A Systematic Review of Multi-Label Feature Selection and a New Method Based on Label Construction

Newton Spolaôr[a,b,*], Maria Carolina Monard[a], Grigorios Tsoumakas[c], Huei Diana Lee[b]

[a] *Laboratory of Computational Intelligence, Institute of Mathematics and Computer Science, University of São Paulo, Brazil. ZIP code: 13560-970. Tel.: +55 16 3373-9646. Fax: +55 16 3373-9751*

[b] *Laboratory of Bioinformatics, State West Paraná University, Foz do Iguaçu, Brazil. ZIP code: 85867-900. Tel.: +55 45 3576-8815. Fax: +55 45 3575-2733*

[c] *Machine Learning and Knowledge Discovery Group, Department of Informatics, Aristotle University of Thessaloniki, Greece. ZIP code: 54124. Tel.: +30 23 1099-8145*

Abstract

Each example in a multi-label dataset is associated with multiple labels, which are often correlated. Learning from this data can be improved when dimensionality reduction tasks, such as feature selection, are applied. The standard approach for multi-label feature selection transforms the multi-label dataset into single-label datasets before using traditional feature selection algorithms. However, this approach often ignores label dependence. In this work, we propose an alternative method, LCFS, that constructs new labels based on relations between the original labels. By doing so, the label set from the data is augmented with second-order information before applying

*Corresponding author

Email addresses: newtonspolaor@gmail.com (Newton Spolaôr), mcmonard@icmc.usp.br (Maria Carolina Monard), greg@csd.auth.gr (Grigorios Tsoumakas), hueidianalee@gmail.com (Huei Diana Lee)

the standard approach. To assess LCFS, an experimental evaluation using Information Gain as a measure to estimate the importance of features was carried out on 10 benchmark multi-label datasets. This evaluation compared four LCFS settings with the standard approach, using random feature selection as a reference. For each dataset, the performance of a feature selection method is estimated by the quality of the classifiers built from the data described by the features selected by the method. The results show that a simple LCFS setting gave rise to classifiers similar to, or better than, the ones built using the standard approach. Furthermore, this work also pioneers the use of the systematic review method to survey the related work on multi-label feature selection. The summary of the 99 papers found promotes the idea that exploring label dependence during feature selection can lead to good results.

Keywords: feature ranking, filter feature selection, binary relevance, information gain, systematic review

---

1. Introduction

In multi-label learning, each example is associated with multiple labels simultaneously. A key difference between multi-label and traditional binary or multi-class single-label learning is that the labels in multi-label learning are not mutually exclusive. Thus, in comparison with traditional single-label learning, multi-label learning is more general and challenging to solve. The issue of learning from multi-label data has attracted significant attention from the community, motivated by an increasing number of new applications in bioinformatics [1, 2], emotion analysis [3], text mining [4, 5] and image

analysis [6], among others.

As other machine learning tasks, multi-label learning also suffers from the "*curse of dimensionality*". Dimensionality reduction (feature selection), which aims to find a small subset of features that describes the dataset as well as, or even better than, the original set of features does [7], is an effective way to mitigate the curse of dimensionality.

The standard approach for multi-label Feature Selection (FS), which transforms the multi-label dataset into single-label datasets before using traditional FS algorithms, is implementable in the Binary Relevance (BR) approach [8]. However, a BR drawback is that label dependence is often ignored. Thus, a significant challenge regarding this approach is how to explore the labels structure to improve multi-label learning performance simultaneously with dimensionality reduction.

An alternative to overcoming this problem would be to construct labels based on relations among the original labels and include the new labels during the feature selection phase. The main idea of variable (label or feature) construction is to gather information about the relations among the original variables from data and infer additional variables [9]. Although feature construction methods are less usual than feature selection methods [10], they have already been used to support single-label [11, 12] and multi-label learning [13, 14, 15]. Nevertheless, to the best of our knowledge, there is little research on *label construction* for multi-label data.

In this work, we present the Label Construction for Feature Selection ($LCFS$) method, originally proposed in [16], to build binary variables (new labels) based on label relations. These variables are then included as new

labels in the original dataset and the standard multi-label FS approach based on *BR* is used in the augmented dataset to select features. Afterwards, the dataset described by the selected features and the original labels can be submitted to any multi-label learning algorithm.

The *LCFS* method was experimentally compared with the standard multi-label FS approach based on *BR* on 10 benchmark datasets. We also used Random Feature Selection (*RFS*) as a reference. Both *LCFS* and the standard approach consider the frequently used measure Information Gain (*IG*) to evaluate features.

The experimental results suggest that setting *LCFS* with simple strategies to build binary variables (new labels) from pairs of labels gives rise to classifiers similar to, or better than, the ones built using the standard approach based on *BR*. Good *LCFS* results are also observed when the number of features selected is small.

Furthermore, we also applied the Systematic Review (SR) method [17] to survey the literature on multi-label FS. The summary of the 99 papers found shows that good results are obtained from research which takes into account label dependence. Another finding is that *IG* is the most frequently used importance measure, as can be observed in 23 out of the 99 papers.

The rest of this paper is organized as follows: Section 2 briefly describes multi-label learning and feature selection. It also presents the systematic review method, the *LCFS* method and the multi-label datasets used in the experimental evaluation. Section 3 describes the experimental setting used to obtain the results discussed in Section 4. Section 5 shows the related work found by applying the SR method. Section 6 concludes the paper and

highlights future work.

2. Material and methods

This section briefly describes multi-label learning and feature selection, as well as the systematic literature review method. It also describes the *LCFS* feature selection method and the characteristics of the 10 benchmark multi-label datasets used in the experimental evaluation.

2.1. Multi-label learning

Let D be a dataset composed of N examples $E_i = (\mathbf{x}_i, Y_i)$, $i = 1 \ldots N$. Each example E_i is associated with a feature vector $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iM})$ described by M features (attributes) X_j, $j = 1 \ldots M$, and its multi-label Y_i, which consists of a subset of labels $Y_i \subseteq L$, where $L = \{y_1, y_2, \ldots, y_q\}$ is the set of q labels. Table 1 shows this representation. In this scenario, the multi-label classification task consists of generating a classifier H which, given an unseen example $E = (\mathbf{x}, ?)$, is capable of accurately predicting its multi-label Y, *i.e.*, $H(E) \rightarrow Y$.

Table 1: Multi-label data

	X_1	X_2	$\ldots$	X_M	Y
E_1	x_{11}	x_{12}	$\ldots$	x_{1M}	Y_1
E_2	x_{21}	x_{22}	$\ldots$	x_{2M}	Y_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
E_N	x_{N1}	x_{N2}	$\ldots$	x_{NM}	Y_N

2.1.1. Categorizing multi-label learning algorithms

Multi-label learning methods can be organized into two main categories [8]: problem transformation and algorithm adaptation. The key philosophy for

the former is to fit data to algorithms, while for the latter is to fit algorithms to data [18]. In particular:

- Problem transformation methods decompose the multi-label learning problem into a set of single-label (binary or multi-class) learning tasks;
- Algorithm adaptation methods adapt specific learning algorithms to handle multi-label datasets directly.

The multi-label learning algorithm *BRkNN*, which modifies the single-label *lazy* k-Nearest Neighbor (*kNN*) algorithm to classify multi-label examples, belongs to the algorithm adaptation category. To better tackle the multi-label problem, the extensions *BRkNN-a* and *BRkNN-b* are proposed in [19]. Both extensions are based on a label confidence score, which is estimated for each label from the percentage of the k-Nearest Neighbors having this label. *BRkNN-a* classifies a new example E using the labels in the multi-labels of the k-Nearest Neighbors which have a confidence score greater than 0.5, *i.e.*, labels included in the multi-labels of at least half of the k-Nearest Neighbors of E. If no label satisfies this condition, it outputs the label with the greatest confidence score. On the other hand, *BRkNN-b* classifies E with the $[s]$ (nearest integer of s) labels that have the greatest confidence score, where s is the average size of the multi-labels of the k-Nearest Neighbors of E. By conducting an experimental comparison with the state-of-the-art *lazy* algorithm *MLkNN* [20], the authors found that *BRkNN-b* achieved competitive results.

As *lazy* learning algorithms are sensitive to irrelevant features, they are a good choice to indicate the quality of a feature selection method. Thus, in

this work, we use *BRkNN-b* to assess the quality of the classifiers built using the original datasets — All Features (AF) — and the classifiers built using the datasets described by the selected features.

As exploring label dependence during learning could improve the classifier performance [21], Zhang and Zhou [18] proposed another categorization of multi-label learning methods which takes into account the degree of label dependence exploration. First-order strategies ignore the co-existence of other labels. The Binary Relevance (*BR*) approach, a problem transformation method, exemplifies this category by transforming a multi-label dataset into q single-label binary datasets, learning from each single-label problem separately and combining the results. Second-order strategies can consider pairwise relations between labels, such as interactions between any pair of labels, or the ranking between relevant and irrelevant labels. High-order strategies consider relations among more labels.

Although high-order strategies potentially model wider label dependences, they are usually computationally more demanding. This work focuses on finding second-order relations between single labels from the multi-label dataset and representing them as new labels. The idea is that, by labeling examples with the original and the constructed labels, feature selection methods based on the *BR* approach to incorporate label pairwise information are feasible.

2.1.2. Evaluation Measures

Unlike single-label classification where the classification of a new example has only two possible outcomes, correct or incorrect, multi-label classification should also take into account *partially* correct classification. As a consequence, multi-label evaluation measures consider the performance of

the classifier from diverse aspects and are, thus, of a different nature. A complete discussion on multi-label evaluation measures is out of the scope of this work and can be found in [8]. In what follows, we describe the four evaluation measures used in this work.

F-measure, *Hamming loss* and *Accuracy*, defined by Equations 1 to 3, are example-based evaluation measures, where Δ represents the symmetric difference of two sets, Y_i and Z_i are the true and the predicted multi-labels respectively.

$$F\text{-}measure(H, D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{2|Y_i \cap Z_i|}{|Z_i| + |Y_i|}. \tag{1}$$

$$Hamming\ loss(H, D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \Delta Z_i|}{|L|}. \tag{2}$$

$$Accuracy(H, D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}. \tag{3}$$

In addition, *Micro-averaged F-measure* (F_b), defined by Equation 4, is a label-based measure, where $T_{P_{y_i}}$, $F_{P_{y_i}}$, $T_{N_{y_i}}$ and $F_{N_{y_i}}$ represent, respectively, the number of true/false positives/negatives for a label $y_j \in L$.

$$F_b(H, D) = \frac{2 \sum_{j=1}^{q} T_{P_{y_j}}}{2 \sum_{j=1}^{q} T_{P_{y_j}} + \sum_{j=1}^{q} F_{P_{y_j}} + \sum_{j=1}^{q} F_{N_{y_j}}}. \tag{4}$$

All these performance measures range in the interval $[0, 1]$. For *Hamming loss*, the smaller the value, the better the multi-label classifier performance is, while for the other measures, greater values indicate better performance.

2.2. Feature selection

Regardless of the multi-label learning approach, any FS method addresses a few relevant issues, such as the interaction with the learning algorithm and the feature importance measure. The first issue is taken into account in different ways by the wrapper, embedded and filter approaches. The wrapper and the embedded approaches involve interaction with the learning algorithm, such that the features are selected for a specific learning algorithm. On the other hand, the filter approach uses general properties of the dataset to remove unimportant features from it, regardless of the learning algorithm. Thus, the features chosen may not be the best ones for a specific learning algorithm. The FS algorithms considered in this work use the filter approach.

Many measures have been proposed to estimate the importance of features based on characteristics of the dataset. As Section 5 reports, a frequently used single-label FS measure is Information Gain (IG), which evaluates each feature according to the dependence between this feature and a single label, as defined by Equation 5 — the higher the IG value for a feature X_j, the stronger is the relationship between X_j and the label.

$$IG(D, X_j) = entropy(D) - \sum_{v} \frac{|D_v| \, entropy(D_v)}{|D|}. \tag{5}$$

In other words, the IG of feature X_j, $j = 1 \ldots M$, calculates the difference between the entropy of dataset D and the weighted sum of the entropy of each subset $D_v \subseteq D$, where D_v consists of the set of examples where X_j has

the value v. Therefore, if X_j has 10 distinct values[1] in D, the sum would be applied to 10 different D_v datasets.

Using the BR approach, any single-label FS measure can be used to select features from multi-label data, as shown in [22], in which the single label FS measures IG and ReliefF are used. The procedure is simple: initially, using the BR approach the multi-label dataset is transformed into q single-label datasets, one per label. Afterwards, the single-label FS measure is applied to each feature X_j having as single-label y_i, $i = 1 \ldots q$, and the q results are averaged to obtain the final importance value of feature X_j. Finally, the importance value of the M features could be ranked to guide the selection of the better subset of features.

In this work, we also use Random Feature Selection (RFS) as a reference, in which the features are randomly selected, *i.e.*, no label or multi-label information is considered. Next , the Systematic Review (SR) method used to survey the literature on multi-label feature selection is briefly described.

2.3. Systematic review

The systematic literature review method provides a rigorous and replicable process to review the evidence relevant to a particular research question [17]. Although this method emerged in areas such as Medicine, currently there are guidelines and applications in other areas. In Computer Science, several applications can be found, including a systematic review that surveys other systematic reviews [23]. Figure 1 summarizes the workflow of the three systematic review steps — planning, conducting and reporting — as well as

[1]Discretization is applied to numerical features before using IG.

its main inputs and outputs.

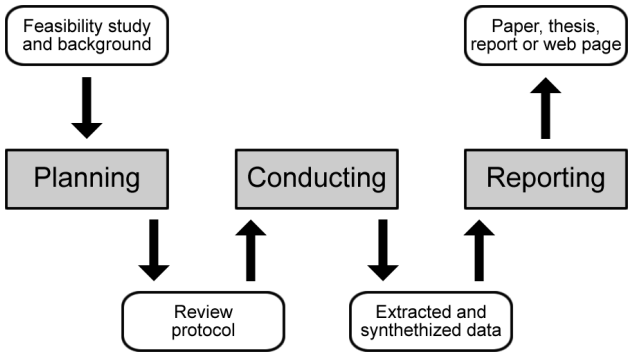


Figure 1: An overview of the systematic literature review method

The input of the planning step consists of a feasibility study about the systematic review method and the background related to the research question. In particular, the feasibility study can be carried out by analyzing whether or not a systematic review is needed concerning the topic of interest.

Planning yields a review protocol, which describes the systematic review components and improves the method replicability. This protocol should be consulted to conduct the method, which provides the data able to answer specific research question(s), which is (are) the SR core. In the last step, reporting, this data is disseminated in papers or a Ph.D. thesis, for example.

Although there are a few surveys on multi-label learning [18, 8], to the best of our knowledge there is no previous SR neither on multi-label learning nor on multi-label feature selection. This motivated us to conduct a pioneering systematic review to rigorously survey the related work on multi-label feature selection.

The systematic review research question we aim to answer is "*what are the publications of feature selection in multi-labeled data?*". Further details

regarding the review protocol and the instantiation of the SR method for multi-label FS are described in [24]. The reporting step to disseminate the related work summary is described in Section 5.

2.4. The LCFS *method*

Given a multi-label dataset D with the set of single labels $L = \{y_1, y_2, y_3, \ldots, y_q\}$, the main idea of *LCFS* is to construct q' new single labels by combining the original labels within pairs (y_i, y_j), $i \neq j$, $y_i \in L$ and $y_j \in L$. In each iteration, *LCFS* selects a pair of labels (y_i, y_j) from L and combines the labels within this pair to generate a new label y_{ij}. After repeating this procedure q' times, the q' new labels are included in the label set L, such that information about pairwise relationships between original labels can be used by the *BR* approach for feature selection.

The *LCFS* method consists of two steps, each one concerned with answering a different question:

1. Selection: which pairs of labels (y_i, y_j) should be chosen?
2. Generation: how to combine these labels to generate the new labels y_{ij}?

Figure 2 illustrates these steps for $q' = 1$.

Thus, instantiating *LCFS* involves choosing a strategy to select label pairs and a strategy to combine the labels within each pair. An additional parameter is the number of new labels q' that will be constructed. In what follows, the two *LCFS* steps are described.

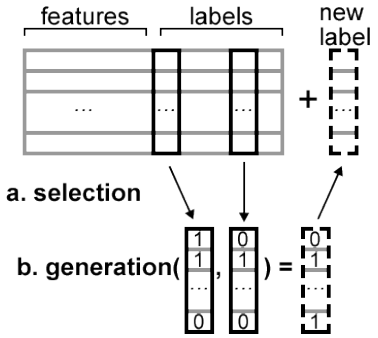


Figure 2: Applying the two steps of *LCFS* to construct $q' = 1$ new labels [16]

2.4.1. Step 1: selection

Given the set of labels $L = \{y_1, y_2, y_3, \ldots, y_q\}$ of dataset D, *LCFS* chooses q' *different pairs* of labels[2] (y_i, y_j), $i \neq j$, according to a selection strategy. The idea is that these pairs capture some pairwise relationships between the labels to be considered by feature selection.

LCFS supports different selection strategies, such as the simple Random Selection (*RS*), as well as heuristic strategies based on the number of examples labeled by each original single label (label frequency). In particular, two strategies considering label frequency are Co-occurrence-based Selection (*CS*) and related Labels Selection (*LS*). *CS* sorts in descending order label pairs according to the co-occurrence c_c, *i.e.*, the number of examples labeled by both labels within a pair — $(1, 1)$ — , and selects the first q' *different pairs*. On the other hand, *LS* counts:

1. The number of examples in which the labels within a pair agree, c_e — $(1, 1)$ or $(0, 0)$;

[2]In this work, two label pairs are considered different if they do not have a common label. For example, (y_3, y_5) and (y_1, y_3) are not considered different pairs because they share the label y_3.

2. The number of examples in which the labels within a pair disagree, c_d — $(1,0)$ or $(0,1)$.

Then, the pairs are sorted, in descending order, into two lists according to the values of c_e and c_d. The pair with the greatest value is selected, removed from the correspondent list and the procedure is repeated until selecting q' *different pairs*.

2.4.2. Step 2: generation

In this step, *LCFS* combines both labels from all previously selected pairs (y_i, y_j), $i \neq j$, to construct the new labels y_{ij}. The idea is that the values of y_{ij} represent a pairwise relationship between y_i and y_j. In the end, all examples in D are labeled by the q original labels and the q' new labels. *LCFS* supports different combination strategies between binary variables (labels). In this work, we use three simple logical operators to generate the values of the new labels of each example in D. The logical operators are:

AND : $y_{ij} = 1$ iff $y_i = y_j = 1$; $y_{ij} = 0$ otherwise.

XOR : $y_{ij} = 1$ iff $y_i \neq y_j$; $y_{ij} = 0$ otherwise.

XNOR : $y_{ij} = 1$ iff $y_i = y_j$; $y_{ij} = 0$ otherwise.

The AND operator clearly highlights co-occurring labels. XNOR, also known as the coincidence function, assigns the value 1 to y_{ij} *iff* the labels y_i and y_j agree, whereas XOR does the opposite. Although other logical operators, such as OR, could be included, we consider that AND, XOR and XNOR are enough to represent relations between the original labels.

Finally, after generating the q' new labels, the traditional *BR* approach for FS can be applied to the dataset now labeled by the $q + q'$ labels. Note that, by combining *BR* with *LCFS*, any single-label FS algorithm can be applied to the augmented dataset with second-order label information [18].

The *LCFS* method was implemented in Mulan [25], a multi-label learning package based on Weka [26], and is available to the community at http://www.labic.icmc.usp.br/pub/mcmonard/Implementations/Multilabel/lcfs.zip.

2.4.3. Illustrative example of the selection strategies

The application of the simple logical operators AND, XOR and XNOR to generate labels is straightforward. To illustrate the strategies to select *different pairs* of labels, consider the multi-label dataset described in Table 2, with $L = \{y_1, y_2, y_3, y_4\}$, and the number of new labels to be constructed $q' = \frac{q}{2} = 2$.

Table 2: Illustrative dataset for the *LCFS* method

	y_1	y_2	y_3	y_4
E_1	1	0	1	0
E_2	1	1	0	0
E_3	0	0	0	1
E_4	1	1	1	0
E_5	1	0	0	1
E_6	1	1	0	1
E_7	0	1	0	1

RS randomly selects $q' = 2$ *different pairs*. On the other hand, *CS* and *LS* sort the pairs of labels (y_i, y_j), $i \neq j$, in descending order according to the number of examples fulfilling a specific condition — c_c, c_e and c_d. Table 3 shows the c_c, c_e and c_d values calculated by *CS* and *LS* for each label pair

involving y_1, y_2, y_3 and y_4.

Table 3: Number of examples fulfilling specific conditions for each label pair

	(y_1, y_2)	(y_1, y_3)	(y_1, y_4)	(y_2, y_3)	(y_2, y_4)	(y_3, y_4)
c_c	3	2	2	1	2	0
c_e	4	4	2	3	3	1
c_d	3	3	5	4	4	6

First, *CS* selects the pair (y_1, y_2), which has the highest co-occurrence (c_c value). Then it considers the next pair in its ordered list. As (y_1, y_3) is not a pair different from (y_1, y_2) due to the label y_1, *CS* goes to the next list element, (y_1, y_4). This procedure is performed successively until finding (y_3, y_4), which is a pair different from the label pair previously selected. As $q' = 2$ pairs of labels were selected, the *CS* selection strategy ends.

LS compares the frequencies (numbers of examples) c_e and c_d from the first label pair in each ordered list: (y_1, y_2) and (y_3, y_4). As $c_d\,(y_3, y_4) > c_e\,(y_1, y_2)$, only (y_3, y_4) is selected. The procedure goes to the next label pair in the list from which (y_3, y_4) was selected, *i.e.*, the list sorted according to c_d. However, as the current label pair, (y_1, y_4), is not different from the pair previously selected due to the label y_4, the procedure moves to the next iteration. As $c_d\,(y_2, y_3) = c_e\,(y_1, y_2)$ and (y_1, y_2) is a *different pair*, the strategy selects (y_1, y_2) before ending.

2.5. Multi-label datasets

Table 4 summarizes the characteristics of the 10 datasets used in this work. For each dataset, it shows: dataset name (Dataset); dataset domain (Domain); number of examples (N); number of features (M); feature type ($Type$); number of labels ($|L|$); label cardinality (LC), which is the average number of labels associated with each example; label density (LD), which is

the cardinality normalized by $|L|$; and the number of different multi-labels ($\#Diff$).

Table 4: Dataset description

Dataset	Domain	N	M	$Type$	$\|L\|$	LC	LD	$\#Diff$
1-*Cal500*	music	502	68	numeric	174	26.044	0.150	502
2-*Corel5k*	image	5000	499	discrete	374	3.522	0.009	3175
3-*Corel16k001*	image	13766	500	discrete	153	2.859	0.019	4803
4-*Emotions*	music	593	72	numeric	6	1.869	0.311	27
5-*Fapesp*	text	332	8669	discrete	66	1.774	0.027	206
6-*Genbase**	biology	662	1185	discrete	27	1.252	0.046	32
7-*Llog-f**	text	1253	1004	discrete	75	1.375	0.018	303
8-*Magtag5k*	music	5260	68	numeric	136	4.839	0.036	4163
9-*Scene*	image	2407	294	numeric	6	1.074	0.179	15
10-*Yeast*	biology	2417	103	numeric	14	4.237	0.303	198

Except for datasets *5-Fapesp* and *8-Magtag5k*, the other datasets are available in the Mulan[3] and Meka[4] repositories. In particular, *5-Fapesp* was built by members of our research laboratory[5] [27]. Dataset *8-Magtag5k*[6] is further described in [28]. Furthermore, *6-Genbase** and *7-Llog-f** are pre-processed versions of the publicly available datasets in which an identification feature and unlabeled examples, respectively, were removed.

Besides the dataset characteristics shown in Table 4, information related to label frequency is also important to characterize multi-label datasets. To this end, we use quartiles[7] to describe the datasets label frequency distribution.

Figure 3 depicts the single label frequencies for each dataset by boxplots. Recall that the bottom and the top of the box are the first and third quartiles,

[3] http://mulan.sourceforge.net/datasets.html

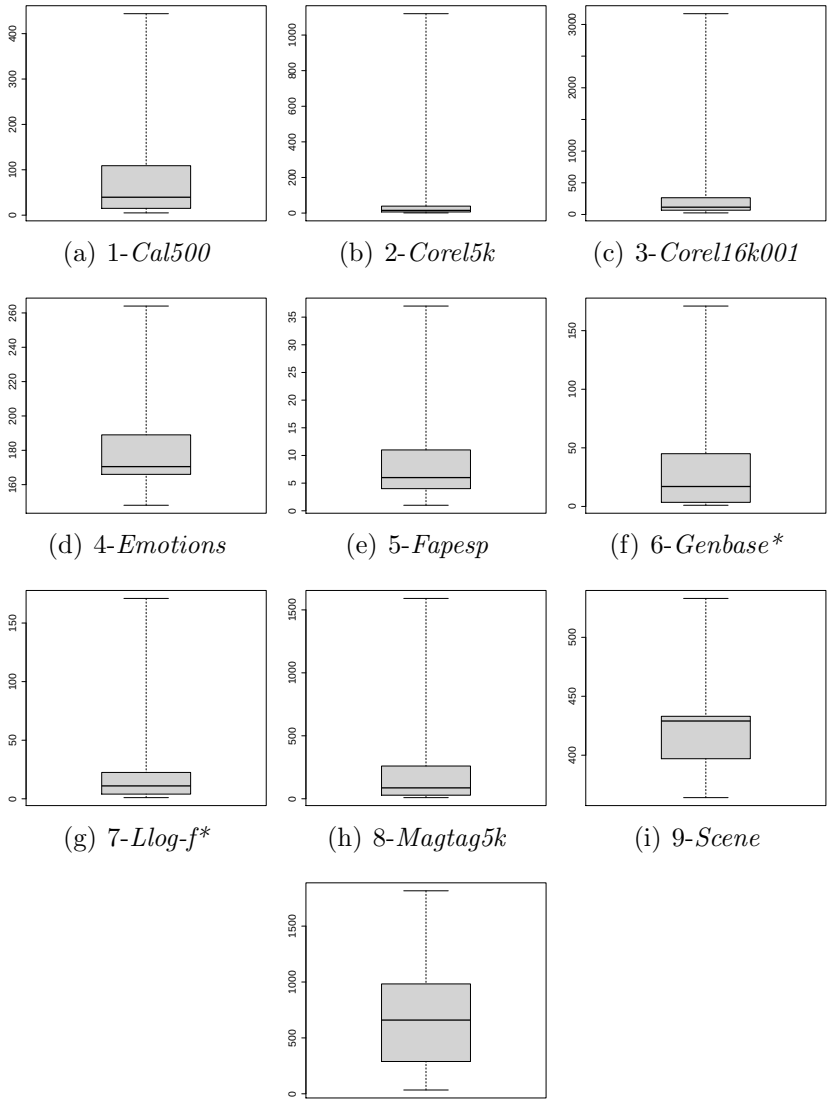
[4] http://meka.sourceforge.net/#datasets

[5] The dataset can be obtained from the authors.

[6] http://tl.di.fc.ul.pt/t/magtag5k.zip

[7] One of the three values that divides a sorted group of data into four equal parts, each one with 25% of the data.

and the band inside the box is the second quartile. Thus, the spacing between the different parts of the box indicates the degree of dispersion, as well as the skewness in the dataset. Moreover, the minimum and maximum label frequencies are also shown.



(a) 1-*Cal500* (b) 2-*Corel5k* (c) 3-*Corel16k001*

(d) 4-*Emotions* (e) 5-*Fapesp* (f) 6-*Genbase**

(g) 7-*Llog-f** (h) 8-*Magtag5k* (i) 9-*Scene*

(j) 10-*Yeast*

Figure 3: Boxplots of the single-label frequencies for each dataset

As can be observed, there is a large variation in the label frequency across the 10 multi-label datasets. It is also worth observing that datasets 3-*Corel16k001* (Figure 3(c), max.=3170, min.=25) and 5-*Fapesp* (Figure 3(e), max.=37, min.=1) show, respectively, the highest (3145) and the smallest (36) absolute difference between the maximum and the minimum label frequencies.

3. Experimental setting

The experiments were carried out on the 10 multi-label datasets described in Table 4. The performance of a FS method was assessed by the *BRkNN-b* classifiers built using the features selected by the method. In particular, the four evaluation measures described in Section 2.1.2 are used to assess the quality of these classifiers, as well as the classifiers built using All Features (AF). The evaluation measures were estimated according to the 10-fold cross-validation strategy.

The number of nearest neighbors k was set as 10 for all datasets. This is a commonly-used value that leads lazy learning algorithms to achieve satisfactory results [20, 29]. On the other hand, this differs from our previous work [16], in which the goal was to find the k value of *BRkNN-b* that maximized the *Example-based F-measure* value achieved by the classifier built from each original dataset. All the remaining parameters related to classification and feature selection were executed with the default values used by the Mulan[8] [25] and Weka[9] [26] frameworks.

[8] http://mulan.sourceforge.net
[9] http://www.cs.waikato.ac.nz/ml/weka

In this work, a combination between Information Gain and Binary Relevance (*IG-BR*) is performed according to the filter approach:

1. In the dataset annotated with the original set of labels (standard approach);
2. In the dataset annotated with the original labels and the ones constructed by a *LCFS* setting.

Figure 4 depicts some differences between these approaches. Moreover, by including second-order label information, *LCFS* is considered as a second-order strategy for multi-label FS, whereas the original *IG-BR* is considered as a first-order strategy.

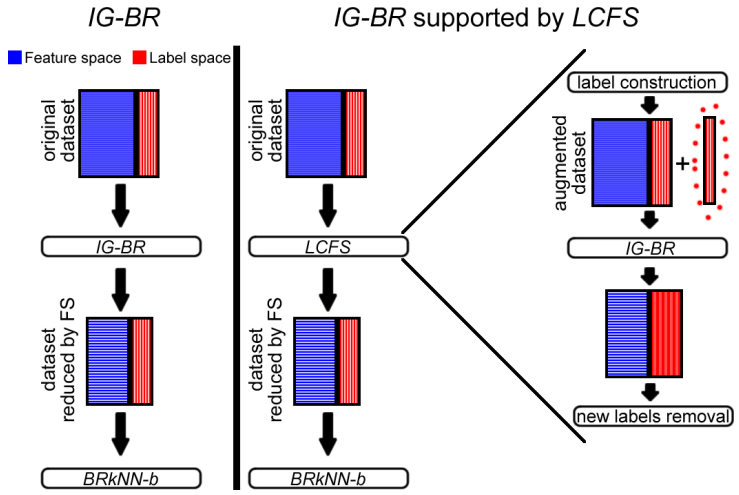


Figure 4: An overview of the FS process conducted by the original *IG-BR* (standard approach) and by *IG-BR* supported by *LCFS*

Regardless of the label set used, *IG-BR* transforms a multi-label dataset into single-label datasets, applies *IG* to each single-label dataset and averages the *IG* score of each feature X_j, $j = 1 \ldots M$, across all labels. The resulting feature ranking sorts the M averaged *IG* values in descending order. Recall

that the labels constructed by *LCFS*, which are only used to select features, are removed before classification.

It should be observed that the averaging strategy used in this work to aggregate *IG* scores in *BR* was highlighted in an experimental comparison on 20 multi-label textual datasets [30].

Table 5 shows the four settings combining different selection and generation strategies considered by *LCFS*.

Table 5: *LCFS* settings evaluated in this work

Setting	Selection	Generation
LS-X	*LS*	XOR or XNOR is chosen based on the lists sorted by the values of c_e and c_d
CS-A	*CS*	AND
RS-A	*RS*	AND
RS-X	*RS*	XOR or XNOR is randomly chosen

Recall that the *LS* selection strategy sorts the label pairs based on the values of c_e and c_d, *i.e.*, label agreement and disagreement. For a given label pair (y_i, y_j), *LS-X* applies the XNOR operator to generate the new label y_{ij} if the pair was selected from the list sorted by c_e; otherwise, it applies the XOR operator. It should be emphasized that *LS-X* and *CS-A* consider a relationship between the selection and the generation strategies, as they take into account label agreement/disagreement and co-occurrence respectively. Finally, *RS-X* randomly selects the XOR or XNOR operator. See Section 2.4.3 for an illustrative example.

We set the number of new labels $q' = \lfloor \frac{q}{2} \rfloor$, *i.e.*, every single label is selected once if q is even, or one single label is left out if q is odd.

In both cases, *IG-BR* and the four *LCFS* settings, the feature subsets $X' \subset X$, $|X'| = 10\%M, 20\%M, \ldots, 90\%M$, ranked by each FS method, are used to describe a dataset. This dataset is then submitted to the multi-label

learning algorithm *BRkNN-b*.

Moreover, the *RFS* method, which was not considered in our previous work [16], is included as a reference. In this case $10\%M$ up to $90\%M$ features are randomly chosen from the M features. *RFS* is executed three times per fold, due to its stochasticity, and the three outputs are averaged to yield the result of each fold.

As previously mentioned, multi-label evaluation measures consider the performance of a classifier from diverse aspects, as most algorithms learn from training examples by explicitly or implicitly optimizing one specific metric. To this end, in this work we also used $General_B$ [31], a simple baseline learning algorithm which learns by only looking at the multi-labels of the dataset. As this algorithm does not necessarily concentrate on optimizing specific loss functions, it can be used as a global baseline for the difficult task of evaluating multi-label predictions.

The rationale behind $General_B$ is very simple. It consists of ranking the q single labels in L according to their individual relative frequencies in the multi-labels in order to include the σ most frequent labels in the predicted multi-label Z. To obtain a representative Z, $General_B$ defines σ as the closest integer value of the label cardinality LC — Section 2.5. In case of ties (single labels with the same frequency), the label co-occurrence measure chooses the label which maximizes its co-occurrence with better ranked labels.

4. Results and discussion

In this section, we compare the learning performance of the *BRkNN-b* classifiers ($k = 10$) built from the datasets described by the features selected

by three groups of FS approaches:

1. The standard *IG-BR*;
2. *IG-BR* after applying the four *LCFS* settings to construct the new sets of labels: *LS-X*, *CS-A*, *RS-A* and *RS-X*;
3. The reference Random Feature Selection (*RFS*).

The main difference between groups (1) and (2) consists in the label space submitted for FS (Figure 4). The four *LCFS* settings take into account the different strategies illustrated in Section 2.4.3. Finally, as is the case with group (1), the method in group (3) selects features directly from the original datasets.

As a *BRkNN-b* classifier is built from a dataset described by the best 10% up to 90% of the features ranked by each FS method, 54 cases, *i.e.*, 9 feature subsets × 6 FS methods, are evaluated for each multi-label dataset and evaluation measure. All the experimental results, as well as tables and graphical representations, can be found in the supplementary material available at http://www.labic.icmc.usp.br/pub/mcmonard/ExperimentalResults/NEUCOM2015.pdf. In what follows, some of the experimental results are summarized and discussed.

4.1. Results overview

First of all, it is worth noticing that the datasets and measures in which the *BRkNN-b* classifiers built using all features failed to improve on the baseline classifier $General_B$ — Table 6. It should be observed that this situation is not unusual in multi-label learning. In [32] we carried out the SR process to find papers reporting experimental evaluation measure values

of classifiers which were constructed using publicly available datasets, and reported on several statistics. From the 10 datasets most frequently used in the selected papers, the statistics show that 12.8% of these published results were worse than or equal to the ones obtained by $General_B$. Moreover, this percentage is unevenly distributed among the datasets. In the "worst" dataset, 43.0% of such results were reported, and in the "best" one only 0.6%.

Table 6: Cases where $General_B$ outperforms the *BRkNN* classifier built using all features

Dataset	*F-measure*	*Hamming loss*	*Accuracy*	F_b
1-*Cal500*	✓	✓	✓	✓
2-*Corel5K*	✓	✓	✓	✓
3-*Corel16k001*	✓	✓	✓	✓
5-*Fapesp*		✓		
7-*Llog-f**		✓		

Nevertheless, dataset 1-*Cal500* is the only one in which the FS methods for all feature subsets considered fails to improve on $General_B$ in the four evaluation measures used in this work. Considering the datasets individually, it can be observed that, as expected, the degree of improvement of each FS method using different feature subsets is dependent on the particular dataset. One of the datasets which obtained good results in three of the four evaluation measures using small feature subsets, is dataset 5-*Fapesp*. Recall that finding a small number of good features is an aim of the FS task. Figure 5 shows the performance of the *BRkNN-b* classifiers according to each evaluation measure (y-axis) built using each feature subset (x-axis) in this dataset.

Figures 5(a), 5(c) and 5(d) show that very good results were obtained by using 10% of the features selected. Moreover, there is a considerable difference with the *RFS* method used as a reference, as well as the classifier

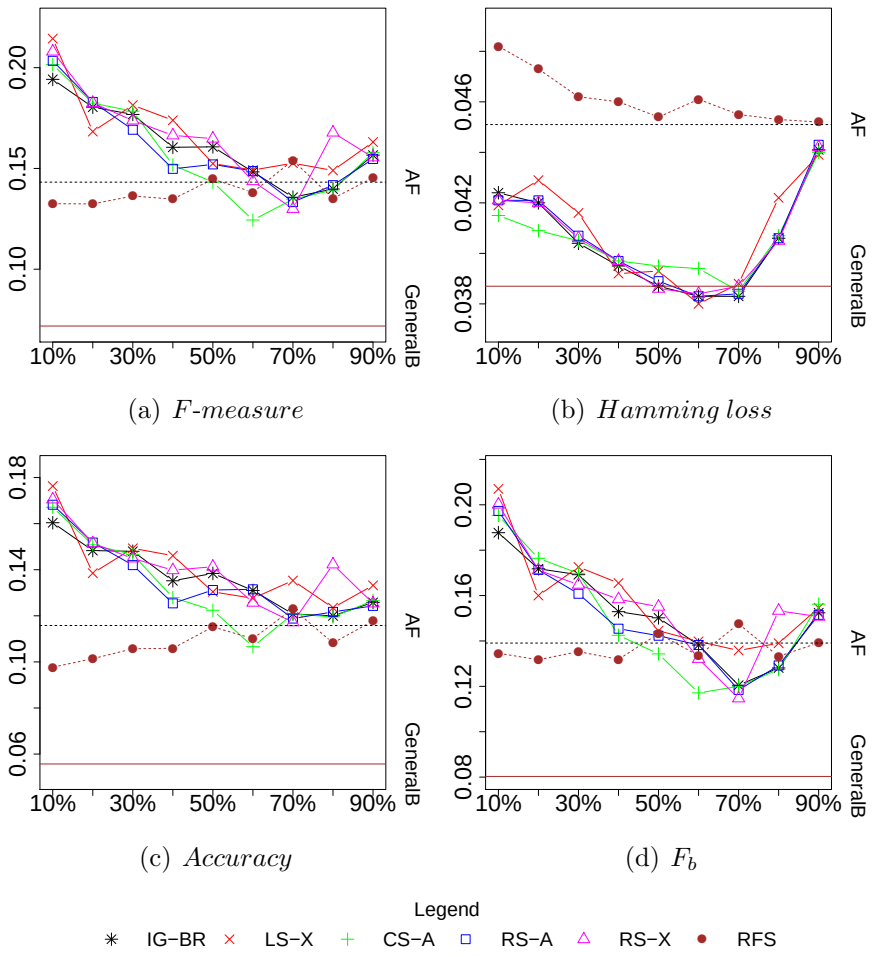


Figure 5: Performance of FS methods assessed by the correspondent *BRkNN-b* classifier in dataset 5-*Fapesp*

built using all features (AF). For *F-measure*, *Accuracy* and F_b, the best results were obtained by the FS methods *LS-X*, followed by *RS-X*, *RS-A*, *CS-A* and *IG-BR* in that order.

Hamming loss results better than the ones obtained by the baseline classifier $General_B$ were achieved only for three feature subsets: $50\%M$, $60\%M$ and $70\%M$ — Figure 5(b). Nevertheless, the heuristic FS methods are still notably better than *RFS* and AF. Thus, regardless of the evaluation measure, the *IG-BR* and *LCFS* methods are highlighted in this dataset.

In fact, *Hamming loss* is the evaluation measure which more often showed worse results than the ones obtained by the baseline classifier $General_B$. Figure 6 shows the number of datasets in which a *BRkNN-b* classifier, built using a feature subset chosen by a FS method, achieved *Hamming loss* values worse than the ones obtained by $General_B$. In this figure, the horizontal thick line shows the number of datasets in which a classifier built using All Features (AF), *i.e.*, without feature selection, was worse than $General_B$.

Figure 6 shows that for small feature subsets ($|X'| \leq 50\%M$), three *LCFS* variations (*CS-A*, *RS-A* and *RS-X*) were able to reduce by one the number of datasets in which the *BRkNN-b* classifiers were worse than the baseline classifier $General_B$. In particular, *RS-X* is the one which obtained that result with the smallest feature subsets ($|X'| = 10\%M$). On the other hand, the original *IG-BR* only achieved that result when using larger feature subsets ($|X'| = 60\%M$ and $|X'| = 70\%M$).

4.2. Statistical comparison among the FS methods

To assess whether the overall differences in performance across the multi-label FS methods are statistically significant, we used the Friedman's test

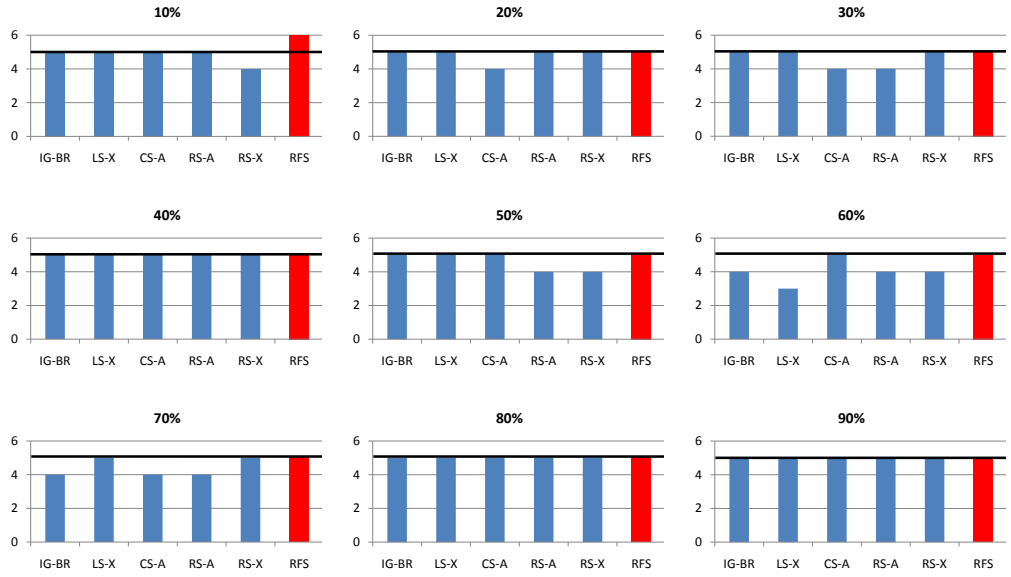


Figure 6: Number of datasets in which a *BRkNN-b* classifier built using a feature subset chosen by a FS method achieved *Hamming loss* values worse than the ones obtained by the *baseline* $General_B$

and the Nemenyi's post-hoc test as recommended by Demšar [33]. The Friedman's test is a non-parametric test for multiple hypotheses testing. It ranks the methods according to their performance for each dataset separately. The best performing method obtains the rank of 1, the second best the rank of 2, and so on. In case of ties, it assigns average ranks. If a statistically significant difference in the performance is detected, the next step is a post-hoc test to detect between which methods those differences appear.

We applied the Friedman's statistical test under the null hypothesis that the performances of the classifiers built using the features selected by each FS method are equivalent. The statistical results can be found in the suplementary material. As the hypothesis was rejected at the significance level $\alpha = 0.05$ for all measures, we proceed with the Nemenyi's multiple comparison

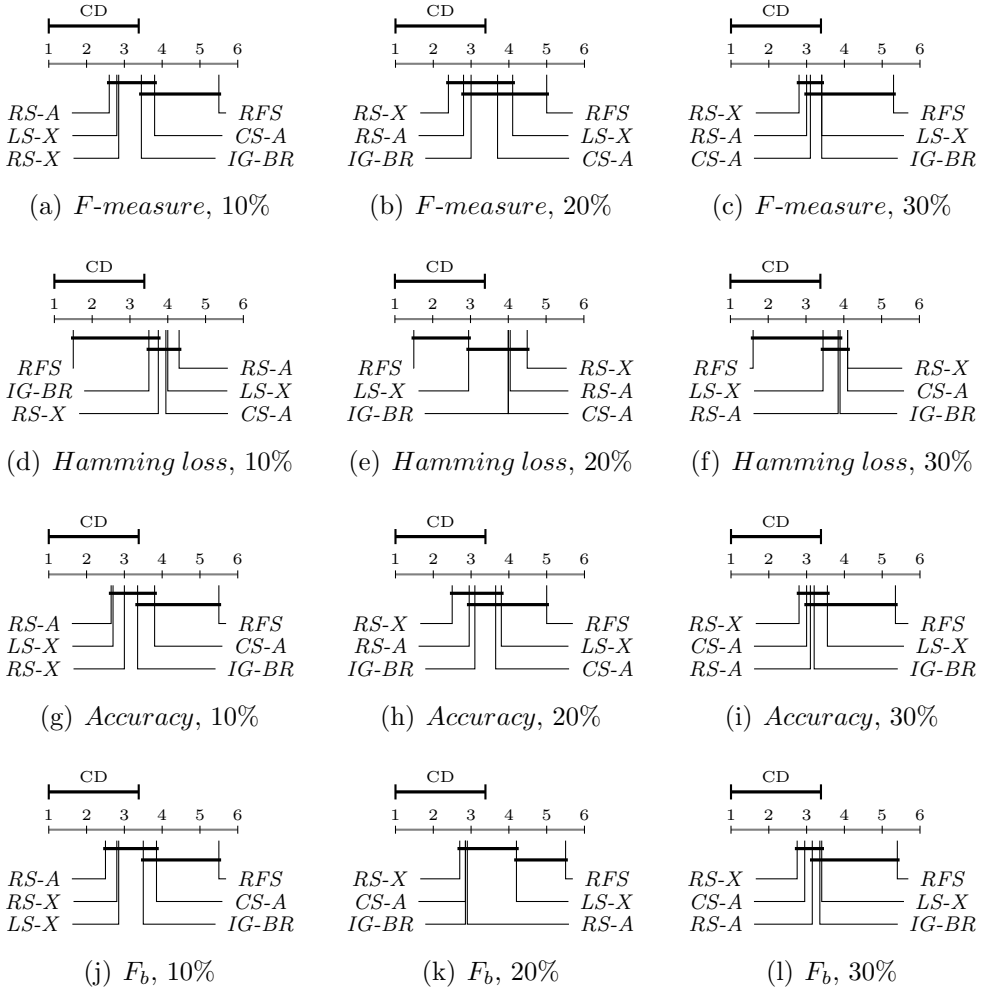
post-hoc test to detect which differences among the methods are significant. This post-hoc test points out a significant difference whenever the average rank of two methods differ by more than a Critical Difference (CD). Furthermore, the results of the post-hoc test can be visually represented with a simple diagram. Figure 7 shows the correspondent diagrams, on the four evaluation measures considered in this work, for the smaller feature subsets evaluated — $|X'| = 10\%M, 20\%M, 30\%M$. The diagrams for other feature subsets evaluated can be found in the supplementary material. The lines for the average ranks of the methods that do not differ significantly (at the significance level of $\alpha = 0.05$) are connected with a line. Observe that for the *Hamming loss* diagrams, the higher the average ranking, the better the FS method is, whereas for the remaining diagrams, the lower the average ranking, the better the method is.

In all cases in which a significant difference was found, at least a heuristic FS method outperformed *RFS*. Moreover, in all cases *RFS* was always ranked last. Table 7 summarizes all algorithms significantly better than *RFS* for each evaluation measure and feature subset size. For example, the first entry in this table shows that *RS-A*, *LS-X* and *RS-X* are significantly better than *RFS* when *F-measure* is considered.

Table 7: Multi-label FS methods significantly better than *RFS* ($\alpha = 0.05$)

$\|X'\|$	*F-measure*	*Hamming loss*	*Accuracy*	F_b
10%	*RS-A*, *LS-X*, *RS-X*	*RS-A*, *LS-X*, *CS-A*	*RS-A*, *LS-X*, *RS-X*	*RS-A*, *RS-X*, *LS-X*
20%	*RS-X*	*RS-X*, *RS-A*, *CS-A*, *IG-BR*	*RS-X*	*RS-X*, *CS-A*, *IG-BR*, *RS-A*
30%	*RS-X*	*RS-X*, *CS-A*	*RS-X*	*RS-X*, *CS-A*
70%		*LS-X*, *RS-X*, *IG-BR*		
80%	*RS-X*	*RS-X*	*RS-X*	*RS-X*

Regardless of the evaluation measure, from the 17 out of a total of 36



(a) *F-measure*, 10% (b) *F-measure*, 20% (c) *F-measure*, 30%

(d) *Hamming loss*, 10% (e) *Hamming loss*, 20% (f) *Hamming loss*, 30%

(g) *Accuracy*, 10% (h) *Accuracy*, 20% (i) *Accuracy*, 30%

(j) F_b, 10% (k) F_b, 20% (l) F_b, 30%

Figure 7: Nemenyi's test comparison of the performance measure values achieved by *BRkNN-b* classifiers built after selecting the best $|X'| = 10\%M, 20\%M, 30\%M$ of the features ranked by each FS method. Groups of classifiers that are not significantly different according to the Critical Difference (CD) — at $\alpha = 0.05$ — are connected

cases analyzed in which significant differences were found, it can be observed that the *LCFS* setting *RS-X* was the one highlighted more often as being significantly better than *RFS* (15 times out of 17), followed by *RS-A*, *LS-X* and *CS-A* (5 times out of 17 each), and finally *IG-BR* (3 times out of 17). As expected, significant differences tend to diminish when the feature subset size is large.

The Friedman's test also provides information about the best method built after FS by the rankings averaged across all datasets — Table 8. In this table, each symbol identifies a FS method: $-$ (*IG-BR*), $*$ (*LS-X*), o (*CS-A*), $\times$ (*RS-A*), $+$ (*RS-X*) and $\bullet$ (*RFS*). The last rows and columns sum up the results for each method. Note that there is more than one FS method in some cells, as the average rankings achieved by the correspondent classifiers are equal.

Table 8: Best FS method based on the Friedman's test average rankings calculated for each feature subset size and evaluation measure

	$10\%M$	$20\%M$	$30\%M$	$40\%M$	$50\%M$	$60\%M$	$70\%M$	$80\%M$	$90\%M$	$-$	$*$	o	$\times$	$+$	$\bullet$
F-measure	$\times$	$+$	$+$	$*$	o $\times$	$*$	$*$	$+$	$+$	0	3	1	2	4	0
Hamming loss	$\times$	$+$	o $+$	$*$ $\times$	$\times$	$-$ $+$	$*$	$+$	$*$	1	3	1	3	4	0
Accuracy	$\times$	$+$	$+$	$*$ o $+$	o	$*$	$*$	$+$	$+$	0	3	2	1	5	0
F_b	$\times$	$+$	$+$	$*$	$\times$	$-$ $*$	$*$	$+$	$*$	1	4	0	2	3	0
$-$	0	0	0	0	0	2	0	0	0	2					
$*$	0	0	0	4	0	3	4	0	2		13				
o	0	0	1	1	2	0	0	0	0			4			
$\times$	4	0	0	1	3	0	0	0	0				8		
$+$	0	4	4	1	0	1	0	4	2					16	
$\bullet$	0	0	0	0	0	0	0	0	0						0

Regardless of the evaluation measures, *RS-X* ($+$) achieved the best average rankings more often, mainly when $|X'| < \frac{|X|}{2}$, *i.e.*, less than half of the features are used. This setting was already highlighted in Figure 7. In addition, another *LCFS* setting, *RS-A* ($\times$), obtained the best average ranking for

the smallest feature subsets ($|X'| = 10\%M$). *IG-BR* (−), in turn, obtained the best average ranking in only two cases, when a large number of features ($|X'| = 60\%M$) was selected. On the other hand, no classifier built using the features chosen by *RFS* achieved the best average ranking.

As the *LCFS* setting *RS-X* was prominent in the statistical comparison and *IG-BR* represents the standard approach, we focused on the comparison of both methods.

We applied the Wilcoxon signed-ranks test, recommended for comparisons of two algorithms [33], with the null hypothesis that both methods are equivalent and $\alpha = 0.05$. By applying this test for each evaluation measure, *RS-X* was significantly better than *IG-BR* twice — when the classifiers are built using the feature subset size $|X'| = 80\%M$ and evaluated by *F-measure* and *Accuracy*.

4.3. LCFS RS-X *versus AF*

We compared the performance of the *BRkNN-b* classifiers built using the features selected by *IG-BR* in the original datasets and in the datasets augmented by using the four *LCFS* settings, as well as the classifiers built using features randomly chosen by *RFS*. In this comparison, *RS-X* showed good results when fewer features were selected. However, the quality of the classifiers has not been taken into account. To this end, we compare the performance of the classifiers built by *BRkNN-b*, using up to 30% of the features selected by *RS-X*, with the performance achieved by the *BRkNN-b* classifiers using all features, *i.e.*, the original dataset. Table 9 shows, for each dataset, and for each one of the four evaluation measures used in this work, whenever the classifiers built using the features selected by *RS-X* achieved

evaluation measure values better than or equal to (indicated by ★), or at most 5% worse (indicated by ☆) than the ones obtained by the classifiers using all features. The symbol 0 indicates the other cases.

Table 9: Classifiers built using the features selected by *RS-X vs* the classifiers built using all features

Dataset	$\|X'\| = 10\%M$	$\|X'\| = 20\%M$	$\|X'\| = 30\%M$
1-*Cal500*	☆/☆/☆/☆	☆/☆/☆/☆	☆/☆/☆/☆
2-*Corel5k*	★/★/★/★	★/★/★/★	★/★/★/★
3-*Corel16k001*	★/★/★/★	★/★/★/★	★/★/★/★
4-*Emotions*	0 / 0 / 0 / 0	☆/ 0 /☆/☆	☆/ 0 /☆/☆
5-*Fapesp*	★/★/★/★	★/★/★/★	★/★/★/★
6-*Genbase**	★/★/★/★	★/★/★/★	★/★/★/★
7-*Llog-f**	★/★/★/★	★/★/★/★	★/★/★/★
8-*Magtag5k*	0 / 0 / 0 / 0	0 / 0 / 0 / 0	0 / 0 / 0 / 0
9-*Scene*	0 / 0 / 0 / 0	0 / 0 / 0 / 0	0 / 0 / 0 / 0
10-*Yeast*	0 / 0 / 0 / 0	☆/ 0 /☆/☆	☆/☆/☆/☆

As can be observed, very good results were obtained in 5 out of the 10 datasets, in which the four evaluation measure values of the classifiers based on the *RS-X* setting were better than or equal to the ones obtained by the AF classifiers. Good results were obtained in all cases in dataset 1-*Cal500*, as well as in some cases in datasets 4-*Emotion* and 10-*Yeast* when 20% and 30% of the features are selected, as the results are at most 5% worse than the AF ones. On the other hand, poor results were obtained in datasets 8-*Magtag5k* and 9-*Scene* even when 30% of the features are considered. In fact, it is necessary to consider 70% of the features selected in these datasets in order to obtain (☆/ ☆/ ☆/ ☆).

However, the good results obtained in dataset 1-*Cal500* should be considered with care, as the classifiers built using AF are worse than the ones built by the baseline classifier $General_B$. In fact, this seems to be a difficult dataset to learn from. Table 4 shows that this dataset has $N = 502$ examples, as well as the same number of different multi-labels ($\#Diff$), in which the

average number of labels associated with each example is $LC = 20.044$ from a total of $|L| = 174$ labels.

5. Related work found by the systematic review method

Feature selection has been an active research topic in supervised learning, and there are many related publications and comprehensive surveys [7]. Although most FS publications are related to single-label learning, a number of papers have recently reported results to support multi-label learning [18, 8].

Aiming at capturing a wide, replicable and rigorous overview of the topic, we have instantiated the systematic literature review method [17] for multi-label FS in [24] and updated it in March/2015. Table 10 summarizes the 99 publications found in terms of two categorizations described in Sections 2.1 and 2.2: the order of label dependence and the interaction of the FS method with the learning algorithm. More information regarding the 99 references is described in the supplementary material available at `http://www.labic.icmc.usp.br/pub/mcmonard/ExperimentalResults/NEUCOM2015.pdf`.

Table 10: Number of papers published per approach found by the systematic literature review process (total = 99 related publications)

categorization	approach	#publications
order of label dependence	first-order	53
	second-order	15
	high-order	14
	hybrid	8
	unrecognized	9
interaction with the learning algorithm	filter	70
	embedded	12
	wrapper	8
	hybrid	3
	unrecognized	6

As can be observed, filters and first-order strategies have been the most usual choices in multi-label FS. This behavior could be partly explained by

the relative lower computational cost in comparison with other alternatives. In addition, these strategies can be combined, as exemplified by *IG-BR* and some proposals from the related work [34, 35, 36, 5, 37]. In particular, these filter methods apply the Information Gain importance measure in binary data directly or indirectly transformed by the *BR* approach, a first-order strategy.

Regarding importance measures, Information Gain has been the most often used measure (23 out of 99 papers). Mutual information [38], chi-squared [39], ReliefF [40] and correlation-based feature selection [1] come next.

The method we present, *LCFS*, pioneers label construction as a second-order strategy. Other methods that take into account label relations have also been proposed, reporting good results [41, 42, 43, 44, 45, 46, 47, 48, 49]. By organizing the related work according to the order of label dependence exploration, the SR can be useful for further research on multi-label feature selection.

Although the number of papers with unrecognized and hybrid[10] strategies in Table 10 is relatively low, it indicates the need to consider a taxonomy specific for multi-label FS. Well established taxonomies for single-label feature selection would be useful as a starting point [7].

The SR also provides information regarding the number of papers per publication year. Figure 8 suggests that interest in multi-label feature selection is increasing as time goes by.

[10]In this work, a hybrid strategy is considered whenever the FS method falls into two or more categories.

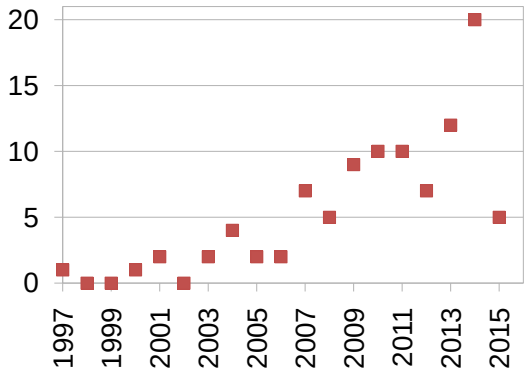


Figure 8: Number of papers published per year found by the systematic literature review process (total = 99 related publications)

6. Conclusion

This work presented and evaluated *LCFS*, a method that constructs new labels based on relations between the original dataset labels. The new labels are included in the dataset before applying the standard multi-label feature selection approach based on binary relevance. By doing so, *LCFS* considers second-order label information during filter feature selection.

The experimental evaluation on 10 benchmark multi-label datasets shows that the *LCFS* setting *RS-X* gave rise to classifiers similar to, or better than, the ones built by simply combining *BR* and Information Gain — *IG-BR*. *RS-X* is a simple alternative that randomly selects a pair of labels and combines them by the XOR or XNOR operator to yield each new label. Thus, the *LCFS* setting is competitive with *IG-BR* by slightly increasing the computational cost due to the application of a binary operator. Moreover, the evaluated method contributed to outperform classifiers built using all features, *i.e.*, without FS, as well as the baseline classifier $General_B$ and random feature selection.

As an additional contribution, this work pioneers the use of the systematic review method to survey the related work on multi-label FS. We organized the 99 papers found in terms of two categorizations proposed for multi-label learning methods and single-label FS algorithms. By doing so, it was observed that most of them consider first-order strategies, *i.e.*, ignore label dependence, and follow the filter approach. In the summary of the 99 papers, it was also found evidence that agrees with *LCFS* experimental achievements, as some related papers reported good results when exploring label dependence.

As future work, we plan to use other multi-label learning algorithms to evaluate FS, as exemplified in previous work [47], aiming to reduce the potential influence of a specific algorithm. Furthermore, we plan to evaluate *LCFS* strategies based on label weighting [50] in benchmark and synthetic datasets [51]. By applying Exploratory Data Analysis [52] in these cases, we expect to find relations among the quality of filter FS methods and multi-label datasets properties.

Acknowledgements

This research was supported by the São Paulo Research Foundation (FAPESP), grant 2011/02393-4. This agency did not have any further involvement in this paper.

Vitae

Newton Spolaôr holds a Ph.D. degree in Computer Science at the Institute of Mathematics and Computer Science from the University of São Paulo, Brazil (2014). He also holds an M.Sc. degree in Information Engineering from the Federal University of ABC (2010) and a B.Sc. degree in Computer Science from the State West Paraná University (2008). His main research interests are feature selection, multi-label learning, systematic review, time series clustering and analysis of biomedical images.

Maria Carolina Monard is Emeritus Professor in Computer Science at the Institute of Mathematics and Computer Science at the University of São Paulo, Brazil. She holds a Ph.D. degree from the Pontifical Catholic University of Rio de Janeiro, Brazil (1980), and an M.Sc. degree from Southampton University, UK (1968). Her research interests are in the field of Artificial Intelligence, more specifically in Machine Learning, Data and Text Mining.

Grigorios Tsoumakas is Assistant Professor at the Department of Informatics at the Aristotle University of Thessaloniki (AUTH), Greece. He holds a Ph.D. degree in Informatics from AUTH (2005), an M.Sc. in Artificial Intelligence from the University of Edinburgh (2000) and a degree in Informatics from AUTH (1999). His research interests include various aspects of machine learning, knowledge discovery and data mining, including ensemble methods, distributed data mining, text classification and multi-label learning.

Huei Diana Lee is Assistant Professor at the Engineering Institute and Exact Sciences at the State West Paraná University, Brazil. She holds a Ph.D. and an M.Sc. degree in Computer Science from the Institute of Mathematics and Computer Science at the University of São Paulo (2005 and 2000). She also holds a B.Sc. degree in Computer Science from the São Paulo State University, Brazil (1994). Her research interests are in bioinformatics, intelligent data analysis and telemedicine.

References

[1] S. Jungjit, A. A. Freitas, M. Michaelis, J. Cinatl, Extending multi-label feature selection with kegg pathway information for microarray data analysis, in: IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology, 2014, pp. 1–8.

[2] D. Heider, R. Senge, W. Cheng, E. Hüllermeier, Multilabel classification for exploiting cross-resistance information in HIV-1 drug resistance prediction, Bioinformatics 29 (2013) 1946–1952.

[3] P. K. Bhowmick, A. Basu, P. Mitra, A. Prasad, Multi-label text classification approach for sentence level news emotion analysis, in: Inter-

national Conference on Pattern Recognition and Machine Intelligence, 2009, pp. 261–266.

[4] A. Esuli, T. Fagni, F. Sebastiani, Boosting multi-label hierarchical text categorization, Information Retrieval 11 (2008) 287–313.

[5] W. Chen, J. Yan, B. Zhang, Z. Chen, Q. Yang, Document transformation for multi-label feature selection in text categorization, in: IEEE International Conference on Data Mining, 2007, pp. 451–456.

[6] M. R. Boutell, J. Luo, X. Shen, C. M. Brown, Learning multi-label scene classification, Pattern Recognition 37 (2004) 1757–1771.

[7] H. Liu, H. Motoda, Computational Methods of Feature Selection, Chapman & Hall/CRC, 2008.

[8] G. Tsoumakas, I. Katakis, I. P. Vlahavas, Mining multi-label data, in: O. Maimon, L. Rokach (Eds.), Data Mining and Knowledge Discovery Handbook, Springer, 2010, pp. 667–685.

[9] H. Motoda, H. Liu, Feature selection, extraction and construction, Communication of Institute of Information and Computing Machinery 5 (2002) 67–72.

[10] K. Lillywhite, D.-J. Lee, B. Tippetts, J. Archibald, A feature construction method for general object recognition, Pattern Recognition 46 (2013) 3300–3314.

[11] D. García, A. González, R. Pérez, A feature construction approach

for genetic iterative rule learning algorithm, Journal of Computer and System Sciences 80 (2014) 101–117.

[12] S. Piramuthu, R. Sikora, Iterative feature construction for improving inductive learning algorithms, Expert Systems with Applications 36 (2009) 3401–3406.

[13] R. Prati, F. Olivetti de Franca, Extending features for multilabel classification with swarm biclustering, in: IEEE Congress on Evolutionary Computation, 2013, pp. 2964–2971.

[14] W. Duivesteijn, E. Loza Mencía, J. Fürnkranz, A. Knobbe, Multi-label LeGo - enhancing multi-label classifiers with local patterns, in: J. Hollmén, F. Klawonn, A. Tucker (Eds.), Advances in Intelligent Data Analysis XI, volume 7619 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 114–125.

[15] Y. Yang, S. Gopal, Multilabel classification with meta-level features in a learning-to-rank framework, Machine Learning 88 (2012) 47–68.

[16] N. Spolaôr, M. C. Monard, G. Tsoumakas, H. D. Lee, Label construction for multi-label feature selection, in: Brazilian Conference on Intelligent Systems, IEEE, 2014, pp. 1–6.

[17] B. A. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering, EBSE-2007-01 Technical Report. 65 pg., 2007. Evidence-based Software Engineering.

[18] M.-L. Zhang, Z.-H. Zhou, A review on multi-label learning algorithms,

IEEE Transactions on Knowledge and Data Engineering 26 (2014) 1819–1837.

[19] E. Spyromitros, G. Tsoumakas, I. Vlahavas, An empirical study of lazy multilabel classification algorithms, in: Hellenic conference on Artificial Intelligence, Springer-Verlag, 2008, pp. 401–406.

[20] M.-L. Zhang, Z.-H. Zhou, A k-nearest neighbor based algorithm for multi-label classification, IEEE International Conference on Granular Computing 2 (2005) 718–721.

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

[22] N. Spolaôr, E. A. Cherman, M. C. Monard, H. D. Lee, A comparison of multi-label feature selection methods using the problem transformation approach, Electronic Notes in Theoretical Computer Science 292 (2013) 135–151.

[23] B. Kitchenham, R. Pretorius, D. Budgen, O. P. Brereton, M. Turner, M. Niazi, S. Linkman, Systematic literature reviews in software engineering - a tertiary study, Information and Software Technology 52 (2010) 792–805.

[24] N. Spolaôr, M. C. Monard, H. D. Lee, A systematic review to identify feature selection publications in multi-labeled data, ICMC Technical Report No 374. 31 pg., 2012. University of São Paulo.

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

[26] I. H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, 2011.

[27] R. G. Rossi, S. O. Rezende, Building a topic hierarchy using the bag-of-related-words representation, in: Symposium on Document Engineering, 2011, pp. 195–204.

[28] G. Marques, M. A. Domingues, T. Langlois, F. Gouyon, Three current issues in music autotagging, in: Conference of the International Society for Music Information Retrieval, 2011, pp. 795–800.

[29] M. Robnik-Šikonja, I. Kononenko, Theoretical and empirical analysis of ReliefF and RReliefF, Machine Learning 53 (2003) 23–69.

[30] N. Spolaôr, G. Tsoumakas, Evaluating feature selection methods for multi-label text classification, in: BioASQ workshop, 2013, pp. 1–12.

[31] J. Metz, L. F. Abreu, E. A. Cherman, M. C. Monard, On the estimation of predictive evaluation measure baselines for multi-label learning, in: J. Pavón, N. D. Duque-Méndez, R. Fuentes-Fernández (Eds.), Advances in Artificial Intelligence - IBERAMIA 2012, volume 7637 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 189–198.

[32] J. Metz, N. Spolaôr, E. A. Cherman, M. C. Monard, Comparing published multi-label classifier performance measures to the ones obtained

by a simple multi-label baseline classifier, in: Arxiv.org, Unpublished results, pp. 1–19. URL: http://arxiv.org/abs/1503.06952.

[33] J. Demšar, Statistical comparison of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.

[34] M. Du, M. Pierce, L. Pivovarova, R. Yangarber, Supervised classification using balanced training, in: L. Besacier, A.-H. Dediu, C. Martín-Vide (Eds.), Statistical Language and Speech Processing, volume 8791 of *Lecture Notes in Computer Science*, Springer International Publishing, 2014, pp. 147–158.

[35] M. Karabulut, Fuzzy unordered rule induction algorithm in text categorization on top of geometric particle swarm optimization term selection, Knowledge-Based Systems 54 (2013) 288–297.

[36] G.-P. Liu, J.-J. Yan, Y.-Q. Wang, J.-J. Fu, Z.-X. Xu, R. Guo, P. Qian, Application of multilabel learning using the relevant feature for each label in chronic syndrome diagnosis, Evidence-Based Complementary and Alternative Medicine 2012 (2012) 1–9.

[37] Z. Zheng, X. Wu, R. Srihari, Feature selection for text categorization on imbalanced data, Special Interest Group on Knowledge Discovery and Data Mining Explorations Newsletter 6 (2004) 80–89.

[38] Y. Yang, J. O. Pedersen, A comparative study on feature selection in text categorization, in: Proceedings of the Fourteenth International Conference on Machine Learning, 1997, pp. 412–420.

[39] E. Spyromitros-Xioufis, K. Sechidis, G. Tsoumakas, I. Vlahavas, Mlkd's participation at the clef 2011 photo annotation and concept-based retrieval tasks, in: ImageCLEF Lab of CLEF 2011 Conference on Multilingual and Multimodal Information Access Evaluation, 2011, pp. 1–15.

[40] D. Kong, C. Ding, H. Huang, H. Zhao, Multi-label ReliefF and F-statistic feature selections for image annotation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2352–2359.

[41] J. Lee, D.-W. Kim, Memetic feature selection algorithm for multi-label classification, Information Sciences 293 (2015) 80–96.

[42] J. Lee, D.-W. Kim, Mutual information-based multi-label feature selection using interaction information, Expert Systems with Applications 42 (2015) 2013–2025.

[43] O. G. R. Pupo, C. Morell, S. V. Soto, Scalable extensions of the relieff algorithm for weighting and selecting features on the multi-label learning context (in press), Neurocomputing (2015).

[44] F. M. Rodrigues, C. J. Camara, A. M. P. Canuto, A. M. Santos, Confidence factor and feature selection for semi-supervised multi-label classification methods, in: International Joint Conference on Neural Networks, 2014, pp. 864–871.

[45] K. Sechidis, N. Nikolaou, G. Brown, Information theoretic feature selection in multi-label data through composite likelihood, in: P. Fränti, G. Brown, M. Loog, F. Escolano, M. Pelillo (Eds.), Structural, Syntac-

tic, and Statistical Pattern Recognition, volume 8621 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2014, pp. 143–152.

[46] I. Slavkov, J. Karcheska, D. Kocev, S. Kalajdziski, S. Džeroski, Relieff for hierarchical multi-label classification, in: A. Appice, M. Ceci, C. Loglisci, G. Manco, E. Masciari, Z. W. Ras (Eds.), New Frontiers in Mining Complex Patterns, volume 8399 of *Lecture Notes in Computer Science*, Springer International Publishing, 2014, pp. 148–161.

[47] N. Spolaôr, M. C. Monard, Evaluating ReliefF-based multi-label feature selection algorithm, in: A. L. C. Bazzan, K. Pichara (Eds.), Advances in Artificial Intelligence – IBERAMIA 2014, volume 8864 of *Lecture Notes in Computer Science*, Springer International Publishing, 2014, pp. 194–205.

[48] Y. Yu, Y. Wang, Feature selection for multi-label learning using mutual information and ga, in: D. Miao, W. Pedrycz, D. Ślęzak, G. Peters, Q. Hu, R. Wang (Eds.), Rough Sets and Knowledge Technology, volume 8818 of *Lecture Notes in Computer Science*, Springer International Publishing, 2014, pp. 454–463.

[49] N. Spolaôr, E. A. Cherman, M. C. Monard, H. D. Lee, ReliefF for multi-label feature selection, in: Brazilian Conference on Intelligent Systems, 2013, pp. 6–11.

[50] S. Jungjit, A. A. Freitas, M. Michaelis, J. Cinatl, Two extensions to multi-label correlation-based feature selection: A case study in bioin-

formatics, in: IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 1519–1524.

[51] J. T. Tomás, N. Spolaôr, E. A. Cherman, M. C. Monard, A framework to generate synthetic multi-label datasets, Electronic Notes in Theoretical Computer Science 302 (2014) 155–176.

[52] G. J. Myatt, W. P. Johnson, Making Sense of Data I - A Practical Guide to Exploratory Data Analysis and Data Mining, John Wiley & Sons, 2014.