entropy

MDPI

Article

M-ary Rank Classifier Combination: A Binary Linear Programming Problem

Vincent Vigneron * and Hichem Maaref *

Informatique, Bio-informatique et Systèmes Complexes (IBISC) EA 4526, univ Evry, Université Paris-Saclay, 40 rue du Pelvoux, 91020 Evry, France

* Correspondence: vincent.vigneron@univ-evry.fr (V.V.); hichem.maaref@univ-evry.fr (H.M.); Tel.: +33-6-635-687-60 (V.V.)

Received: 16 January 2019; Accepted: 18 April 2019; Published: 26 April 2019

Abstract: The goal of classifier combination can be briefly stated as combining the decisions of individual classifiers to obtain a better classifier. In this paper, we propose a method based on the combination of weak rank classifiers because rankings contain more information than unique choices for a many-class problem. The problem of combining the decisions of more than one classifier with raw outputs in the form of candidate class rankings is considered and formulated as a general discrete optimization problem with an objective function based on the distance between the data and the consensus decision. This formulation uses certain performance statistics about the joint behavior of the ensemble of classifiers. Assuming that each classifier produces a ranking list of classes, an initial approach leads to a binary linear programming problem with a simple and global optimum solution. The consensus function can be considered as a mapping from a set of individual rankings to a combined ranking, leading to the most relevant decision. We also propose an information measure that quantifies the degree of consensus between the classifiers to assess the strength of the combination rule that is used. It is easy to implement and does not require any training. The main conclusion is that the classification rate is strongly improved by combining rank classifiers globally. The proposed algorithm is tested on real cytology image data to detect cervical cancer.

Keywords: classifier combination; rank; aggregation; total order; independence; data fusion; mutual information; plurality voting; binary linear programming; cervical cancer; HPV

1. Introduction

Using a single classifier has shown limitations in achieving satisfactory recognition performance, and this leads us to use multiple classifiers, which is now a common practice in machine learning. Classifier combination has been studied in many disciplines such as the social sciences, sensor fusion, pattern recognition, etc. Schapire [1] proved that a strong classifier can be generated by combining weak classifiers. It has been accepted as an effective method to improve classification performances. Many examples of ensemble classifier systems can be found in process engineering or medicine. For a survey of the issues and approaches on classifier combination, readers are referred to Woźniak [2] and Oza and Turner [3]. The same type of approach has also been used, for instance, in remote sensing domains (e.g., for land cover mapping with Landsat Multispectral Scanner, elevation) [4], computer security [5], financial risks [6], proteomics [7].

Classifiers can provide as their final decision only a single class, a ranked list of all the classes, or a score associated with each class as a measure of confidence for the class. In this paper, we focus only on rank-values to perform combination. Rank data are useful when data can not be easily reduced to numbers, such as data that are related to concepts, opinions, feelings, values, and behaviors of people in a social context, genes, characters, etc. Ranking also has the advantage of removing scale

effects while permitting ranking patterns to be compared. But rank-ordering has also its disadvantages: it is difficult to combine data from different rankings, and the information contained in the data is limited [8].

After learning, each classifier of the ensemble has output its own results. Several fusion strategies have been proposed in the literature to combine classifiers at the rank level [9,10]. Among them, one of the most common techniques is certainly the linear combination of the classifier outputs [11,12]. The voting principle is the simplest method of combination, where the top candidate from each classifier constitutes a single vote. The final decisions can be made by majority rule (over half of the votes) [13], plurality (maximum number of votes) [14], weighted sum of significance [15], or other variants. The method of Borda count [16], which sums up the rank values of classifiers, can be considered as a generalization of the voting principle. The Bayesian approach estimates the class posterior probabilities conditioned on classifier decisions by approximating various probability densities [17]. Although decision theory itself does not assume classifiers are independent, this assumption is almost always adopted in practical implementation to reduce the exponential complexity of probability estimation. In summary, classifier combination is an ensemble method that classifies new data by taking a weighted vote of the predictions of a set of classifiers [18]. This is originally a Bayesian averaging, but more recent algorithms include boosting, bagging, random forests, and variants [19–21]. Note that Dempster–Shafer formalism for aggregating beliefs based on uncertainty reasoning lends itself to a more flexible model used to combine multiple pieces of evidence and capable of taking uncertainty and ignorance into account [22].

Finally, a rank classifier provides an ordered list of classes associating each class with a rank integer that indicates its importance in the list. The output of a classifier $\mathcal{K}_k$ is therefore a vector of ranks attributed to the K classes.

An ensemble of classifiers might be a better choice than a single classifier because of the variability of the ensemble errors, which is such that the consensus performance is significantly better than the best individual in the ensemble [23]. This analysis is certainly true when the classifiers of the ensemble "see" different training patterns, and it can be effective even when the classifiers all share the same training set. In a computerized tomography problem to illustrate how the ensemble consensus outperformed the best individuals, Anthimopoulos observed that the marginal benefit obtained by increasing the ensemble size is usually low due to correlation among errors: most classifiers will get the right answer on easy inputs, while many classifiers will make mistakes on difficult inputs [24].

In addition, running several searches and combining the solutions produces a better approximation than many learning techniques that use local searches to converge toward a solution, with the risk of staying stacked in local optima (which may not be true in the case of deep learning classifiers, since Kawaguchi has shown that every local minimum is a global minimum [25]). Thus, we might not be capable of producing the optimal classifier using a training set and a given classifier architecture, compared to a set of several classifiers. Since the number of classifiers can be very high (in the thousands), it is difficult to "understand" the classifier ensemble decision characteristics.

Although general performances are often improved when classifiers are combined, it becomes computationally costly to combine well-trained classifiers [26]. Most of the time, it is believed that the combination of independent classifiers will provide greater performance improvement [27], while combiner decisions could be biased toward duplicated outputs. However, this belief stems from the difficulty of using a dependence assumption. In fact, in practical situations, classifier independence is difficult to assess.

How do multiple rank classifiers improve separation performances when individual classification performances are slightly better than random decision making? And what is "classifier independence" ? This term raises several issues that we will address in Section 4, where we come back to the theory of rank aggregation and propose an algorithm to combine classifiers. The main properties of the classifier are discussed. Section 2 exposes the general framework and the notations used. A classifier ensemble dependence measure is then proposed to evaluate the conditional mutual information in Section 5.

Experimental results are presented in Section 6 for the detection of cervical cancer. Finally, Section 7 gives conclusions on rank classifier combination and further investigations are discussed.

Notations

Set and regions are indicated by double-trace uppercase letters such as $\mathbb{G}, \mathbb{S}, \mathbb{R}$, vectors with bold lowercase such as $\mathbf{x}, \mathbf{y}$, and matrices with uppercase bold letters such as $\mathbf{C}, \mathbf{M}, \mathbf{\Sigma}$. The elements of a matrix $\mathbf{M} = \{m_{ij}\}$ are indexed by the row index i and the column index j. Lowercase letters refer to individual elements in a vector whose position in the vector is indicated by the last subscript. Therefore, x_{ij} refers to the jth element of vector $\mathbf{x}_i$. $p(C_i)$ is the a priori probability of the random value X belonging to class $C_i, 1 \leq i \leq K$, K being the number of classes. M is the number of classifiers used for combination. $|\mathbb{C}|$ denotes the cardinality of set $\mathbb{C}$. ^{T} denotes the transpose operator.

2. Problem Statement and Model

We consider a classification dataset $\mathcal{B}_0$ with n observations

$$\mathcal{B}_0 = \{(\mathbf{x}_i, c_i)\}_{i=1}^n, \tag{1}$$

obtained from a physical signal, or synonymously, explanatory variables, objects, instances, cases, patterns, t-uples, etc. where each $\mathbf{x}_i$ belongs to class $c_i \in \{C_1, \ldots, C_K\}$. The vector $\mathbf{x}_i$ lies in an attribute space $\mathcal{A} \in \mathbf{R}^p$ and each component x_{ij} is a numerical or nominal categorical attribute, also named feature, variable, dimension, component, field, etc.

The output of the M classifiers $\mathcal{K}_1(\mathbf{x}), \ldots, \mathcal{K}_M(\mathbf{x})$ are represented by a K-dimensional vector $\mathbf{u}_i(\mathbf{x}) = (u_{i1}, \ldots, u_{iK})^T, 1 \leq i \leq m$: each component u_{ij} is a certain value associated with class C_j given by $\mathcal{K}_i(\mathbf{x})$. Depending on the nature of the classifier $\mathcal{K}_i(\mathbf{x})$, u_{ij} can be a rank value that reflects a complete or partial ordering of all classes, or a value in $\{0, 1\}$ corresponding to the predicted class assigned to 1 and the others to zero, or a score, e.g., a discriminant value, associated with each class C_j, which serves as a confidence measure for the class to be the true class. The latter can easily be converted into the two former. Therefore, each classifier $\mathcal{K}_i(\mathbf{x})$ defines a mapping function from the image domain $\mathbb{R}^p$ to a K-dimensional vector space defined over a set of values $\mathbb{E}_i$. The general framework is illustrated in Figure 1.

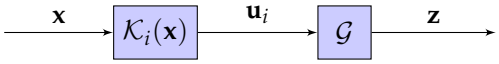


Figure 1. General framework for classifier combination. The classifier $\mathcal{K}_i(\mathbf{x})$ produces output vector $\mathbf{u}_i$. Finally, from $\mathbf{u}_i$ the combination function produces a final decision vector $\mathbf{z}$.

In this paper, u_{ij} is a rank value that reflects a complete or partial ordering of the classes. The objective is to design an optimal combination function $\mathcal{G}$ that takes all the $\mathbf{u}_i$ as input and produces as an output the decision vector $\mathbf{z} = (z_1, \ldots, z_K)^T$, where z_k is the rank associated with the decision on class C_k, that is, $\mathbf{z} = \mathcal{G}(\mathbf{u}_1, \ldots, \mathbf{u}_M)$. Thus, we seek $\mathcal{G}$ as a discriminant function defined over $\mathbb{R}^{K \times m}$.

In the following, it is assumed that (*i*) classifiers have equal individual performances (*ii*) classifiers $\mathcal{K}_i$ are treated as "black boxes". Hence, the combination operator applies only on the real space vectors $\mathbf{u}_i$.

3. Conditional Independence Properties

The term "classifier independence" has been used in an intuitive manner, but what is classifier independence? Formally, two classifiers $\mathcal{K}_1$ and $\mathcal{K}_2$ are said to be independent if

$$p(u_1 = c_j, u_2 = c_\ell) = p(u_1 = c_j,)p(u_2 = c_\ell) \qquad \forall 1 \leq j, \ell \leq K, \tag{2}$$

with u_1 and u_2 being the decision values of $\mathcal{K}_1$ and $\mathcal{K}_2$. The idea is illustrated in the following example.

Example 1 (Independent classifiers)**.** *Consider a binary classification problem (with equiprobable classes C_1 and C_2) and two classifiers $\mathcal{C}_1$ and $\mathcal{C}_2$ with similar performances and whose outputs are $\mathbf{u}_1$ and $\mathbf{u}_2$, i.e., their probabilities of correct classification α_1 and α_2 are equal:*

$$\begin{aligned} p(u_1 = c_1|c_1) &= p(u_1 = c_2|c_2) = \alpha_1 \\ p(u_2 = c_1|c_1) &= p(u_2 = c_2|c_2) = \alpha_2 \\ p(u_1 = c_1|c_2) &= p(u_1 = c_2|c_1) = 1 - \alpha_1 \\ p(u_2 = c_1|c_2) &= p(u_2 = c_2|c_1) = 1 - \alpha_2. \end{aligned} \tag{3}$$

Then the total probability rule helps to find the probability of the outputs:

$$\begin{aligned} p(u_1 = c_1) =& p(u_1 = c_1|c_1)p(c_1) + p(u_1 = c_1|c_2)p(c_2) = \alpha_1/2 + (1 - \alpha_1)/2 = 1/2 \\ p(u_2 = c_1) =& p(u_2 = c_1|c_1)p(c_1) + p(u_2 = c_1|c_2)p(c_2) = \alpha_2/2 + (1 - \alpha_2)/2 = 1/2. \end{aligned} \tag{4}$$

The two classifiers are independent if the joint probability $p(u_1, u_2)$ factorizes

$$p(u_1 = c_1, u_2 = c_1) = p(u_1 = c_1)p(u_2 = c_1) = l\frac{1}{2} \times \frac{1}{2} = \frac{1}{4}. \tag{5}$$

And similarly,

$$p(u_1 = c_1, u_2 = c_2) = p(u_1 = c_2, u_2 = c_1) = p(u_1 = c_1)p(u_2 = c_2) = \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}. \tag{6}$$

In Equations (5)–(6), α_1 and α_2 do not appear anymore. The value of $p(u_1, u_2)$ should be $\frac{1}{4}$, independently of the classifier performances. This is possible only if $\alpha_1 = \alpha_2 = \frac{1}{2}$. Thus, the ensemble performance does not depend of the performance of the individuals. In other words, independent classifiers in the sense of definition (2) are random classifiers (recognition rate of 50%)!

Suppose now that classifiers are very efficient and that α_1 and α_2 are almost identical to 1. In this case, the probability that the two answers are correct is also almost equal to 1 and

$$p(u_1 = c_1, u_2 = c_2) \approx p(c_1) = 1/2 \neq p(u_1 = c_1)p(u_2 = c_2), \tag{7}$$

which is far from the value of $\frac{1}{4}$ required by the condition of independence.

Example (1) suggests that interesting classifiers (non-random!) cannot be independent in the sense of Equation (2). Making the assumption that decision vectors $\mathbf{u}_1, \ldots, \mathbf{u}_M$ are conditionally independent given $\mathbf{x} \in C_j$, the discriminant function $\mathcal{G}$ maximizes the posterior probability $p(C_j) \prod_{i=1}^{M} p(\mathbf{u}_i|C_j) = p(C_j) \prod_{i=1}^{M} \prod_{k=1}^{K} p(u_{ik}|C_j)$, which can be point estimated from the entries of the M KK-confusion matrices, as given, for instance, in Table 1.

Let $\mathbb{1}_{j<k}$ be the indicatrix function for which $\mathbb{1}_{j<k} = 1$ if the rank of the class C_j is less than the alternative class C_k, and 0 otherwise. Then in Table 1, $n_{jk} = \mathbb{1}_{j<k}$ and the line and column marginals are respectively defined by $n_{j.} = \sum_{k=1}^{K} n_{jk}$ and $n_{.k} = \sum_{k=1}^{K} n_{jk}$. If class C_j is the kth choice for classifier $\mathcal{K}_i$, then $p(u_{ik}|C_j) = \frac{n_{jk}}{n_{j.}}$.

Example 2 (Conditional independent classifiers)**.** *Consider once again the binary classification case introduced in Example (1) and assume that the classifiers are very efficient: $\alpha_1 = \alpha_2 \approx 1$. Then*

$$\begin{aligned} p(u_1 = c_2, u_2 = c_1|c_1) &\approx 1 \\ p(u_1 = c_1|c_1)p(u_2 = c_1|c_1) &= \alpha_1\alpha_2 \approx 1 \end{aligned} \tag{8}$$

We conclude that two classifiers can be conditionally independent even if they are very efficient. Equation (8) does not indicate that the classifiers are independent. It only suggests that they can be conditionally independent or *conditionally dependent*.

Therefore, conditional independence can be seen as a necessary condition for classifier combination. But the direct use of the confusion matrix as a criterion to derive the optimal combination rule is not feasible since the true classes are unknown.

Table 1. Confusion matrix of a classifier $\mathcal{K}_i$ used to estimate $p(U_{ik}|C_j)$ in the Bayesian approach. $U_i = R_j$ denotes the classifier decision on class being ranked jth.

		Predicted Classes					
		R_1	$\dots$	R_j	$\dots$	R_K	
True classes	C_1	n_{11}	$\dots$	n_{1j}	$\dots$	n_{1K}	$n_{1\cdot}$
	$\vdots$	$\vdots$	$\ddots$	$\vdots$		$\vdots$	$\vdots$
	C_j	n_{j1}	$\dots$	n_{jj}	$\dots$	n_{jK}	$n_{j\cdot}$
	$\vdots$	$\vdots$		$\vdots$		$\vdots$	$\vdots$
	C_K	n_{K1}	$\dots$	n_{Kj}	$\dots$	n_{KK}	$n_{K\cdot}$
		$n_{\cdot 1}$	$\dots$	$n_{\cdot j}$	$\dots$	$n_{\cdot K}$	

4. Rank Class Combination Problem

4.1. Rank-Order Statistic Model

A rank classifier gives an ordered list of classes associating each class with a integer that indicates its importance in the list; in the case of K classes, it is an integer $k \in \{1, 2, \dots, K\}$. The output of a classifier $\mathcal{K}_k$ is a vector of ranks attributed to K classes:

$$\mathbf{u}_k(\mathbf{x}) = \mathbf{r}_k = \begin{pmatrix} r_{1k} \\ r_{2k} \\ \vdots \\ r_{Kk}, \end{pmatrix}, \tag{9}$$

and $r_{jk} = \mathbf{r}_k(C_j)$ is the rank assigned to class C_j by the classifier $\mathcal{K}_k$. By convention, the smaller the rank assigned to a class, the more likely it is. In other words, $r_{ik} < r_{jk}$ if $\mathcal{K}_k$ judges C_i more likely than C_j. The vector $\mathbf{r}^{(k)}$ is therefore a permutation of the first K integers. The matrix $\mathbf{R} = \{r_{ik}\}$ represents the total order ranking of the K classes attributed by the M classifiers, i.e., $r_{ik} \neq r_{i'k}$, $\forall i' \neq i$ [28]. In the following, for ease of writing, we will denote $r_{ik} = r_i^{(k)}$. Then

$$\mathbf{R} = (\mathbf{r}_1\ \mathbf{r}_2\ \dots\ \mathbf{r}_M) = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1M} \\ r_{21} & r_{22} & \dots & r_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ r_{K1} & r_{K2} & \dots & r_{KM}, \end{pmatrix} \tag{10}$$

where $(r_{j1}\ r_{j2}\ \dots\ r_{jM})$ is the set of ranks assigned to class C_j by the M classifiers.

The solution of a rank class combination problem is a total order ranking (TOR) $\mathbf{r}^*$, given by a virtual classifier minimizing the disagreement of opinions between the M classifiers. The optimization problem is defined as follows:

$$r^* = \arg\min_{\mathbf{r}} \sum_{k=1}^{M} f(\mathbf{r}, \mathbf{r}_k), \quad \text{s.t.} \quad \mathbf{r} \in \mathbb{S}_K, \tag{11}$$

where $\mathbf{r}_k$ is the rank distribution on the K classes proposed by the classifier $\mathcal{K}_k$, $\mathbb{S}_K$ is the symmetric group of the $K!$ permutations [29], and $f : \mathbb{S}_K \times \mathbb{S}_K \to \mathbb{R}^+$ is a metric on $\mathbb{S}_K$. Solving Equation (11) is difficult due to the constraint $\mathbf{r} \in \mathbb{S}_K$. In the following subsections, the search for $\mathbf{r}^*$ conducts to a linear optimization program with an exact solution that depends on the metric used, i.e., the disagreement distance or the Condorcet distance.

The choice of these metrics is motivated by a range of properties: (*i*) both have an intuitive and plausible interpretation as a number of pairwise choices, (*ii*) they provide the best possible description of the process of ranking classes as performed by a human, (*iii*) both have a number of appealing mathematical properties such as counting rather than measuring and providing a very good concordance indicator [30,31].

4.2. Total Order Ranking with Disagreement Distance

The disagreement between the rankings from classifiers $\mathcal{K}_k$ and $\mathcal{K}_{k'}$ is measured by $f_d(\mathbf{r}_k, \mathbf{r}_{k'}) = \sum_{i=1}^{K} \operatorname{sgn} |r_{ik} - r_{ik'}|$. The kth permutation $\mathbf{r}_k$ can be represented by a permutation matrix $\mathbf{P}^{(k)} = \{x_{ij}^{(k)}\}, x_{ij}^{(k)} \in \{0,1\}$, with $x_{ij}^{(k)} = 1$ if class i is positioned in place j and 0 otherwise (see Figure 2). Therefore, the constraint $\mathbf{r} \in \mathbb{S}_K$ in Equation (11) imposes $\sum_{j=1}^{K} r_{ij}^* = \sum_{i=1}^{K} r_{ij}^* = 1, \forall i, j$. Let $\phi_d(\mathbf{r}) = \sum_{k=1}^{M} f_d(\mathbf{r}, \mathbf{r}_k) = \|\mathbf{P}, \mathbf{P}^{(k)}\|_d$ with tensor Einstein notation. Equation (11) can then be rewritten:

$$r^* = \arg\min_{\mathbf{r}} \phi_d(\mathbf{r}) = \arg\min_{\mathbf{r} \in \mathbb{S}_K} \sum_{k=1}^{M} \sum_{i=1}^{K} \operatorname{sgn} |r_i - r_{ik}|, \tag{12}$$

where r_i denotes the rank of the ith candidate in the unknown ranking $\mathbf{r}$. As $\mathbf{r}$ can be represented by its permutation matrix $\mathbf{P} = \{x_{ij}\}$, it comes from the rewriting of $r_i = \sum_j j x_{ij}$ in Equation (12):

$$\phi_d(\mathbf{r}) = \sum_{k=1}^{M} \sum_{i=1}^{K} \operatorname{sgn} |\sum_j j x_{ij} - r_{ik}| \quad \text{s.t.} \quad \sum_j x_{ij} = 1, \tag{13}$$

which is equivalent to:

$$\phi_d(\mathbf{r}) = \sum_{k=1}^{M} \sum_{i=1}^{K} \operatorname{sgn} |\sum_j (j - r_{ik}) x_{ij}| \tag{14}$$

Taking into account the summation on j and the fact that x_{ij} only takes the value 1 once (and 0 elsewhere), only $(j - r_{ik})$ corresponding to the value j for which $x_{ij} = 1$ is considered.Then

$$\phi_d(\mathbf{r}) = \sum_{k=1}^{M} \sum_{i=1}^{K} \operatorname{sgn}(\sum_{j}^{K} |j - r_{ik}| x_{ij}) = \sum_{k=1}^{M} \sum_{i=1}^{K} \sum_{j=1}^{K} (\operatorname{sgn} |j - r_{ik}|) x_{ij}. \tag{15}$$

Let us define by

$$\kappa_{ij}(\mathbf{r}) = \sum_{k=1}^{M} \operatorname{sgn} |j - r_{ik}| = \sum_{k=1}^{M} \left| x_{ij} - x_{ij}^{(k)} \right| \tag{16}$$

the cost of attributing the alternative i in position j. κ_{ij} is also the number of classifiers that don't position the alternative i in place j. $\kappa_{ij}(\mathbf{r})$ is equivalent to $m - \pi_{ij}$, where π_{ij} is the number of classifiers who do position the alternative i in place j. Given that $|x_{ij} - x_{ij}^{(k)}| = (x_{ij} - x_{ij}^{(k)})^2$ because $|x_{ij} - x_{ij}^{(k)}| \in \{0,1\}$, we obtain

$$\phi_d(\mathbf{r}) = \frac{1}{2} \sum_{k=1}^{M} \sum_{i=1}^{K} \sum_{j=1}^{K} (x_{ij} - x_{ij}^{(k)})^2 = \sum_{k=1}^{M} (K - \sum_{i=1}^{K} \sum_{j=1}^{K} x_{ij} x_{ij}^{(k)}), \tag{17}$$

and then

$$\phi_d(\mathbf{r}) = \sum_{i=1}^{K} \sum_{i=1}^{K} (m - \sum_{k=1}^{M} x_{ij}^{(k)}) x_{ij}. \tag{18}$$

In Equation (18), considering that $\pi_{ij} = \sum_{k=1}^{M} x_{ij}^{(k)}$ is the number of classifiers that position class C_i in place j, the linear objective function associated with Equation (12) is finally formulated as

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} \sum_{i=1}^{K}\sum_{j=1}^{K}(M-\pi_{ij})x_{ij} \quad \text{s.t.} \quad \pi_{ij} = \sum_{k=1}^{K} x_{ij}^{(k)}, \quad \sum_{i=1}^{M} x_{ij} = \sum_{j=1}^{M} x_{ij} = 1, \text{ and } x_{ij} \in \{0,1\}, \quad (19)$$

constrained by $\sum_i^K \pi_{ij} = \sum_j^K \pi_{ij} = K$. The form to be minimized in Equation (19) recodes the classifier combination rule, which is reduced to solve an NP-hard binary linear programming problem (see [32] for some resolution strategies).

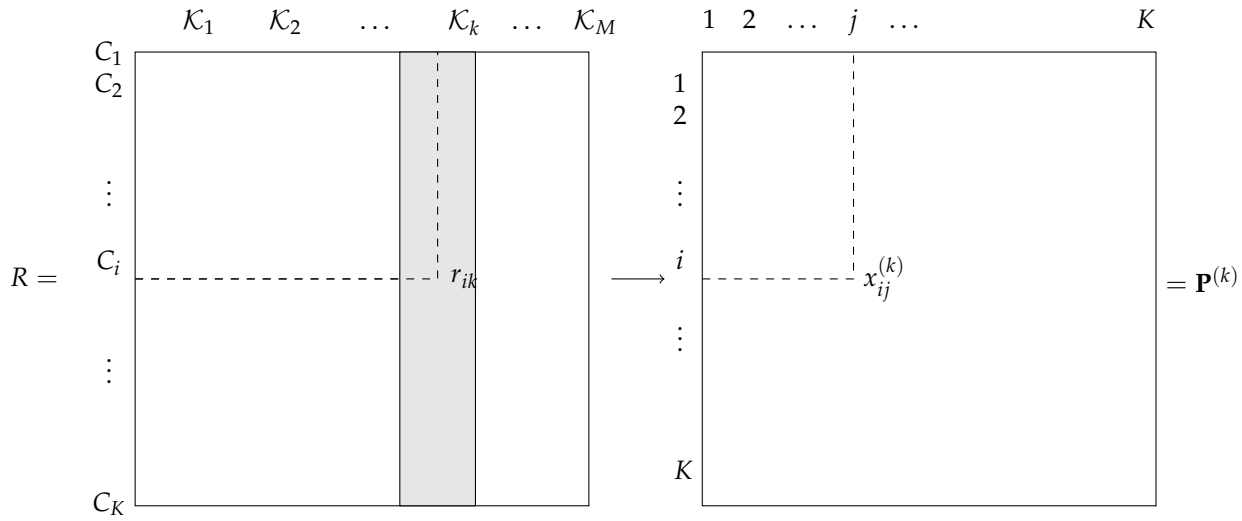


Figure 2. Permutation matrix put for the ranking of classifier $\mathcal{K}_k$.

4.3. Total Order Ranking with Condorcet Distance

To define this distance, we define a new set of matrices $\{\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(m)}\}$, where $\mathbf{Y}_{ij}^{(k)} = \{y_{ij}\} = \mathbb{1}_{i<j}$ is put for the indicator matrix of classifier $\mathcal{K}_k$ with the convention $y_{ij}^{(k)} = 1$ if the rank of class C_i is less than that of class C_j and 0 otherwise (see Figure 3).

Using the tables $\mathbf{Y}^{(k)}$ as in Section 4.2

$$f_C(\mathbf{r}_k, \mathbf{r}_{k'}) = f(\mathbf{Y}^{(k)}, \mathbf{Y}^{(k')}) = \frac{1}{2}\sum_{i}^{K}\sum_{j}^{K}|y_{ij}^{(k)} - y_{ij}^{(k')}|, \quad k, k' = 1, \ldots, M, \quad (20)$$

which can be simplified as follows in the case of total order:

$$f_C(\mathbf{r}_k, \mathbf{r}_{k'}) = \frac{1}{2}\sum_{i}^{K}\sum_{j}^{K}(y_{ik}^{(k)} - y_{ik}^{(k')})^2 = \sum_{i}\sum_{j} y_{ij}^{(k)} y_{ji}^{(k')}. \quad (21)$$

As $(y_{ij}^{(k)})^2 = y_{ij}^{(k)} = 0$ or 1, the consensus function associated with the Condorcet distance is given by

$$\phi_C(\mathbf{r}) = \frac{1}{2}\left[\sum_{i=1}^{K}\sum_{j=1}^{K} M y_{ij} + \sum_{i=1}^{K}\sum_{j=1}^{K}\left(\sum_{k=1}^{M} y_{ij}\right) - 2\sum_{i=1}^{K}\sum_{j=1}^{K} y_{ij}\sum_{k=1}^{M} y_{ij}^{(k)}\right]. \quad (22)$$

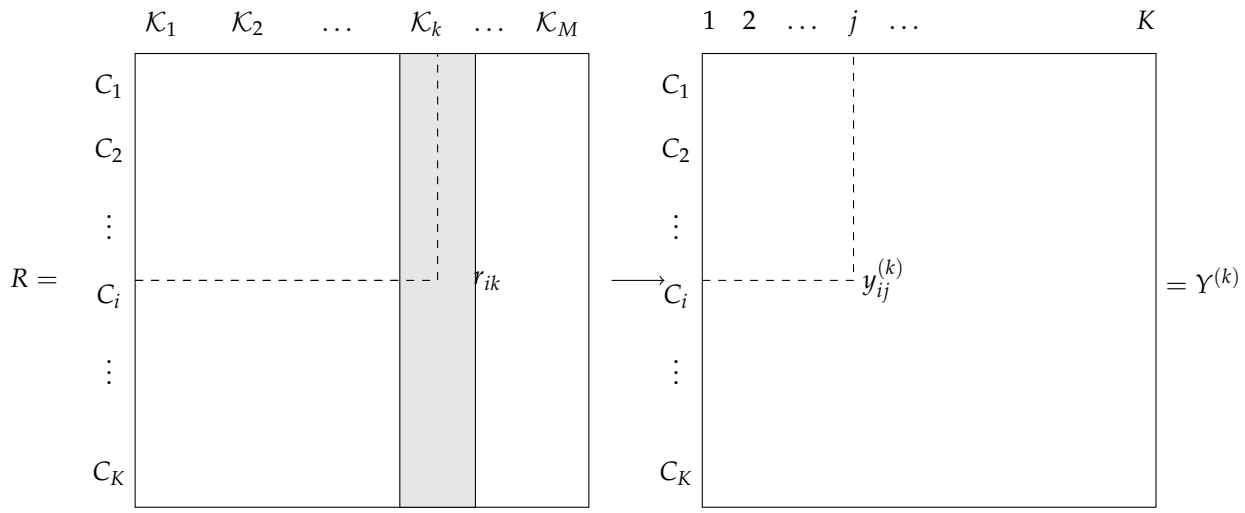


Figure 3. Condorcet matrices.

Let $\delta_{ij} = \sum_{k=1}^{K} y_{ij}^{(k)}$ be the total number of classifiers preferring class C_i to C_j. Defining $\Delta = \{\delta_{ij}\}$ as a matrix summing the M matrices $\mathbf{Y}^{(k)}$ associated with the rankings $\mathbf{r}_k$ of the classifier $\mathcal{K}_k$ allows us to rewrite ϕ_C as

$$\phi_C(\mathbf{r}) = \frac{1}{2}\left[\sum_{i=1}^{K}\sum_{j=1}^{K} M y_{ij} + \sum_{i=1}^{K}\sum_{j=1}^{K}\delta_{ij} - 2\sum_{i=1}^{K}\sum_{j=1}^{K}\delta_{ij}y_{ij}\right]. \tag{23}$$

As $\mathbf{r}$ defines a total order, $\sum_{i=1}^{K}\sum_{j=1}^{K} y_{ij} = \dfrac{K(K-1)}{2}$ and $\sum_{i=1}^{K}\sum_{j=1}^{K}\delta_{ij} < M\dfrac{K(K-1)}{2}$.
Let $\theta = \frac{1}{2}\left(M\dfrac{K(K-1)}{2} + \sum_{i=1}^{K}\sum_{j=1}^{K}\delta_{ij}\right)$. Then *theta* is constant and $\phi_C(\mathbf{r})$ is

$$\phi_C(\mathbf{r}) = \theta - \sum_{i=1}^{K}\sum_{j=1}^{K}\delta_{ij}y_{ij}. \tag{24}$$

Finally, the search for an optimal rank classifier combination conducts to the following binary linear program:

$$\begin{aligned}\max_{Y}\sum_{i=1}^{K}\sum_{j=1}^{K}\delta_{ij}y_{ij} \quad \text{s.t.} \quad \delta_{ij} = \sum_{k=1}^{K} y_{ij}^{(k)}, \quad y_{ij} + y_{ji} = 1, i < j, \quad y_{ii} = 0 \forall i \\ y_{ij} + y_{ji} - y_{ik} \leq 1, \forall i \neq j \neq k,\ y_{ij} \in \{0,1\}.\end{aligned} \tag{25}$$

From a machine learning perspective, solving Equations (19) and (25) provides deterministic matrix solutions $\mathbf{P}^*$ and $\mathbf{Y}^*$, respectively, from which $\mathbf{r}*$ is easily reconstructed, but these solutions are not necessarily identical [28].

Example 3 (Classifier ensemble aggregation rule)**.** *The problem selected to illustrate our theory is that of combining four classifiers for recognizing handwritten digits 0 to 9. Binary images from the MNIST database are used [33]. The four classifiers are tested on a sample and proposed rankings are collected in Table 2.*

The two rankings are concordant except for the predictions for digits 5 and 6.

Table 2. Proposed rank classifier combination using disagreement and Condorcet distances.

Digits	Classifier Ranks				Proposed Rank	
	$\mathcal{K}_1$	$\mathcal{K}_2$	$\mathcal{K}_3$	$\mathcal{K}_4$	Disag.	Condorcet
0	1	4	3	10	3	3
1	2	2	1	2	2	2
2	3	1	2	1	1	1
3	4	6	4	3	4	4
4	5	5	7	5	5	6
5	6	3	6	4	6	5
6	7	8	5	6	7	7
7	8	7	8	9	8	8
8	9	10	10	8	10	10
9	10	9	9	7	9	9

5. Classifier Ensemble Information Measure

Since $\operatorname{sgn}|x| \leq |x|$, then

$$f_d(\mathbf{r}_k, \mathbf{r}_{k'}) = \sum_{i=1}^{K} \operatorname{sgn}|r_{ik} - r_{ik'}| \leq \sum_{i=1}^{K} |r_{ik} - r_{ik'}|. \tag{26}$$

If $r_{ik} = K - \sum_{i=1}^{K} y_{ij}^{(k)}$, then from Equation (26),

$$\sum_{i=1}^{K} |r_{ik} - r_{ik'}| = \sum_{i}^{K} \left| \sum_{j}^{K} (y_{ij}^{(k)} - y_{ij}^{(k')}) \right| \leq \sum_{i}^{K} \sum_{j}^{K} \left| y_{ij}^{(k)} - y_{ij}^{(k')} \right| = 2 f_C(\mathbf{r}_k, \mathbf{r}_{k'}). \tag{27}$$

In summary, $f_d(\mathbf{r}_k, \mathbf{r}_{k'}) \leq f_C(\mathbf{r}_k, \mathbf{r}_{k'})$, which means that f_C is more uncertain than f_D and could be preferred for a classifier ensemble agreement. The question is, how precisely can we measure this voting conjunction?

Section 4.3 introduced a matrix representation of the information. By summing for all the tables $Y^{(k)}$, one obtains the matrix Δ defined previously. If we arrange the classifiers according to a permutation order $\Sigma = (\sigma(1), \sigma(2), \ldots, \sigma(K))$, Δ can be represented from matrix $\Delta(\Sigma)$ obtained by the permutation of rows and columns.

The objective function to minimized is given in the general case by:

$$F_C(\mathbf{r}) = \theta - (\text{sum of the elements of the upper triangular part of the matrix}), \tag{28}$$

and as follows in the case of total orders:

$$F_C(\mathbf{r}) = (\text{sum of the elements of the lower triangular part of the matrix}). \tag{29}$$

A measure of classifier ensemble agreement is a coefficient between 0 and 1 measuring the intensity of the link between the set of classifier votes. The closer its value is to 1, the more the opinions of the classifiers are in agreement. Conversely, the closer their value is to 0, the greater the disagreement between the votes. Here, we give the coefficients of concordance for the two metrics.

5.1. Disagreement Distance

Theorem 1 (Conjunction coefficient interval for the disagreement metric)**.** *Let $\{\mathcal{K}_i\}_{i=1}^{M}$ be an ensemble of conditionally independent classifiers voting on K classes. Then, the interval of variation of the conjunction coefficient I_d is* $[0, 1]$.

See Appendix A for the proof.

5.2. Condorcet Distance

If M classifiers vote on K classes with pairing order comparison matrices $\mathbf{Y}^{(k)}$, the sum of which makes it possible to obtain $\Delta = \{\delta_{ij}\}$ with $\delta_{ij} = \sum_{k=1}^{K} y_{ij}^{(k)}$, as defined in Section 4.3, the conjunction coefficient is defined as

$$I_C = 4\frac{\sum_{j=1}^{K}\sum_{j=1}^{K}\delta_{ij}(\delta_{ij}-1)}{M(M-1)K(K-1)} - 1. \tag{30}$$

Theorem 2 (Conjunction coefficient interval for the Condorcet metric)**.** *Let $\{\mathcal{K}_i\}_{i=1}^{M}$ be an ensemble of conditionally independent classifiers voting on K classes. Then the interval of variation of the conjunction coefficient I_C defined by (30) is*

$$I_C \in \begin{cases} [-\frac{1}{M};1] & \text{if } M \text{ is even,} \\ [-\frac{1}{M-1};1] & \text{otherwise.} \end{cases} \tag{31}$$

See Appendix B for the proof.

6. Experiments

6.1. The Detection of Cervical Cancer

Many studies have shown evidence that cervical cancer may be imputed to a subset of DNA viruses called *human papillomavirus* (HPV) (referred to as risky patients)that infect cutaneous and mucosal epithelia, and in which acute infection causes benign cutaneous lesions [34,35]. Some of these viruses infect the genital tract and cause malignant tumors, which are most commonly located in the cervix. Even though most of these infections are controlled by the immune system, some remain persistent and are ascribed to different types of cancers and particularly, to cervical cancer. In 2016, cervical cancer represented the 12th most lethal female cancer in the European Union, accounting for 13,500 deaths a year and 30,400 new cases a year. Therefore, cervical cancer screening still continues to play a critical role in the control of cervical cancer. However, the screening of a smear is nowadays mostly made manually: a pathologist inspects each cell of a smear with a microscope to check if it is atypical or not. Consequently, human error is always possible, and in particular, mistakenly diagnosing atypical cells as normal. This situation can occur because of the practitioner's fatigue or a lack of experience or concentration. In addition, diagnosis is also linked to the preparation of cells, and in some situations, atypical cells can be partially hidden by others, which makes their interpretation or classification difficult. In addition, the presence of atypical cells in the entire studied population is very uncommon (up to 1‰) which makes the detection task even more difficult. Therefore, an error is easily possible. This could have irreversible effects on the evolution of the cancer and can impact treatment. The introduction of an automatic procedure, able to point out the pathological cells, would both help the practitioner in his diagnosis and improve or strengthen it.

Depending on the morphology of the nuclei of the cells, the diagnosis varies: if a nucleus is considered normal and all of the cells removed have the same diagnosis, then the cervix is considered normal. On the other hand, if a nucleus is considered abnormal, the diagnosis is not automatically associated with a risky smear.

We propose to test our classifier combination strategy to cluster cells into three different classes (normal cells, atypical cells, and debris) using a certain number of classifiers.

6.2. The Dataset

The cytological dataset is constituted of smear images from 14 different women. They generally comprise more than one hundred cells characterized by 42 morphological or textural variables. Nine showed a negative HPV test and the other five, a positive test. In addition, few observations were labeled by an expert who pointed out some atypical cells and noisy objects. The dataset is presented

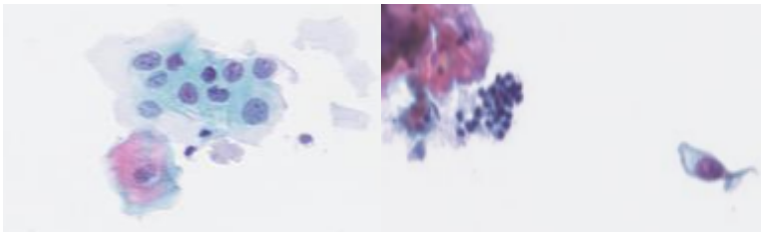
in detail in Table 3. Among the most recurrent patterns of abnormal cells are nucleus regularity or a swollen aspect, nucleus size, important optical density, number of nucleoli, high core/cytoplasm ratio, ratio of minimum/maximum width of the nucleus, etc.

The images were colored with Papanicolaou stain, which is the most widely used reference color for the screening of cervical cancers; it makes it possible to distinguish the different nuclei, which are colored in blue, the mother cells in dark purple to black, and the keratinized and squamous epithelium. The images were then segmented into thumbnail images of 16×16 pixels which correspond a priori to objects. Most of the time, these objects are nuclei, but they may sometimes be non-identified objects that we call "noise". Indeed, they can correspond, for example, to a poor segmentation, a superimposed nuclei, etc.

Table 3. Dataset characteristics.

HPV Test	Total Number of Cells	Number (or %) of			
		– Debris –		– Cancer –	
positive	405	78	*(0.19)*	49	*(0.12)*
positive	114	19	*(0.17)*	8	*(0.07)*
positive	206	31	*(0.15)*	13	*(0.06)*
positive	448	30	*(0.06)*	2	*(0.004)*
positive	519	70	*(0.13)*	33	(0.06)
negative	137	13	*(0.09)*	–	–
negative	76	5	*(0.06)*	–	–
negative	211	84	*(0.39)*	–	–
negative	251	31	*(0.12)*	–	–
negative	251	52	*(0.20)*	–	–
negative	257	40	*(0.15)*	–	–
negative	223	24	*(0.11)*	–	–
negative	691	155	*(0.22)*	–	–
negative	67	23	*(0.24)*	–	–
Total	3857	655	*(0.17)*	105	*(0.02)*

A few observations were labeled by an expert who pointed out some atypical cells and noisy objects. The fact that a nucleus has one of these characteristics does not always imply its malignancy. In fact, a cell can have a singular morphology but not be infected, and others may present abnormalities that correspond to pre-cancerous lesions such as dysplastic cells and in situ carcinomas or to cancerous cells. Figure 4a shows a cluster of abnormal cells (with large nuclei) that are not yet cancerous, because of their low density, unlike Figure 4b, where one can observe a set of abnormal cells with dense nuclei.



(**a**) (**b**)

Figure 4. Images of cervical cells colored with Papanicolaou stain. (**a**) Clumps of abnormal cells with large nuclei. (**b**) Abnormal cells with dense nuclei.

Table 4 summarizes the characteristics of the dataset. First, the observed data come from samples of 14 different smears, which supposes the existence of inter-individual variability (confirmed by tests of variance between the HPV negatives, the HPV positives, or between the two types of population; the 5% risk threshold tests rejected the assumption of equality of means for all variables). However,

it is possible that this variability is simply relative to the studied dataset, in the sense that the study was done on a small number of smear samples. This assumption remains to be verified on larger databases. It can also be noted in Table 3 that the known population of "abnormal" cells remains very low in proportion to the other classes, and, in contrast, the recognized "default/waste" class represents more than 15% of the data. The low proportion of the target class and the heterogeneity of the debris present obstacles for clustering. This means that, among the cells belonging to risky patient smears, there exists a non-null risk that some nuclei are atypical. Iin practice, this proportion is usually very low (0.1% to 5%).

Table 4. Overview of the studied dataset.

	No. of Patients	No. of Nuclei	No./Ype of Data
control patients	9	2165	427/noisy objects
risky patients	5	1692	105/atypical nuclei
	–	–	228 / noisy objects
Total	14	3857	760 objects

From this image segmentation, morphological and photometric features are extracted and computed. In total, the studied dataset has 3857 cell samples belonging to 14 different smears and consists of 42 variables: variables 1 to 19 represent morphological variables, and the rest corresponds to textural and photometric characters. The channel of treatments from the smear image to the dataset is reported in Figure 5.

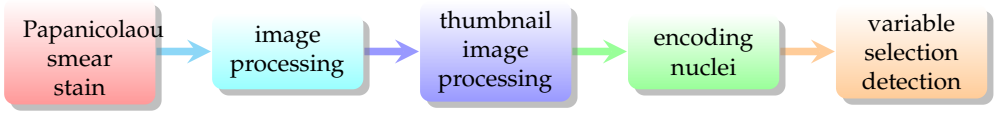


Figure 5. Overview of the processing chain.

Each smear was pre-processed according to a standardized protocol: cell collection, spreading a thin layer on slides, and the staining of these slides. Each slide was then scanned, segmented cell by cell, and finally, underwent an extraction of 42 morphological and textural characteristics.

6.3. Experimental Protocol

Two-layer multilayer perceptrons (MLPs) were chosen as classifiers to produce the desired outputs, which were ordered to produce the ranks. Each multilayer perceptron (MLP) contains 42 input units, 10 hidden units, and 3 output units. Training was achieved using a learning rate of 0.1 and a momentum of 0.9 for two epochs on the training set. We deliberately trained the MLPs without optimization of a validation set. It is important to stress that the training set for the classifier was not the same set as the test set, the ensure that the experiments would be unbiased. The best results obtained for an MLP were a classification error rate of 0.159 ± 0.022 and a false positive rate (FPR) (or false alarm ratio) of 0.133 ± 0.050. The FPR is the number of false positives divided by the total number of negatives N, i.e., FP/N. The false negative rate (FNR) is the number of false negatives divided by the number of real positive cases in the data, i.e., FN/P. In practice, this is a test result that indicates that a condition does not hold, while in fact it does.

In order to assess the efficiency of the rank classifier combination algorithms, error rates were computed from a certain percentage of nuclei whose labels were known. This represented 70% of the observations in a subsample, as we took into account the 20 labeled atypical nuclei randomly selected, and we also assumed that those coming from control patients (120) were all normal nuclei. We proceeded in the same manner to compute the FPR which stands for the percentage of actual atypical nuclei mis-classified.

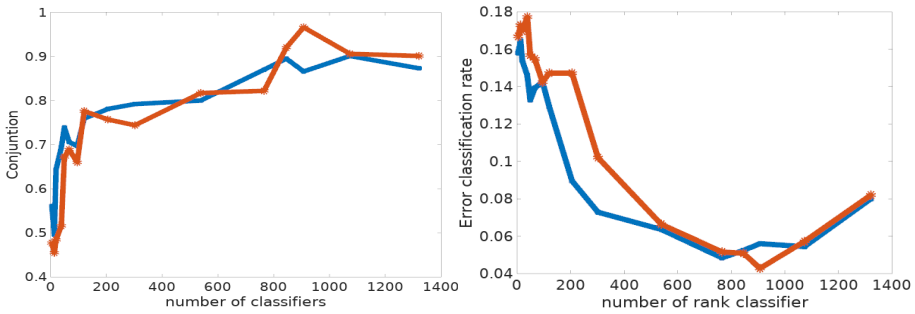
In Table 5 and in Figure 6a, we report the classification error rate computed from the data with known labels and its corresponding FPR, for the two procedures. First of all, we can observe that the Condorcet combination rule shows the best performances in terms of classification error rate and FPR (see also Figure 6c). Indeed, only 4.28% of cells are mis-classified, whereas the disagreement combination rule has a mis-classification rate of 4.84% in the best case, with 765 classifiers. The main conclusion is that the success ratio is strongly improved when combining classifiers. However, it is disappointing to see that the Condorcet algorithm results in a significant number of false negatives (pathological cells classified as normal ones); the FNR also remains relatively high, around 10%, in many simulations. Indeed, the classification risk is not symmetric here: the detection of pathological cells activates the decision for treatment, and their absence implies an absence of treatment.

Table 5. Classification results with disagreement and Condorcet combination rules using a set of M classifiers (with $4 \leq M \leq 1321$).

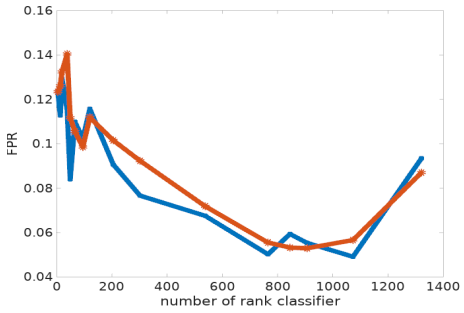
Disagreement Distance					**Condorcet Distance**				
I_d	M	**Error Rate**	**FPR**	**FNR**	I_C	M	**Error Rate**	**FPR**	**FNR**
0.873	1321	0.0800	0.0934	0.0644	0.901	1321	0.0820	0.0870	0.0777
0.901	1073	0.0544	**0.0491**	0.0576	0.906	1073	0.0572	0.0566	0.0561
0.866	907	0.0560	0.0553	0.0562	0.966	907	**0.0428**	0.0529	**0.0313**
0.895	845	0.0524	0.0593	**0.0464**	0.920	845	0.0508	**0.0532**	0.0465
0.870	765	**0.0484**	0.0502	0.0500	0.822	765	0.0516	0.0555	0.0493
0.800	538	0.0636	0.0675	0.0600	0.817	538	0.0664	0.0718	0.0592
0.792	302	0.0728	0.0766	0.0710	0.744	302	0.1020	0.0923	0.1126
0.781	205	0.0896	0.0906	0.0864	0.757	205	0.1472	0.1015	0.1968
0.759	120	0.1292	0.1157	0.1439	0.776	120	0.1472	0.1118	0.1846
0.697	95	0.1424	0.1023	0.1809	0.660	95	0.1424	0.0985	0.1888
0.706	66	0.1392	0.1098	0.1667	0.689	66	0.1548	0.1055	0.2090
0.739	49	0.1328	0.0840	0.1854	0.672	49	0.1568	0.1117	0.2067
0.694	38	0.1460	0.1152	0.1770	0.516	38	0.1772	0.1404	0.2233
0.643	19	0.1540	0.1294	0.1801	0.484	19	0.1696	0.1323	0.2032
0.496	13	0.1644	0.1127	0.2224	0.455	13	0.1728	0.1261	0.2248
0.561	4	0.1580	0.1238	0.1962	0.477	4	0.1668	0.1234	0.2130

We compared the clustering partition obtained by the three competitors: sparse k-means (SkM) proposed by Witten and Tibshirani [36,37], general sparse multi-class linear discriminant analysis (GSM-LDA) [38], and sparse EM (sEM)by Zhong et al. [39]. First, we can observe that among these algorithms, the sEM shows the best performance in terms of clustering accuracy. Only 9% of observations are mis-classified, on average, whereas the GSM-LDA algorithm has a mis-classification rate of 15.9%, and the SkM algorithm mis-classifies 19.2% of nuclei. However, the sparse approaches provide a better clustering results from a medical point of view since the results can be interpreted conversely to the LDA-type algorithm, for which the fitted discriminative axis is a linear combination of the original variables. Therefore, SkM and sEM provide information which can be interpreted to better understand both the data and the phenomenon.

The rank classifier combination provides the best classification results. We can observe that the global clustering error rates are considerably reduced (Table 6). Indeed, the best error rate reaches 4.28% with 907 classifiers and a conjunction coefficient of 96.6%.



(**a**) Conjunction coefficients (**b**) Error classification rate



(**c**) False positive rate

Figure 6. Graphic representations of the classification results for disagreement (blue) and Condorcet (red) distances.

Table 6. Results obtained for the sparse *k*-means (SkM), general sparse multi-class linear discriminant analysis (GSM-LDA), and sparse EM (sEM) algorithms: Average and standard error of clustering error rate, false positive rate FPR, and false negative rate FNR on 20 simulations.

Algorithm	Error Rate	FPR	FNR
SkM [36]	0.192 ± 0.016	0.205 ± 0.044	0.165 ± 0.084
GSM-LDA [38]	0.159 ± 0.022	0.133 ± 0.050	0.118 ± 0.099
sEM [39]	0.090 ± 0.047	0.077 ± 0.022	0.062 ± 0.061

7. Conclusions and Future Research

In this paper, we show that an exact optimal combination rule for a rank classifier ensemble can be computed as the solution to a binary linear programming problem. This rule can be seen as a total order ranking attributed to K classes by a virtual voter resuming the points of view of M voters. One could also stand the dual problem of the previous one, i.e., is there a distribution of marks or values that could have been attributed to a virtual class C by the m voters? The first problem is related to the idea of aggregating points of view, the second with the idea of summarizing profiles.

We compared disagreement and Condorcet metrics, making it possible to quantify the consensus between the classifiers with a conjunction coefficient. The optimal rankings are not the same, i.e., the solution depends of the metric used. But they have shown their efficiency, in addition to the appealing property of being deterministic algorithms: they improve the classification results and ease the interpretation and the understanding of the results. Another point worth mentioning is the theoretical capability of handling the reject option. A weak point of this technique is that it treats all classifiers equally and does not take into account individual classifier capabilities. This disadvantage can be reduced to a certain degree by applying weights. The weights can be different for every classifier, which in turn requires additional training. This idea deserves to be further explored.

The role of variable selection appears to be significant, as it enables the improvement of both the clustering partition and the modeling of the atypical cells in the cancer detection smear (see Figure 5). In the future, we propose including a rule to rank the selected features and to investigate how the number and nature of classifiers influence the results of the rank classifier combination.

Author Contributions: In this research paper H.M.'s contributions are in investigation and conceptualization. V.V. made substantial contributions to conception and design and in interpretation of data. Both participated in drafting the article.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Conjunction Coefficient Extreme Values for Disagreement Metric

As a proof, consider the matrix $\mathbf{ß} = \{\pi_{ij}\}$ defined in Section 4.2 in the case when the rankings $\mathbf{r}_k$ are total orders. Then

$$\sum_{j=1}^{K}\sum_{j=1}^{K}\pi_{ij}(\pi_{ij}-M)=0, \tag{A1}$$

if all of the classifiers propose the same rankings, because $\pi_{ij}=0$ or $\pi_{ij}=M$. Then $\pi_{ij}(\pi_{ij}-M)=0, \quad \forall i,j$. From (A1), $\sum_{j=1}^{K}\sum_{j=1}^{K}\pi_{ij}^2 = KM^2$. Therefore, we define the conjunction coefficient as

$$I_d = \frac{\sum_{j=1}^{K}\sum_{j=1}^{K}\pi_{ij}^2}{KM^2}. \tag{A2}$$

$I_d \leq 1$ because $\sum_{j=1}^{K}\sum_{j=1}^{K}\pi_{ij}^2 \leq \sum_{j=1}^{K}\sum_{j=1}^{K}\pi_{ij}M$ since $\pi_{ij}\leq M \ \ \forall(i,j)$.

Appendix B. Conjunction Coefficient Extreme Values for the Condorcet Metric

The minimum value is obtained for a maximum disagreement, i.e., if most classifiers prefer i to j than the opposite, or, mathematically, if $\delta_{ij}+\delta_{ji}=0$ or $\delta_{ij}=\delta_{ji}=\frac{M}{2}, \ \forall i,j$.

In the case of M being even

$$\sum_{i<j}^{K}(\delta_{ij}-\delta_{ji})^2 = \sum_{i<j}^{K}\delta_{ij}^2 + \sum_{i<j}^{K}\delta_{ji}^2 - 2\sum_{i<j}^{K}\delta_{ji}\delta_{ij}. \tag{A3}$$

As $\sum_{i}^{K}\sum_{j}^{K}(\delta_{ij}-\delta_{ji})^2=0$ and $2\sum_{i}^{K}\sum_{j}^{K}\delta_{ij}\delta_{ji}=\frac{M^2K(K-1)}{4}$, then

$$\sum_{i}^{K}\sum_{j}^{K}\delta_{ij}^2 = \frac{M^2K(K-1)}{4}. \tag{A4}$$

Therefore,

$$\sum_{i}^{K}\sum_{j}^{K}\delta_{ij}(\delta_{ij}-1) = \frac{K(K-1)}{2}\left[\frac{M^2}{2}-M.\right]. \tag{A5}$$

Finally,

$$I_C = \frac{M-2}{M-1}-1 = -\frac{1}{M-1}. \tag{A6}$$

The case of M being odd. Let $M=2m+1$. In the case of maximum disagreement, then $\sum_{i<j}^{K}(\delta_{ij}-\delta_{ji})=\pm1$ and $\delta_{ij}\delta_{ji}=m(m+1), \forall i,j$. It comes $\sum_{i<j}^{K}\delta_{ij}(\delta_{ij}-1)=\sum_{i<j}1=\frac{K(K-1)}{2}$. Moreover,

$$\sum_{i}^{K}\sum_{j}^{K}\delta_{ij}^2 = \frac{K(K-1)}{2}[m(m+1)+1] \tag{A7}$$

and

$$\sum_{i}^{K}\sum_{j}^{K}\delta_{ij}(\delta_{ij}-1)=\frac{K(K-1)}{2}[m(m+1)+1-2m-1]=\frac{K(K-1)}{2}m^2. \tag{A8}$$

Finally,

$$I_C=\frac{4m^2}{2m(2m+1)}-1=-\frac{1}{M}. \tag{A9}$$

References

1. Schapire, R.E. Using output codes to boost multiclass learning problems. In Proceedings of the Fourteenth International Conference on Machine Learning, Nashville, TN, USA, 8–12 July 1997; pp. 313–321.
2. Woźniak, M.; Graña, M.; Corchado, E. A survey of multiple classifier systems as hybrid systems. Special Issue on Information Fusion in Hybrid Intelligent Fusion Systems. *Inf. Fusion* **2014**, *16*, 3–17. [CrossRef]
3. Oza, N.; Tumer, L. Classifier ensembles: Select real-world applications. *Inf. Fusion* **2008**, *9*, 4–20. [CrossRef]
4. Han, M.; Zhu, X.; Yao, W. Remote sensing image classification based on neural network ensemble algorithm. *Neurocomputing* **2012**, *78*, 133–138. [CrossRef]
5. Raj Kumar, P.A.; Selvakumar, S. Distributed Denial of Service Attack Detection Using an Ensemble of Neural Classifier. *Comput. Commun.* **2011**, *34*, 1328–1341. [CrossRef]
6. Bolton, R.J.; Hand, D.J. Statistical Fraud Detection: A Review. *Stat. Sci.* **2002**, *17*, 235–255.
7. Nanni, L. Ensemble of classifiers for protein fold recognition. *Neurocomputing* **2006**, *69*, 850–853. [CrossRef]
8. Vigneron, V.; Duarte, L.T. Rank-order principal components. A separation algorithm for ordinal data exploration. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio, Brazil, 8–13 July 2018; pp. 1036–1041.
9. Altinçay, H.; Demirekler, M. An information theoretic framework for weight estimation in the combination of probabilistic classifiers for speaker identification. *Speech Commun.* **2000**, *30*, 255–272. [CrossRef]
10. Yang, S.; Browne, A. Neural network ensembles: Combining multiple models for enhanced performance using a multistage approach. *Expert Syst.* **2004**, *21*, 279–288. [CrossRef]
11. Wozniak, M. *Hybrid Classifiers: Methods of Data, Knowledge, and Classifier Combination*; Number 519 in Studies in Computational Intelligence; Springer: Berlin/Heidelberg, Germany, 2014.
12. Kuncheva, L.I. Classifier Ensembles for Changing Environments. In Proceedings of the 5th International Workshop on Multiple Classifier Systems, Cagliari, Italy, 9–11 June 2004; pp. 1–15.
13. Bhatt, N.; Thakkar, A.; Ganatra, A.; Bhatt, N. Ranking of Classifiers based on Dataset Characteristics using Active Meta Learning. *Int.J. Comput. Appl.* **2013**, *69*, 31–36. [CrossRef]
14. Abaza, A.; Ross, A. Quality Based Rank-level Fusion in Multibiometric Systems. In Proceedings of the 3rd IEEE International Conference on Biometrics: Theory, Applications and Systems, Washington, DC, USA, 28–30 September 2009; pp. 459–464.
15. Li, Y.; Wang, N.; Perkins, E.; Zhang, C.; Gong, P. Identification and optimization of classifier genes from multi-class earthworm microarray dataset. *PLoS ONE* **2010**, *5*, e13715. [CrossRef]
16. García-Lapresta, J.L.; Martínez-Panero, M. Borda Count Versus Approval Voting: A Fuzzy Approach. *Public Choice* **2002**, *112*, 167–184. [CrossRef]
17. Zhang, H.; Su, J. Naive Bayesian Classifiers for Ranking. In Proceedings of the 15th European Conference on Machine Learning, Pisa, Italy, 20–24 September 2004; Volume 3201, pp. 501–512.
18. Dietterich, T.G. Ensemble Methods in Machine Learning. In Proceedings of the First International Workshop on Multiple Classifier Systems, Cagliari, Italy, 21–23 June 2000; Springer: London, UK, 2000; pp. 1–15.
19. Denison, D.D.; Hansen, M.; Holmes, C.C.; Mallick, B.; Yu, B. *Nonlinear Estimation and Classification*; Number 171 in Lecture Notes in Statistic; Springer: New York, NY, USA, 2003.
20. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [CrossRef]
21. Lee, S.; Kouzani, A.; Hu, E. Random forest based lung nodule classification aided by clustering. *Comput. Med. Imaging Graph.* **2010**, *34*, 535–542. [CrossRef] [PubMed]
22. Panigrahi, S.; Kundu, A.; Sural, S.; Majumdar, A. Credit card fraud detection: A fusion approach using Dempster–Shafer theory and Bayesian learning. *Inf. Fusion* **2009**, *10*, 354–363. [CrossRef]

23. Hansen, L.; Salamon, P. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.* **1990**, *12*, 993–1001. [CrossRef]
24. Anthimopoulos, M.; Christodoulidis, S.; Ebner, L.; Christe, A.; Mougiakakou, S. Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network. *IEEE Trans. Med. Imaging* **2016**, *35*, 1207–1216. [CrossRef]
25. Kawaguchi, K. Deep Learning without Poor Local Minima. In *Advances in Neural Information Processing Systems 29*; Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2016; pp. 586–594.
26. Datta, S.; Pihur, V.; Datta, S. An adaptive optimal ensemble classifier via bagging and rank aggregation with applications to high dimensional data. *BMC Bioinform.* **2010**, *11*, 427. [CrossRef] [PubMed]
27. Nadal, J.; Legault, R.; Suen, C. Complementary algorithms for the recognition of totally unconstrained handwritten numerals. In Proceedings of the 10th International Conference on Pattern Recognition, Atlantic City, NJ, USA, 16–21 June 1990; pp. 443–446.
28. Brüggemann, R.; Patil, G. *Ranking and Prioritization for Multi-Indicator Systems: Introduction to Partial Order Applications*; Environmental and Ecological Statistics; Springer: New York, NY, USA, 2011.
29. Benson, D. *Representations of Elementary Abelian p-Groups and Vector Bundles*, 1st ed.; Cambridge Tracts in Mathematics; Cambridge University Press: Cambridge, UK, 2016.
30. Vigneron, V.; Duarte, L. Toward Rank Disaggregation: An Approach Based on Linear Programming and Latent Variable Analysis. In *Latent Variable Analysis and Signal Separation*; Tichavský, P., Babaie-Zadeh, M., Michel, O.J., Thirion-Moreau, N., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 192–200.
31. Gehrlein, W.; Lepelley, D. *Voting Paradoxes and Group Coherence: The Condorcet Efficiency of Voting Rules*, 1st ed.; Studies in Choice and Welfare; Springer: Berlin/Heidelberg, Germany, 2011.
32. Korte, B.; Vygen, J. *Combinatorial Optimization: Theory and Algorithms*, 4th ed.; Springer Publishing Company, Incorporated: Berlin/Heidelberg, Germany, 2007.
33. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
34. Li, G.; Guillaud, M.; Follen, M.; MacAulay, C. Double staining cytologic samples with quantitative Feulgen-thionin and anti-Ki-67 immunocytochemistry as a method of distinguishing cells with abnormal DNA content from normal cycling cells. *Anal. Quant. Cytopathol. Histopathol.* **2012**, *34*, 273–284.
35. Scheurer, M.; Guillaud, M.; Tortolero-Luna, G.; McAulay, C.; Follen, M.; Adler-Storthz, K. Human papillomavirus-related cellular changes measured by cytometric analysis of DNA ploidy and chromatin texture. *Cytom. Part B Clin. Cytom.* **2007**, *72*, 324–331. [CrossRef]
36. Witten, D.M.; Tibshirani, R. A framework for feature selection in clustering. *J. Am. Stat. Assoc.* **2010**, *105*, 713–726. [CrossRef] [PubMed]
37. Kondo, Y.; Salibian-Barrera, M.; Zamar, R. RSKC: An R Package for a Robust and Sparse K-Means Clustering Algorithm. *J. Stat. Softw. Artic.* **2016**, *72*, 1–26. [CrossRef]
38. Safo, S.E.; Ahn, J. General Sparse Multi-class Linear Discriminant Analysis. *Comput. Stat. Data Anal.* **2016**, *99*, 81–90. [CrossRef]
39. Zhong, M.; Tang, H.; Chen, H.; Tang, Y. An EM algorithm for learning sparse and overcomplete representations. *Neurocomputing* **2004**, *57*, 469–476. [CrossRef]