Classification With Reject Option Using Conformal Prediction

Henrik Linusson[1], Ulf Johansson[2], Henrik Boström[3], and Tuve Löfström[2]

[1] Dept. of Information Technology, University of Borås, Sweden
henrik.linusson@hb.se
[2] Dept. of Computer Science and Informatics, Jönköping University, Sweden
{ulf.johansson, tuve.lofstrom}@ju.se
[3] School of Electrical Engineering and Computer Science, Royal Institute of Technology, Kista, Sweden
bostromh@kth.se

Abstract. In this paper, we propose a practically useful means of interpreting the predictions produced by a conformal classifier. The proposed interpretation leads to a classifier with a reject option, that allows the user to limit the number of erroneous predictions made on the test set, without any need to reveal the true labels of the test objects. The method described in this paper works by estimating the cumulative error count on a set of predictions provided by a conformal classifier, ordered by their confidence. Given a test set and a user-specified parameter k, the proposed classification procedure outputs the largest possible amount of predictions containing on average at most k errors, while refusing to make predictions for test objects where it is too uncertain. We conduct an empirical evaluation using benchmark datasets, and show that we are able to provide accurate estimates for the error rate on the test set.

1 Introduction

Conformal predictors [13] are predictive models that associate each of their predictions with a measure of statistically valid confidence. Given a test object x_j, a conformal classifier outputs a *prediction set*—a class label set $\Gamma_j^\epsilon \subseteq Y$—where the probability of making an erroneous prediction (i.e., excluding the correct class label y_j) is at most $\epsilon \in (0, 1)$, where ϵ is a user-specified significance level. Importantly, conformal predictors are automatically well-calibrated, in that the error probability ϵ is guaranteed to correspond with the empirical error asymptotically [13].

Due to their ability to provide users with accurate confidence measures, conformal predictors are particularly useful in risk-sensitive applications, where poor predictions might incur large costs (monetary or otherwise), e.g., stroke risk assessment [4], diagnosis of acute abdominal pain [9] or drug development [3].

However, while conformal predictors are able to supply users with an appropriate estimate of error probability, the validity of a conformal classifier holds

only *a priori*, i.e., before the prediction is made. After observing a particular prediction, it is no longer automatically correct to interpret ϵ as a well-calibrated error probability for any particular prediction, which leads to conformal classifiers instead requiring predictions to be interpreted in a manner that is potentially counter-intuitive to a user less familiar with p-value statistics [6]. Specifically, some prediction regions are always guaranteed to be correct (because they contain all possible labels) whereas others are always guaranteed to be incorrect (because they contain no class labels); since the overall error rate is asymptotically ϵ, this leads to the more interesting prediction regions (containing, e.g., only a single class label) potentially having an error rate that is not immediately related to ϵ.

In [6], a method was proposed for providing a more practical interpretation of the predictions provided by a conformal classifier, by producing adjusted confidence values specifically for predictions containing only a single class label (in a binary classification context). The method proposed in [6] relied on using posterior information regarding the frequencies of predictions containing one, two or zero class labels (estimated from the test set, without knowledge of the true labels). While that method showed promising results, i.e., such that the updated estimates appeared empirically well-calibrated, it does show obvious limitations; specifically, it retains a particular dependency on ϵ that is far from intuitive.

In this paper, we further refine the work presented in [6], and propose a more flexible method of producing an intuitive interpretation of the predictions produced by a conformal classifier. We remove the dependency on ϵ, and replace it with a new parameter, k, that denotes the maximum expected number of errors that we wish the classifier to make on the test set. The result is a classifier that can accurately estimate the error rate for ordered subsets of the test set (without any need to reveal the true test set class labels); by choosing a value for k, we are able to output predictions for a subset of the test objects (while refusing to make predictions when the underlying conformal predictor is too uncertain), where the predictions that are made contain on average at most k errors.

In the next section, we briefly describe the conformal classification framework. In Section 3, we outline the proposed approach for making predictions with a bound on the expected number of errors. In Section 4, we empirically evaluate the approach using 20 publicly available datasets. Finally, in Section 5, we summarize the main findings and discuss some directions for future research.

2 Conformal Classifiers

In order to produce confidence predictions, a conformal classifier depends on a *nonconformity function*—a function $f(z, \zeta) \rightarrow \mathbb{R}$ that scores a pattern $z = (x, y)$ based on how well it corresponds with a sequence of patterns $\zeta = z_1, \ldots, z_n$, such that nonconforming (i.e., strange or unlikely) patterns obtain larger nonconformity scores than more common patterns. A standard way of defining nonconformity functions is to base them on the predictions made by a traditional

classification model, as

$$f(z_i, \zeta) = \Delta\left[h(x_i), y_i\right], \tag{1}$$

where h is a classifier (often called the *underlying model*) induced from ζ, and Δ is a function that measure the prediction errors of h. A common choice of Δ, for classification problems, is the margin error function,

$$\Delta[h(x_i), y_i] = \max_{y \neq y_i} \hat{P}_h(y \mid x_i) - \hat{P}_h(y_i \mid x_i), \tag{2}$$

where $\hat{P}_h(y \mid x)$ denotes the probability estimate provided by h for the class y.

Once a suitable underlying model and nonconformity function have been selected, a conformal classifier can be constructed in a few different manners. One of the more popular conformal classifier variants is the *inductive conformal predictor* [8, 10, 13], premiered in particular for its low computational overhead. In order to train an inductive conformal predictor for classification, the following training procedure is used:

1. Divide the training set Z into two disjoint subsets:
   - A *proper training set* Z_t.
   - A *calibration set* Z_c, where $|Z| = q$.
2. Train a classifier h using Z_t as the training data.
3. Let $\{\alpha_1, \ldots, \alpha_q\} = \{f\left(z_i, Z_t\right) : z_i \in Z_c\}$.

When a new test object x_j is observed, the standard way of obtaining a prediction from the conformal classifier is to produce a prediction region $\Gamma_j^\epsilon \subseteq Y$ as follows:

1. Fix a significance level $\epsilon \in (0, 1)$.
2. For each class $\tilde{y} \in Y$:
   (a) Tentatively label x_j as $(x_j, \tilde{y})$.
   (b) Let $\alpha_j^{\tilde{y}} = f\left[(x_j, \tilde{y}), Z_t\right]$.
   (c) Calculate $p_j^{\tilde{y}}$ as

$$p_j^{\tilde{y}} = \frac{\left|\left\{z_i \in Z_c : \alpha_i > \alpha_j^{\tilde{y}}\right\}\right|}{q+1} + \theta_j \frac{\left|\left\{z_i \in Z_c : \alpha_i = \alpha_j^{\tilde{y}}\right\}\right| + 1}{q+1}, \tag{3}$$

   where $\theta_j \sim U[0, 1]$.
   (d) Let $\Gamma_j^\epsilon = \left\{\tilde{y} \in Y : p_j^{\tilde{y}} > \epsilon\right\}$.

The resulting class label set Γ_j^ϵ contains the true label y_j with probability $1 - \epsilon$, i.e., an error (meaning that $y_j \notin \Gamma_j^\epsilon$) occurs with probability ϵ.

An alternative way of producing predictions with a conformal classifier is to output what we will refer to as *confidence-credibility predictions* [8]. Here, the output for a test object x_j takes the form $(\hat{y}_j, \gamma_j, \mu_j)$, where

- $\hat{y}_j$ is the most likely class label (i.e., the class label for which $p_j^{\tilde{y}}$ is greatest),
- γ_j is the *confidence*, which is one minus the second largest p-value, and

– μ_j is the *credibility*, which is the largest p-value.

Here, we are effectively forcing the conformal predictor to output the most confident prediction set containing only a single class label (if we were to increase the confidence of the prediction, at least one other class label would be included). Credibility corresponds with the significance level at which all class labels are rejected, and the prediction becomes empty—if credibility is very low, the conformal classifier considers all potential class labels as unsuitable for the test object.

Since conformal predictors are unconditionally valid by default, there is often a need to take the true class labels into consideration when evaluating their predictions [6, 7, 12, 13]; specifically, it is possible that the error probability of a conformal predictor is greater (or smaller) than ϵ, depending on the test object's true label. In practice, this effectively means that, depending on properties of the dataset, it is possible that the most confident predictions are made only for test objects pertaining to a particular class (usually the majority class). This behaviour is easily rectified by employing a label-conditional (Mondrian) conformal classifier [12, 13], where the p-values are additionally conditioned on the class labels using

$$p_j^{\tilde{y}} = \frac{\left|\left\{z_i \in Z^{\tilde{y}} : \alpha_i > \alpha_j^{\tilde{y}}\right\}\right|}{|Z^{\tilde{y}}| + 1} + \theta_j \frac{\left|\left\{z_i \in Z^{\tilde{y}} : \alpha_i = \alpha_j^{\tilde{y}}\right\}\right| + 1}{|Z^{\tilde{y}}| + 1}, \tag{4}$$

where $Z^{\tilde{y}} \subseteq Z_c$ are the calibration patterns that belong to the class $\tilde{y}$.

3 Error Probabilities Using Posterior Information

The confidence measures supplied by a conformal classifier are valid in the sense that the observed error rate, over a test sequence, is guaranteed to converge to ϵ (when the predictor is allowed to output prediction sets). This probability is by default unconditional, in the sense that we might not make any assertions regarding the distribution of errors with regard to the problem space; this can be contrasted to, e.g., label-conditional validity, where we might assert that error probability is independent of y_j or object-conditional validity, where the error probability is independent of x_j [12, 13].

More importantly, the confidence measure of a conformal predictor is valid in an *a priori* sense, meaning the error probability before making a prediction, and the error probability after observing a prediction, are not necessarily the same [6]. In particular, when a conformal predictor is applied in a batch prediction setting (i.e., we are making predictions for a test set whose size is greater than one, and those predictions are obtained simultaneously in a batch), it is easy to see that *a priori* and *a posteriori* error probabilities are unequal: consider a binary classification problem, where we are predicting the output labels of a test set containing 100 objects, with a significance level 0.1 (i.e., we are expecting an overall error rate of 10%); if 90 of our predictions contain both class labels,

while the remaining 10 predictions contain only a single class label, we are likely to fool ourselves if we were to trust the 10 "interesting" singleton predictions—we are expecting the conformal classifier to make approximately 10 errors over the entire test set, and none of the predictions containing both class labels can possibly be erroneous.

Of course, this seemingly counter-intuitive result stems from a forced misunderstanding of the conformal prediction procedure (we are effectively trying to interpret p-values as true probabilities); nonetheless, it is not clear how an end-user should interpret the predictions in an appropriate manner.

In [6], an attempt was made to utilize posterior information (empirical estimates of the rates of empty, singleton and double predictions, coupled with the knowledge that empty and double predictions cannot be correct or erroneous, respectively) in order to produce more reliable estimates of the error probability of singleton predictions (which, in a binary classification scenario, arguably make out the most interesting predictions that can be made). An unconditional (w.r.t. labels and objects) adjusted estimate was defined as

$$\hat{\epsilon}_s = \frac{\epsilon}{P(s) + P(e)} \tag{5}$$

where $P(s)$ and $P(e)$ are the rates of singleton predictions and empty predictions observed in the test set (without any need to consider the true output labels of the test patterns). A label conditional variant was also developed, but is omitted here.

3.1 Getting Rid of ϵ

The adjusted estimates proposed in [6] were intended to provide a better assessment of the quality of singleton predictions; unfortunately, these estimates (both unconditional and label-conditional variants) retain a dependency on the user-specified ϵ-parameter, which is rather unintuitive, since the final estimate $\hat{\epsilon}$ is only loosely related to ϵ. As an alternative, we propose an updated procedure, that is not dependent on ϵ, but instead operates on top of predictions made on the confidence-credibility form.

Table 1. Example of confidence-credibility predictions (credibility scores are omitted).

idx	0	1	2	3	4	5	6	7	8	9
$\hat{y}$	0	0	1	1	0	1	0	1	0	1
confidence	0.60	0.62	0.63	0.65	0.72	0.78	0.82	0.90	0.97	0.99

Suppose we are given a batch of confidence-credibility predictions, where we have sorted the predictions with respect to their confidence, e.g., as in Table 1. The appropriate manner of interpreting these confidence scores is: all predictions with confidence at least $c \in (0, 1)$ contain on average $n(1-c)$ errors, where n is the

total number of predictions made (in this case 10). Hence, from the predictions in Table 1, we should expect approximately four errors across the entire test set, and approximately one error among the predictions for indices 7-9.

Based on this information, we propose the following: given a test set $x_1, \ldots, x_n$, obtain from a conformal classifier the predicted labels and their confidence, $(\hat{y}_1, \gamma_1), \ldots, (\hat{y}_n, \gamma_n)$. For each prediction, compute $\hat{k}_j = n(1 - \gamma_j)$, and construct the tentative prediction set $\hat{Y} = (\hat{y}_1, \gamma_1, \hat{k}_1), \ldots, (\hat{y}_n, \gamma_n, \hat{k}_n)$; here $\hat{k}_j$ is the expected error rate for all predictions with confidence γ_j or greater. Note that $\hat{k}_j$ has a anti-monotonic property with respect to γ_j, i.e., $\gamma_i \leq \gamma_j \rightarrow \hat{k}_i \geq \hat{k}_j$. Finally, output the predictions

$$\left\{\gamma_j \in \hat{Y} : \hat{k}_j \leq k\right\}, \tag{6}$$

where k is user-specified, and denotes the maximum number of expected errors that we allow on the test set. Any prediction where $\hat{k} > k$ is rejected.

The main reason for constructing this new estimate $\hat{k}_j$ is that the confidence value γ_j, by itself, has no clear intuitive interpretation; the formal interpretation given above, i.e., among all test objects, $n(1 - c)$ errors are distributed among those predictions where $\gamma_j \geq c$, is inherently dependent on n. Here, we are simply coding this information into the new estimate $\hat{k}_j$, so that a much more intuitive interpretation can be obtained.

4 Experiments

In order to assess how well our proposed procedure is able to estimate the error rate on the test set, an experimental evaluation was performed using 20 datasets taken from the UCI repository [1], listed in Table 2.

Table 2. Datasets used in the experiments. #inst denotes the number of instances contained in the dataset; #min and #maj denote the number of examples belonging to the minority and majority classes, respectively. %min is the percentage of examples that belong to the minority class.

Dataset	#inst	#min	#maj	%min	Dataset	#inst	#min	#maj	%min
balance-scale	576	288	288	50.0	hepatitis	155	32	123	20.6
breast-cancer	286	85	201	29.7	ionosphere	351	126	225	35.9
breast-w	699	241	458	34.5	kr-vs-kp	3196	1527	1669	47.8
credit-a	690	307	383	44.5	labor	57	20	37	35.1
credit-g	1000	300	700	30.0	liver-disorders	345	145	200	42.0
diabetes	768	268	500	34.9	mushroom	8124	3916	4208	48.2
haberman	306	81	225	26.5	sick	3772	231	3541	6.1
heart-c	303	138	165	45.5	sonar	208	97	111	46.6
heart-h	294	106	188	36.1	spambase	4601	1813	2788	39.4
heart-s	270	120	150	44.4	tic-tac-toe	958	332	626	34.7

The underlying conformal predictor used a random forest classifier [2], containing 100 decision trees, with a margin error nonconformity function (Equation 2). The experiments were implemented in Python using the scikit-learn machine learning library [11], as well as the nonconformist[4] library for conformal prediction. In the experiments, a 10x10-fold cross-validation was performed, and the results presented are averaged across the 10 iterations. In each fold, 25% of the training data was used as the calibration set for the inductive conformal classifier, as suggested in [5].

Table 3. Average number of predictions and errors made per iteration, over the entire dataset (using 10 folds), using an unconditional conformal classifier. #pred denotes the total number of predictions made and %pred denotes the size of the prediction set as a percentage of the total test set. Finally, #err denotes the number of erroneous predictions. k is the user-specified expected error count.

k	1			5			10		
dataset	#pred	%pred	#err	#pred	%pred	#err	#pred	%pred	#err
balance-scale	105.0	18.2	1.0	461.7	80.2	4.4	496.3	86.2	10.2
breast-cancer	7.1	2.5	1.3	30.3	10.6	4.8	51.6	18.0	8.8
breast-w	109.8	15.7	0.8	519.0	74.2	4.7	615.4	88.0	8.4
credit-a	23.1	3.3	1.4	114.9	16.7	5.1	210.0	30.4	9.9
credit-g	21.8	2.2	0.9	103.7	10.4	4.5	171.3	17.1	8.8
diabetes	23.6	3.1	0.7	112.9	14.7	4.3	170.7	22.2	9.6
haberman	8.4	2.7	1.2	49.4	16.1	5.0	77.3	25.3	8.7
heart-c	20.2	6.7	0.6	92.6	30.6	4.4	143.6	47.4	9.6
heart-h	20.8	7.1	1.1	87.8	29.9	4.9	143.7	48.9	10.4
heart-s	17.5	6.5	0.9	81.2	30.1	4.1	129.5	48.0	9.9
hepatitis	18.6	12.0	1.2	82.0	52.9	4.4	106.4	68.6	8.9
ionoshere	49.8	14.2	1.0	221.9	63.2	4.8	286.8	81.7	9.5
kr-vs-kp	549.0	17.2	1.2	2482.2	77.7	5.6	2954.9	92.5	10.4
labor	13.1	23.0	0.5	53.4	93.7	3.6	56.6	99.3	4.7
liver-disorders	5.5	1.6	0.6	27.0	7.8	3.6	52.2	15.1	8.3
mushroom	1820.6	22.4	1.1	8124.0	100.0	2.5	8124.0	100.0	2.5
sick	708.9	18.8	0.7	3184.5	84.4	4.9	3444.8	91.3	9.9
sonar	24.6	11.8	1.3	108.0	51.9	5.1	131.3	63.1	10.1
spambase	229.3	5.0	1.0	1112.9	24.2	5.7	1761.6	38.3	11.4
tic-tac-toe	182.6	19.1	1.0	799.8	83.5	4.2	868.6	90.7	8.1
mean	197.96	10.66	0.98	892.46	47.64	4.53	999.83	58.61	8.91

Table 3 lists the number of predictions made, as well as the number of errors among those predictions, for the 20 datasets. Here, an unconditional conformal classifier is used (Equation 3). Results are averaged over 10 iterations. The maximum number of predictions possible (per dataset) is given by #inst in Table 2.

[4] https://github.com/donlnz/nonconformist

For each dataset, the procedure was applied with $k = 1$, $k = 5$ as well as $k = 10$, i.e., we are asking to make the maximum number of predictions containing on average 1, 5 or 10 errors. From the results in Table 3, it is evident that the proposed procedure is able to estimate the error count on the test set rather well, although the estimates appear to be somewhat conservative in general, in particular as k increases. In all cases, the procedure is able to output a non-trivial number of predictions (i.e., the prediction set is substantially greater than k), while still limiting the number of erroneous predictions.

Table 4. Average number of predictions made per iteration, using an unconditional conformal classifier. #pred is the total number of predictions made, and #min is the number of predictions made for test objects where the true label is the minority class. %min is #min expressed as a percentage of #pred.

k	1			5			10		
Dataset	#pred	#min	%min	#pred	#min	%min	#pred	#min	%min
balance-scale	105.0	52.2	49.7	461.7	230.3	49.9	496.3	247.7	49.9
breast-cancer	7.1	1.5	21.1	30.3	5.0	16.5	51.6	8.9	17.2
breast-w	109.8	29.3	26.7	519.0	133.1	25.6	615.4	184.9	30.0
credit-a	23.1	8.9	38.5	114.9	46.5	40.5	210.0	82.7	39.4
credit-g	21.8	0.9	4.1	103.7	4.5	4.3	171.3	9.0	5.3
diabetes	23.6	1.1	4.7	112.9	6.6	5.8	170.7	14.9	8.7
haberman	8.4	1.1	13.1	49.4	5.0	10.1	77.3	8.9	11.5
heart-c	20.2	7.4	36.6	92.6	40.1	43.3	143.6	62.9	43.8
heart-h	20.8	4.1	19.7	87.8	19.6	22.3	143.7	39.6	27.6
heart-s	17.5	6.5	37.1	81.2	31.9	39.3	129.5	53.0	40.9
hepatitis	18.6	1.3	7.0	82.0	5.5	6.7	106.4	11.5	10.8
ionosphere	49.8	13.6	27.3	221.9	62.0	27.9	286.8	93.0	32.4
kr-vs-kp	549.0	260.6	47.5	2482.2	1159.6	46.7	2954.9	1399.3	47.4
labor	13.1	3.9	29.8	53.4	17.8	33.3	56.6	19.7	34.8
liver-disorders	5.5	1.5	27.3	27.0	9.1	33.7	52.2	18.4	35.2
mushroom	1820.6	866.0	47.6	8124.0	3916.0	48.2	8124.0	3916.0	48.2
sick	708.9	7.5	1.1	3184.5	38.9	1.2	3444.8	70.4	2.0
sonar	24.6	12.2	49.6	108.0	46.1	42.7	131.3	55.7	42.4
spambase	229.3	85.3	37.2	1112.9	423.7	38.1	1761.6	667.5	37.9
tic-tac-toe	182.6	51.9	28.4	799.8	222.0	27.8	868.6	264.7	30.5
mean	198.0	70.8	27.7	892.5	321.2	28.2	999.8	361.4	29.8

Since we are using an unconditional conformal classifier, it becomes interesting to evaluate not only the number of predictions output by our proposed process, but also the number of predictions output for the minority and majority class test objects, respectively. The results shown in Table 4 indicate that, while there appears to be a bias towards premiering the majority class among the output predictions (see, e.g., sick, credit-g, diabetes and hepatitis), this bias is never so strong as to cause the classifier to make predictions only for test ob-

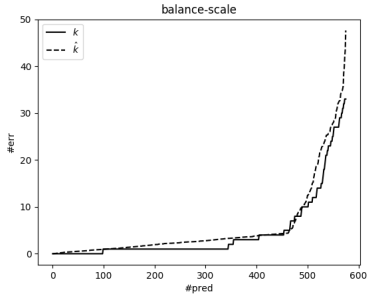
jects belonging to one of the two possible classes. This behaviour—displaying a (sometimes substantial) bias towards the majority class—is common in unconditional conformal predictors when the dataset is heavily imbalanced. The issue is easily alleviated however, by employing a label-conditional Mondrian conformal classifier (Equation 4) instead; results from such a classifier are shown in Table 5.

Table 5. Average number of predictions and errors made per iteration, over the entire dataset (using 10 folds), using a label-conditional conformal predictor. #pred denotes the total number of predictions made and %min denotes the percentage of predictions made for test objects belonging to the minority class. Finally, #err denotes the number of erroneous predictions. k is the user-specified expected error count.

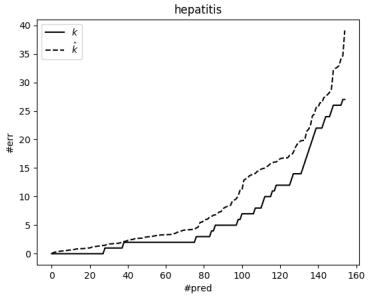
k	1			5			10		
dataset	#pred	%min	#err	#pred	%min	#err	#pred	%min	#err
balance-scale	54.2	47.2	1.0	275.4	49.7	4.3	487.7	49.7	8.7
breast-cancer	3.7	35.1	1.1	17.4	39.1	4.4	34.7	35.4	8.9
breast-w	57.1	35.7	1.0	291.2	39.9	4.5	512.3	34.7	10.0
credit-a	20.1	42.8	1.3	101.8	41.3	5.6	195.9	45.0	10.6
credit-g	12.8	32.0	1.4	59.9	31.2	4.7	109.6	29.7	10.2
diabetes	15.1	27.2	1.5	63.3	27.5	5.8	122.0	25.2	9.9
haberman	3.2	21.9	0.6	19.0	17.4	4.6	37.9	16.9	9.2
heart-c	14.9	45.6	0.7	73.8	46.3	6.1	134.9	48.6	10.6
heart-h	13.6	48.5	1.3	63.4	45.9	5.3	114.3	40.3	10.1
heart-s	13.8	48.6	1.0	70.7	46.0	5.3	128.0	49.1	11.5
hepatitis	7.5	37.3	1.2	33.7	30.3	4.3	64.4	23.8	9.9
ionosphere	30.1	47.8	0.6	149.3	48.6	4.5	247.8	43.5	9.8
kr-vs-kp	306.0	47.7	1.2	1560.4	46.1	5.5	2762.9	48.5	9.4
labor	7.2	40.3	1.1	35.1	39.6	3.4	51.8	36.1	5.3
liver-disorders	6.4	28.1	1.0	28.5	33.7	4.0	54.5	36.9	9.1
mushroom	910.4	49.6	1.2	4579.6	50.0	4.1	8124.0	48.2	5.3
sick	81.2	39.8	1.3	388.8	39.8	5.5	624.5	26.9	10.2
sonar	14.0	48.6	0.8	70.3	49.1	4.1	129.5	47.4	9.7
spambase	147.3	41.0	1.9	751.8	41.1	5.6	1388.4	44.0	10.0
tic-tac-toe	94.6	49.6	1.4	464.6	49.8	5.4	775.6	40.8	9.5
mean	90.7	40.7	1.1	454.9	40.6	4.9	805.0	38.5	9.4

Table 5 shows results analogous to those in Tables 3 ad 4, but instead using an underlying label-conditional conformal classifier (Equation 4). The results correspond well with what is normally expected from a label-conditional conformal predictor: the overall error rate remains relatively untouched (we are still seeing a good correspondence between k and the empirical error rate), but the sensitivity of the classifier is reduced (the model is able to output far fewer predictions). The main benefit shown by the label-conditional variant, however, is that there is a clear reduction in bias with respect to the true class labels of the

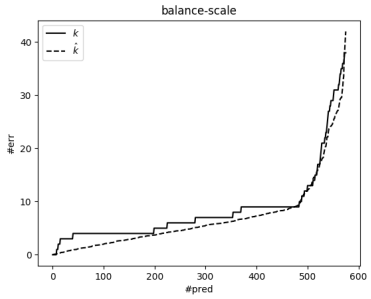
test objects. The classifier is able to output a more even distribution of positive and negative predictions, without any substantial negative effect on the error count among the predictions that are made.



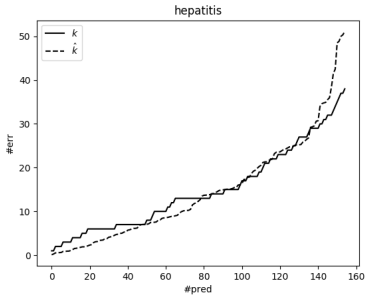
(a) Cumulative errors (real and predicted) on the balance-scale dataset; unconditional.



(b) Cumulative errors (real and predicted) on the hepatitis dataset; unconditional.



(c) Cumulative errors (real and predicted) on the balance-scale dataset; label-conditional.



(d) Cumulative errors (real and predicted) on the hepatitis dataset; label-conditional.

Fig. 1. Cumulative errors (real, k, solid lines; and predicted, $\hat{k}$, dashed lines) on the balance-scale and hepatitis datastets. Results are shown for a single iteration, with the x-axis showing the number of predictions made (with decreasing confidence) and the y-axis showing the cumulative error count among the output predictions.

Finally, in order to provide some insight into how well the proposed procedure functions over a larger selection of values for k, Figure 1 shows the true and predicted cumulative error rates over the full test set for two of the datasets, taken from a single iteration. Figures 1a and 1b show results using an unconditional conformal classifier, and Figures 1c and 1d show results using a label-conditional conformal classifier. The x-axis displays the number of predictions made (in order of decreasing confidence), and the y-axis displays the cumulative

error count. The two lines—dashed and solid—correspond to the predicted and actual cumulative error counts respectively. In each case, the predictive cumulative error count ($\hat{k}$) closely follows the true cumulative error count (k); with respect to their calibration, there is no clear difference between the unconditional and label-conditional variants.

5 Concluding Remarks

In this paper, we offer an interpretation of the conformal classification procedure, that is able to estimate the number of errors made by a classifier on the test set, without needing to reveal the true test set class labels. The procedure described results in a classifier with a reject option, that outputs predictions for a subset of the test set, where the expected error count is limited by a user-specified parameter k; given a test set and a choice of k, the proposed procedure outputs the largest possible number of predictions containing on average at most k errors.

We evaluate the procedure empirically using 20 benchmark datasets, and obtain very promising results, indicating that we are able to provide accurate estimates of the error rate on the test set.

It is not obvious how well the proposed procedure will perform on multi-class datasets or heavily imbalanced datasets; as such, evaluating the procedure on a more diverse selection of datasets would be of great interest. Naturally, it would also be of great interest to evaluate our proposed procedure to alternative methods for constructing classifiers with a reject option.

Additionally, it would be interesting to evaluate the proposed procedure with respect to specific applications—in particular heavily imbalanced problems where identifying the minority test patterns is the key objective. Extending the procedure so that errors are only allowed for one of two classes (normally the minority class) might be beneficial in several applications.

Acknowledgements

This work was supported by the Swedish Knowledge Foundation through the project Data Analytics for Research and Development (20150185).

References

1. Bache, K., Lichman, M.: UCI machine learning repository. URL http://archive.ics.uci.edu/ml (2013)
2. Breiman, L.: Random forests. Machine learning 45(1), 5–32 (2001)
3. Eklund, M., Norinder, U., Boyer, S., Carlsson, L.: The application of conformal prediction to the drug discovery process. Annals of Mathematics and Artificial Intelligence 74(1-2), 117–132 (2015)
4. Lambrou, A., Papadopoulos, H., Kyriacou, E., Pattichis, C.S., Pattichis, M.S., Gammerman, A., Nicolaides, A.: Assessment of stroke risk based on morphological ultrasound image analysis with conformal prediction. In: Artificial Intelligence Applications and Innovations, pp. 146–153. Springer (2010)
5. Linusson, H., Johansson, U., Boström, H., Löfström, T.: Efficiency comparison of unstable transductive and inductive conformal classifiers. In: Artificial Intelligence Applications and Innovations, pp. 261–270. Springer (2014)
6. Linusson, H., Johansson, U., Boström, H., Löfström, T.: Reliable confidence predictions using conformal prediction. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. pp. 77–88. Springer (2016)
7. Löfström, T., Boström, H., Linusson, H., Johansson, U.: Bias reduction through conditional conformal prediction. Intelligent Data Analysis 9(6) (2015)
8. Papadopoulos, H.: Inductive conformal prediction: Theory and application to neural networks. Tools in artificial intelligence 18(315-330), 2 (2008)
9. Papadopoulos, H., Gammerman, A., Vovk, V.: Reliable diagnosis of acute abdominal pain with conformal prediction. Engineering Intelligent Systems 17(2), 127 (2009)
10. Papadopoulos, H., Proedrou, K., Vovk, V., Gammerman, A.: Inductive confidence machines for regression. In: Machine Learning: ECML 2002, pp. 345–356. Springer (2002)
11. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in python. The Journal of Machine Learning Research 12, 2825–2830 (2011)
12. Vovk, V.: Conditional validity of inductive conformal predictors. Machine learning 92(2-3), 349–376 (2013)
13. Vovk, V., Gammerman, A., Shafer, G.: Algorithmic learning in a random world. Springer Verlag, DE (2006)