An Ordinal Data Method for the Classification with Reject Option

Ricardo Sousa, Beatriz Mora, Jaime S. Cardoso
INESC Porto, Faculdade Engenharia, Universidade Porto
Campus da FEUP, Rua Dr. Roberto Frias, n. 378
4200-465 Porto, Portugal
{rsousa,bbarbero,jaime.cardoso}@inescporto.pt

***Abstract*—In this work we consider the problem of binary classification where the classifier may abstain instead of classifying each observation, leaving the critical items for human evaluation. This article motivates and presents a novel method to learn the reject region on complex data. Observations are replicated and then a single binary classifier determines the decision plane. The proposed method is an extension of a method available in the literature for the classification of ordinal data. Our method is compared with standard techniques on synthetic and real datasets, emphasizing the advantages of the proposed approach.**

***Index Terms*—decision support systems, machine learning, reject option, svm**

I. Introduction

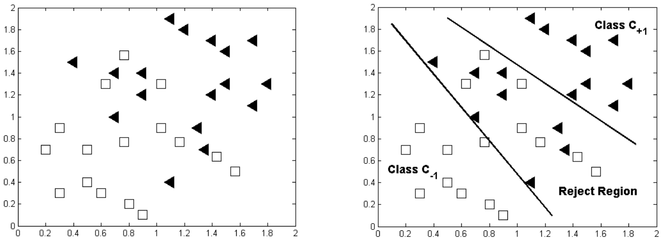
Decision support systems are becoming ubiquitous in many human activities, most notably in finance and medicine. Automatic models are being developed to imitate, as closely as possible, the usual human decision [1]. Within this context, classification is one of the most representative predictive learning tasks. Classification predicts a categorical value for a specific data item. The most well studied scenario is when the class to be predicted can assume only two values—binary setting. The classifier is developed to partition the feature space in two regions, discriminating between the two classes.

One of the problems with classifying complex items is that many items from distinct classes have similar structures in a feature space, resulting in a setting with overlapping classes. The automation of decisions in this region leads invariably to many wrong predictions. On the other hand, and although items in the historical data are labelled *only* as 'good' or 'bad', the deployment of a decision support system in many environments has the opportunity to label critical items for manual revision, instead of trying to automatically classify every and each item. The system automates only those decisions which can be reliably predicted, labelling the critical ones for a human expert to analyse. Therefore, the development of tripartite classifiers, with a third output class, the reject class, in-between the good and bad classes, is attractive.

II. Problem Statement and Standard Solutions

Predictive modelling tries to find good rules (models) for guessing (predicting) the values of one or more variables (target) in a dataset from the values of other variables. Our target can assume only two values, represented by 'good' and 'bad' classes. When in possession of a "complex" dataset, a simple separator is bound to misclassify some points. Two types of errors are possible, 'false positives' and 'false negatives'. The construction (training) of a model can be conducted to optimise some adopted measure of business performance, be it profit, loss, volume of acquisitions, market share, etc, by giving appropriate weights to the two types of errors. When the weights of the two types of errors are heavily asymmetric, the boundary between the two classes will be pushed near values where the most costly error seldom happens.

This fact suggests a simple procedure to construct a three-class output classifier: training a first binary classifier with a set of weights heavily penalising the false negative errors, we expect that when this classifier predicts an item as negative, it will be truly negative. Likewise, training a second binary classifier with a set of weights heavily penalising the false positive errors, we expect that when this classifier predicts an item as positive, it will be truly positive. When a item is predicted as positive by the first classifier and negative by the second, it will be labelled for review. This setting is illustrated in Figure 1. A problem arises when an item is predicted as positive by the first classifier and negative by the second classifier as in Figure 2(a). That can happen because the two separator lines intersect each other. A convenient workaround is then to avoid this problematic state by imposing that the two boundaries of the classifiers do not intersect, Figure 2(b).

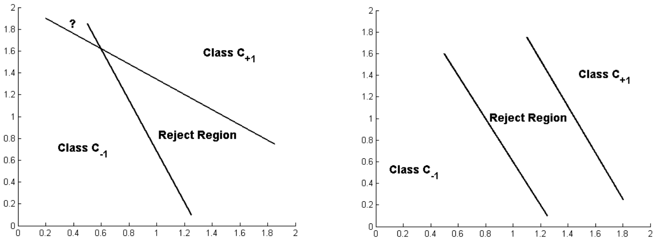


(a) Overlapping regions.

(b) Typical separator lines, obtained with two independent binary classifiers.

Fig. 1. Illustrative setting with overlap classes.

Before delving into the proposed method, it is worth discussing the simple solution of using a single classifier. If more



(a) Intersecting separating lines.

(b) Non-intersecting separating lines.

Fig. 2. Potential discriminative boundaries.

than just discriminating between the two classes, the model to use yields a posterior probability for each target class, then two cutoffs can be defined on this value. All items with predicted probability of belonging to class $\mathcal{C}_{-1}$ less than a low threshold are labelled as $\mathcal{C}_{+1}$, items with predicted probability of belonging to class $\mathcal{C}_{-1}$ higher than a high threshold are labelled as $\mathcal{C}_{-1}$, items with predicted probability of belonging to class $\mathcal{C}_{-1}$ in-between the low and high threshold are labelled for review. Two issues were identified with this approach. First, we need to estimate the probability of each class, which is by itself a problem harder than the problem of discriminating classes. Second, the estimation of the two cutoffs is not straightforward nor can be easily fitted in standard frameworks. The design of classifiers with reject option can be systematised in three different approaches:

- the design of two, *independent*, classifiers. A first classifier is trained to output $\mathcal{C}_{-1}$ only when the probability of $\mathcal{C}_{-1}$ is high and a second classifier trained to output $\mathcal{C}_{+1}$ only when the probability of $\mathcal{C}_{+1}$ is high. The simplicity of this strategy has the weakness of producing intersecting boundaries, leading to regions with a non-logical decision.
- the design of a single, standard binary classifier. This approach already provides non-intersecting boundaries. If the classifier provides some approximation to the a posterior class probabilities, then a pattern is rejected if the maximum of the two posterior probabilities is lower than a given threshold. If the classifier does not provide probabilistic outputs, then a rejection threshold targeted to the particular classifier is used. For example, the rejection techniques proposed with support vector machines consist in rejecting patterns those distance from the optimal separating hyperplane is lower than a predefined threshold. The rejection region is determined *after* the training of the classifier, by defining appropriate threshold values on the output of the classifier.
- the design of a single classifier with embedded reject option. This approach is consisted in the design of algorithms specifically adapted for this kind of problems [2], [3].

III. An Ordinal Data Approach for Detecting Reject Regions

The rejection method to be proposed is an extension of a method already proposed in the literature but for the classification of ordinal data. Therefore, and for completeness, we start by reviewing the data replication method; next, we present the novel aspects introduced in this article.

A. The Data Replication Method for Ordinal Data

The data replication method for ordinal data can be framed under the single binary classifier reduction (SBC), an approach for solving multiclass problems via binary classification relying on a single, standard binary classifier. SBC reductions can be obtained by embedding the original problem in a higher-dimensional space consisting of the original features, as well as one or more extension features. This embedding is implemented by replicating the training set points so that a copy of the original point is concatenated with each of the extension features' vectors. The binary labels of the replicated points are set to maintain a particular structure in the extended space. This construction results in an instance of an artificial binary problem, which is fed to a single binary learning algorithm. To classify a new point, the point is replicated and extended similarly and the resulting replicas are fed to the binary classifier, which generates a number of signals, one for each replica. The class is determined as a function of these signals [4].

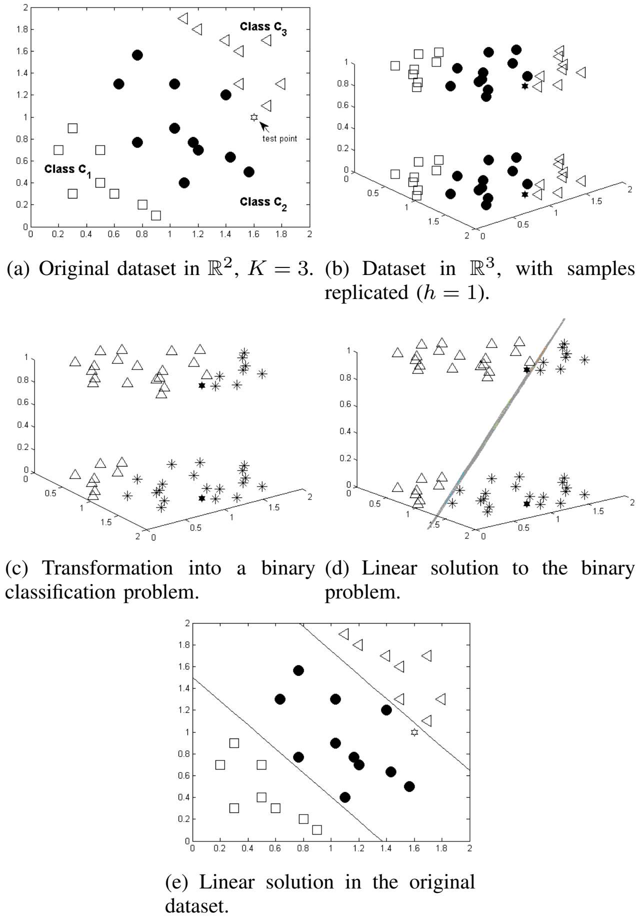
To present the data replication method, assume that examples in a classification problem come from one of K ordered classes, labelled from $\mathcal{C}_1$ to $\mathcal{C}_K$, corresponding to their natural order. Consider the training set $\{\mathbf{x}_i^{(k)}\}$, where $k = 1, \ldots, K$ denotes the class number, $i = 1, \ldots, \ell_k$ is the index within each class, and $\mathbf{x}_i^{(k)} \in \mathbb{R}^p$, with p the dimension of the feature space.

Let us consider a very simplified toy example with just three classes, as depicted in Figure 3(a). Here, the task is to find two parallel hyperplanes, the first one discriminating class $\mathcal{C}_1$ against classes $\{\mathcal{C}_2, \mathcal{C}_3\}$ and the second hyperplane discriminating classes $\{\mathcal{C}_1, \mathcal{C}_2\}$ against class $\mathcal{C}_3$. These hyperplanes will correspond to the solution of two binary classification problems but with the additional constraint of parallelism. The data replication method suggests solving both problems simultaneously in an augmented feature space [5].

In the toy example, using a transformation from the $\mathbb{R}^2$ initial feature-space to a $\mathbb{R}^3$ feature space, replicate each original point, according to the rule (see Figure 3(b)):

$$\begin{tikzcd}[row sep=tiny] & \left[\begin{smallmatrix}\mathbf{x}\\ h\end{smallmatrix}\right] \in \mathbb{R}^3 \\ \mathbf{x} \in \mathbb{R}^2 \arrow[ur] \arrow[dr] & & \text{, where } h = \text{const} \in \mathbb{R}^+ \\ & \left[\begin{smallmatrix}\mathbf{x}\\ 0\end{smallmatrix}\right] \in \mathbb{R}^3 \end{tikzcd}$$

Observe that any two points created from the same original point differ only in the extension feature. Define now a binary training set in the new (higher dimensional) space according



(a) Original dataset in $\mathbb{R}^2$, $K = 3$.

(b) Dataset in $\mathbb{R}^3$, with samples replicated ($h = 1$).

(c) Transformation into a binary classification problem.

(d) Linear solution to the binary problem.

(e) Linear solution in the original dataset.

Fig. 3. Data replication model in a toy example (from [5]).

to (see Figure 3(c)):

$$\begin{bmatrix}\mathbf{x}_i^{(1)}\\ 0\end{bmatrix} \in \overline{\mathcal{C}}_1,\ \begin{bmatrix}\mathbf{x}_i^{(2)}\\ 0\end{bmatrix}, \begin{bmatrix}\mathbf{x}_i^{(3)}\\ 0\end{bmatrix} \in \overline{\mathcal{C}}_2$$
$$\begin{bmatrix}\mathbf{x}_i^{(1)}\\ h\end{bmatrix}, \begin{bmatrix}\mathbf{x}_i^{(2)}\\ h\end{bmatrix} \in \overline{\mathcal{C}}_1,\ \begin{bmatrix}\mathbf{x}_i^{(3)}\\ h\end{bmatrix} \in \overline{\mathcal{C}}_2 \qquad (1)$$

In this step we are defining the two binary problems as a single binary problem in the augmented feature space. A linear two-class classifier can now be applied on the extended dataset, yielding a hyperplane separating the two classes, see Figure 3(d). The intersection of this hyperplane with each of the subspace replicas can be used to derive the boundaries in the original dataset, as illustrated in Figure 3(e).

To predict the class of an unseen example, classify both replicas of the example in the extended dataset with the binary classifier. From the sequence of binary labels one can infer the predicted label on the original ordinal classes

$$\overline{\mathcal{C}}_1, \overline{\mathcal{C}}_1 \Longrightarrow \mathcal{C}_1 \qquad \overline{\mathcal{C}}_2, \overline{\mathcal{C}}_1 \Longrightarrow \mathcal{C}_2 \qquad \overline{\mathcal{C}}_2, \overline{\mathcal{C}}_2 \Longrightarrow \mathcal{C}_3$$

Note that only three sequences are possible [5]. The generalisation for any problem in $\mathbb{R}^p$, with K ordinal classes and nonlinear boundaries can be found in [5].

Summing up, $(K-1)$ replicas in a $\mathbb{R}^{p+K-2}$ dimensional space are used to train a binary classifier. The target class of an unseen example can be obtained by adding one to the number of $\mathcal{C}_2$ labels in the sequence of binary labels resulting from the classification of the $(K-1)$ replicas of the example.

Replica #	points from $\mathcal{C}_1$	points from $\mathcal{C}_2$
1	$-1; C_\ell$	$+1; C_h$
2	$-1; C_h$	$+1; C_\ell$

TABLE I
LABELS AND COSTS (C_ℓ AND C_h REPRESENT A LOW AND A HIGH COST VALUE, RESPECTIVELY) FOR POINTS IN DIFFERENT REPLICAS IN THE EXTENDED DATASET.

B. The Data Replication Method for Detecting Reject Regions

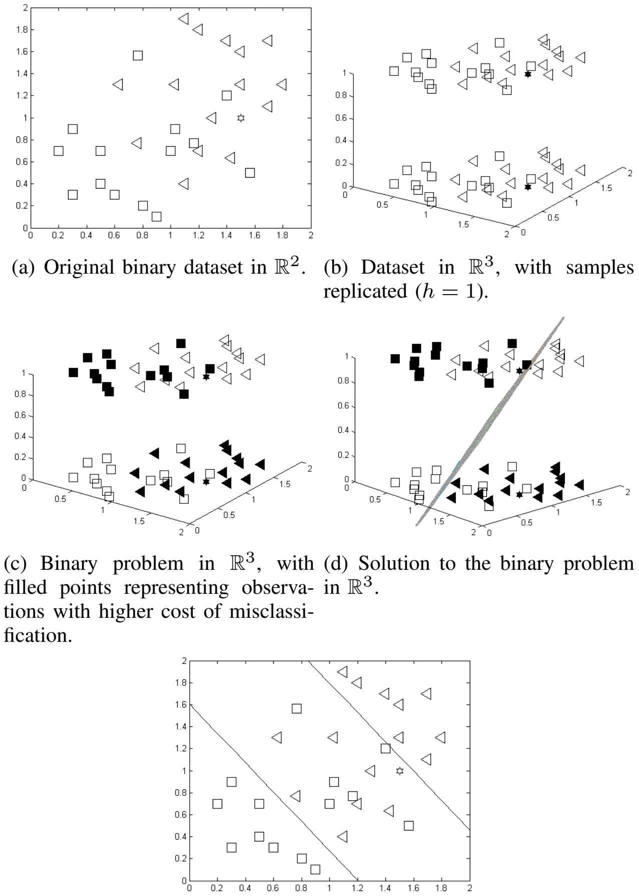
The scenario of designing a classifier with reject option shares many characteristics with the classification of ordinal data. It is also reasonable to assume for the reject option scenario that the three output classes are naturally ordered as $\mathcal{C}_1, \mathcal{C}_{reject}, \mathcal{C}_2$. As the intersection point of the two boundaries would indicate an example with the three classes equally probable—one would be equally uncertain between assigning $\mathcal{C}_1$ or $\mathcal{C}_{reject}$ and between assigning $\mathcal{C}_{reject}$ or $\mathcal{C}_2$—it is plausible to adopt a strategy imposing non-intersecting boundaries. In fact, as reviewed in Section II, methods have been proposed with exactly such assumption. In the scenario of designing a classifier with reject option, we are interested on finding two boundaries: a boundary discriminating $\mathcal{C}_1$ from $\{\mathcal{C}_{reject}, \mathcal{C}_2\}$ and a boundary discriminating $\{\mathcal{C}_1, \mathcal{C}_{reject}\}$ from $\mathcal{C}_2$.

We proceed exactly as in the data replication method for ordinal data. We start by transforming the data from the initial space to an extended space, replicating the data, according to the rule (see Figure 4(b)):

$$\mathbf{x} \in \mathbb{R}^d \begin{array}{l} \nearrow \begin{bmatrix}\mathbf{x}\\ h\end{bmatrix} \in \mathbb{R}^{d+1} \\ \searrow \begin{bmatrix}\mathbf{x}\\ 0\end{bmatrix} \in \mathbb{R}^{d+1} \end{array}, \quad \text{where } h = \text{const} \in \mathbb{R}^+$$

If we design a binary classifier on the extended training data, without further considerations, one would obtain the same classification boundary in both data replicas. Therefore, we modify the misclassification cost of the observations according to the data replica they belong to. In the first replica (the extension feature assumes the value zero), we will discriminate $\mathcal{C}_1$ from $\{\mathcal{C}_{reject}, \mathcal{C}_2\}$; therefore we give higher costs to observations belonging to class $\mathcal{C}_2$ than to observations belonging to class $\mathcal{C}_1$. This will bias the boundary towards the minimisation of errors in $\mathcal{C}_2$. In the second replica (the extension feature assumes the value h), we will discriminate $\{\mathcal{C}_1, \mathcal{C}_{reject}\}$ from $\mathcal{C}_2$; therefore we give higher costs to observations belonging to class $\mathcal{C}_1$ than to observations belonging to class $\mathcal{C}_2$. This will bias the boundary towards the minimisation of errors in $\mathcal{C}_1$. In Figure 4(c) this procedure is illustrated by filling the marks of the observations with higher costs. Table I summarises this procedure.

A two-class classifier can now be applied on the extended dataset, yielding a boundary separating the two classes, see Figure 4(d). The intersection of this boundary with each of the subspace replicas can be used to derive the boundaries in the original dataset, as illustrated in Figure 4(e).



(a) Original binary dataset in $\mathbb{R}^2$.

(b) Dataset in $\mathbb{R}^3$, with samples replicated ($h = 1$).

(c) Binary problem in $\mathbb{R}^3$, with filled points representing observations with higher cost of misclassification.

(d) Solution to the binary problem in $\mathbb{R}^3$.

(e) Solution with reject region in the original dataset.

Fig. 4. Proposed reject option model in a toy example.

Summing up, with a proper choice of costs, the data replication method can be used to learn a reject region, defined by two non-intersecting boundaries. Note that the reject region is optimised during training and not heuristically defined afterwards. Nonlinear (and non-intersecting) boundaries are treated exactly as the ordinal data scenario. Likewise, prediction follows the same rationale.

1) Selecting the Misclassification Costs: In the reject option scheme, one desires to obtain a minimum error while minimising the number of rejected cases. However, when the number of rejected cases decreases the classification error increases, and to decrease the classification error one typically has to increase the reject region. The right balance between these two conflicting goals depends on the relation of the associated costs.

Let $C_{i,q}^{(k)}$ represent the cost of erring a point $\mathbf{x}_i$ from class k in data replica q (or, equivalently, by hyperplane q). Points from class $\mathcal{C}_1$ misclassified by the hyperplane 1 ($\mathbf{w}^t\mathbf{x}+b_1 = 0$) but correctly classified by the second hyperplane ($\mathbf{w}^t\mathbf{x} + b_2 = 0$) incur in a loss $C_{i,1}^{(1)}$; points from class $\mathcal{C}_1$ misclassified by both hyperplanes incur in a loss $C_{i,1}^{(1)} + C_{i,2}^{(1)}$. Likewise, points from class $\mathcal{C}_2$ misclassified by the hyperplane 2 ($\mathbf{w}^t\mathbf{x}+b_2 = 0$) but correctly classified by the first hyperplane ($\mathbf{w}^t\mathbf{x} + b_1 = 0$) incur in a loss $C_{i,2}^{(2)}$; points from class $\mathcal{C}_2$ misclassified by both hyperplanes incur in a loss $C_{i,1}^{(2)} + C_{i,2}^{(2)}$. The resulting loss matrix is given by

		predicted		
		$\mathcal{C}_1$	$\mathcal{C}_{reject}$	$\mathcal{C}_2$
true	$\mathcal{C}_1$	0	$C_{i,1}^{(1)}$	$C_{i,1}^{(1)} + C_{i,2}^{(1)}$
	$\mathcal{C}_2$	$C_{i,1}^{(2)} + C_{i,2}^{(2)}$	$C_{i,2}^{(2)}$	0

The typical adoption of the same cost for erring and rejecting on the two classes leads to the following simplified loss matrix:

		predicted		
		$\mathcal{C}_1$	$\mathcal{C}_{reject}$	$\mathcal{C}_2$
true	$\mathcal{C}_1$	0	C_{low}	C_{high}
	$\mathcal{C}_2$	C_{high}	C_{low}	0

Therefore, $C_{reject} = \frac{C_{low}}{C_{high}} = w_r$ is the cost of rejecting (normalised by the cost of erring). The data replication method with reject option tries to minimizes the empirical risk $w_r R + E$, where R accounts for the rejection rate and E for the misclassification rate.

2) Prediction: To predict the class of an unseen example, classify both replicas of the example in the extended dataset with the binary classifier. From the sequence of binary labels one can infer the predicted label on the original ordinal classes

$$\overline{\mathcal{C}}_1, \overline{\mathcal{C}}_1 \Longrightarrow \mathcal{C}_1 \qquad \overline{\mathcal{C}}_2, \overline{\mathcal{C}}_1 \Longrightarrow \mathcal{C}_{reject} \qquad \overline{\mathcal{C}}_2, \overline{\mathcal{C}}_2 \Longrightarrow \mathcal{C}_2$$

IV. Experimental Results

The aim of our experimental study is to compare the performance of the rejoSVM algorithm with standard approaches described in Section II.

The performance of the classification methods were assessed over two datasets. The first was synthetically generated; the second dataset includes real data from a medical application.

As in [5], for the synthetic dataset, we began by generating 400 example points $\mathbf{x} = [x_1\ x_2]^t$ in the unit square $[0, 1] \times [0, 1] \subset \mathbb{R}^2$ according to a uniform distribution. Then, we assigned to each example $\mathbf{x}$ a class $y \in \{-1, +1\}$ corresponding to

$$\begin{aligned}
(b_{-2}, b_{-1}, b_0, b_1) &= (-\infty; -0.5; 0.25; +\infty) \\
\varepsilon_1 &\sim N(0, 0.125^2) \\
\alpha &= 10(x_1 - 0.5)(x_2 - 0.5) \\
t &= \min_{r \in \{-1,0,+1\}} \{r : b_{r-1} < \alpha + \varepsilon_1 < b_r\} \\
\varepsilon_2 &\sim Uniform(b_{-1}, b_0) \\
y &= \begin{cases} t & t \neq 0 \\ +1 & t = 0 \wedge \varepsilon_2 < \alpha \\ -1 & t = 0 \wedge \varepsilon_2 > \alpha \end{cases}
\end{aligned} \tag{2}$$

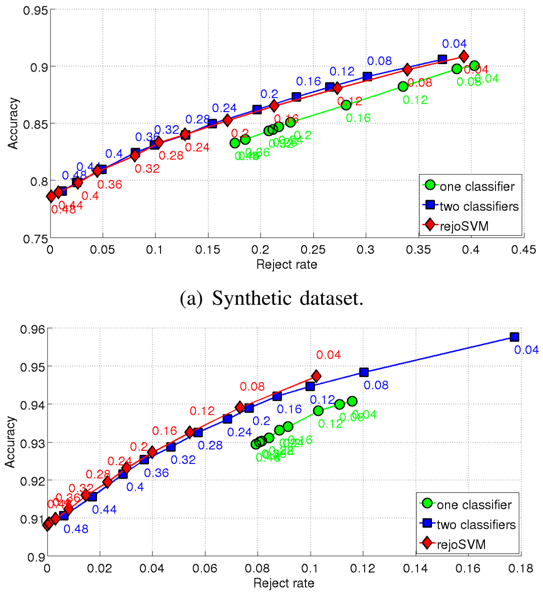
This distribution creates two plateau uniformly distributed and a transition zone of linearly decreasing probability, delimited by hyperbolic boundaries.

The second dataset encompassing 960 observations was taken from previous works [1] and expresses the aesthetic evaluation of Breast Cancer Conservative Treatment. For each patient submitted to BCCT, 30 measurements were recorded, capturing visible skin alterations or changes in breast volume or shape. The aesthetic outcome of the treatment for each and every patient was classified in one of the four categories: Excellent, Good, Fair and Poor. For the purposed of this work, the multiclass problem was transformed into a binary one, by aggregated Excellent and Good in one class, and the Fair and Poor cases in another class.

We randomly split each dataset into training and test sets, with 5% and 95% of the data, respectively. The splitting of the data into training and test sets was repeated 100 times in order to obtain more stable results for accuracy by averaging and also to assess the variability of this measure. The best parameterization of each model was found by 'grid-search', based on a 5-fold cross validation scheme conducted on the training set. Finally, the error of the model was estimated on the test set.

The performance of a classifier with reject option can be represented by the classification accuracy achieved for any value of the reject rate (the so-called Accuracy-Reject curve). The trade-off between errors and rejections depends on the cost of a rejection w_r. This implies that different points of the A-R curve correspond to different values of w_r. We considered values of w_r less than 0.5, as above this value it is preferable to just try to guess randomly.

Figure 5 summarises the results obtained for all three methods on the datasets. A first main assertion is that rejoSVM performs better than the simpler solution based on a single classifier. In all experiments, the performance of rejoSVM was superior to the single classifier approach, over the full range of values for w_r. Comparing the rejoSVM with the two independent classifiers approach, neither of the two techniques outperformed the other one. Indeed, both techniques exhibited almost the same behaviour over the two datasets. It is important to emphasise that rejoSVM has the advantage of simplicity, using a single direction for both boundaries, and interpretability.



(a) Synthetic dataset.

(b) BCCT dataset.

Fig. 5. The A-R curves for the three datasets.

Despite our method does not clearly outperform standard approaches, some considerations should be made concerned to our proposal: 1) the capability to detect reject regions with a single standard binary classifier; 2) it does not need the addition of any confidence level, or thresholds, to define the trust regions; and 3) it does not generate ambiguity regions as the two classifier can produce as it was presented in Figure 2(a). A feature of our proposal is a straightforward extension to multiclass classification which will be developed in further studies.

V. Conclusion

In this paper, we proposed an extension of the data replication method [5] that directly embeds reject option. This extension was derived by taken a new perspective of the classification with reject option problem, viewing the three output classes as naturally ordered. A pair of non-intersecting boundaries delimits the rejection region provided by our model. The same holds for the rejection region provided by the commonly used rejection technique. Our proposal has the advantages of using a standard binary classifier and embedding the design of the reject region during the training process. Moreover, the method allows a flexible definition of the position and orientation of the boundaries, which can change for different values of the cost of rejections w_r. Finally, further studies will be made on the extension of this work applied to a multiclass reject problem and the mapping to additional learning frameworks, such as neural networks. We also plan to conduct a complete comparison study with state of the art methods.

Acknowledgments

This work has been partially supported by Fundação para a Ciência e a Tecnologia (FCT) - Portugal through project PTDC/EIA/64914/2006.

References

[1] J. S. Cardoso and M. J. Cardoso, "Towards an intelligent medical system for the aesthetic evaluation of breast cancer conservative treatment," *Artificial Intelligence in Medicine*, vol. 40, pp. 115–126, 2007.

[2] G. Fumera and F. Roli, "Support Vector Machines with Embedded Reject Option," in *SVM '02: Proceedings of the First International Workshop on Pattern Recognition with Support Vector Machines*. London, UK: Springer-Verlag, 2002, pp. 68–82.

[3] A. Bounsiar, P. Beauseroy, and E. Grall-Maës, "General solution and learning method for binary classification with performance constraints," *Pattern Recognition Letters*, vol. 29, no. 10, pp. 1455–1465, 2008.

[4] R. El-Yaniv, D. Pechyony, and E. Yom-Tov, "Better multiclass classification via a margin-optimized single binary problem," *Pattern Recognition Letters*, vol. 29, pp. 1954–1959, 2008.

[5] J. S. Cardoso and J. F. P. da Costa, "Learning to classify ordinal data: the data replication method," *Journal of Machine Learning Research*, vol. 8, pp. 1393–1429, 2007.