South African Statistical Journal
2019, Vol. 53, No. 1, 1–13

INTERPRETABLE MULTI-LABEL CLASSIFICATION BY MEANS OF MULTIVARIATE LINEAR REGRESSION

Surette Bierman
Department of Statistics and Actuarial Science, Stellenbosch University, South Africa
e-mail: *surette@sun.ac.za*

In this paper, the potential of using a multivariate regression approach in order to obtain interpretable output in a multi-label classification problem is investigated. We focus in our analysis on extensions of ordinary multivariate regression which take into account informative dependencies amongst labels. It is found that the regression approaches make a valuable contribution insofar as the importance of input variables for given labels can be evaluated. An empirical study facilitates comparison of the performance of the regression approaches in multi-label classification and, in terms of several evaluation measures, shows that they are also largely competitive with state-of-the-art multi-label classification procedures.

Key words: Canonical shrinkage, Curds-and-whey regression, Reduced rank regression.

1. Introduction

Multi-label classification (MLC) is an extension of binary and multi-class classification to scenarios where several labels are associated simultaneously with each data instance. The training data are $\{(\boldsymbol{x}_i, \boldsymbol{y}_i), i = 1, 2, \ldots, N\}$, where $\boldsymbol{x}_i$ is a p-component vector of observations of input variables $X_1, X_2, \ldots, X_p$, and where $\boldsymbol{y}_i$ is a K-component vector of observations of binary label variables $Y_1, Y_2, \ldots, Y_K$. It is convenient to summarise the training data in an $N \times (p + K)$ matrix $[\mathbf{X}\ \mathbf{Y}]$, where typically it is assumed that each row in $\mathbf{Y}$ contains at least a single 1, the remaining entries being 0. This assumption seems reasonable since its violation implies an incomplete set of labels.

The dual objective in multi-label classification is accurate prediction of the labels corresponding to new unseen test observations and interpretation of these predictions. For this purpose, a multi-label classifier is fitted to the training data, and its performance typically evaluated using a test dataset. We denote the latter dataset by the $M \times (p + K)$ matrix $[\tilde{\mathbf{X}}\ \tilde{\mathbf{Y}}]$, consisting of observations $\{(\tilde{\boldsymbol{x}}_i, \tilde{\boldsymbol{y}}_i), i = 1, 2, \ldots, M\}$, where $\tilde{\boldsymbol{x}}_i$ denotes previously unseen observations of the inputs $X_1, X_2, \ldots, X_p$, and where $\tilde{\boldsymbol{y}}_i$ contains corresponding test observations of $Y_1, Y_2, \ldots, Y_K$. Again, it is assumed that each row in $\mathbf{Y}$ contains at least a single 1. Examples of MLC are found in direct marketing, identifying the musical instruments playing together in a performance, and in text and image annotation.

In the literature, methods to perform MLC are divided into three groups: problem transformation approaches, algorithm adaptation approaches and ensemble methods. Madjarov, Kocev, Gjorgjevikj and Džeroski (2012) provide a good overview of these different approaches. A problem transformation

MSC2010 subject classifications. 62J05.

approach transforms the single multi-label (ML) problem into several binary or multi-class classification problems. Examples of this approach include binary relevance, label-powerset and classifier chains. An algorithm adaptation approach modifies a binary or multi-class classifier, such as an SVM or k-nearest neighbours, for direct implementation in the ML context. Ensemble methods combine the classifications obtained from a number of multi-label classifiers. The construction of each ML classifier involves some random element, which causes the classifiers to predict differently. Examples of ensemble methods for MLC include random k-labelsets, introduced by Tsoumakas, Katakis and Vlahavas (2011a), random forests of predictive clustering trees (Kocev, Vens, Struyf and Džeroski, 2007), and ensembles of classifier chains in Read, Pfahringer, Holmes and Frank (2011).

Another useful distinction is between methods taking label correlations into account, and those ignoring label correlations (cf. Dembczyński, Waegeman, Cheng and Hüllermeier, 2012). Although it may seem that methods from the first category should be superior to those in the second category, there are scenarios where they perform very similarly. Examples of the former approach include label-powerset and classifier chains, while the primary example of the latter is binary relevance.

In MLC research, the focus currently falls mainly on developing new procedures that perform well in terms of at least some of the measures commonly used to evaluate multi-label classification approaches. Interpretation of the output from an MLC method usually receives little attention. A facility to assess the importance of input variables for the different labels should prove to be valuable in practical problems. Toward such a contribution, we investigate several multivariate regression approaches to multi-label classification. These approaches are evaluated in terms of two aspects. Firstly, an approach should be competitive in terms of its performance on the standard measures used in MLC. Secondly, and of primary importance in this paper, interpretable output should be obtained, especially regarding the influence of the different inputs on the label variables.

Ordinary multivariate linear regression ignores label dependencies. In order to provide for scenarios where label dependencies are informative, one of several extensions of multivariate linear regression may be used. These extensions were proposed with the aim of exploiting the information provided by the dependence structure amongst the labels. In this regard the extensions proposed by Izenman (1975), Van der Merwe and Zidek (1980), and Breiman and Friedman (1997) will be investigated. Our main objectives are to compare the performance of MLC procedures based on these extensions with that of state-of-the-art MLC methods, and to illustrate the useful information provided by a regression approach to multi-label classification.

In connection to our work, it should be noted that Breiman and Friedman (1997) contains a comparison of the three multivariate regression extensions, but not in an MLC context. Also, the 'curds-and-whey' idea has previously been used in Wu, Han, Tian and Zhuang (2010) in order to obtain a boosted multi-label image annotation method. Zhang and Schneider (2011) exploit the relationship between canonical correlation analysis and MLC in order to investigate an encoding approach to the latter. Finally, Borchani, Varando, Bielza and Larrañaga (2015) provide a comprehensive review of multi-output regression without, however, considering multi-label classification.

The remainder of the paper is organised as follows. In Section 2 the different regression approaches to be compared are described. Section 3 is devoted to a discussion of several thresholding options. Empirical results are reported in Section 4, followed by main findings and avenues for further research in Section 5.

2. Regression approaches to ML classification

Multivariate regression concerns itself with fitting a linear model to $\{(\boldsymbol{x}_i, \boldsymbol{y}_i), i = 1, 2, \ldots, N\}$, where $\boldsymbol{x}_i$ contains values of p input variables and $\boldsymbol{y}_i$ contains values of K response variables. In matrix notation the model to be fitted is

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E}, \tag{1}$$

where $\mathbf{Y} : N \times K$ contains the response observations, and $\mathbf{X} : N \times p$ contains the inputs, while $\mathbf{B} : p \times K$ has to be estimated. The matrix $\mathbf{E} : N \times K$ provides for error terms. Note that in all our analyses the data are standardised and we therefore do not provide for an intercept. Least squares, a very common way of fitting the model in (1), yields

$$\widehat{\mathbf{B}}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y},$$

where $\boldsymbol{x}_i^T$ and $\boldsymbol{y}_i^T$ are the ith rows in $\mathbf{X}$ and $\mathbf{Y}$ respectively. It is easy to see that least squares fitting partitions the problem into K separate multiple linear regression problems, treating the K responses completely separately.

It is clear that the above approach suffers from a disadvantage: in each separate regression, the information possibly contained in the other responses is not utilised. Toward alleviating this disadvantage, the three regression approaches mentioned above, are based on the following basic underlying idea. The data are transformed to canonical coordinates, shrinkage is applied to the (unimportant) coordinates corresponding to small canonical correlations, and the results are then transformed back to the original coordinates. The only difference between the regression approaches is the manner in which shrinkage is accomplished. We proceed with a brief description of each regression method. More detailed information may be found in the respective papers.

Transforming the data to canonical coordinates entails the following. The first pair of canonical coordinates is the linear combinations $\sum_{k=1}^{K} a_k Y_k$ and $\sum_{j=1}^{p} b_j X_j$, maximising the correlation between any two such linear combinations. Further coordinates, up to a maximum of $q = \min(p, K)$ are defined similarly, subject to orthogonality restrictions. Let $\mathbf{A}$ be the $q \times q$ matrix with columns defining the response canonical coordinates. All three multivariate shrinkage procedures estimate $\mathbf{B}$ by a matrix of the form

$$\widehat{\mathbf{B}}_{OLS} = \mathbf{ADA}^{-1}.$$

The three methods differ only with respect to the diagonal matrix $\mathbf{D}$. In Izenman (1975) the diagonal entries in $\mathbf{D}$ are given by

$$d_i = \begin{cases} 1 & \text{if } i \leq m, \\ 0 & \text{otherwise,} \end{cases}$$

where m is a tuning parameter of the procedure which should be determined in a data-dependent manner. This procedure is called reduced rank (RR) multivariate regression. In Van der Merwe and Zidek (1980), the diagonal entries are given by

$$d_i = \begin{cases} \dfrac{c_i^2 - (p - K - 1)/N}{c_i^2\,(1 - (p - K - 1)/N)}, \end{cases}$$

where $c_1^2 \geq c_2^2 \geq \ldots \geq c_q^2$ denote the squared canonical correlations. This procedure is called FICYREG (filtered canonical Y-variate regression). Finally, Breiman and Friedman (1997) propose the use of

$$d_i = \frac{(1-r)(c_i^2 - r)}{(1-r)^2 c_i^2 + r^2(1-c_i^2)},$$

where $r = p/N$. This procedure is called curds-and-whey (CW) multivariate regression.

3. Thresholding the regression output

Several approaches for fitting a multivariate regression model to multi-label data were discussed in Section 2. In each case, an estimate $\widehat{\mathbf{B}}$ of the parameter matrix $\mathbf{B}$ in (1) is obtained. An important objective of the analysis is to predict the label vector $\boldsymbol{y}(\boldsymbol{x})$ corresponding to an input vector $\boldsymbol{x}$. This may be achieved by thresholding the values in $\widehat{f}(\boldsymbol{x}) = \widehat{\mathbf{B}}^T\boldsymbol{x}$, which will be real numbers not restricted to $\{0, 1\}$. Consider therefore the problem of determining $\boldsymbol{y}(\boldsymbol{x})$, a vector consisting of zeros and ones, from $\widehat{f}(\boldsymbol{x})$. Different approaches for this purpose were investigated. Let $t_1, t_2, ..., t_K$ denote values such that

$$\widehat{y}_k(\boldsymbol{x}) = I\left[\widehat{f}_k(\boldsymbol{x}) > t_k\right],$$

$k = 1, 2, \ldots, K$, where $I(.)$ denotes the indicator function. A naïve fixed threshold approach may take $t_k = 0.5$ for $k = 1, 2, \ldots, K$. This would be based upon the interpretation of $\widehat{f}_k(\boldsymbol{x})$ as an approximation of the posterior probability $P(Y_k = 1|\mathbf{X} = \boldsymbol{x})$. However, two questions arise: should the same threshold be used for all labels, and how appropriate is the value 0.5, given that $\widehat{f}_k(\boldsymbol{x})$ is not restricted to the interval $[0, 1]$?

Given these reservations regarding the use of a fixed threshold, as well as empirical evidence when using $t_k = 0.5$, it is clear that other approaches should be investigated. We considered three approaches for data-dependent specification of label-specific thresholds. A description of these approaches requires further notation. We write $\widehat{\mathbf{F}} = \mathbf{X}\widehat{\mathbf{B}}$ for the $N \times K$ matrix of values which we have to threshold to label the training data cases. Recall that the matrix containing the inputs corresponding to the test data cases, unseen during the training phase, is denoted by $\tilde{\mathbf{X}}$, an $M \times p$ matrix. In turn, the $M \times K$ matrix $\widehat{\mathbf{G}} = \tilde{\mathbf{X}}\widehat{\mathbf{B}}$ contains the values which we have to threshold to label the test cases.

The first approach for determining label-specific thresholds, called the quantile approach, is based on the assumption that the training and test data are generated by the same mechanism. It would then seem sensible to threshold in such a way that the proportions of test cases for which $\widehat{Y}_k = 1$, $k = 1, 2, ..., K$, are as close as possible to these proportions in the training data. Recall that $\mathbf{Y} = [y_{ik}]$ is the matrix of training data labels. For $k = 1, 2, ..., K$, let $p_k = N^{-1}\sum_{i=1}^N y_{ik}$ denote the proportion of training data cases having $Y_k = 1$, and write $\widehat{g}_{(1)k} < \widehat{g}_{(2)k} < ... < \widehat{g}_{(M)k}$ for the ordered values in the kth column of $\widehat{\mathbf{G}}$. Then the quantile approach specifies $\widehat{t}_k = \widehat{g}_{(r_k),k}$, where $r_k = [M(1-p_k)]$, $k = 1, 2, ..., K$.

A second approach for determining label-specific thresholds is to determine $t_1, t_2, ..., t_K$ such that an estimate of some performance measure on the training data is optimised. Consider for example label k and the Hamming Loss performance measure (see Section 4.2 for definitions of performance measures). Denote by $\{t_{kj}, j = 1, 2, ...h\}$, a set of h candidate threshold values, fixed beforehand, and distributed evenly between the smallest and largest values in the kth column of $\widehat{\mathbf{F}}$. The thresholds based on Hamming Loss optimisation are the values

$$\widehat{t}_k = \underset{\{t=t_{kj}, j=1,2,\ldots,h\}}{\arg\min} \left(\frac{1}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} I(y_{ik} \neq \widehat{y}_{ik}) \right),$$

$k = 1, 2, \ldots, K$. Here $\widehat{y}_{ik}(t) = I(\widehat{f}_{ik} > t)$, where $\widehat{f}_{ik}$ denotes the ith element in $\widehat{f}_k(\boldsymbol{x})$. Note also that in our empirical work, the value of h was set to

$$h = \begin{cases} N-1 & \text{if } NK < 1500, \\ 150 & \text{otherwise.} \end{cases}$$

In addition to Hamming Loss, two other performance measures were also used to determine label-specific thresholds in this way, viz. the F-score and accuracy.

To motivate a third data-dependent method for determining thresholds, we argue that the former approach will break down in cases where there is a large difference between the range of the $\widehat{f}$-values (computed on the training data) and the range of the $\widehat{g}$-values (computed on the test data). In such cases it seems sensible to find the quantile corresponding to $\widehat{t}_k$ (determined by optimising a performance measure on the training data), and to set as threshold the same quantile on the test data. In more detail, let q_k denote the proportion of training data cases having $\widehat{f}_{ik}$-values less than $\widehat{t}_k$, i.e. $q_k = N^{-1} \sum_{i=1}^{N} I(\widehat{f}_{ik} < \widehat{t}_k)$. We use as thresholds the values $\widehat{t}_k = \widehat{g}_{(s_k)k}$, where $s_k = [Mq_k]$, $k = 1, 2, \ldots K$.

It should be noted that using any of the above thresholding methods may lead to predicting all of the labels of some test case i' to be zero. This occurs when none of the K thresholds are exceeded by the values in row i' of the $\widehat{\mathbf{G}}$-matrix. In such a case, note that we entered a 1 in position $\widehat{k} = \arg\max_k \widehat{g}_{i'k}$ in the label vector. This is in line with our assumption of at least one label being present in both the training and test datasets.

Each of the three methods of thresholding introduced above may be combined with any one of the regression approaches discussed in Section 2. We will use the following notation to denote the resulting procedures: CW_j will represent the curds-and-whey procedure using the jth threshold method, $j = 1, 2, 3$, with similar interpretations for FICY, OLS and RR.

4. Data experiments

We evaluated the use of CW, FICY, OLS and RR regression for multi-label classification by applying these techniques to seven datasets from the Mulan MLC library (Tsoumakas, Spyromitros-Xioufis, Vilcek and Vlahavas, 2011b). These datasets have become well known in the multi-label learning literature, serving as benchmark datasets in most empirical work, and were also analysed by Madjarov et al. (2012) in an extensive empirical investigation of MLC procedures. A few characteristics of the datasets render them suitable for comparative studies: they originate from a broad spectrum of application fields, have wide-ranging sizes, differing not only with respect to types of input variables, but also regarding both the number of labels and the average number of labels per data case (cardinality).

We compare the regression approaches to three of the approaches investigated by Madjarov et al. (2012), viz. binary relevance (BR), hierarchy of multi-label classifiers (HR), and random forests for

Table 1. Benchmark datasets and their attributes.

Name	**Domain**	$N + M$	**Nominal**	**Numeric**	K	**Card**	**Dens**
Bibtex	Text	7395	1836	0	159	2.402	0.015
Corel5k	Images	5000	499	0	374	3.522	0.009
Emotions	Music	593	0	72	6	1.869	0.311
Enron	Text	1702	1001	0	53	3.378	0.064
Mediamill	Video	43 907	0	120	101	4.376	0.043
Scene	Images	2407	0	294	6	1.074	0.179
Yeast	Biology	2417	0	103	14	4.237	0.303

predictive clustering trees (RF). Note that we include BR since it is one of the most widely used ML classification approaches, while HR and RF are included since in their study, the above authors consider these procedures to be the overall best.

4.1 Benchmark datasets

Properties of the benchmark datasets are summarised in Table 1. Some remarks on the entries in the table are in order. Two of the datasets are from the field of text annotation, two from image annotation, and one each from the fields of music, video annotation and biology. Regarding the input variables, for three of the seven datasets these are all nominal (binary), while in the other four cases, all are numeric. The number of labels, K, is also given in Table 1 for each dataset, as are the cardinality, and the density (cardinality divided by K).

4.2 Performance measures

Several measures have been proposed in the literature to evaluate the performance of ML classification procedures (for a detailed exposition, see Tsoumakas, Katakis and Vlahavas, 2010). This complicates comparison of different methods, since often an approach that outperforms others in terms of a specific measure may perform worse in terms of another measure. In this paper we restrict attention to the Hamming Loss (HL), F-score (F) and accuracy (A) measures. For a definition of each measure, consider the following notation. As before, suppose $\tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2, \ldots, \tilde{\boldsymbol{y}}_M$ are the true labels of a test data set. Also, let $\widehat{\boldsymbol{y}}_1, \widehat{\boldsymbol{y}}_2, \ldots, \widehat{\boldsymbol{y}}_M$ be the set of corresponding predicted label vectors obtained from an ML classification procedure. Then the Hamming Loss for these predicted label vectors is defined by

$$HL = \frac{1}{MK} \sum_{i=1}^{M} \sum_{k=1}^{K} I(\tilde{y}_{ik} \neq \widehat{y}_{ik}),$$

i.e. the average proportion of misclassifications, computed over test cases and labels. The precision is defined by

$$precision = \frac{1}{M} \sum_{i=1}^{M} \left\{ \frac{\sum_{k=1}^{K} \tilde{y}_{ik} \widehat{y}_{ik}}{\sum_{k=1}^{K} \widehat{y}_{ik}} \right\},$$

i.e. the average proportion of predicted labels which are correct. Similarly,

$$recall = \frac{1}{M}\sum_{i=1}^{M}\left\{\frac{\sum_{k=1}^{K}\tilde{y}_{ik}\widehat{y}_{ik}}{\sum_{k=1}^{K}\tilde{y}_{ik}}\right\},$$

which is the average proportion of true labels predicted as such. Precision and recall are conflicting measures in the sense that modifying a procedure to improve precision leads to a deterioration in recall, and vice versa. A quantity that accounts for both precision and recall is the F-score, defined by

$$F = \frac{2 \times precision \times recall}{precision + recall},$$

which is simply the harmonic mean of precision and recall. Finally with regard to the measures in this paper, accuracy is defined by

$$A = \frac{1}{M}\sum_{i=1}^{M}\left\{\frac{\sum_{k=1}^{K}\tilde{y}_{ik}\widehat{y}_{ik}}{K - \sum_{k=1}^{K}(1-\tilde{y}_{ik})(1-\widehat{y}_{ik})}\right\}.$$

This measure is also known as the (average) Jaccard similarity coefficient (cf. Gouk, Pfahringer and Cree, 2016).

4.3 Pre-processing

In order to apply the multivariate regression approaches, a few preliminary processing steps on the Bibtex, Corel5k, Enron and Mediamill datasets were first required. We distinguish between pre-processing steps required to deal with problems stemming from the observed variables, and pre-processing steps required to handle problems that occurred as a result of the observed labels. All pre-processing steps on the training data were duplicated on the test data.

In terms of the observed variables, note that some of the pairs of nominal variables in the Bibtex and Enron training datasets had correlations equal to 1, causing $\mathbf{X}^T\mathbf{X}$ to be singular. Specifically, in the Bibtex training data, variables 274 and 1118 were perfectly correlated, as well as variables 952 and 1824. In the case of the Bibtex data, we therefore omitted variables 274 and 952. Furthermore, in the Enron data, variables 37 and 52 were perfectly correlated, causing variable 37 to be omitted.

With regard to the observed labels, three kinds of pre-processing steps were required. The first step involved omission of all label columns in the training data that had only 0 or 1 entries. This seems sensible since such a label would be independent of the input variables. Hence we omitted labels 167, 325 and 330 from the Corel5k dataset, and label 46 from the Enron data. The second step involved omission of redundant labels in the training data that were perfectly correlated. These were labels 262, 350 and 366 in the Corel5k dataset. In the third pre-processing step in terms of the observed labels, we omitted rows in the training and test datasets having only 0 entries, which is in line with the assumption stated in Section 1. This was only required in the case of the Mediamill data, where we discarded 1189 out of 30993 training observations, and 541 out of 12914 test observations.

4.4 Results

In this section we compare the following procedures in terms of Hamming Loss, the F-score and accuracy: CW1-3, FICY1-3, OLS1-3, RR1-3, BR, HR and RF. For each of the datasets, each of

Table 2. Hamming Loss values for each multi-label classification procedure.

Appr	Bibtex	Corel5k	Emotions	Enron	Mediamill	Scene	Yeast	Ave1	Ave2
CW1	0.0189	0.0154	0.2162	0.0904	0.0401	0.1180	0.2482	0.1067	0.1095
CW2	0.0165	0.0119	0.2137	0.2765	0.0317	0.1399	0.2045	0.1278	0.1030
CW3	0.0173	0.0126	0.2120	0.0891	0.0320	0.1392	0.2068	0.1013	0.1033
FICY1	0.0185	0.0156	0.2252	0.0890	0.0401	0.1194	0.2476	0.1079	0.1111
FICY2	0.0171	0.0122	0.2145	0.3025	0.0316	0.1385	0.2047	0.1316	0.1031
FICY3	0.0176	0.0128	0.2195	0.0887	0.0320	0.1405	0.2059	0.1024	0.1047
OLS1	0.0187	0.0157	0.2459	0.0886	0.0400	0.1217	0.2471	0.1111	0.1149
OLS2	0.0205	0.0126	0.2228	0.3365	0.0316	0.1444	0.2073	0.1394	0.1065
OLS3	0.0182	0.0131	0.2269	0.0895	0.0320	0.1434	0.2084	0.1045	0.1070
RR1	0.0191	0.0156	0.2384	0.0899	0.0401	0.1665	0.2486	0.1169	0.1214
RR2	0.0176	0.0124	0.2178	0.3025	0.0316	0.1604	0.2060	0.1355	0.1076
RR3	0.0176	0.013	0.2211	0.0900	0.0319	0.1575	0.2049	0.1051	0.1077
BR	**0.012**	0.017	0.257	**0.045**	0.032	**0.079**	**0.190**	0.0903	0.0978
HR	0.014	0.012	0.361	0.051	0.038	0.082	0.207	0.1093	0.1190
RF	0.013	**0.009**	**0.189**	0.046	**0.029**	0.094	0.197	**0.0824**	**0.0885**

the procedures was implemented on the training data, and thereafter applied to the test data. The results are summarised in Tables 2–5. Performances of the methods which were found to be the best over the seven datasets considered, are denoted in bold, whereas best performances amongst the regression approaches are underlined. Note that two columns have been added to each table. The penultimate column (Ave1) provides the average measure obtained over all of the datasets, whereas the final column (Ave2) contains the averages having omitted the Enron data. We include the latter because for this dataset, the regression procedures perform out of line relative to their performance on the other datasets.

Consider first the performances in terms of their Hamming Loss values. These are reported in Table 2. In terms of the average Hamming Loss values for the regression procedures, it is clear that for every threshold method, CW performs best, followed by FICY, OLS and RR. In terms of performances on the individual datasets, CW is best in five of the seven cases. The relatively poor performance of RR may in part be attributable to the fact that we did not attempt to optimise the performance of this approach with respect to its tuning parameter m. The picture is more complex regarding the relative merits of the three thresholding methods. This is because of the exceptionally bad performance of the second thresholding method on the Enron data. For all four of the regression procedures, if the Enron data is included in a comparison, the third threshold method is best, followed by the first and second methods. If Enron is omitted, the second method performs best in all cases. Clearly the Enron dataset is an example of what we had in mind in Section 3 when introducing the third threshold method. Overall we recommend the third method since it does not exhibit this undesirable behaviour. We next consider all the procedures, including BR, HR and RF. In this case RF is best, followed by BR and CW3.

We proceed by considering the F-scores in Table 3. With three exceptions (the Bibtex, Enron and Yeast datasets), we see that once again CW performs best, followed by FICY, OLS and RR. We also note that in the case of the Bibtex, Emotions and Mediamill data, one of the regression

Table 3. F scores for each multi-label classification procedure.

Appr	**Bibtex**	**Corel5k**	**Emotions**	**Enron**	**Mediamill**	**Scene**	**Yeast**	**Ave1**	**Ave2**
CW1	0.4206	0.2406	0.6607	0.3472	**0.6022**	0.7308	0.6028	0.5150	0.5430
CW2	0.4416	0.2606	**0.6859**	0.2661	0.5896	0.6685	0.6395	0.5074	0.5476
CW3	0.4301	0.2487	0.6850	0.3546	0.5880	0.6645	0.6385	0.5156	0.5425
FICY1	0.4354	0.2269	0.6436	0.3633	0.6019	0.7264	0.6023	0.5143	0.5394
FICY2	**0.4492**	0.2421	0.6737	0.2605	0.5922	0.6667	0.6414	0.5037	0.5442
FICY3	0.4363	0.2575	0.6675	0.3642	0.5904	0.6637	0.6408	0.5172	0.5427
OLS1	0.4390	0.2212	0.6103	0.3646	0.6020	0.7155	0.5984	0.5073	0.5311
OLS2	0.4409	0.2293	0.6521	0.2485	0.5945	0.6498	0.6418	0.4938	0.5347
OLS3	0.4381	0.2523	0.6509	0.3662	0.5925	0.6456	0.6443	0.5128	0.5373
RR1	0.4166	0.2263	0.6134	0.3529	0.6018	0.6356	0.6027	0.4928	0.5161
RR2	0.4326	0.2304	0.6770	0.2553	0.5885	0.5776	0.6367	0.4854	0.5238
RR3	0.4236	0.2527	0.6658	0.3518	0.5864	0.5795	0.6390	0.4998	0.5245
BR	0.433	0.047	0.469	0.582	0.557	0.714	0.650	0.4931	0.4783
HR	0.426	**0.280**	0.614	**0.613**	0.579	**0.745**	**0.687**	**0.5634**	**0.5552**
RF	0.212	0.014	0.611	0.552	0.589	0.553	0.614	0.4493	0.4322

approaches yields the best performance overall. Regarding the merits of the threshold approaches, the conclusions are largely unchanged from those for Hamming Loss. With Enron included, the third method is best, while the second method improves if Enron is omitted. Taking into account also BR, HR and RF, we see that HR performs best, followed by FICY3 and CW3. It should be noted that RF and BR, which performed well in terms of Hamming Loss, now perform quite poorly.

The accuracies of the regression approaches are presented in Table 4. The pattern is largely the same as previously. Amongst the regression approaches, CW does best, followed by FICY, OLS and RR. With regard to the threshold methods, the first approach is best for three of the four regression methods if the Enron dataset is included. If Enron is excluded, no clear pattern emerges, although the second approach is now again competitive. In terms of all the procedures, HR performs the best, followed by BR and CW1.

In summary thus far, it is evident that the CW and FICY approaches (and to a lesser extent OLS and RR) are quite competitive. For example, CW finishes second or third overall in terms of all three performance measures, followed closely by FICY. Although RF does very well in terms of Hamming Loss, it performs poorly when judged by the F-score and accuracy. Similarly, HR does well as measured by the F-score and accuracy, but not as well in terms of Hamming Loss. Regarding a threshold method, the third approach is recommended.

We extend our comparison by reporting average values of the measures $1 - HL$, F and A over the seven datasets considered. These combined performance measures are provided in Table 5. From this table it is once again clear that especially the CW and FICY approaches are very competitive when compared to the best performing approaches in Madjarov et al. (2012). In fact, in terms of the average combined performance measures, even the OLS and RR approaches compare very well with the second and third best approaches overall. It is only HR that stands out from the rest.

Table 4. Accuracy values for each multi-label classification procedure.

Appr	Bibtex	Corel5k	Emotions	Enron	Mediamill	Scene	Yeast	Ave1	Ave2
CW1	0.3087	0.1479	0.5499	0.2133	0.4277	0.6679	0.4589	0.3963	0.4268
CW2	0.3221	0.1631	0.5635	0.1539	0.4146	0.6003	0.4850	0.3861	0.4248
CW3	0.3128	0.1542	**0.5697**	0.2184	0.4130	0.5991	0.4844	0.3931	0.4222
FICY1	0.3207	0.1410	0.5334	0.2204	0.4278	0.6644	0.4606	0.3955	0.4247
FICY2	0.3321	0.1512	0.5639	0.1472	0.4176	0.5956	0.4890	0.3852	0.4249
FICY3	0.3214	0.1586	0.5540	0.2218	0.4161	0.5980	0.4874	0.3939	0.4226
OLS1	0.3213	0.1372	0.4913	0.2245	0.4277	0.6572	0.4579	0.3882	0.4154
OLS2	0.3111	0.1427	0.5305	0.1380	0.4195	0.5871	0.4931	0.3746	0.4140
OLS3	0.3194	0.1550	0.5330	0.2252	0.4178	0.5890	0.4947	0.3906	0.4182
RR1	0.3043	0.1402	0.5050	0.2210	0.4276	0.5598	0.4581	0.3737	0.3992
RR2	0.3098	0.1435	0.5638	0.1460	0.4136	0.4684	0.4813	0.3609	0.3967
RR3	0.3069	0.1565	0.5517	0.2212	0.4118	0.4608	0.4839	0.3704	0.3953
BR	**0.348**	0.030	0.361	0.446	0.403	0.689	0.520	0.3996	0.3918
HR	0.330	**0.179**	0.471	**0.478**	0.413	**0.717**	**0.559**	**0.4496**	**0.4448**
RF	0.166	0.009	0.519	0.416	**0.441**	0.541	0.478	0.3671	0.3590

4.5 Estimated regression coefficients

In the previous section it was shown that in terms of prediction accuracy there is worth in a multivariate regression approach to ML classification. Perhaps a more important contribution of regression in this context lies in the estimated regression coefficients which form part of the regression output. The estimated coefficients contain information that facilitates interpretation of the corresponding ML classifier. In this section we provide illustrations of this aspect.

We consider the Emotions dataset. Since standard errors of the regression coefficient estimates can easily be obtained for OLS, our focus in this example is interpretation of the (standardised) OLS regression coefficients. By also obtaining standard errors of regression coefficient estimates also in the case of CW, FICY and RR, similar interpretations are also possible in their case. We proceed with a summary of the absolute standardised OLS coefficients in the form of a heatmap in Figure 1.

The most prominent feature in Figure 1 is the relatively small number of large values, all grouped together at the bottom of the heatmap. It is clear that the same small set of variables is significant for all of the labels. Table 6, showing the inputs with absolute standardised coefficients exceeding 2, provides a summary in this regard.

Interpretation of the variables in Table 6 is interesting. Variables 4, 20 and 36 are all related to the first mel-frequency cepstral coefficient, which is well known to play an important role in sound digitisation. Similarly, variable 5 is related to the second cepstral coefficient, while variables 66 and 68 are related to low and high peak beats per minute. Furthermore, only variable 4 is found to be significant for all of the labels. In contrast, the number of labels for which variables 68, 66, 5, 36 and 20 are significant, decreases from 5 labels (in the case of variable 68) to only a single label (in the case of variable 20). Information such as this can be useful when addressing the problem in ML scenarios of a variable being significant for some labels, but not for others.

Table 5. Combined performance measures for each multi-label classification procedure.

Appr	Bibtex	Corel5k	Emotions	Enron	Mediamill	Scene	Yeast	Ave1	Ave2
CW1	0.5701	0.4577	0.6648	0.4900	0.6633	0.7602	0.6045	0.6015	0.6201
CW2	0.5824	0.4706	0.6786	0.3812	0.6575	0.7096	0.6400	0.5886	0.6231
CW3	0.5752	0.4634	**0.6809**	0.4946	0.6563	0.7081	0.6387	0.6025	0.6205
FICY1	0.5792	0.4508	0.6506	0.4982	0.6632	0.7571	0.6051	0.6006	0.6177
FICY2	0.5881	0.4604	0.6744	0.3684	0.6594	0.7079	0.6419	0.5858	0.6220
FICY3	0.5800	0.4678	0.6673	0.4991	0.6582	0.7071	0.6408	0.6029	0.6202
OLS1	0.5805	0.4476	0.6186	0.5002	0.6632	0.7503	0.6031	0.5948	0.6106
OLS2	0.5772	0.4531	0.6533	0.3500	0.6608	0.6975	0.6425	0.5763	0.6141
OLS3	0.5798	0.4647	0.6523	0.5006	0.6594	0.6971	0.6435	0.5996	0.6161
RR1	0.5673	0.4503	0.6267	0.4947	0.6631	0.6763	0.6041	0.5832	0.5980
RR2	0.5749	0.4538	0.6743	0.3663	0.6568	0.6285	0.6373	0.5703	0.6043
RR3	0.5710	0.4654	0.6655	0.4943	0.6554	0.6276	0.6393	0.5884	0.6040
BR	**0.5897**	0.3533	0.5243	0.6610	0.6427	0.7747	0.6600	0.6008	0.5908
HR	0.5807	**0.4823**	0.5747	**0.6800**	0.6513	**0.7933**	**0.6797**	**0.6346**	**0.6270**
RF	0.4550	0.3380	0.6470	0.6407	**0.6670**	0.6667	0.6317	0.5780	0.5676

Table 6. Significant input variables in the Emotions data.

Inputs	Labels
4	1, 2, 3, 4, 5, 6
5	3, 4, 5, 6
20	2
36	1, 3, 4
66	1, 3, 4, 6
68	1, 2, 3, 4, 6

5. Conclusions and further research

In this paper we introduced the use of several multivariate regression approaches to ML classification. Three of the four approaches considered, exploit information pertaining to dependence amongst the labels. In an empirical study, the regression procedures were compared with current state-of-the-art ML classification procedures. Two of the regression approaches were found to perform competitively in terms of three measures of ML prediction accuracy. Applying regression for classification requires thresholding. We investigated three approaches in this regard, the last of which, a new proposal, was found to perform particularly well. A data example highlighted the advantage of the regression approaches in terms of interpreting the relative importance of the input variables. Overall, we deem a multivariate regression approach to be a serious contender in ML scenarios. Several avenues for further research may be pursued. Regarding the procedures proposed in this paper, replacing the 0-1 response values by appropriate numerical values should be investigated. Furthermore, there are

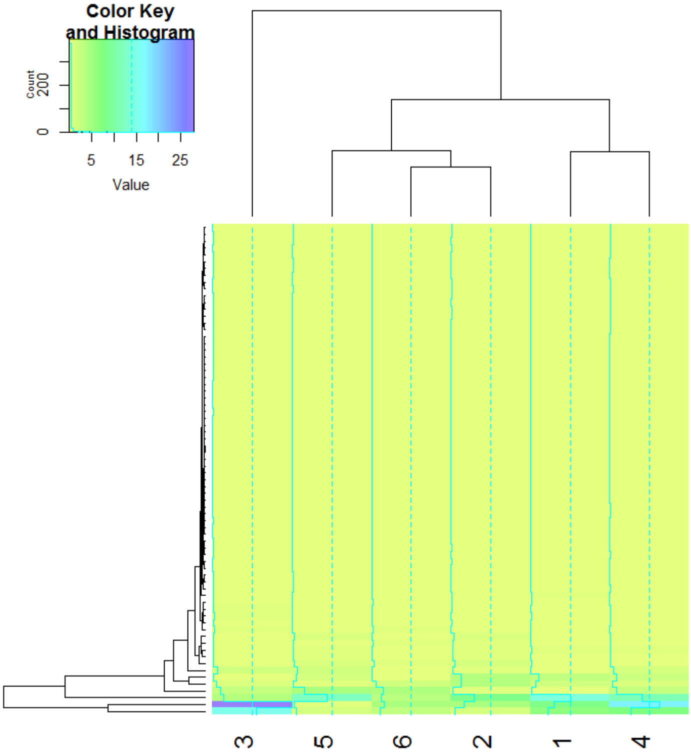

Figure 1. Heatmap of absolute standardised OLS regression coefficients for the Emotions data.

other multivariate regression approaches incorporating different forms of shrinkage which may be useful in ML problems. Finally, extending the interpretation of standardised regression coefficients to formally address the problem of variable selection in MLC, should be a worthwhile enterprise.

Acknowledgements. The author would like to sincerely thank Prof. Sarel Steel, for his invaluable contributions to the research, and the referee, for helpful suggestions that led to improvements of the paper.

References

Borchani, H., Varando, G., Bielza, C., and Larrañaga, P. (2015). A survey on multi-output regression. *WIREs Data Mining and Knowledge Discovery*, **5**, 216–233.

Breiman, L. and Friedman, J. H. (1997). Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society B*, **59**, 3–54.

Dembczyński, K., Waegeman, W., Cheng, W., and Hüllermeier, E. (2012). On label dependence and loss minimization in multi-label classification. *Machine Learning*, **88**, 5–45.

Gouk, H., Pfahringer, B., and Cree, M. J. (2016). Learning distance metrics for multi-label classification. *In Proceedings of the 8th Asian Conference on Machine Learning*, volume 63. 318–333.

Izenman, A. J. (1975). Reduced-rank regression for the multivariate linear model. *Journal of Multivariate Analysis*, **5**, 248–264.

Kocev, D., Vens, C., Struyf, J., and Džeroski, S. (2007). Ensembles of multi-objective decision trees. *In European Conference on Machine Learning*, volume 4701. 624–631.

Madjarov, G., Kocev, D., Gjorgjevikj, D., and Džeroski, S. (2012). An extensive experimental comparison of methods for multi-label learning. *Pattern Recognition*, **45**, 3084–3104.

Read, J., Pfahringer, B., Holmes, G., and Frank, E. (2011). Classifier chains for multi-label classification. *Machine Learning*, **85**, 333–359.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2010). Mining multi-label data. *Data Mining and Knowledge Discovery Handbook, 2nd ed.*, 667–685.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2011a). Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, **23**, 1079–1089.

Tsoumakas, G., Spyromitros-Xioufis, E., Vilcek, J., and Vlahavas, I. (2011b). Mulan: A Java library for multi-label learning. *Journal of Machine Learning Research*, **12**, 2411–2414.

Van der Merwe, A. and Zidek, J. V. (1980). Multivariate regression analysis and canonical variates. *The Canadian Journal of Statistics*, **8**, 27–39.

Wu, F., Han, Y., Tian, Q., and Zhuang, Y. (2010). Multi-label boosting for image annotation by structural grouping sparsity. *In Proceedings of the 18th ACM International Conference on Multimedia*, 15–24.

Zhang, Y. and Schneider, J. (2011). Multi-label output codes using canonical correlation analysis. *In Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, volume 15. 873–882.

Manuscript received 2018-09-05, revised 2019-02-10, accepted 2019-02-13.