Restricted Set Classification: Who is there?

Ludmila I. Kuncheva[a], Juan J. Rodríguez[b], Aaron S. Jackson[c]

[a] *Bangor University, Dean Street, Bangor Gwynedd, LL57 1UT, United Kingdom*
[b] *University of Burgos, Escuela Politécnica Superior, Avda. de Cantabria s/n, 09006 Burgos, Spain*
[c] *The University of Nottingham, NG8 1BB, United Kingdom*

Abstract

We consider a problem where a set X of N objects (instances) coming from c classes have to be classified simultaneously. A restriction is imposed on X in that the maximum possible number of objects from each class is known, hence we dubbed the problem who-is-there? We compare three approaches to this problem: (1) Independent classification whereby each object is labelled in the class with the largest posterior probability; (2) A greedy approach which enforces the restriction; and (3) A theoretical approach which, in addition, maximises the likelihood of the label assignment, implemented through the Hungarian assignment algorithm. Our experimental study consists of two parts. The first part includes a custom-made chess data set where the pieces on the chess board must be recognised together from an image of the board. In the second part, we simulate the restricted set classification scenario using 96 datasets from a recently collated repository (University of Santiago de Compostela, USC). Our results show that the proposed approach (3) outperforms approaches (1) and (2).

Keywords: pattern recognition, object classification, restricted set classification, compound decision problem, chess pieces classification

1. Introduction

One of the standard assumptions in classical pattern recognition is that the data points to be classified come as an independent, identically distributed (iid) sequence. In many problems, this assumption does not hold. As an example, imagine the task of classifying all the pieces on a chess board from a bird-view snapshot, without knowledge of the course of the game up to that position. A classifier trained to recognise each piece individually will not be aware that, say, there cannot be more than two white bishops on the board. Thus a white pawn could be misclassified as a white bishop without a penalty. Should the classifier 'know' the restriction, a mistake of this type will be less likely.

Email addresses: l.i.kuncheva@bangor.ac.uk (Ludmila I. Kuncheva), jjrodriguez@ubu.es (Juan J. Rodríguez), aaron.jackson@nottingham.ac.uk (Aaron S. Jackson)

Consider a classification problem where an instance $\mathbf{x}$ may come from one of the c classes in the set $\Omega = \{\omega_1, \ldots, \omega_c\}$. Every instance is described by the values of n features, so without loss of generality, $\mathbf{x} \in \mathbb{R}^n$. Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_c\}$ be a set containing exactly one instance from each class. A set-classifier D_{set} will label X with a permutation of the c class labels and ensure the best match in terms of classification accuracy [1, 2]. We refer to this task as "who-is-who".

This paper extends the above model to the more general case where X consists of m instances, and it is known that at most k_i instances may belong to class ω_i, $i = 1, \ldots, c$. Denoting $k = k_1 + \ldots + k_c$, we require that $m \leq k$. The who-is-who task is a special case where $k_i = 1$, $i = 1, \ldots, c$, and $m = c$.

Simultaneous classification of instances has been studied in various contexts for non-i.i.d data:

1. Compound decision problem. Duda et al. [3] formulate the problem where each class is represented in X by a specific number of objects but without offering a solution. Taking inspiration from labelling the chromosomes in a cell (karyotyping), Slot [4] proposes a solution to this problem through maximising the log-likelihood of the labelling of X by using 0-1 integer programming.

2. Multiple-instance classification. This problem arises in complex machine learning applications where the information about the instances is incomplete or ambiguous [5, 6, 7, 8, 9], for example, in drug activity prediction [5]. The training examples come in "bags" labelled either positive or negative. For a positive bag, it is known that at least one instance in the bag has a true positive label. For a bag labelled negative, all instances are known to be negative. The problem is to design a classifier that can label as accurately as possible an unseen bag of instances.

3. Set classification. In this problem, all the instances in a set are assumed to have come from the same unknown class [10]. This problem may arise in face recognition where multiple images of the same person's face are submitted as a set.

4. Relaxation labelling. The m instances in set X should be labelled using the label set Ω. There are relationships between the classes which are specified by the so-called compatibility coefficients. Iterative relaxation labelling algorithms have been developed to solve this problem [11, 12].

Collective recognition. Collective recognition [13, 14] can be thought of as a special case of relaxation labelling. The instances within the set are related, so that the dependencies can be used to improve the classification accuracy. For example, in classifying web pages into topic categories, hyperlinked web pages are more likely to share common class labels than non-linked pages [14].

Who-is-who can be cast as a relaxation labelling problem. However, the compatibility coefficients enforcing the constraint of one-per-class are such that we cannot take advantage of the existing algorithms. In fact, there is an exact algorithm to solve who-is-who, which is preferable to the iterative alternatives.

5. Tracking of multiple objects. Simultaneous classification of a set of instances is used in tracking algorithms for video sequences. For example, a moving object can be regarded as a patchwork of parts [15], a set of tracklets [16] or a structure with connected pieces such as parts of a human face or body [17, 18, 19, 20]. The parts are matched from one image frame to the next. Tracking several people in a video [17, 21, 22] also relies on simultaneous classification. The classification in tracking is dominated by assessing the spatial location of the object/part using algorithms such as Kalman filter, Probabilistic Data Association Filter (PDAF) [23, 24], mixture modelling, AdaBoost, particle filters [25, 26], temporal templates [17], the Hungarian algorithm [27, 28], a game-theory approach [22], a locomotion model [21] and so on.

The appearance-based component (which is the trained classifier in our model) is deemed much less important [19]. Indeed, sometimes the objects are indistinguishable, and the only way to identify them is using their predicted and observed locations (for example, monitoring fruit flies [28]). Typically, the appearance-based classifier uses silhouette, texture [17], HSV colour histograms and edge detection [25, 23]. Tracking piglets has been attempted by marking each piglet on the back by a dye pattern [29, 30] thereby empowering the appearance-based classifier. The simultaneous classification model proposed here can be regarded as an additional tool for improving the tracking accuracy by making a better use of the objects' appearance.

While close, none of the problems and solutions above matches exactly our formulation of the restricted set classification problem. The closest set-up is the compound decision problem but in our case we allow for *up to* k_i objects from each class instead of a fixed number. Potential applications of the restricted set classification scenario include automatic attendance registration of students, karyotyping [31, 32, 33, 4], monitoring of animal behaviour (fruit flies [28], piglets [29, 30]), real-time labelling of the players in a game video stream (football [26], hockey [25]).

The rest of the paper is organised as follows. Section 2 lays out the theory behind the restricted set classification problem. Experimental results are shown in Section 3, and a conclusion is offered in Section 4.

2. The restricted set classification problem

Definition 1. The *restricted set classification problem* is defined as follows. Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ be a set of instances such that at most k_i instances come from class $\omega_i \in \Omega = \{\omega_1, \ldots, \omega_c\}$. Find labels for all elements of X so that the restriction holds.

Note that $k_1 + \ldots + k_c = k \geq m$.

Definition 2. A *base classifier* D is a classifier that assigns a class label to an instance $\mathbf{x} \in \mathbb{R}^n$

$$D : \mathbb{R}^n \to \Omega. \tag{1}$$

We also require that D provides estimates of the posterior probabilities $P(\omega_1|\mathbf{x}), \ldots, P(\omega_c|\mathbf{x})$.

Definition 3. A *super-label* for set X is any collection of m labels from Ω so that any instance $\mathbf{x} \in X$ receives a single label. A super-label will be called *consistent* if it satisfies the requirement that at most k_i labels are equal to ω_i, $i = 1, \ldots, c$.

Denote by $\mathcal{S}$ the set of all possible super-labels of X. Let $P = [p_{ij}]$ be a matrix of size $m \times c$ that contains the posterior probability estimates obtained from the base classifier D applied to X. Entry p_{ij} is the estimate of $P(\omega_j|\mathbf{x}_i)$. Let $\mathcal{P}$ be the set of all matrices P.

Definition 4. A *set classifier* D_{set} assigns a super-label to any set X using the output of classifier D, that is

$$D_{\text{set}}(X, D) : \mathcal{P} \to \mathcal{S}. \tag{2}$$

We consider two type of estimates of the accuracy of D_{set} for a given set X:

- A_T, total accuracy: $A_T = 1$ if *all* labels are correctly assigned to the instances in X, and $A_T = 0$, otherwise;
- A_P, partial accuracy: A_P is the *proportion* correctly labelled instances in X.

Definition 5. Independent (Baseline) Set Classifier D^i_{set}. This classifier takes the labels suggested by D without any modification.

Definition 6. Greedy Set Classifier D^g_{set}. Assume that D outputs the true posterior probabilities $P(\omega_i|\mathbf{x})$, for $i = 1 \ldots, c$ and any $\mathbf{x} \in \mathbb{R}^n$. This classifier labels the set X according to the following algorithm:

1. Initialise a set $V = \emptyset$ to store the assigned object-class pairs.
2. Identify the largest posterior probability $P(\omega^*_j|\mathbf{x}^*_j)$ among the objects and classes not assigned so far.
3. Remove ω^*_j from the list of available classes, and $\mathbf{x}^*_j$ from the list of available objects, and add the pair to set V.
4. If there are no class-object pairs left, stop and return V. Else, continue from step 2.

To derive the extended model for simultaneous classification we will first introduce the two special cases: the who-is-who [1] and who-is-missing [2].

2.1. Who-is-who?

A set of c objects have to be labelled into c classes so that there is exactly one object in each class. Let p be the probability that D will label correctly a randomly chosen instance $\mathbf{x}$.

D^i_{set} assumes that all instances are labelled independently. Then the accuracy measures of D^i_{set} are

$$A_T(D^i_{\text{set}}) = p^c \tag{3}$$

and

$$A_P(D^i_{\text{set}}) = p. \tag{4}$$

The super-label assigned by D^i_{set} may not be consistent, more likely so for larger number of classes c. It is possible to improve especially on $A_P(D^i_{\text{set}})$ by ensuring that the super label is consistent, for example by applying D^g_{set}.

2.1.1. Two classes

Let $c = 2$ and $X = \{\mathbf{x}_1, \mathbf{x}_2\}$. Without loss of generality, assume that $\mathbf{x}_1$ was drawn from the distribution of class ω_1, and $\mathbf{x}_2$ from ω_2, hence the true super-label is $\langle \omega_1, \omega_2 \rangle$. Suppose again that D is the perfect classifier for the chosen feature space, and therefore we have knowledge of the true posterior probabilities. To simplify notation, denote $P_1 = P(\omega_1|\mathbf{x}_1)$ and $P_2 = P(\omega_2|\mathbf{x}_2)$.

The probability that D^i_{set} will give the correct super-label of X is

$$A_T(D^i_{\text{set}}(X)) = Pr(P_1 > 0.5 \ \ \& \ \ P_2 > 0.5) \ . \tag{5}$$

However, D^g_{set} will add to this two more cases. The super-label will be correct also when $P_1 < 0.5$ and $P_2 > 1 - P_1$, ensuring that ω_2 will be assigned first to $\mathbf{x}_2$, leaving the free label ω_1 for $\mathbf{x}_1$. By the same logic, D^g_{set} will be right when $P_2 < 0.5$ and $P_1 > 1 - P_2$. Since the cases are mutually exclusive, the probability that D^g_{set} will give the correct super-label of X is

$$\begin{aligned} A_T(D^g_{\text{set}}(X)) \quad = & \\ & Pr(P_1 > 0.5 \ \ \& \ \ P_2 > 0.5) \\ + \quad & Pr(P_1 < 0.5 \ \ \& \ \ P_2 > (1 - P_1)) \\ + \quad & Pr(P_1 > (1 - P_2) \ \ \& \ \ P_2 < 0.5) \qquad (6) \\ \geq \quad & A_T(D^i_{\text{set}}(X)) \qquad (7) \end{aligned}$$

The above expression reduces to

$$A_T(D^g_{\text{set}}(X)) = Pr(P_1 + P_2 > 0.5) \ . \tag{8}$$

As $A_T(D^g_{\text{set}}(X)) \geq A_T(D^i_{\text{set}}(X))$ for any X, the inequality is valid across the whole space of pairs $(\mathbf{x}_1, \mathbf{x}_2)$.

To visualise the improvement due to the greedy strategy, consider the two-dimensional data set shown in Figure 1. We drew 10,000 random pairs $X = \{\mathbf{x}_1, \mathbf{x}_2\}$; $\mathbf{x}_1$ from ω_1 and $\mathbf{x}_2$ from ω_2.

The true probabilities $P_1 = P(\omega_1|\mathbf{x}_1)$ and $P_2 = P(\omega_2|\mathbf{x}_2)$ are used as the coordinate axes in Figure 2 where points corresponding to the 10,000 pairs are scattered. The region where D^i_{set} gives the correct super-label is shaded in light grey, and the number of points is shown. The regions where D^g_{set} adds accuracy to that of D^i_{set} are shaded in dark grey.

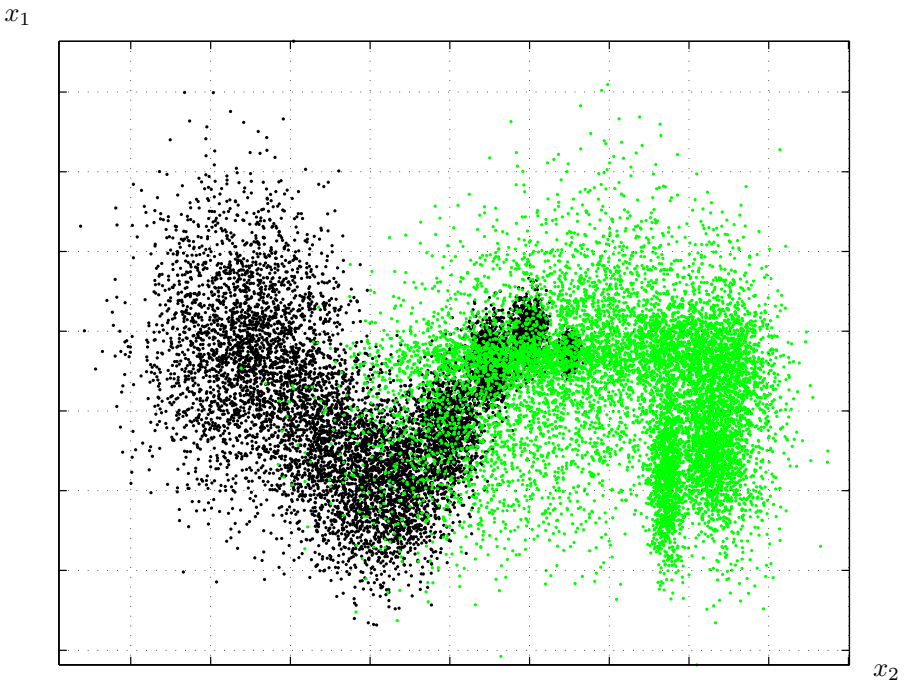


Figure 1: A two-dimensional data set with 10,000 points in each class.

For this example, $A_T(D^i_{\text{set}}) = 80.80\%$ and $A_T(D^g_{\text{set}}) = 7.65 + 7.90 + 80.80 = 96.35\%$.

Proposition 1. [1] For 2-class problems,

$$A_P(D^g_{\text{set}}) > A_P(D^i_{\text{set}}) \; . \tag{9}$$

The proof is shown in the Appendix.

The theory about the two-class who-is-who problem has been illustrated by an earlier experimental study to demonstrate the results' validity when D is not a Bayes classifier and the posterior probabilities are only estimates [1].

2.1.2. c classes

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_c\}$ be the set of c objects where object $\mathbf{x}_i$ is drawn from the distribution of class ω_i, independently of the other $c-1$ objects. Then the likelihood of a super label $S = \langle s_1, \ldots, s_c \rangle$, $s_i \in \Omega$ is

$$L(S) = \prod_{i=1}^{c} P(s_i|\mathbf{x}_i) p(\mathbf{x}_i|\omega_i) \; . \tag{10}$$

Since $p(\mathbf{x}_i|\omega_i)$ does not depend on the super-label S, we can organise it into a multiplicative constant $P_X = \prod_{i=1}^{c} p(\mathbf{x}_i|\omega_i)$, and rewrite the likelihood as

$$L(S) = P_X \prod_{i=1}^{c} P(s_i|\mathbf{x}_i) \; . \tag{11}$$

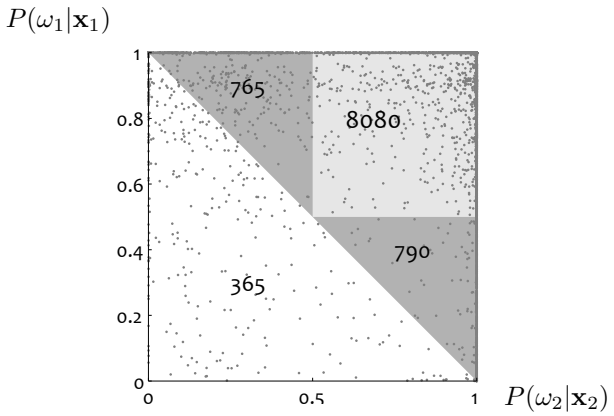


Figure 2: 10,000 random pairs of points with super-label $\langle \omega_1, \omega_2 \rangle$ drawn from the problem in Figure 1. The points are plotted the space of the true posterior probabilities $P(\omega_1|\mathbf{x}_1)$ and $P(\omega_2|\mathbf{x}_2)$. The number of points in the respective regions are displayed.

The set of feasible super-labels is the set of all permutations of the elements of Ω. The optimal super-label S^* will be the one maximising the L (equivalently $\log(L)$), that is

$$S^* = \arg\max_{S \in \mathcal{I}} \sum_{i=1}^{c} \log(P(s_i|\mathbf{x}_i)) \ , \tag{12}$$

where $\mathcal{I}$ is the set of all permutations of the class labels in Ω. S^*. Note that the greedy set classifier D^g_{set} will not guarantee the optimal solution.

Definition 7. Hungarian Set Classifier D^h_{set}. This classifier uses the Hungarian assignment algorithm [34] to find S^*.[1]. The input to the algorithm is the matrix LP with the *logarithms* of the posterior probabilities obtained from the individual classifier, and the output is the optimal permutation S^*, guaranteeing the maximum sum of logarithms, as in equation (12).

Previous experiments have shown significant improvement of $A_T(D^h_{\text{set}})$ over both $A_T(D^g_{\text{set}})$ and $A_T(D^i_{\text{set}})$ [1].

2.2. Who-is-Missing?

In this scenario, a set X of $k < c$ objects have to be labelled into c classes so that there is at most one object in each class [2]. As the question of interest here is "who-is-missing", it may be thought that the correct assignment of the elements of X is not strictly necessary. However, in order to maximise the likelihood of discovering the identity of the missing classes, we still need to maximise the likelihood of the assignments of the objects in X. Let $S_{(-)} \subset \Omega$ and $S_{(+)} = \Omega \setminus S_{(-)}$ be respectively the set of missing and the set of present classes. The probability that $S_{(-)}$ is missing is the same as the probability that $S_{(+)}$ is present.

[1] Further developed by Kuhn and Munkres, also known as Kuhn-Munkres algorithm. Proposed originally for $c \times c$ matrices, the Hungarian algorithm has been extended for rectangular matrices [35].

Therefore, by maximising the log-likelihood of the collection of labels in $S_{(+)}$, we maximise the probability of discovering the correct $S_{(-)}$. This allows for the who-is-missing problem to be cast as who-is-who.

We create $v = c - k$ dummy objects $\mathbf{z}_1, \ldots, \mathbf{z}_v$, and assign probabilities $P(\omega_i|\mathbf{z}_j) = 1/c$ for all $i = 1, \ldots, c$ and $j = 1, \ldots, v$. Thus the matrix with posterior probabilities P is of size $c \times c$, and has v identical rows with values $1/c$. The optimal labelling (eqn (12)) becomes

$$S^* = \arg\max_{S \in \mathcal{I}} \left(\sum_{i=1}^{k} \log(P(s_i|\mathbf{x}_i)) + \sum_{j=1}^{v} \log\left(\frac{1}{c}\right) \right). \tag{13}$$

The second term can be absorbed into a constant which does not depend on S, and does not affect the optimal assignment of labels to X. The Hungarian set algorithm D^h_{set} can be applied to P for finding S^*. The labels assigned to the dummy objects are the missing classes. Note that the value of the constant we assign in place of the posterior probabilities for the dummy objects does not matter. The same assignment will be obtained for any value.

We carried a set of experiments for the who-is-missing problem [2] with the UCI letter data set and an image data set of LEGO parts. The results again strongly favoured the Hungarian set algorithm D^h_{set} before the greedy D^g_{set} and the independent D^i_{set} set algorithms.

2.3. Solution to the restricted set classification problem

Following the naming convention for the two special cases above, we call the restricted set classification problem (Definition 1) 'who-is-there'.

Recall the example of who-is-there where the chess pieces on a board are to be recognised from a bird-view snapshot. In this case, there are 12 possible classes (pawn, bishop, knight, castle, king, queen; $\times$ 2 for black and white) and up to 32 objects. The maximum number of objects from each class is fixed by context. We assume that we don't have prior knowledge of the moves leading to the current board configuration.

To solve the who-is-there problem, we should be looking to maximise the log-likelihood of the super-label. However, this time the set of classes has to be augmented with k_i copies of each class. The posterior probabilities for the copies are the same as the one in the original column. In addition, the set of objects also has to be expanded to contain k objects altogether. The $k - m$ dummy objects are assigned $1/c$ posterior probabilities for all k possible class labels in the super-label. The resultant matrix P is of size $k \times k$. The example below illustrates this arrangement.

Consider three classes denoted respectively $\bullet$, $\triangle$ and $\square$. It is known that X contains at most $k = 5$ objects where at most 2 are from class $\bullet$, at most 1 is from class $\triangle$ and at most 2 are from class $\square$. Suppose that the observed set X contains $m = 4$ objects with the following posterior probabilities provided by D:

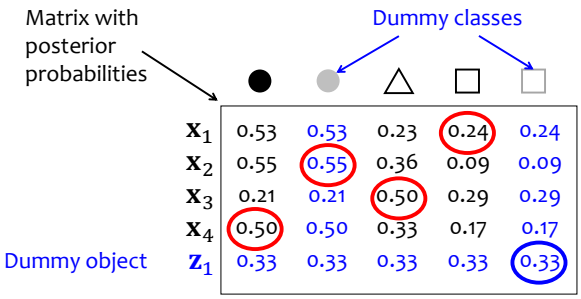


Figure 3: Construction of matrix P (8×8) for the numerical example. The ellipses show the assignment by D^h_{set}.

Object	$P(\bullet\|\mathbf{x})$	$P(\triangle\|\mathbf{x})$	$P(\square\|\mathbf{x})$
$\mathbf{x}_1$	0.53	0.23	0.24
$\mathbf{x}_2$	0.55	0.36	0.09
$\mathbf{x}_3$	0.21	0.50	0.29
$\mathbf{x}_4$	0.50	0.33	0.17

We construct P as shown in Figure 3. The assignment resulting from applying D^h_{set} to P is also indicated in the figure.

The Hungarian set classifier D^h_{set} will assign super-label $\langle \square \bullet \triangle \bullet \rangle$ to the four objects in X, amounting to criterion value

$$\ln(0.24) + \ln(0.55) + \ln(0.50) + \ln(0.50) = -3.4112 \,.$$

The Greedy set classifier D^g_{set}, on the other hand, will assign super-label $\langle \bullet \bullet \triangle \square \rangle$, which gives

$$\ln(0.53) + \ln(0.55) + \ln(0.50) + \ln(0.17) = -3.6978 \,.$$

Both super-labels satisfy the constraints but D^h_{set} leads to a higher (better) log-likelihood value compared to D^g_{set}. This is to be expected as D^h_{set} guarantees the optimal assignment of labels with respect to the log-likelihood criterion.

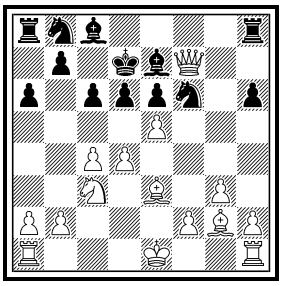
3. Experiments

Our hypothesis is that, for the restricted set classification problem, both accuracy measures A_T and A_P should be maximised by applying D^h_{set} compared to applying D^g_{set} and D^i_{set}, for various models of the base classifier D.

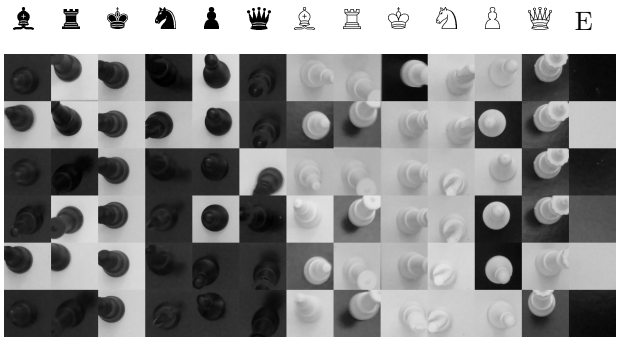
3.1. Chess pieces

For this experiment we used 46 bird-view snapshots of a chess board. Each image was cropped and squared (Figure 4 (a)), and subsequently split into 64 squares. The task is to identify the chess pieces and their positions (Figure 4 (b)). Figure 4 (c) shows examples of the training data for the 13 classes: 12 classes for the chess pieces and one class for empty squares (denoted by E). Five instances (image tiles) are shown from each class.

(a) Bird-view snapshot



(b) True positions



(c) Examples of the training data

Figure 4: A chess board example

We decided to use the following features on each tile:

- Entropy of the grey-level image.
- Standard deviation of the grey-level image.
- Difference between the mean grey intensity of (i) a square centred at the tile centre, with side equal to half of the tile side, and (ii) the tile area outside the square.
- 100 grey-level values obtained by resizing the image to 10×10 pixels.

The experiment was organised as 46-fold cross-validation where one board was left aside in each fold. A classifier was trained for each fold, and the posterior probabilities of the testing data were subsequently calculated. The classifier models which we tried out as base classifier D are shown in Table 1.[2]

The matrix with the posterior probabilities P was expanded as demonstrated in Figure 3. The three set classifiers: D^i_{set} (B), D^g_{set} (G) and D^h_{set} (H) were then applied giving the accuracies shown in Table 1.

Table 1: Averaged partial accuracy A_P and averaged total accuracy A_T in % for the Baseline (B), Greedy (G) and the Hungarian (H) set classifiers for the 46-cross-validation experiment for the Chess data. Symbol • after the value in column A_p-H indicates that H is significantly different from G at $p < 0.001$, and symbol − means that there is no significant difference. The largest A_P for each classifier type is underlined.

	Partial accuracy A_P			Total accuracy A_T		
Classifier	B	G	H	B	G	H
Nearest neighbour	95.75	96.23	<u>96.43</u> −	32.61	43.48	47.83
Linear discriminant analysis	79.42	80.74	<u>83.19</u> •	0.00	0.00	0.00
Naïve Bayes	39.64	59.68	<u>64.20</u> •	0.00	0.00	0.00
Decision tree	<u>87.13</u>	85.67	86.01 −	4.35	4.35	4.35
SVM – linear	89.57	90.18	<u>90.52</u> −	0.00	2.17	2.17
SVM – radial basis function (RBF)	87.67	88.52	<u>89.50</u> •	2.17	2.17	2.17
SVM – sigmoid kernel	67.46	<u>67.53</u>	67.46 −	2.17	2.17	2.17
SVM – polynomial kernel	87.57	88.76	<u>88.82</u> −	0.00	0.00	0.00
Random Forest	91.88	93.00	<u>93.58</u> •	2.17	19.57	26.09

Paired t-test was used to identify any significant differences between A_P for the proposed D^h_{set} against D^g_{set} at $p < 0.001$ for the classifier models. The results favour D^h_{set} which is often the most accurate set classifier. The greedy set classifier is the next best, and the baseline set classifier (independent application of D) wins only once, for the decision tree classifier. This is likely a result from the notoriously poor approximation of posterior probabilities by the standard decision tree classifier.

Looking at A_T, the only base classifier which can be of any use in recognising the whole set of chess pieces for this experiment is the nearest neighbour. Even its accuracy A_T is not sufficiently high. For the purposes of demonstrating the advantage of using a proper set classifier, we chose a difficult task. This can be observed in the small example of the training set in Figure 4 (c). The position of the camera was such that the same type of piece could be seen to "lean" up, down, left or right, so much so that a distinguishing part of the top of the piece is missing. We did not attempt to analyse whether the background of the piece is

[2]Own MATLAB code was used for the nearest neighbour classifier, the naïve Bayes classifier and the random forest ensemble (RF). The MATLAB Statistic Toolbox was used for the linear discriminant classifier, the decision tree classifier and the random tree base classifier for RF. LibSVM library [36] was used for the multi-class SVM classifiers.

black or white, and proceed to apply different classifiers accordingly. If the aim of the paper was to recognise correctly all the pieces in this restricted set classification problem, we would have opted for multiple cameras, more elaborate and context-related features, advanced classifiers such as deep learning neural networks, and calibration of the posterior probabilities. Our experiment is a proof of concept. It demonstrates that the restricted set classification problem needs special treatment beyond training a standard classifier D and applying it independently to the elements of the set of instances X.

3.2. USC data

3.2.1. The data collection

In the second set of experiments we used a collection of datasets chosen as a testbed for a comprehensive experimental evaluation of 179 classifiers from 17 families [37][3]. Many of the datasets within the collection are from the UCI Machine Learning Repository [38]. We refer to this repository as USC after the host university (University of Santiago de Compostela, Spain).

To enable a reasonable formulation of the restricted set classification problem we had to ensure that there is sufficient variability within each class we sampled from. Otherwise the results would depend too much on a few instances. Therefore, we reduced the original 121 datasets to 96 datasets, and modified those retained according to the following rule. Classes containing less than 50 examples were removed. Hence, datasets without at least two classes with at least 50 examples in each were discarded from the collection. The description of the data collection that we used is shown in Table 2.

3.2.2. Experimental protocol

Two fold cross validation, repeated five times, was used to partition the data into training and initial testing sets. From each of the ten initial testing sets, 100 runs were carried out. Thus, for each dataset, we carried out 1000 runs.

In each run, we commence by initialising the set to be labelled by $X = \emptyset$. A random integer between 1 and 10 is drawn for each class to serve as the limit on the number of possible objects from that class. Denote by k_i the number of objects allowed for class ω_i. Then a second random integer, r_i, is drawn between 0 and k_i, to determine how many objects from ω_i will *actually* be present in the set X. Note that we include the possibility of a completely absent class. Next, we add to X r_i randomly selected testing objects whose true label is ω_i. After constructing the set X, D^i_{set} (Baseline), D^g_{set} (Greedy) and D^h_{set} (Hungarian) are applied to find the respective super-labels.

The following set of classifiers were tested as D:

- Nearest neighbour (1-NN)

[3]The repository is available at `http://persoal.citius.usc.es/manuel.fernandez.delgado/papers/jmlr/`

Table 2: Characteristics of the datasets (#E: examples, #F: feautures, #C: classes).

dataset	#E	#F	#C
abalone	4177	8	3
acute-inflammation	120	6	2
acute-nephritis	120	6	2
adult	48842	14	2
annealing	850	31	3
arrhythmia	295	262	2
balance-scale	576	4	2
bank	4521	16	2
blood	748	4	2
breast-cancer	286	9	2
breast-cancer-wisc	699	9	2
breast-cancer-wisc-diag	569	30	2
car	1728	6	4
cardiotocography-10clases	2126	21	10
cardiotocography-3clases	2126	21	3
chess-krvk	28029	6	17
chess-krvkp	3196	36	2
congressional-voting	435	16	2
conn-bench-sonar-mines-rocks	208	60	2
conn-bench-vowel-deterding	990	11	11
connect-4	67557	42	2
contrac	1473	9	3
credit-approval	690	15	2
cylinder-bands	512	35	2
dermatology	297	34	4
ecoli	272	7	3
energy-y1	768	8	3
energy-y2	768	8	3
glass	146	9	2
haberman-survival	306	3	2
hayes-roth	129	3	2
heart-cleveland	219	13	2
heart-hungarian	294	12	2
heart-va	107	12	2
hill-valley	1212	100	2
horse-colic	368	25	2
ilpd-indian-liver	583	9	2
image-segmentation	2310	18	7
ionosphere	351	33	2
iris	150	4	3
led-display	1000	7	10
letter	20000	16	26
low-res-spect	469	100	3
lymphography	142	18	2
magic	19020	10	2
mammographic	961	5	2
miniboone	130064	50	2
molec-biol-promoter	106	57	2
molec-biol-splice	3190	60	3
monks-1	556	6	2
monks-2	601	6	2
monks-3	554	6	2
mushroom	8124	21	2
musk-1	476	166	2
musk-2	6598	166	2
nursery	12958	8	4
oocytes-merluccius-nucleus-4d	1022	41	2
oocytes-merluccius-states-2f	1022	25	3
oocytes-trisopterus-nucleus-2f	912	25	2
oocytes-trisopterus-states-5b	898	32	2
optical	5620	62	10
ozone	2536	72	2
page-blocks	5445	10	4
pendigits	10992	16	10
pima	768	8	2
planning	182	12	2
ringnorm	7400	20	2
seeds	210	7	3
semeion	1593	256	10
soybean	362	35	4
spambase	4601	57	2
spect	265	22	2
spectf	267	44	2
statlog-australian-credit	690	14	2
statlog-german-credit	1000	24	2
statlog-heart	270	13	2
statlog-image	2310	18	7
statlog-landsat	6435	36	6
statlog-shuttle	57977	9	5
statlog-vehicle	846	18	4
steel-plates	1941	27	7
synthetic-control	600	60	6
teaching	102	5	2
thyroid	7200	21	3
tic-tac-toe	958	9	2
titanic	2201	3	2
twonorm	7400	20	2
vertebral-column-2clases	310	6	2
vertebral-column-3clases	310	6	3
wall-following	5456	24	4
waveform	5000	21	3
waveform-noise	5000	40	3
wine	130	13	2
wine-quality-red	1571	11	4
wine-quality-white	4873	11	5
yeast	1350	8	5

- Linear discriminant analysis (LDA)
- Naïve Bayes (NB)
- Logistic Regression (LOG)
- Decision tree (DT)
- Random Forest (RF)
- Rotation Forest (ROT)

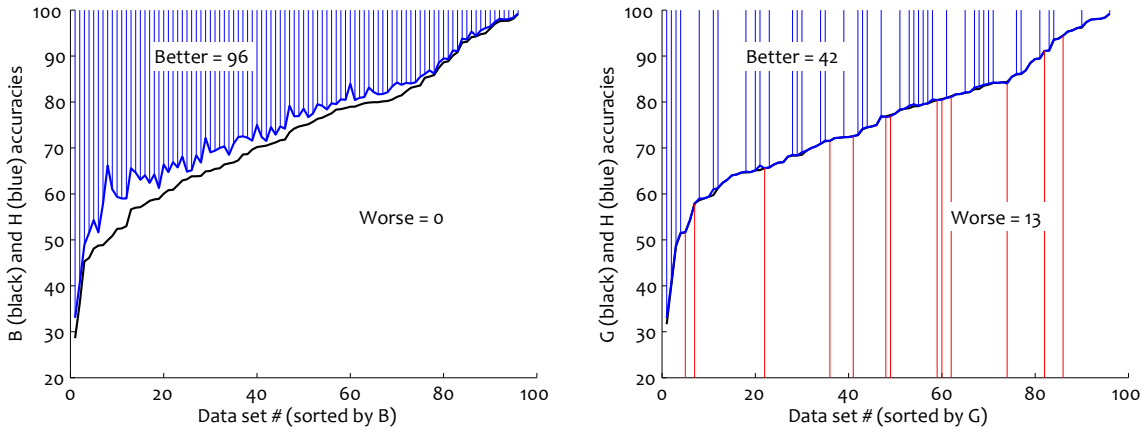
3.2.3. Results

We have prepared a supplementary document which contains the full numerical results from the experiments in seven tables, one for each classifier. Here we reproduce only the table for the decision tree classifier (Table 4).

The best and the worst base classifiers. We next give a graphical illustration of the two methods which appeared to be the weakest and the strongest in our experiment: Naïve Bayes (NB, the weakest) and Rotation Forest ensemble (ROT, the strongest). Figure 5 (a) shows the improvement on A_P of D^i_{set} achieved by applying D^h_{set} to the restricted set classification problem. To prepare the plot, we arranged the datasets in ascending order of $A_P(D^i_{\text{set}})$. Then we plotted $A_P(D^i_{\text{set}})$ and $A_P(D^h_{\text{set}})$ versus the dataset index. The differences between the two curves are small and not clearly visible, especially for larger values of $A_P(D^i_{\text{set}})$. Therefore, in order to show the consistency of the improvement, we drew a vertical line upwards from each point (dataset) where the strict inequality $A_P(D^i_{\text{set}}) < A_P(D^h_{\text{set}})$ held. Shown above the curves is the number of datasets out of 96 which satisfy the inequality. For NB, the partial accuracy $A_P(D^h_{\text{set}})$ was higher than $A_P(D^i_{\text{set}})$ for all 96 datasets.

In the same way, we plot together the curves for $A_P(D^g_{\text{set}})$ and $A_P(D^h_{\text{set}})$, this time sorting the datasets on $A_P(D^g_{\text{set}})$. The graph is shown in Figure 5 (b). This time, there were datasets for which the opposite strict inequality held, that is, $A_P(D^g_{\text{set}}) > A_P(D^h_{\text{set}})$. For these datasets, we drew the vertical lines downward, and show the number of datasets under the curves. For this case, Method H was better than method G in 42 comparisons and worse in 13 comparisons, leaving 41 ties. We can apply the sign statistical test whose p-value signifies reflects the probability that such difference may happen by chance if the two methods are, in fact, equivalent. For the results plotted in Figure 5 (b), $p = 6.66 \times 10^{-5}$, suggesting that $A_P(D^h_{\text{set}})$ is significantly better than $A_P(D^g_{\text{set}})$.

Figure 5 suggests that, even though A_P is not as high as for the remaining baseline classifiers D, the set classifier is able to improve on it consistently. Moreover, the Hungarian algorithm should be preferred to the greedy algorithm for its, albeit small, provably superior accuracy.



(a) (b)

Figure 5: Comparison between the partial accuracy A_P of: (a) D^i_{set} and D^h_{set}, and (b) D^g_{set} and D^h_{set} for the **Naïve Bayes** classifier and USC data collection. Upward vertical lines indicate that method H is strictly better than its counterpart for the respective dataset, and downward vertical line indicate that it is strictly worse. The numbers of datasets which satisfy the inequalities are shown in the respective parts of the plot.

Figure 6 is a matching example to Figure 5 where the based classifier is the Rotation Forest ensemble, the best overall set classifier. Method H is again making a "clean sweep" against I. This time, however, the number of results where method H is worse than method G rose to 38 versus 55 comparisons in favour of method H. This gives $p = 0.0180$ for the *two-tailed* sign test, which still supports the claim that there is significant difference in favour of method H at $p < 0.01$.

The scaling of the figures was kept the same to allow for a visual comparison of the accuracies. Clearly, ROT leads to better A_P than NB as all curves run higher in Figure 6. The improvement on A_P, however, is more pronounced for NB. Part of this may be due to the fact that NB is meant to approximate posterior probabilities (under the feature independence assumption), and gives a ready-made matrix with probabilities P. Rotation Forest, on the other hand, uses the voting scores as approximation of the posterior probabilities, which is not ideal, and may compromise the expected improvement on the accuracy of the independent set classifier I,

Figure 7 contains all the results for the *total* accuracy A_T whereby we require that all elements of X are classified correctly (correct super-label). The p-values for the results in subplots (b) and (d) are respectively 0.0820 and 0.4484. If we consider the right-tailed test with null hypothesis about the medians $G \geq H$, and alternative $H > G$, then the p-value for subplot (b) is 0.0410, hence method H is better than method G at $p < 0.05$.

Statistical testing for all results. Table 3 presents a *summary* of all the results. Given the large number of datasets, we considered it reasonable to display the average accuracies to support the statistical test

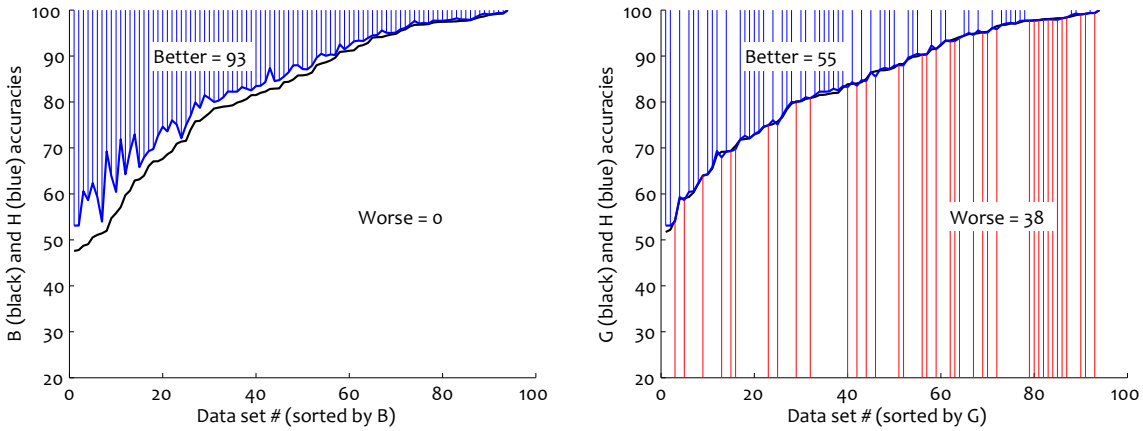


Figure 6: Comparison between the partial accuracy A_P of: (a) D^i_{set} and D^h_{set}, and (b) D^g_{set} and D^h_{set} for the **Rotation Forest** ensemble and USC data collection. Upward vertical lines indicate that method H is strictly better than its counterpart for the respective dataset, and downward vertical line indicate that it is strictly worse. The numbers of datasets which satisfy the inequalities are shown in the respective parts of the plot.

results. Note that we still used the non-parametric sign test for the comparison. The results support the claim that the set classifier which uses the Hungarian assignment algorithm is consistently better than the two rival algorithms. Admittedly, while consistent, the improvement over method G is fairly small, as evidenced by Figures 5 – 7, where the curves for G and H visibly coincide. This raises up the question of whether the Hungarian algorithm is really needed for this task or satisfactory results can be obtained with the greedy algorithm?

Note that this paper is about the definition of the restricted set classification problem. The crucial part of the proposed solution is the expansion of the probability matrix P. This can be followed by the optimal assignment algorithm (H) or a good suboptimal assignment algorithm (G).

Analysis of the improvement of H over B. Figure 8 illustrates the improvement offered by method H over the baseline set classifier B for both partial accuracy A_P and total accuracy A_T. For this example we used all the results for the seven base classifiers. Each point corresponds to a dataset, so there are $96 \times 7 = 672$ points scattered in total in each sub-plot. The diagonal line is where the points should lie if H and B had identical accuracies. The figure shows that, for both A_P and A_T, H is better than B. Interestingly, the improvement on A_P (sub-plot (a)) depends on the base accuracy while the improvement on A_T (sub-plot (b)) is more uniform on the total accuracy of B. The set classifier can correct individual errors better when the base classifier is not very accurate, and less so when the individual accuracy increases. This tendency exists but is less pronounced for the total accuracy A_T.

In Figure 9 we examine the relationship between the improvement on the partial and the total accuracy. The 672 points obtained from all seven classifier models and all datasets are scattered in the space

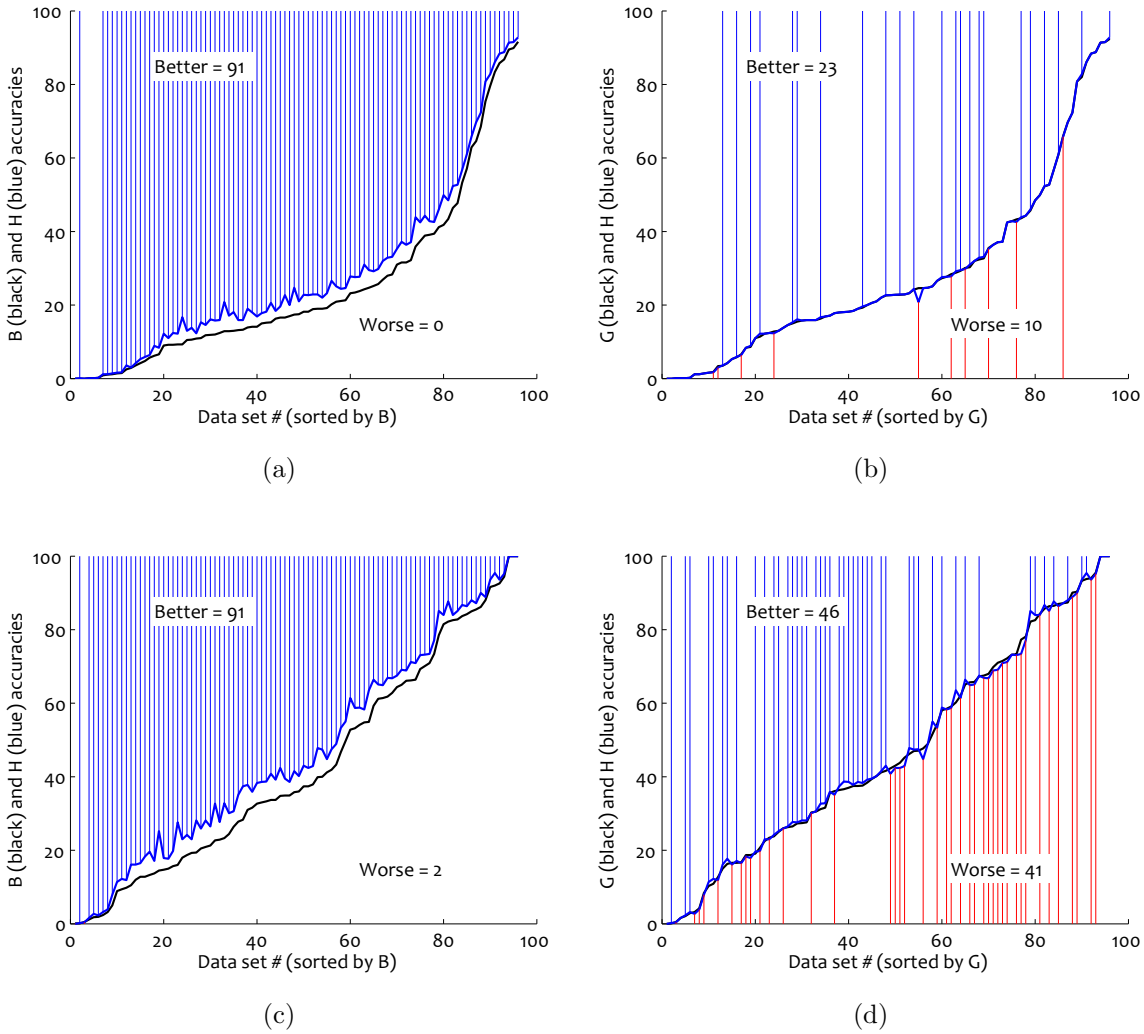
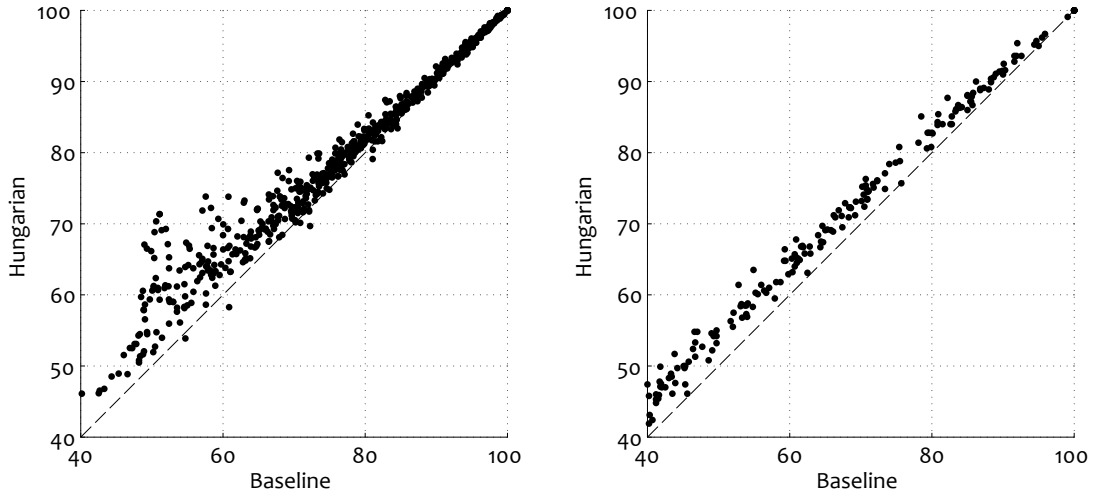


Figure 7: Comparison between the total accuracy A_T of: (a) method H versus I, (b) method H versus method G for the **Naïve Bayes** classifier; (c) method H versus I, (d) method H versus method G for the **Rotation Forest** ensemble, all for the USC data collection. Upward vertical lines indicate that method H is strictly better than its counterpart for the respective dataset, and downward vertical line indicate that it is strictly worse. The numbers of datasets which satisfy the inequalities are shown in the respective parts of the plot.

Table 3: Average partial accuracy A_P and total accuracy A_T for the three set classifiers (B Baseline, G Greedy and H Hungarian) for seven base classifier models with the USC data collection. Method H was compared with I and G using the sign test at significance level $p < 0.001$. The results are shown next to the H columns. The first symbol is the result of the H/I comparison, and the second symbol, the result from the H/G comparison. A bullet indicates that H is significantly better that the other set classifier, and a dash indicates that there is no difference at the chosen significance level.

	Partial accuracy A_P			Total accuracy A_T		
Classifier	B	G	H	B	G	H
Nearest neighbour (1-NN)	76.05	77.57	78.14 • •	30.03	32.55	33.45 • •
Linear discriminant analysis (LDA)	73.74	76.86	76.99 • •	27.67	30.97	31.06 • •
Naïve Bayes (NB)	72.83	75.86	75.95 • •	25.10	28.50	28.51 • –
Logistic Regression (LOG)	75.13	78.43	78.74 • •	29.74	33.65	33.95 • •
Decision tree (DT)	77.81	79.85	80.27 • •	32.63	35.60	36.29 • •
Random Forest (RF)	81.25	83.57	83.62 • •	40.16	44.03	44.14 • •
Rotation Forest (ROT)	82.09	84.69	84.79 • –	42.80	47.05	47.14 • –



(a) Partial accuracy A_P (b) Total accuracy A_T

Figure 8: Improvement of method H over method B for the 96 USC datasets using the all 7 base classifiers.

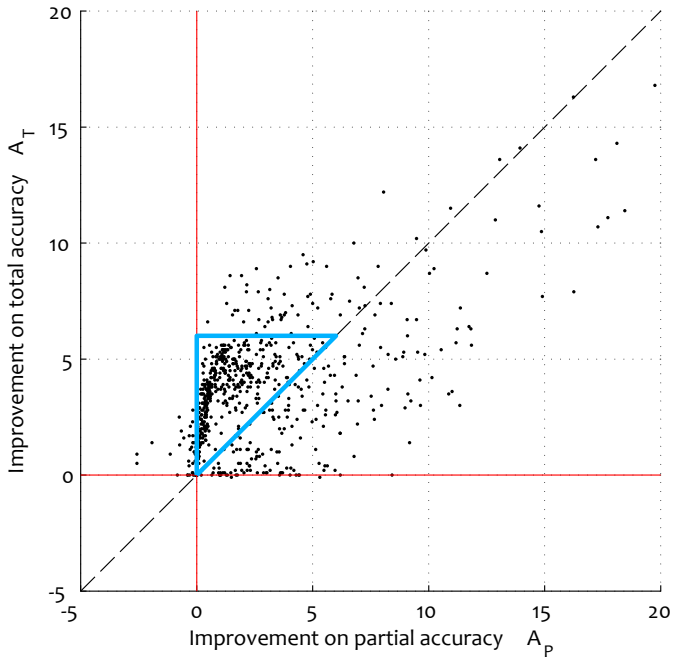


Figure 9: Scatterplot of all datasets for all seven base classifiers in the plane $(\Delta A_P, \Delta A_T)$. The blue triangle contains 54% of all points

$(\Delta A_P, \Delta A_T)$ where $\Delta A_P = A_P(D^h_{\text{set}}) - A_P(D^i_{\text{set}})$ and $\Delta A_T = A_T(D^h_{\text{set}}) - A_T(D^i_{\text{set}})$. The lines of no improvement are depicted in red. The figure shows that the *total* accuracy benefits even more than the *partial* accuracy (68.6% of the points are above the diagonal line, that is $\Delta A_T > \Delta A_P$). We identified and outlined by a triangle a dense region in the diagram containing approximately 54% of all points plotted. For these points, a fairly small improvement on the partial accuracy was sufficient to ensure a greater improvement in the total accuracy, justifying overall the restricted set classification approach. Larger improvements on A_P lead to larger improvement on A_T too, but for most of these points, $\Delta A_P > \Delta A_T$ (that is, the points lie under the diagonal).

Another interesting observation from the figure is that even for datasets where the partial accuracy of method H was worse than that of method method B (left from the line $\Delta A_P = 0$), the *total* accuracy of method H was still better than that of method B ($\Delta A_T > 0$). Overall, $\Delta A_T > 0$ for 93.75% of the points, $\Delta A_T = 0$ for 5.95% of the points, and $\Delta A_T < 0$ for 0.3% of the points. Note that this set of points includes all seven base classifiers and all datasets.

Relationship of the improvement of H over B and the number of classes.. Figure 10 plots the improvement ΔA_P and ΔA_T versus the number of classes. The points in the scatterplots again correspond to the datasets using all seven base classifiers.

There is no visible relationship between the number of classes and the improvement of method H against

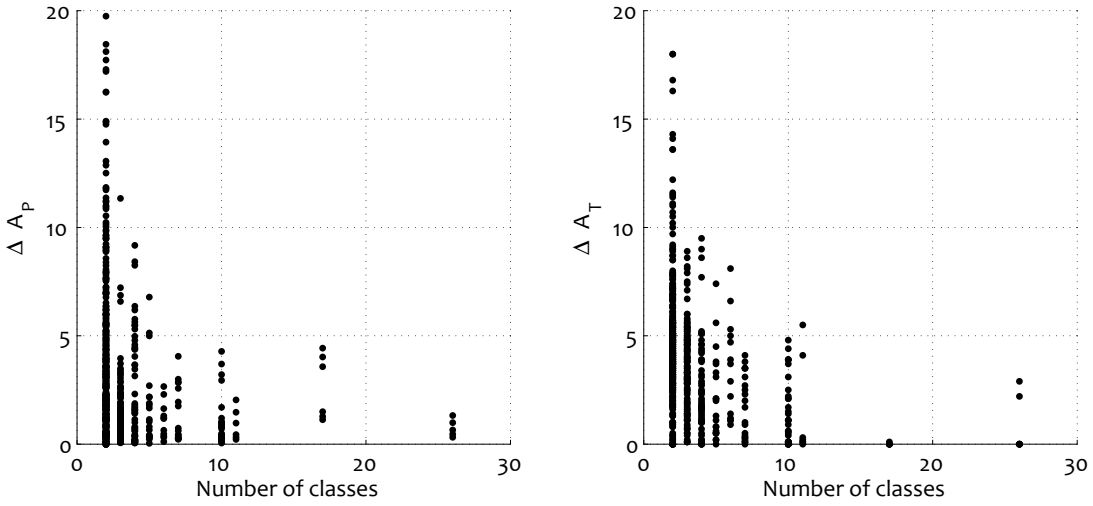


Figure 10: Improvement of H over B for the partial accuracy (ΔA_P) and the total accuracy ΔA_T) against the number of classes for the respective problems.

method B. The problems with two and three classes make up 75% of the USC collection. It can be expected that more is to be gained if we have a large number of classes compared to fewer classes, but this could not be verified with this collection.

4. Conclusion

We propose a solution for the *restricted set classification problem* which we also call "who-is-there". The problem is to classify simultaneously a set of objects into c classes, $\omega_1, \ldots, \omega_c$, knowing that there is at most k_i objects from class ω_i. The values of k_i are specified in advance and fixed. Our solution is to expand the matrix with posterior probabilities given by the classifier so that we cover the possibility for $k = \sum_{i=1}^{c} k_i$ objects and labels, and apply the Hungarian assignment algorithm on the logarithms of the posterior probabilities in the expanded matrix. The simpler alternatives which we considered were the standard approach of applying the trained classifier D individually to each instance in the set X, and a greedy approach where objects and classes are paired and eliminated from the set. Our experimental study validates the proposed approach for various choices of base classifier model D.

We demonstrated that the set classifier is better than the individual classifier for the restricted set classification problem through a two-part experiment. In the first part, we formulated the problem on a chess dataset where the all pieces on a chessboard had to be recognised from a bird-view snapshot. The second experiment was carried out using the University of Santiago de Compostela collection of 96 data. We constructed restricted set classification problems and applied the proposed solution. The experiments

favoured ensemble classifiers as the base classifier D, and subsequently the Hungarian set classifier D^h_{set} on the expanded probability matrix.

The scalability of the proposed framework is affected only by the complexity of the Hungarian algorithm. Training of the base classifier D is the same with or without the restricted set classification framework. The complexity of the Hungarian algorithm for our case is $O(k^3)$, where k is the maximum number of objects in the set. The Hungarian algorithm may be impractical for very large k or when rapid classification is needed, for example in tracking. In such cases, the greedy set classifier can be used, sacrificing some accuracy A_P and A_T.

An interesting question which we intend to address in the future is that about possible dependencies in the set of objects X. The instances in X may not be independently drawn from their respective classes. For example, the snapshot of the chessboard has a certain illumination. When classifying the pieces in the individual squares cropped from the snapshot, we assume that they are drawn independently from their respective classes. In other words, the fact that these sub-images have similar illumination is not currently accounted for in the theoretical model we propose here.

Acknowledgements

This work was partially supported by the Spanish Ministry of Economy and Competitiveness through project TIN 2015-67534-P.

Appendix

Proof of Proposition 1

Proposition 1. For 2-class problems,

$$A_P(D^g_{\text{set}}) > A_P(D^i_{\text{set}}) \, . \tag{14}$$

Proof. For D^i_{set},

$$\begin{aligned} A_P(D^i_{\text{set}}) &= Pr(P_1 > 0.5)Pr(P_2 > 0.5) \times \frac{2}{2} \\ &+ Pr(P_1 > 0.5)Pr(P_2 \leq 0.5) \times \frac{1}{2} \\ &+ Pr(P_1 \leq 0.5)Pr(P_2 > 0.5) \times \frac{1}{2} \qquad (15) \\ &= \frac{1}{2}\left(Pr(P_1 > 0.5) + Pr(P_2 > 0.5)\right) \qquad (16) \\ &= p \, , \qquad (17) \end{aligned}$$

where p is the accuracy of D. For D^g_{set},

$$A_P(D^g_{\text{set}}) = Pr(P_1 > 0.5)Pr(P_2 > 0.5) \times \frac{2}{2}$$

$$+Pr(P_1 \leq 0.5)Pr(P_2 > 0.5)Pr(P_2 > 1 - P_1) \times \frac{2}{2}$$

$$+Pr(P_1 > 0.5)Pr(P_2 \leq 0.5)Pr(P_1 > 1 - P_2) \times \frac{2}{2} \; . \tag{18}$$

The ROC curve of a classifier for a two-class problem is constructed by nominating any of the two classes to be the 'positive' class and the other to be the 'negative' class. The area under the ROC curve, AUC, gives the probability that the classifier will rank a randomly chosen positive instance higher than randomly chosen negative instance [39]. Phrased differently, this is the probability that the classifier will make errors with less certainty compared to the certainty when assigning a correct label. Formally,

$$AUC = Pr(P_1 > 1 - P_2) = Pr(P_2 > 1 - P_1) \; . \tag{19}$$

Denote by $S_1 = Pr(P_1 > 0.5)$ the sensitivity of D assuming that class ω_1 is the positive class, and by $S_2 = Pr(P_2 > 0.5)$ the sensitivity of D assuming that class ω_2 is the positive class. Then

$$A_P(D^g_{\text{set}}) = S_1S_2 + (1 - S_1)S_2AUC + S_1(1 - S_2)AUC$$

$$= S_1S_2 + (S_1 + S_2 - 2S_1S_2)AUC \; . \tag{20}$$

For the Bayes classifier D, whereby the labelling is done by the largest posterior probability, $AUC > 0.5$. Therefore,

$$A_P(D^g_{\text{set}}) \; > \; S_1S_2 + (S_1 + S_2 - 2S_1S_2) \times 0.5$$

$$= \frac{1}{2}(S_1 + S_2) = p = A_P(D^i_{\text{set}}) \; . \tag{21}$$

■

Results for the USC data for the decision tree classifier

Table 4: Averaged partial accuracy A_P and averaged total accuracy A_T in % for the Baseline (B), Greedy (G) and the Hungarian (H) set classifiers for the **decision tree** classifier. Symbol • after the value in column H indicates that H is significantly different from G at $p < 0.01$. The highest accuracy for each dataset is indicated in boldface.

	A_P			A_T		
Classifier	B	G	H	B	G	H
abalone	61.01	63.06	**63.23** –	6.50	**7.60**	**7.60** –
acute-inflammation	**100.00**	**100.00**	**100.00** –	**100.00**	**100.00**	**100.00** –
acute-nephritis	**100.00**	**100.00**	**100.00** –	**100.00**	**100.00**	**100.00** –
adult	76.34	79.90	**79.95** –	28.10	34.80	**35.00** –
annealing	90.64	91.60	**91.99** –	52.10	56.10	**57.50** –
arrhythmia	82.84	84.79	**85.41** –	41.70	46.40	**47.80** –
balance-scale	85.11	85.52	**85.52** –	43.50	45.90	**46.10** –
bank	64.79	71.92	**71.94** –	18.60	23.90	**24.70** –
blood	60.04	68.82	**69.92** –	15.90	23.10	**25.60** –
breast-cancer-wisc-diag	92.85	92.61	**92.95** –	64.70	66.50	**67.40** –
breast-cancer-wisc	94.26	**94.34**	94.32 –	70.50	**72.50**	72.40 –
breast-cancer	58.66	65.34	**66.58** –	11.90	17.50	**19.30** –
car	92.04	92.58	**92.78** –	46.70	50.20	**51.30** –
cardiotocography-10clases	74.96	**74.96**	74.77 –	0.50	**0.60**	**0.60** –
cardiotocography-3clases	85.53	**86.14**	86.09 –	34.10	**37.60**	37.40 –
chess-krvk	73.36	74.25	**74.48** •	0.00	**0.10**	**0.10** –
chess-krvkp	98.99	99.04	**99.09** –	94.70	95.30	**95.70** –
congressional-voting	48.95	62.61	**67.07** •	11.80	21.10	**26.10** •
conn-bench-sonar-mines-ro	69.99	72.29	**72.30** –	19.70	23.10	**23.50** –
conn-bench-vowel-deterdin	81.38	81.22	**81.60** •	1.70	1.80	**1.90** –
connect-4	74.75	77.78	**77.79** –	24.70	29.00	**29.10** –
contrac	48.18	50.17	**50.79** •	2.80	**2.90**	**2.90** –
credit-approval	84.22	84.32	**84.32** –	37.90	**40.80**	**40.80** –
cylinder-bands	68.71	70.59	**70.67** –	20.20	**23.00**	22.90 –
dermatology	96.86	96.81	**96.93** –	73.40	74.70	**74.90** –
ecoli	91.02	91.01	**92.10** •	51.70	53.20	**56.30** •
energy-y1	93.55	93.90	**94.11** –	59.20	63.40	**64.80** –
energy-y2	88.81	89.57	**89.83** –	39.90	44.20	**45.00** –
glass	73.44	75.24	**75.70** –	23.10	26.80	**27.90** –
haberman-survival	60.01	66.97	**68.40** –	14.00	19.30	**21.40** •
hayes-roth	61.75	65.71	**66.39** –	15.50	19.50	**20.30** –
heart-cleveland	68.21	73.01	**73.63** •	19.70	24.40	**25.40** •
heart-hungarian	76.13	78.80	**79.00** –	26.80	31.20	**31.90** –
heart-va	54.55	58.22	**58.32** –	9.40	11.10	**11.40** –
hill-valley	50.59	65.25	**70.34** •	14.90	26.10	**31.70** •
horse-colic	81.21	82.36	**82.58** –	34.40	38.00	**38.50** –
ilpd-indian-liver	52.41	63.11	**65.28** –	11.60	19.70	**22.60** –
image-segmentation	**93.57**	93.40	93.39 –	29.90	32.00	**32.20** –
ionosphere	**83.95**	83.71	83.78 –	40.70	**42.40**	**42.40** –
iris	**92.99**	92.73	92.97 –	60.40	62.20	**63.20** –
led-display	70.21	**70.25**	70.20 –	**0.10**	**0.10**	**0.10** –

Continued on next page

Table 4 – continued from previous page

	A_P				A_T			
Classifier	B	G	H		B	G	H	
letter	**84.33**	84.04	84.03	–	**0.00**	**0.00**	**0.00**	–
low-res-spect	80.94	81.65	**81.76**	–	23.60	26.40	**27.20**	–
lymphography	75.82	78.33	**78.61**	–	27.30	32.00	**33.10**	–
magic	82.35	**83.81**	83.78	–	38.60	**42.80**	**42.80**	–
mammographic	83.16	84.03	**84.27**	–	37.10	41.50	**42.20**	•
miniboone	87.53	**87.72**	**87.72**	–	48.60	**50.80**	**50.80**	–
molec-biol-promoter	74.64	76.13	**76.48**	–	28.10	31.00	**31.90**	–
molec-biol-splice	**89.73**	89.55	89.45	–	45.30	**47.40**	**47.40**	–
monks-1	83.45	86.07	**87.22**	–	47.00	52.70	**54.80**	–
monks-2	58.44	67.65	**69.38**	–	13.60	22.70	**25.10**	–
monks-3	96.97	97.26	**97.32**	–	83.30	85.60	**85.80**	–
mushroom	**100.00**	**100.00**	**100.00**	–	**100.00**	**100.00**	**100.00**	–
musk-1	76.64	77.30	**77.79**	–	27.80	31.20	**32.30**	–
musk-2	90.16	90.58	**90.64**	–	56.70	60.20	**60.30**	–
nursery	98.36	98.54	**98.57**	–	83.90	86.10	**86.30**	–
oocytes-merluccius-nucleu	64.59	68.07	**68.11**	–	16.20	19.10	**19.60**	–
oocytes-merluccius-states	**83.63**	83.30	83.31	–	28.90	30.30	**30.60**	–
oocytes-trisopterus-nucle	69.67	71.35	**71.48**	–	18.30	20.10	**20.30**	–
oocytes-trisopterus-state	85.77	86.15	**86.19**	–	43.90	47.20	**47.60**	–
optical	**88.81**	88.44	88.47	–	5.70	6.20	**6.30**	–
ozone	57.82	66.25	**66.89**	–	16.00	21.60	**22.70**	–
page-blocks	84.40	85.49	**86.00**	•	26.20	29.30	**30.10**	–
pendigits	**95.13**	94.83	94.84	–	27.90	**29.10**	29.00	–
pima	69.17	72.39	**72.77**	–	20.20	24.50	**25.10**	–
planning	51.08	65.87	**71.35**	•	12.80	25.40	**30.80**	•
ringnorm	89.81	90.05	**90.18**	–	53.90	**57.30**	**57.30**	–
seeds	90.40	**90.55**	90.49	–	49.10	51.80	**52.20**	–
semeion	**72.11**	71.62	71.73	–	**0.10**	**0.10**	**0.10**	–
soybean	86.94	87.47	**87.94**	•	28.10	30.90	**32.50**	•
spambase	90.87	**91.19**	91.14	–	58.10	61.70	**61.80**	–
spect	65.84	68.54	**69.22**	–	15.50	18.90	**19.60**	–
spectf	70.13	**71.11**	70.51	–	21.70	23.30	**23.70**	–
statlog-australian-credit	54.57	57.59	**58.14**	–	10.30	11.80	**12.30**	–
statlog-german-credit	62.43	**66.31**	66.25	–	13.80	**16.60**	**16.60**	–
statlog-heart	74.53	**76.02**	75.77	–	24.10	27.20	**27.30**	–
statlog-image	**95.05**	94.96	95.02	–	40.20	41.80	**41.90**	–
statlog-landsat	**82.25**	82.04	82.19	–	8.60	9.10	**9.50**	–
statlog-shuttle	98.15	**98.25**	98.18	–	79.30	**81.00**	80.60	–
statlog-vehicle	70.48	71.81	**72.00**	–	7.30	8.40	**8.60**	–
steel-plates	71.98	72.57	**72.72**	–	0.80	**1.10**	**1.10**	–
synthetic-control	**87.35**	86.84	87.24	–	14.20	15.00	**15.60**	–
teaching	58.33	62.62	**62.83**	–	9.60	12.70	**13.30**	–
thyroid	**97.59**	97.57	97.55	–	82.80	**84.20**	84.00	–
tic-tac-toe	91.11	**91.69**	91.64	–	61.00	**64.60**	64.50	–
titanic	68.37	75.28	**76.43**	•	22.40	32.50	**34.60**	•
twonorm	84.40	84.54	**84.62**	–	40.30	42.90	**43.10**	–
vertebral-column-2clases	76.37	78.38	**78.63**	–	29.30	33.30	**33.70**	–
vertebral-column-3clases	74.11	**75.90**	75.77	–	16.70	19.50	**19.60**	–
wall-following	99.13	99.13	**99.18**	•	90.00	90.50	**91.00**	–

Continued on next page

Table 4 – continued from previous page

	A_P			A_T		
Classifier	B	G	H	B	G	H
waveform-noise	74.58	74.70	**74.97** –	14.20	15.30	**15.70** –
waveform	76.16	76.40	**76.68** –	17.10	18.20	**18.40** –
wine-quality-red	42.66	46.18	**46.54** –	1.00	**1.30**	**1.30** –
wine-quality-white	42.51	44.73	**46.16** •	0.20	0.20	**0.30** –
wine	94.15	94.03	**94.48** –	71.80	73.60	**75.10** •
yeast	57.58	58.41	**58.62** –	**1.20**	**1.20**	**1.20** –

References

[1] L. I. Kuncheva, Full-class set classification using the Hungarian algorithm, International Journal of Machine Learning and Cybernetics 1 (1) (2010) 53–61. doi:DOI10.1007/s13042-010-0002-z.

[2] L. I. Kuncheva, A. S. Jackson, Who is missing? A new pattern recognition puzzle, in: Intenrational Conference on Statistical, Structural and Syntactic Pattern Recognition (S+SSPR), Vol. LNCS 8621, Springer, Joensuu, Funland, 2014, pp. 243–252.

[3] R. O. Duda, P. E. Hart, D. G. Stork, Pattern Classification, 2nd Edition, John Wiley & Sons, NY, 2001.

[4] R. E. Slot, On the profit of taking into account the known number of objects per class in classification methods IT–25 (1979) 484–488.

[5] T. G. Dietterich, R. H. Lathrop, T. Lozano-Perez, Solving the multiple-instance problem with axis-parallel rectangles, Artificial Intelligence 89 (1997) 31–71.

[6] J. Wang, J.-D. Zucker, Solving the multiple-instance problem: A lazy learning approach, in: In Proceedings 17th International Conference on Machine Learning, 2000, pp. 1119–1125.

[7] O. L. Mangasarian, E. W. Wild, Multiple instance classification via successive linear programming, Journal of Optimization Theory and Applications 137 (2008) 555–568.

[8] Z. Fu, A. Robles-Kelly, J. Zhou, MILIS: multiple instance learning with instance selection, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (5) (2011) 958–977.

[9] X. Xu, Statistical learning in multiple instance problems, Master's thesis, University of Waikato, Department of Computer Science, New Zealand (2003).

[10] X. Ning, G. Karypis, The set classification problem and solution methods, in: In Proceedings of SIAM Data Mining, 2009, pp. 847–858.

[11] M. Pelillo, M. Refice, Learning compatibility coefficients for relaxation labeling processes, Pattern Analysis and Machine Intelligence, IEEE Transactions on 16 (9) (1994) 933–945.

[12] A. Rosenfeld, R. A. Hummel, S. Zucker, Scene labeling by relaxation operations, IEEE Transactions on Systems, Man and Cybernetics SMC-6 (6) (1976) 420–433.

[13] L. K. McDowell, K. M. Gupta, D. W. Aha, Cautious inference in collective classification, in: In Processdings of AAAI, 2007, pp. 596–601.

[14] P. Sen, G. M. Namata, Bilgic, L. Getoor, B. Gallagher, T. Eliassi-Rad, Collective classification in network data, AI Magazine 29 (2008) 93106.

[15] Y. Amit, A. Trouvé, POP: patchwork of parts models for object recognition, International Journal of Computer Vision 75 (2007) 267–282.

[16] R. Kaucic, A. G. A. Perera, G. B. J., Kaufhold, A. Hoogs, A unified framework for tracking through occlusions and across sensor gaps, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR, Vol. 1, 2005, pp. 1063–1069.

[17] I. Haritaoglu, D. Harwood, L. S. Davis, W 4: Real-time surveillance of people and their activities, Pattern Analysis and Machine Intelligence, IEEE Transactions on 22 (8) (2000) 809–830.

[18] A. Mohan, C. Papageorgiou, T. Poggio, Example-based object detection in images by components, Pattern Analysis and Machine Intelligence, IEEE Transactions on 23 (4) (2001) 349–361.

[19] P. F. Felzenszwalb, D. P. Huttenlocher, Pictorial structures for object recognition, International Journal of Computer Vision 61 (1) (2005) 55–79.

[20] B. Heisele, T. Serre, M. Pontil, T. Poggio, Component-based face detection, in: Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, Vol. 1, IEEE, 2001, pp. I–657.

[21] T. Zhao, R. Nevatia, Tracking multiple humans in complex situations, Pattern Analysis and Machine Intelligence, IEEE Transactions on 26 (9) (2004) 1208–1221.

[22] X. Zhou, Y. Li, B. He, Game-theoretical occlusion handling for multi-target visual tracking, Pattern Recognition 46 (10) (2013) 2670–2684.

[23] C. Rasmussen, G. D. Hager, Probabilistic data association methods for tracking complex visual objects, Pattern Analysis and Machine Intelligence, IEEE Transactions on 23 (6) (2001) 560–576.

[24] Y. Bar-Shalom, F. Daum, J. Huang, The probabilistic data association filter, Control Systems, IEEE 29 (6) (2009) 82–100.

[25] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, D. G. Lowe, A boosted particle filter: Multitarget detection and tracking, in: Computer Vision-ECCV 2004, Springer, 2004, pp. 28–39.

[26] A. Dearden, Y. Demiris, O. Grau, Tracking football player movement from a single moving camera using particle filters, in: Proceedings of CVMP-2006, IET Press, 2006, pp. 29–37.

[27] P. Konstantinova, A. Udvarev, T. Semerdjiev, A study of a target tracking algorithm using global nearest neighbor approach, in: Proceedings of the International Conference on Computer Systems and Technologies (CompSysTech03), 2003.

[28] R. Ardekani, A. Biyani, J. E. Dalton, J. B. Saltz, M. N. Arbeitman, J. Tower, S. Nuzhdin, S. Tavaré, Three-dimensional tracking and behaviour monitoring of multiple fruit flies, Journal of The Royal Society Interface (2012) rsif20120547.

[29] J. N. Jover, M. Alcañiz-Raya, V. Gómez, S. Balasch, J. Moreno, V. G. Colomer, A. Torres, An automatic colour-based computer vision algorithm for tracking the position of piglets, Spanish Journal of Agricultural Research 7 (3) (2009) 535–549.

[30] M. Kashiha, C. Bahr, S. Ott, C. P. Moons, T. A. Niewold, F. O. Ödberg, D. Berckmans, Automatic identification of marked pigs in a pen using image pattern recognition, Computers and Electronics in Agriculture 93 (2013) 111–120.

[31] Y. Kamisugi, N. Furuya, K. Iijima, K. Fukui, Computer-aided automatic identification of rice chromosomes by image parameters 1 (3) (1993) 189–196.

[32] D. Ming, J. Tian, Automatic pattern extraction and classification for chromosome images, Journal of Infrared Milli Terahz Waves 31 (2010) 866–877.

[33] X. Wang, S. Li, H. Liu, M. Wood, W. R. Chen, B. Zheng, Automated identification of analyzable metaphase chromosomes depicted on microscopic digital images, J Biomed Inform 41 (2) (2008) 264–271.

[34] H. W. Kuhn, The Hungarian Method for the assignment problem, Naval Research Logistic Quarterly 2 (1955) 83–97.

[35] F. Bourgeois, J.-C. Lassalle, An extension of the Munkres algorithm for the assignment problem to rectangular matrices, Communications ACM 14 (12) (1971) 802–804.

[36] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (2011) 27:1–27:27, software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[37] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems?, Journal of Machine Learning Research 15 (2014) 3133–3181.

[38] K. Bache, M. Lichman, UCI machine learning repository (2013).

URL http://archive.ics.uci.edu/ml

[39] T. Fawcett, ROC graphs: Notes and practical considerations for researchers, Tech. Rep. HPL-2003-4, HP Labs, Palo Alto, http://www.hpl.hp.com/techreports/2003/HPL-2003-4.pdf (2003).

Vitae

Ludmila I. Kuncheva is a Professor of Computer Science at Bangor University, UK. Her interests include pattern recognition, and specifically classifier ensembles. She has published two monographs and over 150 research papers. Lucy has won two Best Paper Awards (2006 IEEE TFS and 2003 IEEE TSMC.) She is a Fellow of IAPR.

Juan J. Rodríguez is an Associate Professor of Computer Science at University of Burgos, Spain. His interests include data science, pattern recognition and specifically classifier ensembles.

Aaron S. Jackson is currently working toward his PhD degree in Computer Science at Nottingham University, UK. His interests include pattern recognition, deep learning and computer vision.