Background Check: A general technique to build more reliable and versatile classifiers. Supplementary material

Miquel Perello Nieto*†, Telmo M. Silva Filho*‡, Meelis Kull† and Peter Flach†
†Intelligent Systems Laboratory, University of Bristol, UK
‡Centro de Informatica, Universidade Federal de Pernambuco, Brazil
Email: †firstname.lastname@bristol.ac.uk, ‡tmsf@cin.ufpe.br

References

[1] M. Lichman, "UCI machine learning repository," 2013.
[2] K. Hempstalk, E. Frank, and I. H. Witten, *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2008, Antwerp, Belgium, September 15-19, 2008, Proceedings, Part I.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, ch. One-Class Classification by Combining Density and Class Probability Estimation, pp. 505–519.
[3] C. Ferri and J. Hernández-Orallo, "Cautious classifiers." *Proceedings of ROC Analysis in Artificial Intelligence, 1st International Workshop (ROCAI- 2004)*, vol. 4, pp. 27–36, 2004.
[4] L. Li, Q. Hu, X. Wu, and D. Yu, "Exploration of classification confidence in ensemble learning," *Pattern Recognition*, vol. 47, no. 9, pp. 3120 – 3131, 2014.
[5] D. Tax and R. Duin, "Growing a multi-class classifier with a reject option," *Pattern Recognition Letters*, vol. 29, no. 10, pp. 1565–1570, jul 2008.
[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

I. Proofs

Proposition 1.

$$p(\mathsf{b}|x) = \frac{1}{1+r(x)},$$
$$p(\mathsf{f}_c|x) = \frac{p(\mathsf{f}_c|\mathsf{f},x)r(x)}{1+r(x)} \qquad \textit{for } c = 1,\dots,k$$

Proof. $\frac{1}{1+r(x)} = \frac{1}{(p(\mathsf{b}|x)+p(\mathsf{f}|x))/p(\mathsf{b}|x)} = \frac{1}{1/p(\mathsf{b}|x)} = p(\mathsf{b}|x)$; $\frac{p(\mathsf{f}_c|\mathsf{f},x)r(x)}{1+r(x)} = p(\mathsf{f}_c|\mathsf{f},x)\frac{p(\mathsf{f}|x)}{p(\mathsf{b}|x)}p(\mathsf{b}|x) = p(\mathsf{f}_c,\mathsf{f}|x) = p(\mathsf{f}_c|x)$. □

Proposition 2.

$$q_\mathsf{f}(x) = \frac{p(x|\mathsf{f})}{\max_x p(x|\mathsf{f})}, p(x|\mathsf{f}) \qquad = \frac{q_\mathsf{f}(x)}{\int_x q_\mathsf{f}(x)dx}.$$

Proof. $q_\mathsf{f}(x) = \frac{p(x,\mathsf{f})}{\max_x p(x,\mathsf{f})} = \frac{p(x|\mathsf{f})p(\mathsf{f})}{\max_x p(x|\mathsf{f})p(\mathsf{f})} = \frac{p(x|\mathsf{f})}{\max_x p(x|\mathsf{f})}$, $\int_x q_\mathsf{f}(x)dx = \int_x \frac{p(x|\mathsf{f})}{\max_x p(x|\mathsf{f})}dx = \frac{\int_x p(x|\mathsf{f})dx}{\max_x p(x|\mathsf{f})} = \frac{1}{\max_x p(x|\mathsf{f})}$, $p(x|\mathsf{f}) = q_\mathsf{f}(x)\max_x p(x|\mathsf{f}) = q_\mathsf{f}(x)\frac{1}{\int_x q_\mathsf{f}(x)dx}$. □

Proposition 3. *If μ is the affine background bias with $\mu(0) = 0$, then $p(\mathsf{f}|x)$ is a monotonically decreasing function of $\mu(1)$ of the form $p(\mathsf{f}|x) = 1/(\mu(1)+1)$.*

Proof. We have that:

$$p(\mathsf{f}|x) = \frac{q_\mathsf{f}}{q_\mathsf{b}+q_\mathsf{f}}.$$

Applying the affine background bias we get:

$$p(\mathsf{f}|x) = \frac{q_\mathsf{f}}{(1-q_\mathsf{f})\mu(0)+q_\mathsf{f}\mu(1)+q_\mathsf{f}}.$$

Finally, with $\mu(0) = 0$ and eliminating q_f, we arrive at:

$$p(\mathsf{f}|x) = \frac{1}{\mu(1)+1}.$$

□

Proposition 4. *Let μ be the affine background bias with $\mu(0) = 0$, then for a given rejection threshold θ, $\mu(1) = \theta$.*

Proof. Following Chow's rule, for a k-class cautious classification problem, the minimum condition such that an instance x can be accepted and classified by the model is:

$$p(\mathsf{f}_c|x) = \theta,$$

where f_c represents the foreground class with the highest class conditional probability for instance x. In our $(k+1)$-class setting, with the extra class being background, this condition is rewritten as:

$$p(\mathsf{f}_c|\mathsf{f},x) = p(\mathsf{b}|x), \text{ and } p(\mathsf{f}_c|x)p(\mathsf{f}|x) = p(\mathsf{b}|x)$$

Substituting $p(\mathsf{f}_c|x) = \theta$ and $p(\mathsf{b}|x) = 1 - p(\mathsf{f}|x)$ and isolating $p(\mathsf{f}|x)$, we get:

$$p(\mathsf{f}|x) = \frac{1}{\theta+1}$$

Then, from Proposition 3, we arrive at $\mu(1) = \theta$ □

II. Algorithms

III. Data preprocessing

In order to demonstrate the versatility of BC we selected 41 datasets from UCI [1]. Half of them have been used previously in publications [2], [3], [4], [5] which we cite and/or compare against. Because of our interest in multiclass classification problems we selected 20 additional datasets with more than 3 classes. When the datasets were available from the dataset repository mldata.org we used the Python library scikit-learn

Algorithm 1 Training BCD

Require:
Number of *foreground* classes k;
If $k > 1$, the k-class *foreground* classifier;
Algorithm:
Uniformly generate artificial *background* data around *foreground* data;
Train a binary discriminative classifier of *foreground* vs *background*;
if $k > 1$ **then**
    Combine classifiers into a $(k+1)$-class posterior probability estimator;
end if
return $(k+1)$-class posterior probability estimator.

Algorithm 2 Testing BCR

Require:
Number of *foreground* classes k;
If $k > 1$, the k-class *foreground* classifier;
background bias μ;
One-class model trained on *foreground* data;
Algorithm:
Obtain q_{f} from the one-class model;
Estimate q_{b} as $\mu(q_{\mathsf{f}})$;
Estimate posterior probabilities $p(\mathsf{b}|x)$ and $p(\mathsf{f}|x)$;
if $k > 1$ **then**
    Obtain k-class probability vector from *foreground* classifier;
    Combine calibrated probabilities into a $(k+1)$-vector;
end if
return $(k+1)$-class posterior probability estimates.

Algorithm 3 Cautious classification with BC

Require:
k-class *foreground* classifier;
k-class rejection threshold θ;
Algorithm:
Set $\mu(1) = \theta$;
Estimate posterior probabilities $p(\mathsf{b}|x)$ and $p(\mathsf{f}|x)$;
Obtain k-class probability vector from *foreground* classifier;
Combine probabilities into a $(k+1)$-vector;
For every instance x predict $\hat{y} = \text{argmax}_i(p(y = i|x))$;
Reject x if $\hat{y} = (k+1)$;
return Predictions.

Algorithm 4 Outlier detection with BC–training phase

Require:
Number of *foreground* classes k;
k-class *foreground* classifier;
background bias μ;
Algorithm:
if $\mu(0) = \mu(1) = 0.5$ **then**
    Obtain $(k+1)$-class posterior probability estimator with BCD;
else
    Obtain $(k+1)$-class posterior probability estimator with BCR;
end if
return $(k+1)$-class posterior probability estimator.

[6] to download them directly from that repository. Otherwise, we downloaded the data from the UCI webpage.

Because of the large variety of datasets, we had to preprocess and standardise them to run all our experiments. First,

Algorithm 5 Outlier detection with BC–test phase

Require:
Number of *foreground* classes k;
$(k+1)$-class posterior probability estimator BC;
Algorithm:
Obtain $(k+1)$-class posterior probability estimates from BC;
For every instance x predict $\hat{y} = \text{argmax}_i(p(y = i|x))$;
Mark x as outlier if $\hat{y} = (k+1)$;
return Predictions.

nominal features were transformed into numerical values. We chose this option instead of transforming them into sparse binary representations in order to reduce the computational cost of the experiments. For each nominal feature all its values were sorted alphabetically, next they were substituted by their corresponding index in the sorted list, starting from zero. In case of missing values a special number was assigned to them and they were preprocessed as missing values in the next step.

Secondly, all samples that contained missing values were preprocessed in two different ways, depending on the proportion of instances with missing values. In datasets where less than 25% of the instances had missing values these samples were removed (46 samples from autos, 6 from cleveland, 31 from credit-aproval, 8 from dermatology and 4 from wpbc). In the other cases the missing values were substituted by the mean of their corresponding feature (167 values from hepatitis, 1605 from horse and 2480 from mushroom).

Thirdly, datasets with more than 30 000 instances were reduced to 10% of their original size (letter and shuttle) in order to reduce the computational cost. Finally, all features were standardised with mean zero and variance one. Table I summarises the datasets in terms of number of samples, features and classes after preprocessing.

IV. TABLES

Name	Samples	Features	Classes
abalone	4177	8	3
autos	159	25	6
balance-scale	625	4	3
car	1728	6	4
cleveland	297	13	5
credit-approval	653	15	2
dermatology	358	34	6
diabetes	768	8	2
ecoli	336	7	8
flare	1389	10	6
german	1000	20	2
glass	214	9	6
heart-statlog	270	13	2
hepatitis	155	19	2
horse	300	27	2
ionosphere	351	34	2
iris	150	4	3
landsat-satellite	6435	36	6
letter	3511	16	26
libras-movement	360	90	15
lung-cancer	96	7129	2
mfeat-karhunen	2000	64	10
mfeat-morphological	2000	6	10
mfeat-zernike	2000	47	10
mushroom	8124	22	2
optdigits	5620	64	10
page-blocks	5473	10	5
pendigits	10992	16	10
scene-classification	2407	294	2
segment	2310	19	7
shuttle	10154	9	7
sonar	208	60	2
spambase	4601	57	2
tic-tac	958	9	2
vehicle	846	18	4
vowel	990	10	11
waveform-5000	5000	40	3
wdbc	569	30	2
wpbc	194	33	2
yeast	1484	8	10
zoo	101	16	7

TABLE I: Description of the 41 classification datasets from UCI used for the experiments

	BC	O-norm	T-norm
abalone	48.90(3)	49.08(2)	**49.94(1)**
autos	71.75(3)	**74.96(1)**	74.75(2)
balance-	62.36(3)	93.68(2)	**93.86(1)**
car	88.54(2)	**91.53(1)**	79.96(3)
clevelan	**67.54(1)**	44.76(3)	63.63(2)
credit-a	64.27(3)	79.90(2)	**81.01(1)**
dermatol	**82.51(1)**	82.33(2)	82.26(3)
diabetes	**78.92(1)**	75.10(3)	78.44(2)
ecoli	83.83(2)	82.53(3)	**84.47(1)**
flare	**59.21(1)**	57.36(3)	58.30(2)
german	78.61(2)	77.71(3)	**79.44(1)**
glass	65.08(2)	64.73(3)	**71.07(1)**
heart-st	**80.66(1)**	80.03(2)	78.93(3)
hepatiti	**84.99(1)**	66.16(3)	84.21(2)
horse	78.63(2)	69.48(3)	**82.07(1)**
ionosphe	**87.64(1)**	82.15(3)	83.15(2)
iris	**80.08(1)**	79.66(3)	79.8(2)
landsat-	66.25(3)	**84.13(1)**	83.13(2)
letter	72.01(3)	**79.52(1)**	77.12(2)
libras-m	**46.01(1)**	43.38(2.5)	43.38(2.5)
lung-can	**34.58(1)**	34.20(2.5)	34.20(2.5)
mfeat-ka	**84.11(1)**	33.41(2)	33.39(3)
mfeat-mo	71.42(3)	76.25(2)	**77.45(1)**
mfeat-ze	**75.88(1)**	60.30(2)	59.98(3)
mushroom	88.05(3)	**99.77(1)**	99.61(2)
optdigit	87.25(3)	**90.88(1)**	87.82(2)
page-blo	**94.13(1)**	73.70(3)	90.85(2)
pendigit	78.29(3)	**91.99(1)**	86.58(2)
scene-cl	**84.81(1)**	33.37(2.5)	33.37(2.5)
segment	82.80(3)	**91.80(1)**	90.63(2)
shuttle	78.66(3)	82.43(2)	**83.93(1)**
sonar	**65.00(1)**	36.07(2.5)	36.07(2.5)
spambase	78.36(3)	**85.88(1)**	82.55(2)
tic-tac	75.25(2)	72.81(3)	**77.49(1)**
vehicle	63.89(3)	**72.73(1)**	69.18(2)
vowel	71.58(3)	**74.80(1)**	72.91(2)
waveform	**86.44(1)**	53.54(3)	53.66(2)
wdbc	**88.57(1)**	84.72(2)	82.81(3)
wpbc	**64.29(1)**	61.60(3)	61.92(2)
yeast	59.03(2)	53.74(3)	**67.47(1)**
zoo	**86.75(1)**	86.70(2)	85.34(3)
Average	**74.32(1.90)**	70.95(2.14)	72.59(1.95)

TABLE II: Mean accuracies for each dataset and 20 iterations of 5-fold cross-validation for Background Check, O-norm and T-norm methods [5]. The number in brackets represent the rankings of the three methods per dataset.

method	Accuracy EP-CC	BC	Log-loss EP-CC	BC
abalone	55.06 ± 1.5	$\mathbf{55.36 \pm 1.4}^{*}$	3.94 ± 0.7	$\mathbf{3.32 \pm 0.7}^{***}$
autos	67.54 ± 9.2	$\mathbf{69.49 \pm 7.2}^{*}$	1.03 ± 0.5	$\mathbf{0.46 \pm 0.3}^{***}$
balance-sc	$\mathbf{91.21 \pm 2.3}^{***}$	90.54 ± 2.1	0.96 ± 0.3	$\mathbf{0.54 \pm 0.4}^{***}$
car	71.61 ± 1.7	$\mathbf{71.63 \pm 0.9}$	2.60 ± 0.2	$\mathbf{2.26 \pm 0.3}^{***}$
cleveland	55.95 ± 5.0	$\mathbf{58.44 \pm 3.1}^{***}$	1.97 ± 0.5	$\mathbf{1.36 \pm 0.6}^{***}$
credit-app	85.85 ± 2.9	$\mathbf{86.14 \pm 2.8}^{*}$	$\mathbf{9.40 \pm 0.5}$	9.45 ± 0.6
dermatolog	96.40 ± 2.2	$\mathbf{96.45 \pm 2.2}$	0.21 ± 0.1	$\mathbf{0.05 \pm 0.1}^{***}$
diabetes	76.66 ± 2.6	$\mathbf{77.13 \pm 2.9}^{**}$	10.30 ± 0.6	$\mathbf{10.14 \pm 0.9}$
ecoli	$\mathbf{85.23 \pm 3.6}$	84.20 ± 5.5	0.58 ± 0.2	$\mathbf{0.37 \pm 0.2}^{***}$
flare	39.74 ± 2.8	$\mathbf{42.82 \pm 2.3}^{***}$	2.96 ± 0.5	$\mathbf{2.19 \pm 0.7}^{***}$
german	$\mathbf{75.12 \pm 2.5}$	74.90 ± 2.2	$\mathbf{3.18 \pm 0.7}$	3.30 ± 1.0
glass	$\mathbf{64.50 \pm 6.8}^{***}$	62.02 ± 6.4	1.62 ± 0.5	$\mathbf{0.95 \pm 0.5}^{***}$
heart-stat	81.85 ± 5.1	$\mathbf{83.13 \pm 5.2}^{***}$	6.48 ± 1.0	$\mathbf{5.21 \pm 1.3}^{***}$
hepatitis	82.21 ± 5.9	$\mathbf{83.65 \pm 5.0}^{*}$	11.77 ± 1.8	$\mathbf{10.96 \pm 2.2}^{*}$
horse	78.69 ± 5.5	$\mathbf{80.94 \pm 4.1}^{***}$	4.80 ± 1.0	$\mathbf{2.47 \pm 1.1}^{***}$
ionosphere	86.43 ± 3.4	$\mathbf{88.82 \pm 3.2}^{***}$	11.81 ± 0.1	$\mathbf{9.48 \pm 0.8}^{***}$
iris	96.43 ± 3.2	$\mathbf{96.73 \pm 3.1}$	0.41 ± 0.4	$\mathbf{0.20 \pm 0.3}^{***}$
landsat-sa	86.45 ± 0.8	$\mathbf{86.79 \pm 0.8}^{***}$	0.53 ± 0.1	$\mathbf{0.36 \pm 0.1}^{***}$
letter	80.96 ± 1.5	$\mathbf{81.40 \pm 1.3}^{**}$	0.15 ± 0.0	$\mathbf{0.09 \pm 0.0}^{***}$
libras-mov	$\mathbf{79.31 \pm 4.0}^{***}$	76.54 ± 4.5	0.27 ± 0.1	$\mathbf{0.16 \pm 0.1}^{***}$
lung-cance	98.70 ± 2.8	$\mathbf{99.42 \pm 1.6}^{*}$	$\mathbf{16.50 \pm 0.0}$	$\mathbf{16.50 \pm 0.0}$
mfeat-karh	95.50 ± 1.0	$\mathbf{96.64 \pm 0.9}^{***}$	0.08 ± 0.0	$\mathbf{0.04 \pm 0.0}^{***}$
mfeat-morp	$\mathbf{73.63 \pm 1.8}$	73.46 ± 1.8	0.59 ± 0.1	$\mathbf{0.49 \pm 0.1}^{***}$
mfeat-zern	81.38 ± 1.2	$\mathbf{82.96 \pm 1.3}^{***}$	0.37 ± 0.1	$\mathbf{0.09 \pm 0.0}^{***}$
mushroom	$\mathbf{98.86 \pm 0.3}^{***}$	98.44 ± 0.5	8.83 ± 0.1	$\mathbf{8.65 \pm 0.2}^{***}$
optdigits	97.72 ± 0.5	$\mathbf{98.51 \pm 0.3}^{***}$	0.04 ± 0.0	$\mathbf{0.02 \pm 0.0}^{***}$
page-block	96.00 ± 0.5	$\mathbf{96.14 \pm 0.4}^{**}$	0.28 ± 0.0	$\mathbf{0.22 \pm 0.0}^{***}$
pendigits	98.08 ± 0.3	$\mathbf{98.25 \pm 0.2}^{***}$	0.06 ± 0.0	$\mathbf{0.03 \pm 0.0}^{***}$
scene-clas	76.33 ± 2.0	$\mathbf{80.10 \pm 1.3}^{***}$	4.04 ± 0.2	$\mathbf{1.72 \pm 0.8}^{***}$
segment	94.80 ± 1.0	$\mathbf{94.85 \pm 0.9}$	0.24 ± 0.1	$\mathbf{0.15 \pm 0.1}^{***}$
shuttle	$\mathbf{97.68 \pm 0.3}^{***}$	97.03 ± 0.4	$\mathbf{0.10 \pm 0.0}^{***}$	0.13 ± 0.0
sonar	73.57 ± 6.4	$\mathbf{77.76 \pm 5.7}^{***}$	4.62 ± 1.2	$\mathbf{2.19 \pm 0.8}^{***}$
spambase	92.71 ± 0.8	$\mathbf{92.83 \pm 0.8}$	6.82 ± 0.3	$\mathbf{6.12 \pm 0.4}^{***}$
tic-tac	$\mathbf{67.44 \pm 3.0}^{***}$	65.42 ± 0.9	$\mathbf{9.44 \pm 1.6}^{***}$	11.70 ± 0.8
vehicle	78.70 ± 2.7	$\mathbf{79.64 \pm 2.6}^{***}$	1.21 ± 0.4	$\mathbf{0.60 \pm 0.3}^{***}$
vowel	77.58 ± 3.3	$\mathbf{78.62 \pm 2.6}^{***}$	0.38 ± 0.2	$\mathbf{0.14 \pm 0.1}^{***}$
waveform-5	84.95 ± 1.0	$\mathbf{86.17 \pm 0.9}^{***}$	1.40 ± 0.4	$\mathbf{0.59 \pm 0.2}^{***}$
wdbc	96.07 ± 1.7	$\mathbf{97.29 \pm 1.5}^{***}$	6.61 ± 0.3	$\mathbf{6.05 \pm 0.4}^{***}$
wpbc	74.68 ± 7.1	$\mathbf{78.52 \pm 4.2}^{***}$	$\mathbf{2.07 \pm 1.0}$	2.37 ± 1.3
yeast	58.89 ± 2.4	$\mathbf{59.15 \pm 2.4}$	1.29 ± 0.1	$\mathbf{1.02 \pm 0.2}^{***}$
zoo	92.71 ± 3.1	$\mathbf{95.14 \pm 3.6}^{***}$	0.38 ± 0.2	$\mathbf{0.08 \pm 0.1}^{***}$
Average	81.54 ± 14.19	$\mathbf{82.27 \pm 13.89}^{***}$	3.42 ± 4.12	$\mathbf{2.98 \pm 4.12}^{***}$

TABLE III: Mean and standard deviation of accuracy and log-loss on 41 datasets. Obtained from 20 iterations of 5-fold cross-validation. A Wilcoxon signed rank-sum test was performed for each metric and dataset; * significant at $p < 0.05$; ** significant at $p < 0.005$; *** significant at $p < 0.001$.