F-Measure Optimisation in Multi-label Classifiers

Ignazio Pillai, Giorgio Fumera, Fabio Roli
Department of Electrical and Electronic Engineering, University of Cagliari (Italy)
{*pillai,fumera,roli*}*@diee.unica.it*

Abstract

When a multi-label classifier outputs a real-valued score for each class, a well known design strategy consists of tuning the corresponding decision thresholds by optimising the performance measure of interest on validation data. In this paper we focus on the F-measure, which is widely used in multi-label problems. We derive two properties of the micro-averaged F measure, viewed as a function of the threshold values, which allow its global maximum to be found by an optimisation strategy with an upper bound on computational complexity of $O(n^2N^2)$, where N and n are respectively the number of classes and of validation samples. So far, only a suboptimal threshold selection rule and a greedy algorithm without any optimality guarantee were known for this task. We then devise a possible optimisation algorithm based on our strategy, and evaluate it on three benchmark, multi-label data sets.

1 Introduction

In a multi-label classification problem, a sample can belong to more than one class. Such kind of problem occurs in several applications, like text categorisation, image annotation, protein function classification and music classification, and is receiving an increasing interest in the pattern recognition and machine learning literature [4, 5, 2, 6]. Let us denote the number of classes with N, and a sample with $(\mathbf{x}, \mathbf{y})$, where $\mathbf{x} \in X$ is a feature vector in a given feature space X, and $\mathbf{y} \in Y = \{+1, -1\}^N$ encodes the set of its class labels, where $y_k = +1(-1)$ means that the sample (does not) belong to the k-th class. Accordingly, a multi-label classifier implements a decision function $f : X \rightarrow Y$. Performance measures for multi-label problems are based on precision and recall. A widely used one is the F measure, which combines precision and recall into a scalar.

In this paper we address an open problem related to the micro-averaged F measure, in the case when a multi-label classifier which outputs a real-valued score $s_k(\mathbf{x})$ for each class is used, and the decision function is obtained by setting a possibily differerent threshold t_k for each class, such that $f_k(\mathbf{x}) = +1(-1)$, if $s_k(\mathbf{x}) \geq t_k (< t_k)$. In this case, a widely used design strategy is to tune the N threshold values after classifier training, by optimising the chosen performance measure on validation data [7, 1]. So far, no optimisation algorithm that guarantees to find the global maximum of the micro-averaged F measure was known, except for the computationally prohibitive exahustive search. Only a suboptimal threshold selection strategy [7], and a greedy search algorithm [1] were proposed so far.

Our contribution consists of deriving two properties of the micro-averaged F measure as a function of the N thresholds, computed on a given set of n samples, which guarantee that its global maximum can be found by an optimisation strategy based on changing a single threshold value at a time, with an upper bound on computational complexity of $O(n^2N^2)$. We also devise a possible implementation of this strategy, and experimentally evaluate it on three benchmark, multi-label data sets, related to different application domains.

In Sect. 2 we describe the F measure, and review related works. The two properties and the resulting optimisation strategy are presented in Sect. 3. The experimental evaluation is reported in section 4.

2 Background and Previous Works

In information retrieval, precision and recall are defined respectively as the probability that a retrieved sample is relevant to a given query, and the probability to retrieve a relevant sample. In a multi-label classification problem, each class is viewed as the set of samples that are relevant to a distinct query. Precision and recall for the k-th class can thus be estimated from a multi-label data set, respectively as:

$$p_k = \frac{TP_k}{TP_k + FP_k} \,, \quad r_k = \frac{TP_k}{TP_k + FN_k} \,, \qquad (1)$$

where TP_k (true positive) is the number of samples that are correctly labelled as belonging to the k-th class, while FP_k (false positive) and FN_k (false negative) are defined analogously. The F measure is often used to obtain a scalar combination of precision and recall, weighted by a parameter $\beta \in [0, +\infty)$:

$$F_{\beta,k} = \frac{1+\beta^2}{\beta^2/r_k + 1/p_k} \,. \tag{2}$$

The overall performance on the N categories can be computed either by macro- or micro-averaging the class-related values, depending on application requirements [4]. We focus here on the micro-averaged F measure, denoted as F^{m}_{β}, which is defined as [7]:

$$F^{\mathrm{m}}_{\beta} = \frac{(1+\beta^2)}{(1+\beta^2) + \frac{\sum_{k=1}^{N}(FP_k+\beta^2 FN_k)}{\sum_{k=1}^{N} TP_k}} \,. \tag{3}$$

Consider now a trained classifier which outputs real-valued scores $s_k(\mathbf{x})$, and a decision function implemented using threshold values t_k, $k = 1, \ldots, N$, as described in Sect. 1. The corresponding F^{m}_{β} computed on a given data set of n samples $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{n}$ (e.g., a validation set) is a piece-wise constant function of $t_1, \ldots, t_N$, which can exhibit discontinuities for $t_k = s_k(\mathbf{x}_i)$, $k = 1, \ldots, N$, $i = 1, \ldots, n$. It can thus take up to $(n+1)^N$ distinct values. Contrary to its macro-averaged version, F^{m}_{β} can not be decomposed into independent functions of individual thresholds [7]. Therefore, no straightforward optimisation strategy exists to find the threshold values that provide its global maximum, except for the computationally prohibitive exhaustive search.

This issue has been addressed so far only in [1, 7]. In [7] a very simple solution was proposed, consisting of using the threshold values that maximise the macro-averaged F measure, which can be computed at very low computational cost. However, the resulting value of F^{m}_{β} can be significantly lower than the one attainable by tuning the thresholds on the same F^{m}_{β} [1]. To this aim, a greedy search algorithm was proposed in [1]. It iteratively finds the local maximum of F^{m}_{β} with respect to a *single* threshold at a time, until the attained improvement falls below a predefined amount. However, no guarantee was provided that this algorithm can attain the global maximum of F^{m}_{β}.

In [7] the issue of overfitting was also addressed. It was argued that the risk of overfitting is higher for rarer classes, and that too low threshold values should hurt F^{m}_{β} to a higher extent than too high values. Based on this argument, two heuristics (named "FBR") were proposed to limit overfitting. They consist of setting the threshold of a rare class either to $+\infty$ (FBR.0), or to the score of the top-ranked sample in that class (FBR.1). Rare classes were defined as the ones for which $F_{\beta,k} < fbr$, where fbr is a predefined value.

3 F^{m}_{β} Optimisation Strategy

In this section we present the main contribution of this work. We first state two properties of F^{m}_{β} as a function of $t_1, \ldots, t_N$, evaluated on a given set of n samples, and exploit them to devise an optimisation strategy that guarantees to attain the global maximum of F^{m}_{β} at low computational compexity. We then provide a possible implementation of this strategy, and derive its computational complexity. We finally discuss the relationship with the optimisation strategy of [1]. Due to lack of space, the proofs are not reported in this paper, and are available at the authors' web site.[1]

Property 1. *Consider any given set of threshold values $t_1, \ldots, t_N$. If, for each $k = 1, \ldots, N$,*
$F^{\mathrm{m}}_{\beta}(t_1, \ldots, t_N) =$
$\max_{\tau} F^{\mathrm{m}}_{\beta}(t_1, .., t_{k-1}, \tau, t_{k+1}, .., t_N)$,
then $t_1, \ldots, t_N$ provides the global maximum of F^{m}_{β}.

This implies that, if a given set of threshold values does not provide the global maximum of F^{m}_{β}, then F^{m}_{β} can be improved by changing the value of at least one of them, *while keeping the other $N-1$ ones fixed*.

The second property states that, after any threshold t_k has been updated once, no further improvement of F^{m}_{β} can be attained in any subsequent step, by values of t_k lower than the current one:

Property 2. *Consider any set of threshold values $t_1, \ldots, t_N$, such that, for a given k:*
$t_k = \arg\max_{\tau} F^{\mathrm{m}}_{\beta}(t_1, \ldots, t_{k-1}, \tau, t_{k+1}, \ldots, t_N)$.
Consider now another set of threshold values $t'_1, \ldots, t'_{k-1}, t_k, t'_{k+1}, \ldots, t'_N$, such that:
$F^{\mathrm{m}}_{\beta}(t'_1, \ldots, t'_{k-1}, t_k, t'_{k+1}, \ldots, t'_N) >$
$F^{\mathrm{m}}_{\beta}(t_1, \ldots, t_{k-1}, t_k, t_{k+1}, \ldots, t_N)$.
For any $\tau < t_k$ the following inequality is always true:
$F^{\mathrm{m}}_{\beta}(t'_1, \ldots, t'_{k-1}, \tau, t'_{k+1}, \ldots, t'_N) <$
$F^{\mathrm{m}}_{\beta}(t'_1, \ldots, t'_{k-1}, t_k, t'_{k+1}, \ldots, t'_N)$.

It is easy to see that properties 3 and 4 guarantee that the global maximum of F^{m}_{β} can be found as follows. First, set the thresholds to their smallest possible value, i.e., any value $t_k < \min_i s_k(\mathbf{x}_i)$. Then, repeatedly scan them, and update each of them to any value which provides an improvement of F^{m}_{β} (if any), keeping the other ones at their current values, until no F^{m}_{β} improvement is attained after a scan over all thresholds. A possible

[1] http://prag.diee.unica.it/pra/bib/pillai_icpr2012_thr

Algorithm 1 F_β^{m} optimisation algorithm

Require: the score values on a validation set V
Ensure: the values of $t_1, \ldots, t_N$ that maximise F_β^{m} on V
  set t_k to any value lower than $\min_i s_k(\mathbf{x}_i)$, $k = 1, \ldots, N$
  repeat
    $updated \leftarrow$ False
    for $k = 1, \ldots, N$ **do**
      $\theta \leftarrow \arg\max_{\tau \geq t_k} F_\beta^{\mathrm{m}}(t_1, \ldots, t_{k-1}, \tau, t_{k+1}, \ldots, t_N)$
      if $\theta \neq t_k$ **then**
        $t_k \leftarrow \theta, \quad updated \leftarrow$ True
      end if
    end for
  until $updated =$ False
  return $t_1, \ldots, t_N$

implementation of this optimisation strategy is given by Algorithm 1: at each scan (corresponding to the repeat-until loop), each threshold is updated to the value which locally maximises F_β^{m}. Note that this requires to evaluate up to $n + 1$ values for each t_k (see Sect. 2).

In the same online appendix mentioned above, we prove that the computational complexity of Algorithm 1 is upper bounded by $\frac{1}{2}\left[N^2(n+1)^2 + N(n+1)\right] = O(n^2N^2)$, in terms of the number of different sets of threshold values $(t_1, \ldots, t_N)$ which are evaluated.

The greedy algorithm of [1] turns out to be another possible implementation of our optimisation strategy, exploiting only Property 3. It can thus provide the global maximum of F_β^{m}, provided that no early-stopping criterion as the one considered in [1] is used.

4 Experimental Evaluation

We experimentally evaluated the computational cost and the tendency to overfit of Algorithm 1. The latter is an obvious concern, since Algorithm 1 finds the global maximum of F_β^{m} without any countermeasure against overfitting, except for the use of validation data instead of training data. We did not make any comparative performance evaluation, since no alternative optimisation algorithm exists. Indeed, we have shown that also the algorithm of [1] attains the global maximum of F_β^{m}, if no early-stopping is used, while the threshold selection strategy of [7] was already found to be less effective than directly optimising F_β^{m} [1].

We used three benchmark multi-label data sets: the "ModApte" version of "Reuters 21578" (text categorization); Yeast (gene function classification), and Scene (image annotation). For Reuters we used the bag-of-words representation, with tf-idf features. Let D denotes the number of training documents, $tf(\tau_k, d)$ the frequency of term τ_k in any document d, and D_k the number of training documents in which τ_k occurs. The corresponding tf-idf feature value for τ_k in document d is defined as $tf(\tau_k, d) \times log(D/D_k)$. After stemming and stop-word removal, a further feature selection was carried out using the information gain criterion. The main characteristics of the data sets, after the above preprocessing steps for Reuters, are reported in Table 1.

		Reuters	Yeast	Scene
N. of training samples		7769	1500	1211
N. of testing samples		3019	917	1196
Feature set size		15000	104	295
N. of classes		90	14	6
Class frequency	Min.	1.3E-4	0.065	0.136
	Max.	0.370	0.752	0.229

Table 1. Characteristics of the data sets.

The well known *binary relevance* (BR) approach was used to implement multi-label classifiers. It consists of independently training N binary classifiers using the one-vs-all strategy [4, 6]. We used as base classifiers the k-nearest neighbours (k-NN), and support vector machines (SVM) with linear kernel for Reuters, and radial-basis function kernel for Scene and Yeast.

Ten runs of the experiments were carried out: the original training set was partitioned into ten disjoint subsets of identical size, and at each run only eight subsets were used for classifier training. Threshold values were computed through a five-fold cross-validation, carried out on the training samples of each run: Algorithm 1 was applied to the union of the scores of the five validation folds. We considered only $\beta = 1$ as in [1, 7]. The average F_1^{m} value over the ten runs was computed on the original testing set.

In Table 2 we report the attained F_β^{m} values, under different experimental settings. First, to assess whether and to what extent overfitting occurs, we compared the testing set F_1^{m} value ("Test set" column) with the value attained on the same cross-validation samples where the thresholds were computed ("Validation set"). It can be seen that the latter values are higher, which means that overfitting occurred, although its extent was rather small. In particular, in the Reuters data set, where $N = 90$ thresholds had to be computed, and several classes were very rare (see Table 1), the difference was less than 0.03 for both classifiers.

We then evaluated whether the FBR heuristic of [7] was able to reduce overfitting. To this aim, we estimated the value of the fbr parameter (see Sect. 2) through an inner five-fold cross validation carried out on each training fold of the outer cross-validation used for computing the decision thresholds, similarly to [1]. The corresponding results on testing samples are reported in the

Data set	Classifier	Validation set	Test set	Test set (1st loop)	FBR.0	FBR.1
Reuters	SVM	0.907±0.001	0.880±0.002	0.689±0.010	0.879±0.002	0.878±0.002
	k-NN	0.854±0.002	0.825±0.003	0.580±0.013	0.825±0.003	0.825±0.003
Yeast	SVM	0.682±0.002	0.678±0.003	0.669±0.003	0.678±0.003	0.678±0.003
	k-NN	0.667±0.003	0.661±0.003	0.651±0.004	0.661±0.003	0.660±0.002
Scene	SVM	0.778±0.007	0.769±0.006	0.757±0.006	0.769±0.006	0.769±0.006
	k-NN	0.739±0.007	0.711±0.004	0.706±0.006	0.711±0.004	0.711±0.004

Table 2. Average F_1^{m} values, and standard deviation, over the ten runs of the experiments.

"FBR.0" and "FBR.1" columns of Table 2. They show that there is no appreciable difference with respect to the results attained without using FBR. This is in agreement with the results of [1], where FBR was found to be effective only for the marco-averaged F measure. A possible reason is that F_β^{m} is mainly affected by FP errors on rare classes (see Eq. 9), whose amount is usually much higher than FNs and TPs. Accordingly, to maximise F_β^{m} it is crucial to reduce FPs errors on rare classes. This is attained by increasing the corresponding thresholds as much as possible. Note now that the optimal values of such thresholds can be reliably estimated by an optimisation algorithm from validation data, due to the relatively large number of FP samples in multi-label problems, especially in rare classes. Increasing the thresholds of rare classes is also what FBR tries to do *afterwards*, which can explain its ineffectiveness.

A very low computational cost was observed in our experiments. The number of different sets of threshold values $(t_1, \ldots, t_N)$ that were evaluated by Algorithm 1 was always smaller than $2(n+1)N$, which is much lower than the upper bound reported in Sect. 3, and of the cost of an exhaustive search, given by $(n+1)^N$. This also provides evidence that Algorithm 1 can scale very well on large data sets with many classes.

Consider finally that in [1] no significant improvement of F_1^{m} was found, after the first scan of the N thresholds. Accordingly, a single scan was suggested, which corresponds to a single repeat-until loop of Algorithm 1. We found instead that more than one repeat-until loop may be required. This was the case of the Reuters data set, for which the testing set F_1^{m} values attained after the first loop, reported in the "Test set (1st loop)" column of Table 2, turned out to be significantly lower than the final ones ("Test" column).

5 Conclusions

We developed an optimisation strategy for the micro-averaged F measure, that allows its global maximum to be found on a given data set, as a function of the class-related decision thresholds, with a low computational cost. Empirical evidence showed that, using validation data, a limited overfitting is incurred, even in problems with many classes, including rare ones.

Our results could also be exploited to evaluate the macro- and micro-averaged precision-recall curves as a function of $t_1, \ldots, t_k$, which is another open issue. In [7] a strategy based on maximising the corresponding F measure for different β values was suggested, but it was not analysed, and no implementation was proposed.

The design strategy we considered consists of training any multi-label classifier using its own objective function, not necessarily related to the F measure (e.g., a standard SVM classifier), and then optimising the F measure by tuning the decision thresholds. It will be interesting to compare its performance with the one of classifiers whose objective function was designed to approximate the F measure of a single class (e.g., [3]).

Acknowledgements. This work was partly supported by a grant from Regione Autonoma della Sardegna awarded to I. Pillai, PO Sardegna FSE 2007-2013, L.R.7/2007 "Promotion of scientific research and technological innovation in Sardinia".

References

[1] R.-E. Fan and C.-J. Lin. A study on threshold selection for multi-label. Tech. rep., National Taiwan Univ., 2007.

[2] J. Fürnkranz, E. Hüllermeier, E. L. Mencía, and K. Brinker. Multilabel classification via calibrated label ranking. *Machine Learning*, 73(2):133–153, 2008.

[3] D. R. Musicant, V. Kumar, and A. Ozgur. Optimizing f-measure with support vector machines. In *FLAIRS Conference*, pages 356–360, 2003.

[4] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[5] G. Tsoumakas and I. Katakis. Multi label classification: An overview. *Int. J. Data Warehousing and Mining*, 3(3):1–13, 2007.

[6] G. Tsoumakas, I. Katakis, and I. Vlahavas. Mining multi-label data. *Data Mining and Knowledge Discovery Handbook*, pages 667–685, 2010.

[7] Y. Yang. A study of thresholding strategies for text categorization. In *Int. Conf. on Research and development in information retrieval*, New York, USA, 2001.

A Auxiliary equivalences.

We report here three equivalences that will be used in the proofs of the next sections.

Equivalence 4.

Given four real values A, B, ΔA and ΔB, with $B > 0$, $\Delta B < 0$, and $B + \Delta B > 0$, the following equivalence holds:

$$\frac{A + \Delta A}{B + \Delta B} < \frac{A}{B} \quad \Leftrightarrow \quad \frac{A}{B} < \frac{\Delta A}{\Delta B}. \tag{4}$$

Proof. Taking into account the constraints on $B, \Delta B$ and $B + \Delta B$, from the first inequality of (4) one obtains:

$$B(A + \Delta A) < A(B + \Delta B),$$

$$AB + \Delta A \times B < AB + A \times \Delta B,$$

$$\Delta A \times B < A \times \Delta B,$$

$$\frac{A}{B} < \frac{\Delta A}{\Delta B}.$$

Equivalences 5 and 6.

Given four real values A, B, ΔA and ΔB, with $B > 0$, $\Delta B > 0$, the following equivalences hold:

$$\frac{\Delta A}{\Delta B} < \frac{A + \Delta A}{B + \Delta B} \quad \Leftrightarrow \quad \frac{A + \Delta A}{B + \Delta B} < \frac{A}{B}, \tag{5}$$

$$\frac{\Delta A}{\Delta B} < \frac{A}{B} \quad \Leftrightarrow \quad \frac{\Delta A}{\Delta B} < \frac{A + \Delta A}{B + \Delta B} < \frac{A}{B}. \tag{6}$$

Proof. We indirectly prove (5) and (6) by showing that the following equivalences (7) hold, under the same constraints given above. This implies that equivalences (5) and (6) hold, as they are implied by (7). Note indeed that the two inequalities of (5) coincide with the first and third inequality of (7), while the ones of (6) coincide with the second inequality of (7), and with the union of the first and third inequality of (7).

$$\frac{\Delta A}{\Delta B} < \frac{A + \Delta A}{B + \Delta B} \quad \Leftrightarrow \quad \frac{\Delta A}{\Delta B} < \frac{A}{B} \quad \Leftrightarrow \quad \frac{A + \Delta A}{B + \Delta B} < \frac{A}{B}. \tag{7}$$

Taking into account the constraints on B and ΔB, from the second inequality of (7) one obtains:

$$\Delta A \times B < A \times \Delta B. \tag{8}$$

Adding $(\Delta A \times \Delta B)$ to both sides of (8), one obtains:

$$\Delta A \times B + \Delta A \times \Delta B < \Delta A \times \Delta B + A \times \Delta B,$$

$$\Delta A(B + \Delta B) < (A + \Delta A)\Delta B,$$

which corresponds to the first inequality of (7).

Adding $(A \times B)$ to both sides of (8), one obtains instead:

$$\Delta A \times B + A \times B < A \times B + A \times \Delta B,$$

$$B(A + \Delta A) < A(B + \Delta B),$$

which corresponds to the last inequality of (7).

B Proof of Property 3

Property 3. *Consider any given set of threshold values $t_1, \ldots, t_N$. If, for each $k = 1, \ldots, N$,*

$$F_\beta^{\mathrm{m}}(t_1, \ldots, t_N) = \max_\tau F_\beta^{\mathrm{m}}(t_1, \ldots, t_{k-1}, \tau, t_{k+1}, \ldots, t_N) \,,$$

then $t_1, \ldots, t_N$ provides the global maximum of F_β^{m}.

We remind the reader that the F_β^{m} measure is defined as:

$$F_\beta^{\mathrm{m}} = \frac{(1+\beta^2)}{(1+\beta^2) + \frac{\sum_{k=1}^N (FP_k + \beta^2 FN_k)}{\sum_{k=1}^N TP_k}} \,. \tag{9}$$

Proof. Consider a set of threshold values $\mathcal{T} = (t_1, \ldots, t_N)$, and another set obtained from $\mathcal{T}$ by changing the values of m thresholds, with $m \leq N$. Without losing generality, we assume that the first m thresholds are changed. We denote the latter set as $\mathcal{T}^{(1,\ldots,m)} = (t'_1, \ldots, t'_{m-1}, t'_m, t_{m+1}, \ldots, t_N)$. Let us also denote with $\mathcal{T}^{(k)}$ the threshold values obtained from $\mathcal{T}$ by changing only the k threshold from t_k to t'_k, $k \in \{1, \ldots, m\}$. In the following we prove that, for any given $m \in \{2, \ldots, N\}$ and any given $\mathcal{T}^{(1,\ldots,m)}$, the following implication holds:

$$\begin{array}{ll} \text{if} & \hat{F}_\beta^{\mathrm{m}}(\mathcal{T}) > \hat{F}_\beta^{\mathrm{m}}(\mathcal{T}^{(k)}) \text{ for each } k \in \{1, \ldots, m\}, \\ \text{then} & \hat{F}_\beta^{\mathrm{m}}(\mathcal{T}) > \hat{F}_\beta^{\mathrm{m}}(\mathcal{T}^{(1,\ldots,m)}). \end{array} \tag{10}$$

Clearly, this implies that Property 3 is true.

Consider first Eq. (9). To simplify the notation, let us denote the values $\sum_{k=1}^N (FP_k + \beta^2 FN_k)$ and $\sum_{k=1}^N TP_k$, corresponding to the thresholds $\mathcal{T}$, respectively as E and TP. We also denote as $E + \Delta E_k$ and $TP + \Delta TP_k$ the corresponding values attained by $\mathcal{T}^{(k)}$ (we remind the reader that FP_k, FN_k and TP_k depend only on the value of the k-th threshold). Obviously, $\Delta E_k = \Delta TP_k = 0$, for any $k > m$.

From Eq. (9) it is easy to see that the inequality $\hat{F}_\beta^{\mathrm{m}}(\mathcal{T}) > \hat{F}_\beta^{\mathrm{m}}(\mathcal{T}^{(k)})$ is equivalent to $\frac{E}{TP} < \frac{E+\Delta E_k}{TP+\Delta TP_k}$, and that inequality $\hat{F}_\beta^{\mathrm{m}}(\mathcal{T}) > \hat{F}_\beta^{\mathrm{m}}(\mathcal{T}^{(1,\ldots,m)})$ is equivalent to $\frac{E}{TP} < \frac{E+\sum_{k=1}^m \Delta E_k}{TP+\sum_{k=1}^m \Delta TP_k}$. Accordingly, implication (10) can be rewritten as:

$$\begin{array}{ll} \text{if} & \dfrac{E}{TP} < \dfrac{E + \Delta E_k}{TP + \Delta TP_k}, \text{ for each } k \in \{1, \ldots, m\}, \\ \text{then} & \dfrac{E}{TP} < \dfrac{E + \sum_{k=1}^m \Delta E_k}{TP + \sum_{k=1}^m \Delta TP_k}. \end{array} \tag{11}$$

If $m = 1$, (11) is trivially true. If $m > 1$, we prove it by induction. First, we prove that it holds when $m = 2$. Then we prove that, if (11) holds for any $m = m^* \in \{2, \ldots, N-1\}$, then it holds also for $m = m^* + 1$.

Base case: $m = 2$

Assume that the consequent part of (11) is false, namely, a point $\mathcal{T}^{(1,2)} = (t'_1, t'_2, t_3, \ldots, t_N)$ exists, such that $F_\beta^{\mathrm{m}}(\mathcal{T}) < F_\beta^{\mathrm{m}}(\mathcal{T}^{(1,2)})$. Using the above notation, this inequality can be rewritten as:

$$\frac{E}{TP} > \frac{E + \Delta E_1 + \Delta E_2}{TP + \Delta TP_1 + \Delta TP_2}.$$

Taking into account also the assumptions of (11) we obtain:

$$\frac{E + \Delta E_1 + \Delta E_2}{TP + \Delta TP_1 + \Delta TP_2} < \frac{E}{TP} < \frac{E + \Delta E_k}{TP + \Delta TP_k}, k = 1, 2. \tag{12}$$

Let us now consider two different cases: $\Delta TP_2 < 0$, and $\Delta TP_2 > 0$ (the case $\Delta TP_2 = 0$ is trivial), with no constraint on ΔTP_1. (Note that the proof can be made also by considering the cases $\Delta TP_1 < 0$ and $\Delta TP_1 > 0$, with no constraint on ΔTP_2.)

If $\Delta TP_2 < 0$, applying (4) to the first and third term of (12),[2] we obtain:

$$\frac{E + \Delta E_1}{TP + \Delta TP_1} < \frac{\Delta E_2}{\Delta TP_2}.$$

From the above expression and the second inequality of (12), we obtain:

$$\frac{E}{TP} < \frac{\Delta E_2}{\Delta TP_2}.$$

Finally, applying (4) to the above inequality,[3] we obtain:

$$\frac{E + \Delta E_2}{TP + \Delta TP_2} < \frac{E}{TP},$$

which contradicts the second inequality of (12) for $k = 2$.

The proof for the case $\Delta TP_2 > 0$ is similar. It can be obtained by applying (5) to the first and third term of Eq. (12),[4] then using the first of the inequalities (12), and finally applying (6),[5] which leads to a contradiction.

Inductive step

Assuming that (11) holds for each $m \leq m^* < N$, we have to prove that it holds also for $m = m^* + 1$, namely, that the following implication holds:

$$\begin{aligned} &\text{if} \quad \frac{E}{TP} < \frac{E + \Delta E_k}{TP + \Delta TP_k}, \ \text{ for each } k \in \{1, \ldots, m^* + 1\}, \\ &\text{then} \quad \frac{E}{TP} < \frac{E + \sum_{k=1}^{m^*+1} \Delta E_k}{TP + \sum_{k=1}^{m^*+1} \Delta TP_k}. \end{aligned} \tag{13}$$

By the above assumption, we know that:

$$\frac{E}{TP} < \frac{E + \sum_{k=1}^{m^*} \Delta E_k}{TP + \sum_{k=1}^{m^*} \Delta TP_k}. \tag{14}$$

Note now that the consequent of (13) can be rewritten as:

$$\frac{E}{TP} < \frac{E + \sum_{k=1}^{m^*} \Delta E_k + \Delta E_{m^*+1}}{TP + \sum_{k=1}^{m^*} \Delta TP_k + \Delta TP_{m^*+1}}. \tag{15}$$

It is now easy to see that (15) is implied by (14) and by the antecedent of (13) for $k = m^* + 1$, which in turn implies that (13) is true. The proof coincides indeed with the one of the basis case above, with a simple change of notation. This completes the proof of Property 3.

C Proof of Property 4

Property 4. *Consider any set of threshold values $t_1, \ldots, t_N$, such that, for a given k:*

$$t_k = \arg\max_{\tau} F_\beta^{\mathrm{m}}(t_1, \ldots, t_{k-1}, \tau, t_{k+1}, \ldots, t_N) \, .$$

Consider now another set of threshold values $t'_1, \ldots, t'_{k-1}, t_k, t'_{k+1}, \ldots, t'_N$, such that:

$$F_\beta^{\mathrm{m}}(t'_1, \ldots, t'_{k-1}, t_k, t'_{k+1}, \ldots, t'_N) > F_\beta^{\mathrm{m}}(t_1, \ldots, t_{k-1}, t_k, t_{k+1}, \ldots, t_N) \, .$$

For any $\tau < t_k$ the following inequality is always true:

$$F_\beta^{\mathrm{m}}(t'_1, \ldots, t'_{k-1}, \tau, t'_{k+1}, \ldots, t'_N) < F_\beta^{\mathrm{m}}(t'_1, \ldots, t'_{k-1}, t_k, t'_{k+1}, \ldots, t'_N) \, .$$

[2] with $A = E + \Delta E_1$, $B = TP + \Delta TP_1$, $\Delta A = \Delta E_2$ and $\Delta B = \Delta TP_2 < 0$

[3] with $A = E$, $B = TP$, $\Delta A = \Delta E_2$, $\Delta B = \Delta TP_2 < 0$

[4] with $A = E + \Delta E_1$, $B = TP + \Delta TP_1$, $\Delta A = \Delta E_2$ and $\Delta B = \Delta TP_2 > 0$

[5] with $A = E$, $B = TP$, $\Delta A = \Delta E_2$, $\Delta B = \Delta TP_2 > 0$

Proof. The first assumption of Property 4 considers a given set of threshold values, such that no higher value of $\hat{F}_\beta^{\mathrm{m}}$ can be attained by changing the k-th threshold, for some given k. Using the same notation of B, this can be rewritten as:

$$\forall(\Delta E_k, \Delta TP_k),\ \ \frac{E}{TP} < \frac{E + \Delta E_k}{TP + \Delta TP_k}, \tag{16}$$

where E and TP correspond to the considered set of threshold values, while ΔE_k and ΔTP_k are *any* change respectively of E and TP attainable by changing the k-th threshold.

The second assumption states that a higher value of $\hat{F}_\beta^{\mathrm{m}}$ is attained by changing any subset of the threshold values but the k-th one. Using the previous notation, this can be rewritten as:

$$\exists(\Delta E_i, \Delta TP_i), i \in \{1, \ldots, N\} - \{k\},\ \ \frac{E}{TP} > \frac{E + \sum_{i\neq k} \Delta E_i}{TP + \sum_{i\neq k} \Delta TP_i}. \tag{17}$$

Under the above assumptions, Property 4 states that no higher values of $\hat{F}_\beta^{\mathrm{m}}$ can be attained from the latter set of threshold values, by decreasing the value of the k-th threshold, namely:

$$\forall(\Delta E'_k, \Delta TP'_k),\ \ \frac{E + \sum_{i\neq k} \Delta E_i}{TP + \sum_{i\neq k} \Delta TP_i} < \frac{E + \sum_{i\neq k} \Delta E_i + \Delta E'_k}{TP + \sum_{i\neq k} \Delta TP_i + \Delta TP'_k}, \tag{18}$$

where $\Delta E'_k$ and $\Delta TP'_k$ denote *any* change of E and TP obtained by decreasing the value of the k-th threshold.

Since ΔE_k and ΔTP_k correspond to *any* change of the k-th threshold, we can consider the particular change that leads to $\Delta E_k = \Delta E'_k$ and $\Delta TP_k = \Delta TP'_k$. Decreasing the value of the k-th threshold implies that $\Delta TP'_k \geq 0$. If we consider a value of T_k such that $\Delta TP'_k = 0$, then $\Delta E'$ must be negative, and Property 4 is trivially true. We will consider therefore only the case $\Delta TP'_k > 0$ in the following.

Inequality (16) implies thus the following one:

$$\frac{E}{TP} < \frac{E + \Delta E'_k}{TP + \Delta TP'_k}.$$

Applying (5) to this inequality,[6] we obtain:

$$\frac{E + \Delta E'_k}{TP + \Delta TP'_k} < \frac{\Delta E'_k}{\Delta TP'_k} \tag{19}$$

Combining inequalities (16), (17) and (19), we obtain:

$$\frac{E + \sum_{i\neq k} \Delta E_i}{TP + \sum_{i\neq k} \Delta TP_i} < \frac{\Delta E'_k}{\Delta TP'_k}.$$

Applying (6),[7] we finally obtain (18), which completes the proof.

D Computational complexity of Algorithm 1

Let us denote with $T^{(0)}$ and $T^{(n)}$ respectively the interval $\big(-\infty, s_k(\mathbf{x}_{(1)})\big)$ and $\big[s_k(\mathbf{x}_{(n)}), +\infty\big)$, and with $T^{(i)}$ the intervals $\big[s_k(\mathbf{x}_{(i)}), s_k(\mathbf{x}_{(i+1)})\big)$, $i = 1, \ldots, n-1$.

An upper bound for the number of threshold values evaluated by Algorithm 1 can be obtained by considering the following conditions:

1. For each class k, the scores $s_k(\mathbf{x}_i)$, $i = 1, \ldots, n$, are all different.
2. In each repeat-until loop, only one threshold is updated.
3. When any threshold t_k is updated, if its current value is in $T_k^{(i)}$, $i < n$, then the new value is in $T_k^{(i+1)}$.

[6] with $A = E$, $B = TP$, $\Delta A = \Delta E'_k$ and $\Delta B = \Delta TP'_k$ (we remind the reader that $\Delta TP'_k > 0$).

[7] with $A = E + \sum_{i\neq k} \Delta E_i$, $B = TP + \sum_{i\neq k} \Delta TP_i$, $\Delta E'_k$ and $\Delta B = \Delta TP'_k$

4. The global maximum of $\hat{F}_\beta^{\mathrm{m}}$ is attained when $t_k \in T_k^{(n)}$, $k = 1, \ldots, N$.

This implies that the repeat-until loop is executed for Nn times, and that in each loop one threshold value less than in the previous loop has to be evaluated. The number of threshold values that are evaluated is thus: Nn in the first loop, $Nn - 1$ in the second loop, $Nn - 2$ in the third loop, ..., 1 in the last loop, which amounts to:

$$\sum_{j=1}^{Nn} j = \frac{1}{2}\left[N^2 n^2 + Nn\right] = O(n^2 N^2)\,.$$