pinto, J.A. Carriço1, M. Ramirez1 and J. S. Almeida, "Ranked Adjusted Rand: integrating distance and partition information in a measure of clustering agreement", BMC Bioinformatics, 2007.