Comparing Multi-class Classifiers: On the Similarity of Confusion Matrices for Predictive Toxicology Applications

Mokhairi Makhtar, Daniel C. Neagu, and Mick J. Ridley

School of Computing, Informatics and Media, University of Bradford, Bradford, BD7 1DP, UK
{M.B.Makhtar,D.Neagu,M.J.Ridley}@Bradford.ac.uk

Abstract. Calculating the similarity of predictive models helps to characterize the models diversity and to identify relevant models from a collection of models. The relevant models are considered based on their performance, calculated using their confusion matrix. In this paper, we propose a methodology to measure the similarity for predictive models performances by comparing their confusion matrices. In this research, we focus on multi-class classifiers for toxicology applications. The performance measures of confusion matrices of multi-class classifiers are regrouped into a binary classification problem. Such approach may result in selecting multi-class classifiers with lower False Negative Rate (FNR) for example. Consequently, the methodology for model comparison based on the similarity of confusion matrices provides a working way to select models from a collection of classifiers.

Keywords: Similarity of Confusion Matrices, Classifiers Comparison, Multi-Class Classifiers.

1 Introduction

Predictive models comparison helps in finding how similar models are. But relying only on standard performance indicators such as accuracy may not give much clue on the overall or specific quality of a predictive model. Sometimes the accuracy might be biased for a certain class and this may not provide a good indication of the overall performance for the predictive model. In this case the accuracy is not necessarily the best measurement for predictive models, whereas the confusion matrix is still the most valuable source of performance indicators from classifiers to be analyzed.

Our motivation is given by the need of analyzing the multi-class classifier models for selected classes. In toxicology, we are mostly interested in the toxic class being predicted correctly. Using the confusion matrix as the information source of classifiers performance, we can adapt more useful measurements related to our objective. The classifiers can be either binary class or multi-class models. In our case, we want to predict if the chemical compound is toxic or non-toxic where all our

H. Yin, W. Wang, and V. Rayward-Smith (Eds.): IDEAL 2011, LNCS 6936, pp. 252–261, 2011.

classifiers are in a multi-class format. The multi-class classifiers can be used as binary class models. It is done by combining the multi-class dataset into a new dataset with only binary classes of toxic and non-toxic output [1, 2] and re-generate new predictive models related to the new datasets. But the solution requires much effort in converting datasets to new binary class sets and retraining the models with the new datasets. To be more practical because there are thousands of models in a collection of models, we propose to use the multi-class classifiers confusion matrices as new binary class classifiers confusion matrices. The practical method is to transform the multi-class confusion matrices into binary confusion matrices without updating the datasets and re-generating the models. This will confirm that the original structures and information the predictive models learned remain unchanged. We will demonstrate the proposed technique in section 3.

In this paper we propose a technique to compare multi-class predictive models' performance measures based on confusion matrices. Our methodology addresses model selection, where comparing the classifiers' performance for each class will lead to usefully diverse predictive models for the class of interest from model ensembles.

The rest of the paper is structured as follows: Section 2 presents related work on reducing multi-class into binary class problem. Section 3 defines the technique proposed for comparison of confusion matrices for multi-class (toxicology) models. In Section 4 we introduce and exemplify the technique to calculate the performance measures of output for multi-class predictive models represented by their confusion matrix. Experiments and results are discussed in Section 5. The paper ends with conclusions on current work and further research directions.

2 Reducing Multi-class to Binary Classification Problems

Sometimes, the multi-class classification problems can still be solved with binary classifiers. Such a solution may divide the original multi-class dataset into two class subsets, learning a different binary model for each subset. These techniques are known as binarisation strategies. There are three main approaches: *One-vs-All* (OVA), *One vs-One* (OVO), and *Error Correcting Output Codes* (ECOC) [2].

All of these techniques decompose a complex multi-class to a simpler binary class problem. Hence this strategy may improve the performance because the classifiers have an easier task to distinguish between only two classes rather than many classes.

In this paper we want to investigate whether there are any differences in performance between binarisation strategies by regenerating new binary classifiers from multi-class classifiers. We calculate the performance measures using multi-class classifiers confusion matrices without retraining new binary classifiers.

In the next section, we will discuss on the performance measures related to binary classification classifiers and propose a methodology to reduce multi-class problems to a binary version while calculating the performance measures of the multi-class classifiers with a focus on lower False Negative Rate (FNR) for example, as required in toxicity prediction problems.

3 Performance Measures and Confusion Matrix for Multi-class Classifiers

The confusion matrix contains information about learned and predicted classifications done by a classification model [3]. Table 1 shows the confusion matrix for a two class classifier. The performance measures for two-class classifiers can be calculated from the confusion matrix [3],[4]: sensitivity *TPR = TP/(TP+FN)* is the rate of correct predictions for the positive output (e.g. Yes or True), *FPR = FP/(FP+TN)* is the rate of incorrect predictions for the positive output (e.g. No or False), specificity *TNR =TN/(TN+FP)* is the rate of correct predictions for the negative output, and *FNR = FN / (TP+FN)* is the rate of incorrect predictions for the negative output. Accuracy *ACC = (TP+TN) / (TP+FP+FN+TN)* measures the correct predictions for all classes.

Table 1. Confusion Matrix of Binary Classification: True Positive (TP), True Negative (TN), False Negative (FN) and False Positive (FP)

		Actual	
	Classes	Toxic	Non-toxic
Predicted	Toxic	TP	FP
	Non-toxic	FN	TN

The confusion matrix for a multi-class classification problem is a generalization of the binary case. Below we discuss the properties and the performance measures derived from a multi-class confusion matrix. Table 2 is an example of a multi-class confusion matrix. For the first column (Class A) the intersection with the first row is the True Positive (TP) value for Class A. The sum of values from remaining cells of the column is the False Negative (FN) value for Class A. True positives for second and third columns are the diagonal values of the confusion matrix.

Table 2. Confusion Matrix for a 3-Class Classifier

	Class A	Class B	Class C
Class A	$TP_{A\,(1,1)}$	$e_{AB\,(1,2)}$	$e_{AC\,(1,3)}$
Class B	$e_{BA\,(2,1)}$	$TP_{B\,(2,2)}$	$e_{BC\,(2,3)}$
Class C	$e_{CA\,(3,1)}$	$e_{CB\,(3,2)}$	$TP_{C\,(3,3)}$

The classification accuracy of a multi-class classifier is the ratio of the sum of the principal diagonal values to the total of values in the confusion matrix. If *C* indicates the confusion matrix, the classification accuracy ACC_c can be defined [5] as:

$$ACC_C = \left(\sum_{i=1}^{N} C_{ii} \Big/ \sum_{i=1}^{N} \sum_{j=1}^{N} C_{ij} \right) \tag{1}$$

where *N* is the number of classes, *i* refers to the rows index and *j* refers to the columns index for the confusion matrix *C*. The Error Rate (ER) for the classifiers is the complement of the accuracy: *ER = (1 – ACC)*.

Beside the accuracy ACC_C and the error rate (*ER*), we can derive other performance measures that can be used to measure multi-class classifiers' quality. Moreover the performance measures of the two-class classification problem can be applied by regrouping the multi-class confusion matrix into two-class classification measures.

In predictive toxicology applications, there is more interest in the false negative rate (*FNR*) measurement of the cases the model fails to correctly classify the instances to the appropriate classes. To give more flexibility for such applications for multi-class classifiers comparison, we propose that the positive (toxic) class and negative (non-toxic) class can be selected by regrouping them into a two class problem. Furthermore this technique is also highly recommended in classifier ensembles where good combination of classes and models will make the binary prediction more accurate [5, 6].

The performance measures for the positive (toxic) class in predictive multi-class classifiers are described below. Let's say the selected toxic classes are Class A (e.g. Very Toxic, column 1) and Class B (e.g. Toxic, column 2) in Table 2. The selected class indexes are stored into the one-row vector *V*. Thus *V= (1, 2).* The proposed *TPR* and *FNR* measures for the selected classes are as follow:

$$TPR_{Ma} = \left(\sum_{\substack{x=1,\\ j=V_x}}^{C} \sum_{\substack{y=1,\\ i=V_y}}^{C} R_{ij} \Big/ \sum_{\substack{x=1,\\ j=V_x}}^{C} \sum_{i=1}^{N} R_{ij} \right) \tag{2}$$

$$FNR_{Ma} = \left(\sum_{\substack{x=1,\\ j=V_x}}^{C} \sum_{\substack{y=1,\\ i\neq V_y}}^{N} R_{ij} \Big/ \sum_{\substack{x=1,\\ j=V_x}}^{C} \sum_{i=1}^{N} R_{ij} \right) \tag{3}$$

where: *N* is the number of samples of all classes in the confusion matrix *R*, *C* is the number of selected class samples for the confusion matrix *R*, *i* is the row index in the confusion matrix *R*, *j* is the column index in the confusion matrix *R*, *x* and *y* are counters for columns and rows, and *V* is a vector of selected class indexes.

The performance measures for the non-toxic class, False Positive Rate (*FPR*) and True Negative Rate (*TNR*), can be derived by adapting equation (2) and equation (3).

Table 3. Confusion Matrix (M_{M1}) for Model M1

	Class A	Class B	Class C
Class A	10 *(1,1)*	21 *(1,2)*	33 *(1,3)*
Class B	24 *(2,1)*	53 *(2,2)*	26 *(2,3)*
Class C	17 *(3,1)*	18 *(3,2)*	19 *(3,3)*

Let's say Model M1 produced a confusion matrix M_{M1} (see Table 3). Referring to the equation 2 and equation 3, we demonstrate how to calculate the *TPR* and *FNR* of toxic classes. For these examples we select two classes as toxic classes (Class A and Class B). The index for Class A is 1 and the index for Class B is 2. Thus the vector

V = (1, 2). For example, from Table 3: TPR_{M1} = ((10 + 24) + (21+53)) / ((10+24+17) + (21+53+18)) = 0.76 and FNR_{M1}= ((17) + (18)) / ((10+24+17) + (21+53+18)) = 0.24.

From the results above, *TPR* and *FNR* complement each other in the confusion matrix. In the next section we will demonstrate the methodology to measure the similarity between confusion matrices for multi-class classifiers.

4 Similarity of Confusion Matrices for Multi-class Classifiers

In this section, we apply the technique proposed in [1] to compare multi-class classifiers' confusion matrices**.** Let's say we have three predictive models generated by different classifiers using the same dataset. The model M1 generates the confusion matrix M_{M1} (see Table 3), the model M2 generates the confusion matrix M_{M2}, and the model M_3 generates confusion matrix M_{M3} (see Table 4).

Table 4. Confusion Matrices for Model M2 and Model M3

Confusion matrix for model M2			
Class	A	B	C
A	$24_{(1,1)}$	$18_{(1,2)}$	$33_{(1,3)}$
B	$10_{(2,1)}$	$53_{(2,2)}$	$19_{(2,3)}$
C	$17_{(3,1)}$	$21_{(3,2)}$	$26_{(3,3)}$

Confusion matrix for model M3			
Class	A	B	C
A	$34_{(1,1)}$	$4_{(1,2)}$	$9_{(1,3)}$
B	$10_{(2,1)}$	$80_{(2,2)}$	$10_{(2,3)}$
C	$7_{(3,1)}$	$8_{(3,2)}$	$59_{(3,3)}$

Table 5 shows the performance measures *TPR*, *FNR* and *ACC* calculated using equations 1, 2 and 3. The values of performance measures were calculated by grouping the selected toxic classes A and B. Thus, V = (1, 2). From the results depicted in Table 5, model M3 is the better model compared to M1 and M2: *TPR* is the highest value and *FNR* is the lowest value for model M3.

Table 5. Performance Measures (*TPR* and *FNR*) for Models M1, M2 and M3

Models	*TPR*	*FNR*	*ACC*
M1	0.76	0.25	0.37
M2	0.73	0.27	0.47
M3	0.90	0.10	0.78

For the similarity measurement, in this example we chose *FNR* to measure the distance between the models' performances. For the performance measures in Table 5, let's use the notations $k_{1...n}$. In this case k_1 is *FNR*. The following steps illustrate the calculation of the distance between the confusion matrices between two predictive models:

Step 1: Save the selected performance measure/s in a 1-dimension (vector).

We save the selected performance measures into two rows vectors; in this case the vectors for M1 (V_{M1}) and M2 (V_{M2}) have just 1 element: V_{M1} = (0.25) and V_{M2} = (0.27).

Step 2: Calculate the distance between the vectors.

The distance between the vectors V_{M1} and V_{M2} is calculated using the Euclidean Distance. The distance O (Output) between model *M1* and model *M2* is the average of distances between the confusion matrix elements. Similarity and distance measures are complementary. In our case, the similarity of output *O (SimO)* between two models will be:

$$SimO_{(M1,M2)} = 1 - \left(\sqrt{\sum_{k=1}^{n} (V_{M1k} - V_{M2k})^2} \Big/ n \right) \quad (4)$$

where: k is the order of performance measures selected, n equals to number of k, V_{M1} is the index vector for model *M1*, and V_{M2} is the index vector for model *M2*. The value for $SimO_{(M2,M2)}$ in the example above is 0.98. Table 6 contains the values for $SimO_{(M1,M2)}$ related to the similarity of the three classifiers using *FNR*. The result shows that models M1 and M2 are 98% similar on their FNR.

Table 6. Similarity Matrix for models M_1, M_2 and M_3

	M_1	M_2	M_3
M_1	1	0.98	0.85
M_2	0.98	1	0.83
M_3	0.85	0.83	1

5 Experiments and Results

For this study, we generated collections of models using a series of classification algorithms implemented in Weka [7], such as k-nearest neighbors classifier (weka.classifiers.lazy.IBk), decision trees (weka.classifiers.trees.J48) and numerical prediction algorithms (weka.classifiers.rules.JRip). The predictive models were applied to various toxicology data sets such as Demetra [8] (Bee, Daphnia, Oral Quails, Dietary Quails and Trout) and TETRATOX [9]. Each dataset had originally more than two classes to predict the toxicity levels for each compound. Table 7 is an example of the confusion matrix. We mapped the old multi-classes onto binary classes. We want to study how the relation of different class categories will affect the performance of classifier algorithms (refer to Table 12 in [10]).

Over 1,300 predictive models were generated with different combinations of datasets, algorithms, and model parameters. The feature selection algorithm applied to the original full datasets was Correlation-based Feature Selection (CFS). We used feature selection to find sets of attributes that are highly correlated with the target classes [10, 11]. Each data set was processed using Weka with 10-fold cross validation and classifiers weka.classifiers.lazy.IBk, weka.classifiers.trees.J48, and weka.classifiers.rules.Jrip). In Table 7 the confusion matrix for a decision tree applied to the Bee dataset with 5 classes is provided.

Table 7. A confusion matrix generated using multi-class dataset with feature selection (CFS), 10-fold cross validation and using classifiers (weka.classifiers.trees.J48)

	Class1	Class2	Class3	Class4	Class5
Class1	7	4	2	3	0
Class2	4	7	4	8	2
Class3	0	2	1	4	0
Class4	2	10	4	23	4
Class5	0	0	2	4	8
Total Instances	13	23	13	42	14

Considering the fusion of Class1, Class2 and Class3 as toxic classes, the performance for a randomly chosen model M154c are as follows:

Table 8. Performance measures calculated based on the confusion matrix using table 7

Performance Measures	Results
TPRate (All Classes) and Accuracy (See Eq. 1 and 2)	0.44
Error Rate (All Classes)	0.56
FNRate (selected toxic class; 1, 2, 3, 4) (See Eq. 3)	0.07

Experiment 1:

In this experiment we want to compare the use of error rate for all classes vs. false negative rate for selected toxic classes in multi-class classifiers. We are interested in FNR because in predictive toxicology good models should have lower rate of false negatives (FN) for toxic class results. For Table 9 (ER vs. FNR results measured using the selected classes) we can find that models with similar ER Rate can exhibit a range of FNR values:

Table 9. Error Rate (ER) and FNR of multi-class classifiers applied to the DEMETRA datasets

Datasets	Toxic Classes (Low FNR)	All Classes (ER)	Toxic Classes (High FNR)	All Classes (ER)
Bee	0.04 *–M304c*	0.60 *–M304c*	0.12 *-M1c*	0.61 *-M1c*
Daphnia	0.07 *-M334c*	0.56 *-M334c*	0.20 *–M31c*	0.56 *–M31c*
Dietary Quail	0.19 *–M364c*	0.59 *–M364c*	0.25 *-M211c*	0.61 *-M211c*
Oral Quail	0.30 *-M91c*	0.60 *-M91c*	0.52 *-M244c*	0.61 *-M244c*
Trout	0.12 *M271c*	0.51 *-271c*	0.17 *-M274c*	0.52 *-M274c*

Experiment 2:

For the second experiment, we want to study if the relationship between the numbers of toxic classes will affect the performance of the classifier. In this experiment we mapped the toxic class into two categories: binary class (Toxic and Non-toxic) and multi-class (class A, class B .. class N). From the results shown in Table 10 we conclude that:

- Datasets with feature selection algorithms (such as CFS) applied are better in FNR performance measurement compared to datasets with no feature selection. Examples of such models are *M4a* and *M1a*.
- The classifiers perform best in Bee dataset and worst in Oral Quail dataset.
- Some performance (FNR) of models with selected class for more than 1 toxic class (e.g. *M4c*) is poor compared to binary model with only 1 toxic class (e.g. *M4a*), but in contrast some of the multi-class classifiers are better than binary classifiers (e.g. M34c vs. M34a and M271c vs. M271a).
- On average, models that applied binarisation strategies (models number ended with *'a'*) are better than multi-class classifiers that apply calculation of FNR to their confusion matrices (models named ending in *'c'*). This proved that multi-class classifiers for Daphnia datasets such as M334c are better than binary classifiers (e.g. M331a). For Oral Quail dataset, both binary and multi-class were having the same performance (0.30) of FNR (e.g. M91c vs. M244a).

From the results shown in Table 10, if the objective is to discriminate between two binary classes, in our case Toxic and Non-toxic, then the classifiers with binary class format have better performance compared to multi-class classifiers. But the difference between both categories is very small (between 0.02 – 0.04). For some models, regrouping classes in a single toxic class may increase the accuracy as compared to re-generating binary class classifiers.

Table 10. Results of FNR for all datasets with feature selection algorithms (CFS) and without CFS generated (None) using classifiers (IBK, J48 and JRip)

	IBK	J48	JRip
Datasets	FNR –Model_ID	FNR – Model_ID	FNR –Model_ID
Bee	0.12 – *M1a*	0.06 - *M151a*	0.06 - *M301a*
(None)	0.12 – *M1c*	0.09 - *M151c*	0.04 – *M301c*
Bee	0.04 – *M4a*	0.02 - *M154a*	0.04 - *M304a*
(CFS)	0.11 – *M4c*	0.07 - *M154c*	0.04 – *M304c*
Daphnia	0.19 – *M31a*	0.19 - *M181a*	0.10 - *M331a*
(None)	0.20 – *M31c*	0.20 - *M181c*	0.11 – *M331c*
Daphnia	0.20 – *M34a*	0.12 - *M184a*	0.12 - *M334a*
(CFS)	0.16 - *M34c*	0.14 - *M184c*	0.07 – *M334c*
Dietary Quail	0.19 - *M61a*	0.20 - *M211a*	0.23 - *M361a*
(None)	0.19 - *M61c*	0.25 - *M211c*	0.24 – *M361c*
Dietary Quail	0.15 - *M64a*	0.13 - *M214a*	0.20 - *M364a*
(CFS)	0.19 - *M64c*	0.15 - *M214c*	0.19 – *M364c*
Oral Quail	0.32 - *M91a*	0.36 - *M241a*	0.54 - *M391a*
(None)	0.30 - *M91c*	0.34 - *M241c*	0.62 – *M391c*
Oral Quail	0.37 - *M94a*	0.30 - *M244a*	0.47 - *M394a*
(CFS)	0.36 - *M94c*	0.52 - *M244c*	0.61 – *M394c*
Trout	0.14 - *M121a*	0.17 - *M271a*	0.10 - *M421a*
(None)	0.16 - *M121c*	0.12 - *M271c*	0.09 – *M421c*
Trout	0.12 - *M124a*	0.07 - *M274a*	0.05 - *M424a*
(CFS)	0.14 - *M124c*	0.17 - *M274c*	0.12 – *M424c*

Experiment 3:

In this experiment, models from Table 10 were selected to calculate their similarity. From the results in Table 11 we can see that the models have a large spread of performance value of FNR. The similarity values between confusion matrices shows that similar FNR values between models indicate similar performance among them although using different classifier algorithms. Example of such models are model M4a and model M304a, and model M31c and model M181c.

However the results only show a single element of the similarity evaluation for predictive models' performance. To have more accurate results of similarity of predictive models, the comparison of multi-class confusion matrices can be applied using our proposed methodology for calculating the similarity of binary predictive models [1].

Table 11. Similarity Matrix for Models (M4a, M304A, M151c and M154c)

Model ID	M4a	M304a	M151c	M154c
M4a	1	1	0.95	0.97
M304a	1	1	0.97	0.97
M151c	0.95	0.97	1	0.98
M154c	0.97	0.97	0,98	1

6 Conclusions

This study shows that comparing predictive models' confusion matrices will help users to choose similar models based on FNR performance measure. We studied whether there are any differences in performance measures between binarisation strategies by converting the multi-class datasets into binary classes, compared to calculating the performance measure on the fly using their confusion matrices.

From the experiments presented, regrouping multi-class classifiers' confusion matrices to binary problem is a simple solution to analyze and categorize the performance of the multi-class classifiers from a collection of models. This methodology can be integrated in ensembles of classifiers by further analysing diversity of classes of selected models.

Our experiments also show that the similarity of confusion matrices will help for further analysis and customized selection of the relevant models according to the user's needs. In future, we will integrate the methodology in a models management system and evaluate various ways to characterize and use their performances.

Acknowledgments. This work is partially supported by BBSRC, TSB and Syngenta through the Knowledge Transfer Partnerships (KTP) Grant "Data and Model Governance with Applications in Predictive Toxicology". The first author acknowledges the financial support received from the University Sultan Zainal Abidin (UniSZA), Malaysia.

References

1. Makhtar, M., Neagu, D.C., Ridley, M.J.: Binary classification models comparison: On the similarity of datasets and confusion matrix for predictive toxicology applications. In: Khuri, S., Lhotská, L., Pisanti, N. (eds.) ITBAM 2011. LNCS, vol. 6865, pp. 108–122. Springer, Heidelberg (2011)
2. Galar, M., Fernández, A., Barrenechea, E., Bustince, H., Herrera, F.: An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes. J. Pattern Recognition 44, 1761–1776 (2011)
3. Kohavi, R., Provost, F.: Glossary of Terms. Editorial for the Special Issue on Applications of Machine Learning and the Knowledge Discovery Process. J. Machine Learning 30, 271–274 (1998)
4. Fawcett, T.: ROC Graphs: Notes and Practical Considerations for Researchers. HP Laboratories, `http://www.hpl.hp.com/techreports/2003/HPL-2003-4.pdf`
5. Prasanna, S.R.M., Yegnanarayana, B., Pinto, J.P., Hermansky, H.: Analysis of Confusion Matrix to Combine Evidence for Phoneme Recognition. IDIAP Research Report, IDIAP-RR-27-2007 (2007)
6. Freitas, C.O.A., Carvalho, J.M.D., Jose Josemar Oliveira, J., Aires, S.B.K., Sabourin, R.: Confusion Matrix Disagreement for Multiple Classifiers. In: Proceedings Of The Congress On Pattern Recognition 12th Iberoamerican Conference on Progress in Pattern Recognition, Image Analysis and Applications, pp. 387–396 (2007)
7. Witten, I.H., Frank, E., Trigg, L., Hall, M., Holmes, G., Cunningham, S.J.: Weka: Practical Machine Learning Tools and Techniques with Java Implementations. In: Proceedings of the ICONIP/ANZIIS/ANNES 1999 Workshop on Emerging Knowledge Engineering and Connectionist-Based Information Systems, pp. 192–196 (1999)
8. DEMETRA Project, `http://www.demetra-tox.net/`
9. TETRATOX, `http://www.vet.utk.edu/TETRATOX/index.php`
10. Neagu, D., Guo, G.: A Data-Driven Approach for Improved Effective Classification in Predictive Toxicology. In: Proceeding of IEEE International Conference on Computational Cybernetics ICCC 2006, pp. 193–198 (2006)
11. Trundle, P.: Hybrid Intelligent Systems Applied to Predict Pesticides Toxicity - a Data Integration Approach. PhD Thesis. School of Informatics. University of Bradford, UK (2008)