The algorithm versus the chimps: On the minima of classifier performance metrics

Joep Burger[1][0000−0002−7298−5561] and
Quinten Meertens[2,3,4][0000−0002−3485−8895]

[1] Statistics Netherlands, Research and Development, CBS-weg 11, PO Box 4481, 6401 CZ Heerlen, the Netherlands
j.burger@cbs.nl
[2] Center for Nonlinear Dynamics in Economics and Finance, University of Amsterdam, Roetersstraat 11, PO Box 15867, 1001 NJ Amsterdam, the Netherlands
[3] Leiden Centre of Data Science, Leiden University, the Netherlands
[4] Statistics Netherlands, Business Statistics (The Hague), the Netherlands
q.a.meertens@uva.nl

Abstract. In this paper we seek the minima of performance metrics for binary classification to facilitate comparison between metrics and applications, and to assess the quality of inferential statistics made from non-probability samples. We use these minima to min-max normalize the performance metrics so that they can be interpreted as a percentage of the perfect classifier relative to the proverbial chimps at the zoo[†] guessing at random. We compare our results with the balanced metrics that have been introduced recently, which are corrected for bias due to class imbalance.

Keywords: Inferential statistics · Non-probability samples · Statistical learning · Supervised machine learning · Binary classification.

1 Introduction

Imagine you have to do a school exam. The test consists of one hundred multiple choice questions. At each question you can choose from four possible answers (A, B, C or D). In the Netherlands, students pass when they score 6 points or more on a scale from 0 to 10. Is it sufficient to answer sixty questions correctly to pass the test? Maybe not, because the proverbial chimps at the zoo that choose answers randomly will on average answer twenty five questions correctly. The teacher could correct for that minimum by grading twenty five correct answers a 0 and then scaling linearly to maintain a perfect score when no mistakes are made by the student, Then, a student answering sixty questions correctly fails the test, having a score of 4.7 on the scale from 0 to 10. The student would now have to answer at least seventy questions correctly to pass the test. From a statistical point of view, an advantage of this so-called min-max normalization,

[†] Inspired by Swedish physician Hans Rosling (1948–2017).

is that the students' grades are now comparable with the grades at a second school where the students can choose between two possible answers (A or B). The proverbial chimps at the zoo that answer randomly will then answer fifty question correctly, on average. After min-max normalization, a student at that second school will only pass the test when answering eighty or more questions correctly.

In supervised machine learning, algorithms instead of students are trained to find the right answer to a multiple choice question. The algorithm learns, for example, that in the case of a drawing the subject is a 'moon', 'rose' or 'fish'. (These are the first words Dutch children learn to read. Many algorithms are still in elementary school and it is nice to use something else than pictures of cats and cars as an example.) What final grade does an algorithm get for its answers to new drawings? For this purpose a considerable list of performance metrics has been developed (see, e.g., [7]). For some of them it is unclear what the score of the control group would be: how well would the chimps at the zoo perform?

In this paper, we will provide an answer to that question. The answer is important from a statistical point of view: as students' grades should preferably be comparable between subjects and schools, we would like the performance of algorithms to be comparable between metrics and applications. Moreover, we would like to have a statistical interpretation of the actual value of a performance metric, such that the interpretation is independent of the metric and application, similar to the min-max normalized grade of a multiple-choice test: a grade equal to 6 can always be interpreted as 60 percent between guessing and perfection. That statistical interpretation is essential when employing supervised machine learning algorithms at national statistical institutes, such as Statistics Netherlands, to produce official statistics.

Official statistics provide quantitative information about the status and development of well-defined populations such as businesses or households. One of the challenges is to produce such information at reasonable accuracy, cost and time. A burning question in official statistics is how to make inference from non-probability (NP) samples [8]. NP samples like social media messages or sensor data can offset some disadvantages of questionnaires sent to units in a probability sample, such as response burden, high costs and a considerable time lag between data collection and dissemination [1]. However, not all units in the population of interest have a positive and known probability of being included in an NP sample. This rules out design-based estimators from sampling theory.

Alternatively, the data-generating mechanism of NP samples can be deduced by modeling the relationship between features and the target variable in the NP sample and use it to predict the missing data, assuming they are missing at random. Statistical models could then be deployed, but more often machine learning algorithms are used, because they are designed for prediction or extrapolation and scale better with the number of features.

To assess the quality of the extrapolations, the quality of the predictions are assessed on test sets for which the actual value is known. A range of performance

metrics exists to this end [7]. However, as noted before, it remains unclear what the proverbial chimps at the zoo would achieve by randomly guessing the value of the target variable. A typical example is high accuracy in imbalanced datasets: if a class has a relative frequency of 95%, it is easy to obtain a seemingly impressive accuracy of 95% by always 'predicting' the most common class.

In this paper we seek the minima of performance metrics for binary classification to facilitate comparison between metrics and to assess the quality of inferential statistics made from NP samples. We use these minima to min-max normalize the performance metrics so that they can be interpreted as percentage of perfection relative to the performance of the proverbial chimps at the zoo. We compare our results with balanced metrics [6], which have been corrected for bias due to class imbalance. In our view, the paper is a methodological contribution with preliminary simulation results that encourage a more thorough experimental study in the future.

2 Imbalanced performance metrics

We assume that we have a test set of n data points, n_1 of which are labeled positive: $y_i = 1$, where y_i is the observed class of instance i. The fraction $\alpha = n_1/n$ is referred to as the base rate. The algorithm trained on a training set of $N - n$ data points predicts for all n instances the probability that instance i belongs to the positive class: $\hat{p}_i = \mathbb{P}(y_i = 1)$. By choosing a cutoff $0 \leq c \leq 1$ above which the probability $\hat{p}_i$ is assigned to the positive class, a 2×2 contingency table or confusion matrix can be constructed (Table 1). Optimizing cutoff c is discussed in Section 4.

Table 1. Confusion matrix for cutoff c. Cells highlighted in gray can be used to derive all other cells and metrics.

	Predicted Positive	Negative	$\sum$	
Actual Positive	X_c	$n_1 - X_c$	n_1	$TPR_c = \frac{X_c}{n_1}$
Negative	Y_c	$n_2 - Y_c$	$n_2 := n - n_1$	$TNR_c = \frac{n_2 - Y_c}{n_2} = 1 - \frac{Y_c}{n_2}$
$\sum$	$X_c + Y_c$	$n - X_c - Y_c$	n	
	$PPV_c = \frac{X_c}{X_c + Y_c}$	$NPV_c = \frac{n_2 - Y_c}{n - X_c - Y_c}$	$\alpha = \frac{n_1}{n}$	

From this confusion matrix the following well-known performance metrics are derived (left two columns of Table 2). Accuracy (ACC_c) is the fraction of all cases that is predicted correctly. The true positive rate (TPR_c), also known as sensitivity or recall, is the fraction of positively labeled cases that is predicted correctly and the true negative rate (TNR_c), also known as specificity, is the fraction of negatively labeled cases that is predicted correctly. The positive predictive value (PPV_c), also known as precision, is the fraction of predicted

positive cases that is actually labeled 'positive' and the negative predictive value (NPV_c) is the fraction of predicted negative cases that is actually labeled 'negative'. The receiver operating characteristic curve, or ROC curve, plots TPR_c against the complement of TNR_c for $0 \leq c \leq 1$. The area under the ROC curve (AUC) is used as a performance metric. Note that TPR_c will decrease with c whereas TNR_c will increase with c. This trade-off is captured by Youden's J index (J_c) or Peirce Skill Score, which is also the vertical distance between the ROC curve and the diagonal. Note also that PPV_c will become unstable at higher values of c, whereas NPV_c will become unstable at lower values of c, because the respective denominators decrease there. This trade-off is captured by markedness (MRK_c). The Matthews correlation coefficient (MCC_c) is the correlation between the actual and predicted binary classifications. The positive F_1 score (PF_{1c}) is the harmonic mean of TPR_c and PPV_c. Analogously, the negative F_1 score (NF_{1c}) is the harmonic mean of TNR_c and NPV_c. The harmonic mean is more sensitive to one of the values being low than the arithmetic mean.

Table 2. Imbalanced performance metrics and their expected value when randomly guessing the positive class with probability g.

Metric Q	Definition [7]	$\mathbb{E}[Q(g)]$
ACC_c	$\frac{n_2+X_c-Y_c}{n}$	$\alpha g + (1-\alpha)(1-g)$
TPR_c	$\frac{X_c}{n_1}$	g
TNR_c	$1-\frac{Y_c}{n_2}$	$1-g$
PPV_c	$\frac{X_c}{X_c+Y_c}$	$\alpha + O(\frac{1}{n^2})$
NPV_c	$\frac{n_2-Y_c}{n-X_c-Y_c}$	$1-\alpha+O(\frac{1}{n^2})$
AUC	$\int_{c=0}^{1} TPR_c dTNR_c$	$\frac{1}{2}$
J_c	$TPR_c + TNR_c - 1$	0
MRK_c	$PPV_c + NPV_c - 1$	$0+O(\frac{1}{n^2})$
MCC_c	$\frac{n_2X_c-n_1Y_c}{\sqrt{n_1n_2(X_c+Y_c)(n-X_c-Y_c)}}$	$0+O(\frac{1}{n^2})$
PF_{1c}	$\frac{2}{\frac{1}{TPR_c}+\frac{1}{PPV_c}}$	
	$=\frac{2X_c}{n_1+X_c+Y_c}$	$2\alpha g\left(\frac{1}{\alpha+g}-\frac{\alpha(1-g)}{n(\alpha+g)^3}\right)+O\left(\frac{1}{n^2}\right)$
NF_{1c}	$\frac{2}{\frac{1}{TNR_c}+\frac{1}{NPV_c}}$	
	$=\frac{2(n_2-Y_c)}{n+n_2-X_c-Yc}$	$2(1-\alpha)(1-g)\left(\frac{1}{2-\alpha-g}-\frac{(1-\alpha)g}{n(2-\alpha-g)^3}\right)+O\left(\frac{1}{n^2}\right)$

We will now formally introduce how to model the outcome of the predictions made by the proverbial chimps at the zoo. To that end, let g be the probability that a chimp at the zoo predicts the positive class. We assume that the chimps will all guess according to one and the same strategy out of the following three: they may toss a fair coin, throw a dice with n sizes, n_1 of which are labeled 'positive', or always guess the most common class (the mode), i.e.:

$$g^{\text{unif}} = \frac{1}{2}$$
$$g^{\text{prop}} = \alpha$$
$$g^{\text{mode}} = \begin{cases} 1 & \text{if } \alpha > \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

Then, let X and Y be independently distributed random variables with binomial distributions $X \sim \text{Bin}(n_1, g)$ and $Y \sim \text{Bin}(n_2, g)$. Each evaluation metric is a random variable as well. Table 3 gives the expected confusion matrix. The third column of Table 2 shows how to compute the expected value of each performance metric. The proofs are provided in Appendix A.1. Five of the metrics are linear functions in the random variables X and Y, hence, it is trivial to compute their expected value. The expected value of the other six metrics take the form $\mathbb{E}[f(X, Y)]$ for a nonlinear, real-valued function f. If n is very small, these expectations could in theory be computed by the closed form expression

$$\mathbb{E}[f(X,Y)] = \sum_{l=0}^{n_1} \sum_{m=0}^{n_2} f(l,m)\mathbb{P}(X=l)\mathbb{P}(Y=m).$$

In practice, however, it might take a relatively long time to evaluate this expression if n gets large. So, unless n is small, the approximations given in Table 2 should be used. In addition, note that both X and Y have a (very small, but strictly) positive probability of being 0, in which case f might not be defined (e.g., for PPV we find $0/0$). Mathematically, the correct way to deal with this is to exclude the event by conditioning the expectations on its complement. In practice, in particular for larger values of n, the obtained value will be very close to simply skipping the terms in the summation where f is not defined.

Table 3. Expected confusion matrix when randomly guessing the positive class with probability g.

		Predicted Positive	Predicted Negative	$\sum$
Actual	Positive	$n_1 g$	$n_1(1-g)$	n_1
	Negative	$n_2 g$	$n_2(1-g)$	n_2
	$\sum$	ng	$n(1-g)$	n

3 Balanced performance metrics

Some performance metrics are biased due to class imbalance. Balanced performance metrics are obtained by rewriting the imbalanced performance metrics

as a function of the imbalance coefficient $\delta = 2\alpha - 1$ and setting δ to 0, i.e. α to $\frac{1}{2}$ [6]. Table 4 shows the balanced metrics and an approximation of their expected value when randomly guessing the positive class with probability g. The derivations of the formulas in the third column can be found in Appendix A.2.

Table 4. Balanced performance metrics and their expected value when randomly guessing the positive class with probability g.

Metric Q^b	Definition [6]	$\mathbb{E}[Q^b(g)]$
ACC_c^b	$\frac{TPR_c+TNR_c}{2}$	$\frac{1}{2}$
TPR_c^b	TPR_c	g
TNR_c^b	TNR_c	$1-g$
PPV_c^b	$\frac{TPR_c}{TPR_c-TNR_c+1}$	$\frac{1}{2} + \frac{\delta(1-g)}{2n(1+\delta)(1-\delta)g} + O\left(\frac{1}{n^2}\right)$
NPV_c^b	$\frac{TNR_c}{TNR_c-TPR_c+1}$	$\frac{1}{2} - \frac{\delta g}{2n(1+\delta)(1-\delta)(1-g)} + O\left(\frac{1}{n^2}\right)$
AUC^b	$2AUC - 1$	0
J_c^b	J_c	0
MRK_c^b	$PPV_c^b + NPV_c^b - 1$	$\frac{\delta(1-2g)}{2n(1+\delta)(1-\delta)g(1-g)} + O\left(\frac{1}{n^2}\right)$
MCC_c^b	$\frac{TPR_c+TNR_c-1}{\sqrt{(TPR_c-TNR_c+1)(TNR_c-TPR_c+1)}}$	$\frac{\delta(1-2g)}{2n(1+\delta)(1-\delta)\sqrt{g(1-g)}} + O\left(\frac{1}{n^2}\right)$
PF_{1c}^b	$\frac{2TPR_c}{TPR_c-TNR_c+2}$	$2g\left(\frac{1}{1+2g} - \frac{2\left(1-\delta(1+2g)\right)(1-g)}{n(1+\delta)(1-\delta)(1+2g)^3}\right) + O\left(\frac{1}{n^2}\right)$
NF_{1c}^b	$\frac{2TNR_c}{TNR_c-TPR_c+2}$	$2(1-g)\left(\frac{1}{3-2g} - \frac{2\left(1+\delta(3-2g)\right)g}{n(1+\delta)(1-\delta)(3-2g)^3}\right) + O\left(\frac{1}{n^2}\right)$

4 Min-max normalization and optimization

After establishing $\mathbb{E}[Q(g)]$, min-max normalization can be applied to rescale each metric so that the proverbial chimps at the zoo score 0, on average:

$$Q_c^{mmn}(g) = \frac{Q_c - \mathbb{E}[Q(g)]}{1 - \mathbb{E}[Q(g)]}. \tag{1}$$

Note that $ACC^{mmn}(g)$ equals the Heidke Skill Score [4] or Cohen's κ [3] if g is set to $\frac{X+Y}{n}$. This g is, however, not a random guessing probability. The Heidke Skill Score min-max normalizes accuracy with the expected cell frequencies, which depend on the model.

Through K-fold cross validation or bootstrapping, we obtain the quality of K classifiers trained on different partitions or bootstrap samples of the data. We propose to first average Q_{kc}^{mmn} per cutoff to determine the overall optimal cutoff c^*, that is:

$$c^* = \arg\max_c \overline{Q}_c^{mmn}, \tag{2}$$

in which

$$\overline{Q}_c^{mmn} = \frac{1}{K}\sum_{k=1}^{K} Q_{kc}^{mmn}. \tag{3}$$

Then, we propose to use the distribution of $Q_k^{mmn}(c^*)$ as a proxy for the quality of the predictions when applied to unlabeled data for making statistical inference.

5 Example: normalized F_1 scores

Figure 1 shows the F_1 performance of a fictitious binary classifier that predicts 60% of the actual positive instances correctly ($TPR = 0.6$) and 80% of the actual negative cases ($TNR = 0.8$), using g^{unif} for normalization. Similar figures for accuracy and F_1 with g^{prop} can be found in Appendix B. When the test set is balanced ($\delta = 0$, i.e. $\alpha = 0.5$), this classifier scores $PF_1 = \frac{2}{3}$ and $NF_1 = \frac{8}{11}$. The more abundant the positive class relative to the negative class, the higher the classifier scores on PF_1 and the lower on NF_1 (thin red line in left panels). One solution to this sensitivity to class imbalance is to balance the metric (thin red lines in right panels) by correcting for the bias. The alternative we propose is to min-max normalize the metric (thick red line in left panels) by relating it to the expected value when randomly guessing the positive class with probability g (thin blue lines).

Two interesting observations can be made. First, data sets with a different imbalance coefficient can be compared. The classifier performs best at $\delta \approx -0.18$ where $PF_1^{mmn} \approx 0.34$, i.e. 34% from perfection relative to tossing a fair coin. A classifier with the same TPR and TNR in an application with a higher δ scores better on PF_1 (up to $\frac{3}{4}$), the same on PF_1^b but worse on PF_1^{mmn}. Second, metrics can be compared. Before min-max normalization, the classifier scores equally well on PF_1 and NF_1 at $\delta = \frac{1}{7}$ (small white points). After min-max normalization, however, the classifier scores equally well on PF_1^{mmn} and NF_1^{mmn} at $\delta \approx 0.5$ (large white points). Between $\frac{1}{7} < \delta < 0.5$, $PF_1 > NF_1$ but $PF_1^{mmn} < NF_1^{mmn}$.

Note that $\mathbb{E}[F_1]$ is sensitive to sample size n (see Tables 2 and 4), which becomes apparent when the metric is balanced and the sample is highly imbalanced (Fig. 1, right panel, blue line).

6 Conclusion

In this paper, we propose to rescale performance metrics through min-max normalization, where the minimum is set to the expected value when randomly guessing the positive class with probability g. It should be explicitly specified which expected value the algorithm is trying to defeat. Our proposed normalization yields different results than correcting for bias due to class imbalance [6] or balancing the sample [e.g. 2; 5]. The min-max normalized metrics allow for a better comparison between applications and between metrics. Moreover, we propose to use the distribution across test sets of a normalized metric at the overall optimal cutoff as performance metric for inferential statistics. Future research could

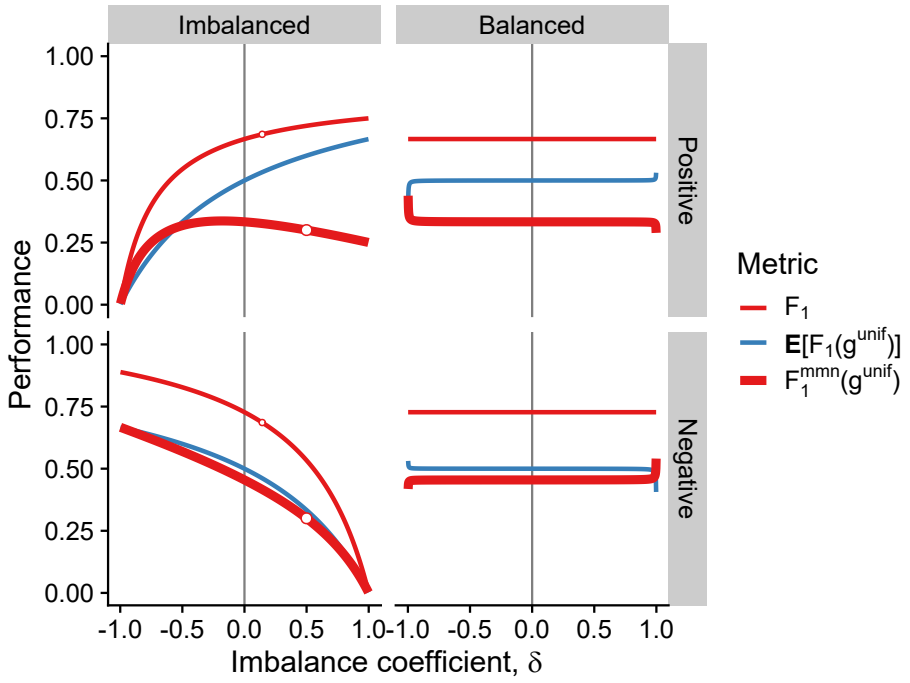

Fig. 1. Performance of a fictitious binary classifier in relation to imbalance coefficient δ. $g = 0.5$, $TPR = 0.6$, $TNR = 0.8$, $n = 1000$. White points show where positive and negative F_1 intersect.

focus on generalizing the results from binary classification to multi-class classification and regression, and on metrics that compare the predicted probability directly with the actual label, without a cutoff for constructing the confusion matrix.

Bibliography

[1] Baker, R., Brick, J.M., Bates, N.A., Battaglia, M., Couper, M.P., Dever, J.A., Gile, K.J., Tourangeau, R.: Summary report of the AAPOR task force on non-probability sampling. Journal of Survey Statistics and Methodology **1**, 90–143 (2013). https://doi.org/doi.org/10.1093/jssam/smt008

[2] Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: Synthetic minority over-sampling technique. J. Artif. Int. Res. **16**(1), 321–357 (Jun 2002). https://doi.org/10.5555/1622407.1622416

[3] Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement **20**(1), 37–46 (1960). https://doi.org/10.1177/001316446002000104

[4] Heidke, P.: Berechnung des erfolges und der gute der windstärkevorhersagen im sturmwarnungsdienst (measures of success and goodness of wind force forecasts by the gale-warning service). Geografiska Annaler **8**, 301–349 (1926). https://doi.org/10.1080/20014422.1926.11881138

[5] Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. Progress in Artificial Intelligence **5**, 221–232 (2016). https://doi.org/10.1007/s13748-016-0094-0

[6] Luque, A., Carrasco, A., Martín, A., de las Heras, A.: The impact of class imbalance in classification performance metrics based on the binary confusion matrix. Pattern Recognition **91**, 216–231 (2019). https://doi.org/10.1016/j.patcog.2019.02.023

[7] Powers, D.M.W.: Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation. Journal of Machine Learning Technologies **2**(1), 37–63 (2011)

[8] Wu, C., Thompson, M.E.: Non-probability survey samples. In: Sampling Theory and Practice, pp. 319–331. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-44246-0$_1$7

A Appendix - Proof of expected values

This appendix belongs to the paper by Burger & Meertens titled "The algorithm versus the chimps: On the minima of classifier performance metrics". It contains the proofs of the formulas provided in the third column of Table 2 and Table 4.

The key idea is to approximate an expectation of the form $\mathbb{E}[f(X,Y)]$, in which X and Y are random variables and in which f is an infinitely differentiable real-valued function, by inserting the Taylor series of f at $(\mathbb{E}[X], \mathbb{E}[Y])$. More specifically, let $x_0 = \mathbb{E}[X]$ and $y_0 = \mathbb{E}[Y]$ and consider the second-order Taylor series of f at (x_0, y_0):

$$\begin{aligned} f(x,y) \approx\ & f(x_0,y_0) + f_x(x_0,y_0)(x-x_0) + f_y(x_0,y_0)(y-y_0) \\ & + \frac{1}{2} f_{xx}(x_0,y_0)(x-x_0)^2 + \frac{1}{2} f_{yy}(x_0,y_0)(y-y_0)^2 \\ & + f_{xy}(x_0,y_0)(x-x_0)(y-y_0). \end{aligned} \tag{4}$$

We then approximate the expectation $\mathbb{E}[f(X,Y)]$ by taking the expectation of the right-hand side of the above equation. Then, assuming that X and Y are uncorrelated, we find

$$\mathbb{E}[f(X,Y)] \approx f(x_0,y_0) + \frac{1}{2} f_{xx}(x_0,y_0)\operatorname{Var}(X) + \frac{1}{2} f_{yy}(x_0,y_0)\operatorname{Var}(y). \tag{5}$$

In the proofs below we will specify the order of the approximation in terms of the size n of the test dataset.

In the remainder of the appendix, n_1 and n_2 are positive integers that sum up to n and $g \in (0,1)$ represents the probability that class 1 is predicted by the proverbial chimps at the zoo. Moreover, X will be a random variable distributed as $\mathrm{Bin}(n_1, g)$ and Y a random variable distributed as $\mathrm{Bin}(n_2, g)$. The random variables X and Y are assumed to be independent. The expectation and variance are given by

$$x_0 = \mathbb{E}[X] = n_1 g, \quad \operatorname{Var}(X) = n_1 g(1-g), \tag{6}$$

and

$$y_0 = \mathbb{E}[Y] = n_2 g, \quad \operatorname{Var}(Y) = n_2 g(1-g). \tag{7}$$

Finally, we will use the notation $\alpha = n_1/n$ (and hence $1-\alpha = n_2/n$) and $\delta = 2\alpha - 1 = (n_1 - n_2)/n$.

A.1 Expected value of imbalanced metrics

This appendix contains the derivations of the approximations of the expected values of the imbalanced performance metrics, as presented in Table 2 of the main text.

Expected Positive Predictive Value ($\mathbb{E}[\boldsymbol{PPV}]$)

The positive predictive value PPV can be written as $f(X, Y)$ with $f(x, y) = x/(x + y)$. Check that

$$f_{xx}(x, y) = \frac{-2y}{(x + y)^3}, \quad f_{yy}(x_0, y_0) = \frac{2x}{(x + y)^3}. \tag{8}$$

It follows that

$$\mathbb{E}[PPV] \approx \frac{n_1 g}{ng} - \frac{n_2 g}{(ng)^3} \cdot n_1 g(1 - g) + \frac{n_1 g}{(ng)^3} \cdot n_2 g(1 - g) = \frac{n_1}{n} = \alpha. \tag{9}$$

The higher order terms in the Taylor series of PPV are in $O(1/n^2)$. It can be shown by looking at the terms or order 3 in the Taylor series. Only f_{xxx} and f_{yyy} remain, which are both $O(1/n^3)$ when evaluated at (x_0, y_0), and the third central moment of both X and Y are $O(n)$.

Expected Negative Predictive Value ($\mathbb{E}[\boldsymbol{NPV}]$) The negative predictive value NPV can be viewed as the positive predictive value for the negative class, i.e., to compute NPV we first swap the roles of n_1 and n_2 and replace g by $1 - g$ and then compute PPV. It follows that

$$\mathbb{E}[NPV] = 1 - \alpha + O\left(\frac{1}{n^2}\right). \tag{10}$$

Expected Area under the ROC curve ($\mathbb{E}[\boldsymbol{AUC}]$) If the threshold value c is equal to 1, then any coin toss prediction by the chimps is considered as tails, corresponding to the point $(0, 0)$ on the ROC curve. Similarly, $c = 0$ corresponds to the point $(1, 1)$ on the ROC curve. For any other value of the threshold value c, the predictions by the chimps do not depend on c, and thus we have $TPR_c = X/n_1$ and $1 - TNR_c = Y/n_2$, for any $0 < c < 1$. The ROC curve can be obtained by connected these three points, resulting in (the random variable!)

$$\begin{aligned} AUC &= \frac{1}{2}\frac{Y}{n_2}\frac{X}{n_1} + \left(1 - \frac{Y}{n_2}\right)\frac{X}{n_1} + \frac{1}{2}\left(1 - \frac{Y}{n_2}\right)\left(1 - \frac{X}{n_1}\right) \\ &= \frac{1}{2}\left(\frac{X}{n_1} + 1 - \frac{Y}{n_2}\right). \end{aligned} \tag{11}$$

It then follows that $\mathbb{E}[AUC] = \frac{1}{2}$.

Expected Matthews Correlation Coefficient ($\mathbb{E}[\boldsymbol{MCC}]$) The Matthews Correlation Coefficient (MCC) can be written as $f(X, Y)$ in which

$$f(x, y) = \frac{n_2 x - n_1 y}{\sqrt{n_1 n_2 (x + y)(n - x - y)}}. \tag{12}$$

Introducing the function $D(x,y) = n(x+y) - (x+y)^2$, the above simplifies to

$$f = (n_1 n_2)^{-\frac{1}{2}}(n_2 x - n_1 y)D^{-\frac{1}{2}}. \tag{13}$$

Both first order partial derivatives of D are equal to $n - 2(x+y)$. The identity $n_2 x_0 - n_1 y_0 = 0$ then implies that only the following term remains in $f_{xx}(x_0, y_0)$:

$$\begin{aligned} f_{xx}(x_0,y_0) &= 2 \cdot (n_1 n_2)^{-\frac{1}{2}} \cdot n_2 \cdot (-\tfrac{1}{2}) \cdot D^{-\frac{3}{2}}(x_0,y_0) \cdot (2 - n(x_0+y_0)) \\ &= \frac{-n_2(1-2g)}{n^2\sqrt{n_1 n_2 g^3 (1-g)^3}}. \end{aligned} \tag{14}$$

Notice that $f_{xx}(x_0,y_0) = O(1/n^2)$, and hence $f_{xx}(x_0,y_0)\operatorname{Var}(X) = O(1/n)$. Similarly, we obtain

$$\begin{aligned} f_{yy}(x_0,y_0) &= 2 \cdot (n_1 n_2)^{-\frac{1}{2}} \cdot (-n_1) \cdot (-\tfrac{1}{2}) \cdot D^{-\frac{3}{2}}(x_0,y_0) \cdot (2 - n(x_0+y_0)) \\ &= \frac{n_1(1-2g)}{n^2\sqrt{n_1 n_2 g^3 (1-g)^3}}. \end{aligned} \tag{15}$$

Interestingly, we have derived that

$$f_{yy}(x_0,y_0)\operatorname{Var}(Y) = -f_{xx}(x_0,y_0)\operatorname{Var}(X). \tag{16}$$

In particular, we have $f_{yy}(x_0,y_0)\operatorname{Var}(Y) = O(1/n)$. Finally, as $f(x_0,y_0) = 0$, we have shown that

$$\mathbb{E}[MCC] = 0 + O\left(\frac{1}{n^2}\right). \tag{17}$$

Expected Positive F_1 ($\mathbb{E}[PF_1]$) The positive F_1 score (PF_1) can be written as $f(X,Y)$ for $f(x,y) = 2x/(n_1 + x + y)$. Check that

$$f_{xx}(x,y) = \frac{-4(n_1+y)}{(n_1+x+y)^3}, \quad f_{yy}(x,y) = \frac{4x}{(n_1+x+y)^3}. \tag{18}$$

We leave it to the reader to check that $f_{xxx}(x_0,y_0) = O(1/n^3)$ and $f_{yyy}(x_0,y_0) = O(1/n^3)$. It follows that

$$\begin{aligned} \mathbb{E}[PF_1] &= \frac{2n_1 g}{n_1+ng} - \frac{2(n_1+n_2 g)}{(n_1+ng)^3} \cdot n_1 g(1-g) + \frac{2n_1 g}{(n_1+ng)^3} \cdot n_2 g(1-g) + O\left(\frac{1}{n^2}\right) \\ &= \frac{2n_1 g}{n_1+ng} - \frac{2n_1^2 g(1-g)}{(n_1+ng)^3} + O\left(\frac{1}{n^2}\right) \\ &= 2n_1 g\left(\frac{1}{n_1+ng} - \frac{n_1(1-g)}{(n_1+ng)^3}\right) + O\left(\frac{1}{n^2}\right) \\ &= 2\alpha g\left(\frac{1}{\alpha+g} - \frac{\alpha(1-g)}{n(\alpha+g)^3}\right) + O\left(\frac{1}{n^2}\right). \end{aligned} \tag{19}$$

Notice that $\mathbb{E}[PF_1(X,Y)] - PF_1(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n)$ and not $O(1/n^2)$. Moreover, the difference is strictly negative.

Expected Negative F_1 ($\mathbb{E}[NF_1]$) The approximation of the expectation of the negative F_1 score (NF_1) can be obtained from that of PF_1 by first swapping n_1 and n_2 and replacing g by $1-g$. In particular, we find

$$\mathbb{E}[NF_1] = 2(1-\alpha)(1-g)\left(\frac{1}{2-\alpha-g} - \frac{(1-\alpha)g}{n(2-\alpha-g)^3}\right) + O\left(\frac{1}{n^2}\right), \quad (20)$$

Again, notice that $\mathbb{E}[NF_1(X,Y)] - NF_1(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n)$ and not $O(1/n^2)$, and that the difference is strictly negative.

A.2 Expected value of balanced metrics

This appendix contains the derivations of the approximations of the expected values of the balanced performance metrics, as presented in Table 4 of the main text. The derivations are similar to those in Appendix A.1, although the outcomes are slightly different.

Expected balanced Positive Predictive Value ($\mathbb{E}[PPV^b]$) The balanced positive predictive value PPV^b can be written as $f(X,Y)$ with $f(x,y) = (x/n_1)/(x/n_1 + y/n_2)$. Check that

$$f_{xx}(x,y) = \frac{-2y/n_2}{n_1^2(x/n_1+y/n_2)^3}, \quad f_{yy}(x_0,y_0) = \frac{2x/n_1}{n_2^2(x/n_1+y/n_2)^3}. \quad (21)$$

It follows that

$$\begin{aligned}\mathbb{E}[PPV^b] &= \frac{1}{2} - \frac{n_1 g^2(1-g)}{n_1^2(2g)^3} + \frac{n_2 g^2(1-g)}{n_2^2(2g)^3} + O\left(\frac{1}{n^2}\right)\\ &= \frac{1}{2} + \frac{(n_1-n_2)(1-g)}{8n_1n_2g} + O\left(\frac{1}{n^2}\right)\\ &= \frac{1}{2} + \frac{\delta(1-g)}{2n(1+\delta)(1-\delta)g} + O\left(\frac{1}{n^2}\right). \quad (22)\end{aligned}$$

Observe that $\mathbb{E}[PPV^b(X,Y)] - PPV^b(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n)$, in contrast to $\mathbb{E}[PPV(X,Y)] - PPV(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n^2)$. However, the absolute value of the term of order $1/n$ can be bounded from above by $(1-g)/(8g)$.

Expected balanced Negative Predictive Value ($\mathbb{E}[NPV^b]$) The balanced negative predictive value NPV^b can be viewed as the balanced positive predictive value for the negative class, i.e., to compute NPV^b we first swap the roles of n_1 and n_2 and replace g by $1-g$ and then compute PPV^b. It follows that

$$\mathbb{E}[NPV^b] = \frac{1}{2} - \frac{\delta g}{2n(1+\delta)(1-\delta)(1-g)} + O\left(\frac{1}{n^2}\right). \quad (23)$$

Again, observe that $\mathbb{E}[NPV^b(X,Y)] - NPV^b(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n)$, in contrast to $\mathbb{E}[NPV(X,Y)] - NPV(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n^2)$. However, the absolute value of the term of order $1/n$ can be bounded from above by $g/(8(1-g))$.

Expected balanced Markedness ($\mathbb{E}[\boldsymbol{MRK^b}]$) The expectation of the balanced markedness (MRK^b) can be approximated as follows:

$$\begin{aligned}\mathbb{E}[MRK^b] &= \mathbb{E}[PPV^b] + \mathbb{E}[NPV^b] - 1 \\ &= \frac{1}{2} + \frac{\delta(1-g)}{2n(1+\delta)(1-\delta)g} + \frac{1}{2} - \frac{\delta g}{2n(1+\delta)(1-\delta)(1-g)} - 1 + O\left(\frac{1}{n^2}\right) \\ &= \frac{\delta(1-2g)}{2n(1+\delta)(1-\delta)g(1-g)} + O\left(\frac{1}{n^2}\right). \end{aligned} \tag{24}$$

It shows that $\mathbb{E}[MRK^b(X,Y)] - MRK^b(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n)$, in contrast to $\mathbb{E}[MRK(X,Y)] - MRK(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n^2)$. However, if $g = \frac{1}{2}$, then the term of order $1/n$ is zero. If $g \neq \frac{1}{2}$, then the absolute value of the term of order $1/n$ can be bounded from above by $(1-2g)/(8g(1-g))$.

Expected balanced Matthews Correlation Coefficient ($\mathbb{E}[\boldsymbol{MCC^b}]$) The balanced Matthews Correlation Coefficient (MCC^b) can be written as $f(X,Y)$ in which

$$f(x,y) = \frac{x/n_1 - y/n_2}{\sqrt{(x/n_1 + y/n_2)(2 - x/n_1 - y/n_1)}}. \tag{25}$$

Introducing the function $D(x,y) = 2(x/n_1 + y/n_2) - (x/n_1 + y/n_2)^2$, the above simplifies to

$$f = (x/n_1 - y/n_2)D^{-\frac{1}{2}}. \tag{26}$$

The first order partial derivatives of D are equal to $2/n_1 \cdot (1 - x/n_1 - y/n_2)$. The identity $x_0/n_1 - y_0/n_2 = 0$ then implies that only the following term remains in $f_{xx}(x_0,y_0)$:

$$\begin{aligned} f_{xx}(x_0,y_0) &= 2 \cdot (1/n_1) \cdot (-\tfrac{1}{2}) \cdot D^{-\frac{3}{2}}(x_0,y_0) \cdot 2/n_1 \cdot (1 - x_0/n_1 - y_0/n_2)) \\ &= \frac{-(1-2g)}{4n_1^2\sqrt{g^3(1-g)^3}} \end{aligned} \tag{27}$$

Similarly, we obtain

$$\begin{aligned} f_{yy}(x_0,y_0) &= 2 \cdot (-1/n_2) \cdot (-\tfrac{1}{2}) \cdot D^{-\frac{3}{2}}(x_0,y_0) \cdot 2/n_2 \cdot (1 - x_0/n_1 - y_0/n_2)) \\ &= \frac{(1-2g)}{4n_2^2\sqrt{g^3(1-g)^3}}. \end{aligned} \tag{28}$$

Finally, as $f(x_0,y_0) = 0$, it follows that

$$\begin{aligned} \mathbb{E}[MCC^b] &= \frac{-(1-2g)n_1 g(1-g)}{8n_1^2\sqrt{g^3(1-g)^3}} + \frac{(1-2g)n_2 g(1-g)}{8n_2^2\sqrt{g^3(1-g)^3}} + O\left(\frac{1}{n^2}\right) \\ &= \frac{(n_1-n_2)(1-2g)}{8n_1 n_2\sqrt{g(1-g)}} + O\left(\frac{1}{n^2}\right) \\ &= \frac{\delta(1-2g)}{2n(1+\delta)(1-\delta)\sqrt{g(1-g)}} + O\left(\frac{1}{n^2}\right). \end{aligned} \tag{29}$$

Once again, observe that $\mathbb{E}[MCC^b(X,Y)] - MCC^b(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n)$, in contrast to $\mathbb{E}[MCC(X,Y)] - MCC(\mathbb{E}[X],\mathbb{E}[Y]) = O(1/n^2)$. However, if $g = \frac{1}{2}$, then the term of order $1/n$ is zero. If $g \neq \frac{1}{2}$, then the absolute value of the term of order $1/n$ can be bounded from above by $(1-2g)/(8\sqrt{g(1-g)})$.

Expected balanced Positive F_1 ($\mathbb{E}[PF_1^b]$) The balanced positive F_1 score (PF_1^b) can be written as $f(X,Y)$ for $f(x,y) = (2x/n_1)/(x/n_1+y/n_2+1)$. Check that

$$f_{xx}(x,y) = \frac{-4(y/n_2+1)}{n_1^2(x/n_1+y/n_2+1)^3}, \quad f_{yy}(x,y) = \frac{4x/n_1}{n_2^2(x/n_1+y/n_2+1)^3}. \quad (30)$$

It follows that

$$\begin{aligned}\mathbb{E}[PF_1^b] &= \frac{2g}{1+2g} - \frac{2n_1(1+g)g(1-g)}{n_1^2(1+2g)^3} + \frac{2n_2g^2(1-g)}{n_2^2(1+2g)^3} + O\left(\frac{1}{n^2}\right) \\ &= 2g\left(\frac{1}{1+2g} - \frac{\Big(n_2-(n_1-n_2)g\Big)(1-g)}{n_1n_2(1+2g)^3}\right) + O\left(\frac{1}{n^2}\right) \\ &= 2g\left(\frac{1}{1+2g} - \frac{2\Big(1-\delta(1+2g)\Big)(1-g)}{n(1+\delta)(1-\delta)(1+2g)^3}\right) + O\left(\frac{1}{n^2}\right) \qquad (31)\end{aligned}$$

The term of order $1/n$ is bounded from above by $2g^2(1-g)/(2g+1)^3$, which is at most $8/243 \approx 0.033$ at $g = 2/5$. Moreover, it is bounded from below by $-2g(1-g^2)/(2g+1)^3$, which is at least $-4/243 \cdot (7\sqrt{7}-10) \approx -0.14$ at $g = (\sqrt{7}-2)/3 \approx 0.22$.

Expected balanced Negative F_1 ($\mathbb{E}[NF_1^b]$) The approximation of the expectation of the balanced negative F_1 score (NF_1^b) can be obtained from that of PF_1^b by first swapping n_1 and n_2 and replacing g by $1-g$. In particular, we find

$$\mathbb{E}[NF_1^b] = 2(1-g)\left(\frac{1}{3-2g} - \frac{2\Big(1+\delta(3-2g)\Big)g}{n(1+\delta)(1-\delta)(3-2g)^3}\right) + O\left(\frac{1}{n^2}\right), \quad (32)$$

The term of order $1/n$ is bounded from above by $2g(1-g)^2)/(2(1-g)+1)^3$, which is at most $8/243 \approx 0.044$ at $g = 3/5$, and bounded from below by $-2(1-g)(1-(1-g)^2)/(2(1-g)+1)^3$, which is at least $-4/243 \cdot (7\sqrt{7}-10) \approx -0.14$ at $g = (5-\sqrt{7})/3 \approx 0.78$.

B Appendix - Performance of fictitious binary classifier

This appendix shows how a fictitious binary classifier with $TPR = 0.6$ and $TNR = 0.8$ performs on accuracy (Figs. 2 and 3) and F_1 (Figs. 1 and 4 when

it is min-max normalized with the expected value when randomly guessing the positive class with probability $g = 0.5$ (Figs. 2 and 1) or $g = \alpha$ (Figs. 3 and 4), as a function of imbalance coefficient δ. Shown are imbalanced metrics (left panels) and balanced metrics (right panels), which have been corrected for bias due to class imbalance.

When $g^{unif} = \frac{1}{2}$ is chosen as control, $\mathbb{E}[ACC(g^{prop})] = \frac{1}{2}$ (Fig. 2, left panel, blue line). When $g^{prop} = \alpha$ is chosen as control, $\mathbb{E}[ACC(g^{prop})]$ is a quadratic function (Fig. 3, left panel, blue line; see Table 2). As a result, the classifier is outperformed ($ACC^{mmn}(g^{prop}) < 0$) by this strategy when imbalance is large (here when $\delta < \frac{-1-\sqrt{41}}{10} \approx -0.74$ or $\delta > \frac{-1+\sqrt{41}}{10} \approx 0.54$).

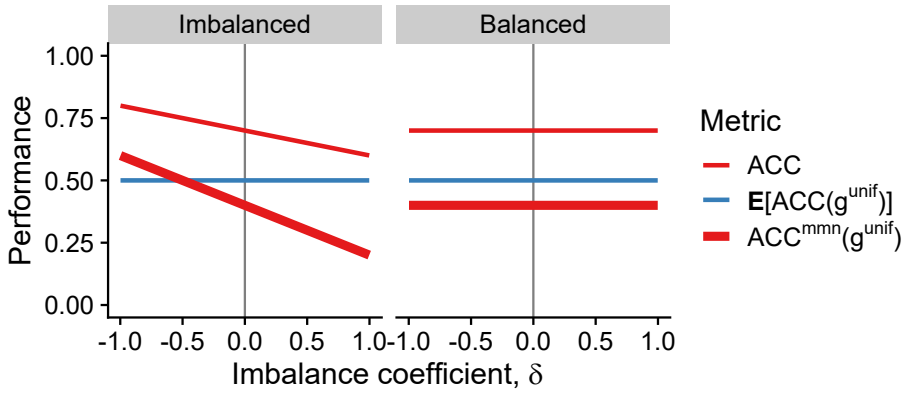

Fig. 2. Accuracy of a fictitious binary classifier in relation to imbalance coefficient δ. $g = 0.5$, $TPR = 0.6$, $TNR = 0.8$, $n = 1000$.

Before min-max normalization, the classifier scores equally well on PF_1 and NF_1 at $\delta = \frac{1}{7}$ (Fig. 4, small white points). After min-max normalization using g^{prop}, however, the classifier scores equally well on PF_1^{mmn} and NF_1^{mmn} at $\delta = -\frac{1}{3}$ (large white points). For $\delta < -\frac{1}{3}$ and $\delta > \frac{1}{7}$, the regular F_1 and normalized $F_1^{mmn}(g^{prop})$ disagree on whether the model performs better on the positive or the negative class.

By definition, the balanced F_1 is insensitive to class imbalance. After min-max normalization with g^{prop}, however, it is sensitive again to class imbalance. The larger the imbalance coefficient, the lower the classifier scores on $PF_1^{mmn,b}(g^{prop})$ and the higher on $NF_1^{mmn,b}(g^{prop})$ (Fig. 4, right panels).

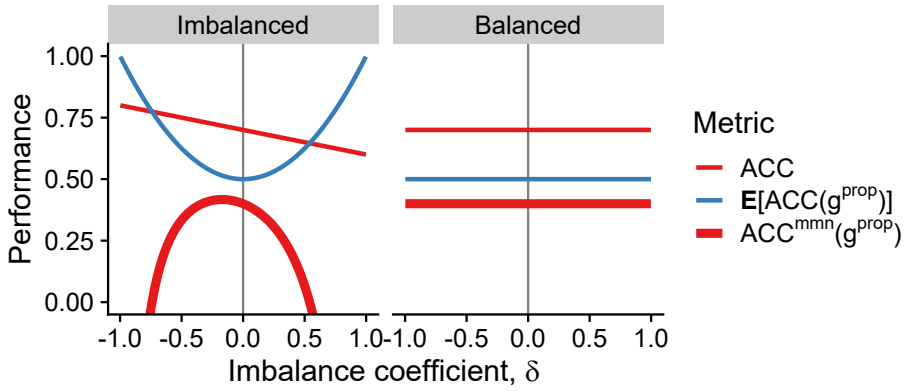

Fig. 3. Accuracy of a fictitious binary classifier in relation to imbalance coefficient δ. $g = \alpha$, $TPR = 0.6$, $TNR = 0.8$, $n = 1000$.

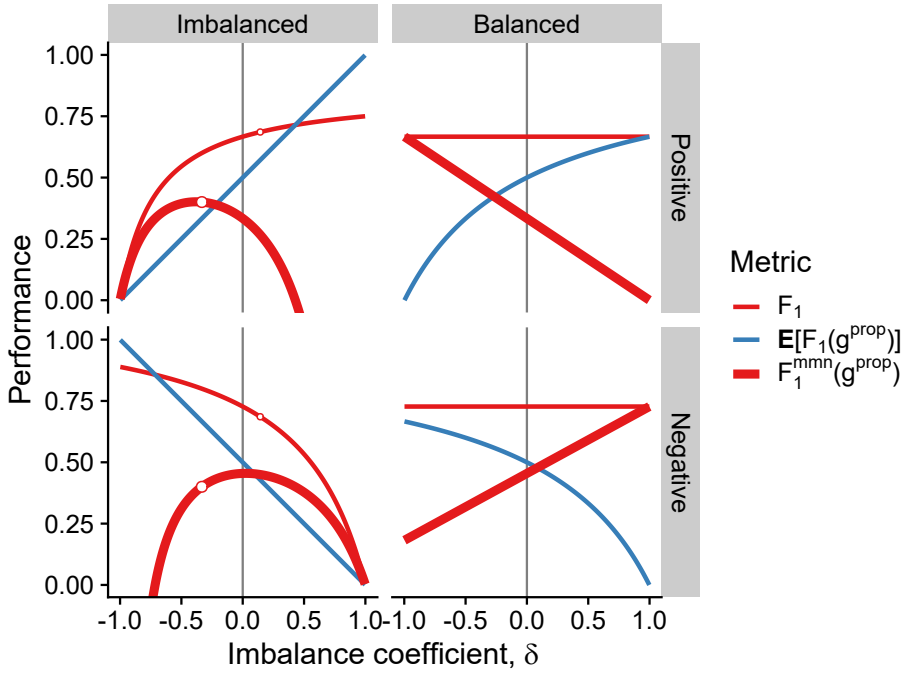

Fig. 4. F_1 of a fictitious binary classifier in relation to imbalance coefficient δ. $g = \alpha$, $TPR = 0.6$, $TNR = 0.8$, $n = 1000$. White points show where positive and negative F_1 intersect.