A Parameter Randomization Approach for Constructing Classifier Ensembles

Enrica Santucci, Luca Didaci, Giorgio Fumera, Fabio Roli

Dept. of Electrical and Electronic Eng., University of Cagliari
Piazza d'Armi, 09123 Cagliari, Italy
Email addresses: enrica.santucci@gmail.com, didaci@diee.unica.it, fumera@diee.unica.it, roli@diee.unica.it
URL: http://pralab.diee.unica.it

Abstract

Randomization-based techniques for classifier ensemble construction, like Bagging and Random Forests, are well known and widely used. They consist of independently training the ensemble members on random perturbations of the training data or random changes of the learning algorithm. We argue that randomization techniques can be defined also by directly manipulating the parameters of the base classifier, i.e., by sampling their values from a given probability distribution. A classifier ensemble can thus be built without manipulating the training data or the learning algorithm, and then running the learning algorithm to obtain the individual classifiers. The key issue is to define a suitable parameter distribution for a given base classifier. This also allows one to re-implement existing randomization techniques by sampling the classifier parameters from the distribution implicitly defined by such techniques, if it is known or can be approximated, instead of explicitly manipulating the training data and running the learning algorithm. In this work we provide a first investigation of our approach, starting from an existing randomization technique (Bagging): we analytically approximate the parameter distribution for three well-known classifiers (nearest-mean, linear and quadratic discriminant), and empirically show that it generates ensembles very similar to Bagging. We also give a first example of the definition of a novel randomization technique based on our approach.

Keywords: Multiple classifier systems, Ensemble construction techniques,

1. Introduction

Ensembles methods have become a state-of-the-art approach for classifier design [1, 2]. Among them, ensemble construction techniques based on randomization are well-known and widely used, e.g., Bagging [6], Random Subspace Method [3], Random Forests [4], and the more recent Rotation Forests [7]. Randomization techniques have been formalized in [4] as independently learning several individual classifiers using a given learning algorithm, after randomly manipulating the training data or the learning algorithm itself. For instance, Bagging and Random Subspace Method consist in learning each individual classifier respectively on a bootstrap replicate of the original training set, and on a random subset of the original features; Random Forests (ensembles of decision trees) combine the bootstrap sampling of the original training set with a random selection of the attribute of each node, among the most discriminative ones.

The main effect of randomization techniques, and in particular Bagging, is generally believed to be the reduction of the variance of the loss function of a base classifier. Accordingly, they are effective especially for *unstable* classifiers, i.e., classifiers that exhibit large changes in their output as a consequence of small changes in the training set, like decision trees and neural networks, as opposed, e.g., to the nearest neighbor classifier [6]. It is worth noting that randomization techniques operate in *parallel*, contrary to another state-of-the-art approach, boosting, which is a *sequential* ensemble construction technique [8].

In this work we propose a new approach for defining randomization techniques, inspired by the fact that existing ones can be seen as implicitly inducing a probability distribution on the parameters of a base classifier. Accordingly, we propose that new randomization techniques can be obtained by directly defining a *suitable* parameter distribution for a given classifier, as a function of the training set at hand; an ensemble can therefore be built by directly sampling the parameter values of its members from such a distribution, without actually

manipulating the available training data nor running the learning algorithm. In this way, an ensemble can be obtained even without having access to the training set, but having access only to a pre-trained classifier. Some information about the training set, such as mean and covariance matrix, is enough to apply our method, and it could be obtained from a pre-trained classifier.

Our approach also allows a different implementation of existing randomization techniques. If the distribution induced by a given technique on the parameters of a given base classifier is known or can be approximated, one could build an ensemble as described above, instead of running the corresponding procedure and then the learning algorithm.

As mentioned above, the key issue of our approach is to define a suitable parameter distribution for a given base classifier, i.e., capable of providing a trade-off between accuracy and diversity of the resulting classifiers which is advantageous in terms of ensemble performance. To our knowledge no previous work investigated the distribution of classifier parameters induced by randomization techniques, which is not a straightforward problem. To take a first step in this direction, in this work we start from the analysis and modelling of the distribution induced by one of the most popular techniques, Bagging, on base classifiers that can be dealt with analytically: the nearest mean, linear discriminant, and quadratic discriminant classifiers. We then assess the accuracy of our model by comparing the corresponding, empirical parameter distribution with the one produced by Bagging. The results of our analysis, that have to be extended in future work to other base classifiers and randomization techniques, are aimed at obtaining insights on the parameter distributions induced by existing randomization techniques, and thus hints and guidelines for the definition of *novel* techniques based on our approach. We give a first example of the definition of a new randomization technique, starting from our model of the distribution induced by Bagging on the classifiers mentioned above.

The rest of this paper is structured as follows. In Sect. 2 we summarize the main relevant concepts about randomization techniques and Bagging. We then present our approach and describe the considered base classifiers in Sect. 3.

In Sect. 4 we model the parameter distribution induced by Bagging on such classifiers. In Sect. 5 we empirically evaluate the accuracy of our model, and give an example of the definition of new randomization techniques based on our approach. In Sect. 6 we discuss limitations and extensions of our work.

2. Background

The notation used in this paper is summarized in Table 1. We shall use Greek letters to denote probability distribution parameters, and Roman letters for other quantities, including estimated distribution parameters (statistics); vectors in Roman letters will be written in bold. For a given statistic $\mathbf{a}$ estimated from a training set we shall denote by $\mathbf{a}^*(j)$ its j-th bootstrap replicate, and with $\mathbf{a}^*$ the corresponding random variable.

Randomization techniques for ensemble construction can be formalized as follows [4]. Given a feature space $\mathcal{X} \subseteq \mathbb{R}^d$, a set of class labels $\mathcal{Y}$, a training set $T = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, a base classifier and its learning algorithm $\mathcal{L}$, a randomization technique R independently learns N different classifiers $h_j(\cdot;\theta_j)$, $j = 1, \ldots, N$, by repeatedly calling $\mathcal{L}$, where $\theta_1, \ldots, \theta_N$ are independent and identically distributed (i.i.d.) realizations of some random variable Θ_{R}. In practice, the above idea can be implemented by introducing some randomness into the training process of the individual classifiers, by manipulating either the training data or the learning algorithm, or both.

As an example, we focus here on the popular Bagging technique. It has been originally devised for regression tasks, with the aim of reducing the variance of the expected error (mean squared error) of a given regression algorithm, and has been extended to classification algorithms [6]. According to the above formalization, the corresponding random variable Θ_{R} is associated with the bootstrap sampling procedure: its values correspond to the possible bootstrap replicates T^* of the original training set T of size n, obtained by randomly drawing with replacement n instances from it (hence the name "Bagging", which is an acronym for "bootstrap aggregating"). Each base classifier h_j, $j = 1, \ldots, N$, is learned

Table 1: Summary of the notation used in this paper

Symbol	Meaning
$\mathcal{X},\ \mathcal{Y}$	Feature space and class label set
$(\mathbf{x}_i, y_i) \in \mathcal{X} \times \mathcal{Y}$	Feature vector and label of the i-th instance
T	Training set
$\mathcal{L}$	Learning algorithm
$h : \mathcal{X} \mapsto \mathcal{Y}$	Individual classifier
R	Randomization technique
Θ_{R}	Random variable associated to R
$\Psi(\Theta_{\mathrm{R}})$	Random variable of the classifier parameters associated to R
$\mu,\ \Sigma$	True mean and covariance matrix
$\mathbf{m},\ \mathbf{S}$	Sample mean and covariance matrix

on a bootstrap replicate T_j^*, and can be also denoted as $h_j(\cdot; T_j^*)$. The ensemble prediction is usually obtained by majority voting. For base classifiers that output a real-valued score, simple averaging can also be used [6].

As the ensemble size N increases, its output approaches the asymptotic Bagging prediction, which, when majority voting is used, is defined as:

$$y^* = \arg\max_{y \in \mathcal{Y}} \mathbb{P}[h(\mathbf{x}; T^*) = y] \ . \tag{1}$$

Several authors (e.g., [6, 9, 10]) have shown that ensembles of 10 to 25 "bagged" classifiers attain a performance very similar to the one of larger ensembles, and thus of the asymptotic Bagging. This is a useful, practical guideline to attain a trade-off between computational (both space and time) complexity and classification performance.

Since [6], Bagging is known to be effective especially for unstable classifiers like decision trees and neural networks. In particular, it mainly works by reducing the variance component of the loss function (usually, the misclassification probability) of a given base classifier [11, 12]. Other explanations have also been proposed; for instance, in [13] it has been argued that Bagging equalizes the influence of training instances, and thus reduces the effect of outliers; this is due to the fact that every instance in T has a probability of about 0.632 of

appearing in a bootstrap replicate, and thus each outlier is present on average only in 63% of them.

A thorough analysis of the stabilizing effect of Bagging has been carried out in [9, 14] for the Linear Discriminant and the Nearest Mean classifiers. Their degree of instability was found to depend also on the training set size n: the smaller the training set, the higher the instability, which in turn worsens classification performance. In particular, the above classifiers turned out to very unstable (thus exhibiting a maximum of the generalization error) for critical values of n around the number of features d, and Bagging was capable of improving their performance only under this condition.

In Sect. 4 we shall analyze and model the parameter distribution induced by Bagging on some base classifiers, including the ones considered in [9, 14], as a first step toward the development of novel randomization techniques based on the definition of suitable parameter distributions.

3. A parameter randomization approach for ensemble construction

Consider a given classification algorithm, e.g., a parametric linear classifier with discriminant function $\mathbf{w}^\top \cdot \mathbf{x} + w_0$ implemented as the linear discriminant classifier (LDC), or a non-parametric neural network trained with the back-propagation algorithm. Let ψ denote the parameters that are set by the chosen learning algorithm $\mathcal{L}$, e.g., the coefficients of the LDC (in this case, $\psi = (\mathbf{w}, w_0)$), or the connection weights of a neural network.

Consider now any given randomization technique R (e.g., Bagging), defined by some manipulation procedure of the training set T or of $\mathcal{L}$. The classifiers of an ensemble of size N obtained using R can be denoted as $h_1(\mathbf{x}; \psi(\theta_1))$, $\ldots$, $h_N(\mathbf{x}; \psi(\theta_N))$, where θ_j, $j = 1, \ldots, N$, denote N i.i.d. realizations of the corresponding random variable Θ_R, and the $\psi(\theta_j)$'s denote the parameters of the corresponding classifiers, where we explicitly point out their dependence on the θ_j's. For instance, if Bagging is applied to a linear classifier, $\psi(\theta_j)$ denotes the coefficients $(\mathbf{w}_j, w_{0,j})$ obtained by $\mathcal{L}$ on a bootstrap replicate T_j^* of T. In

the above setting, the parameters $\psi(\theta_j)$ can be seen as i.i.d. realizations of a random variable $\Psi = \Psi(\Theta_{\mathrm{R}})$, whose distribution is implicitly defined by R and $\mathcal{L}$, and depends on T. Accordingly, we write such a distribution as $\mathbb{P}_{\mathrm{R},\mathcal{L}}[\Psi]$. Note that the $\psi(\theta_i)$'s are i.i.d. because they are functions of i.i.d. realizations of Θ_{R}. Note also that $\mathbb{P}_{\mathrm{R},\mathcal{L}}[\Psi]$ is the *joint* distribution of the classifier parameters.

The above formalization suggests an alternative implementation of a *known* technique R, in the case when the distribution $\mathbb{P}_{\mathrm{R},\mathcal{L}}[\Psi]$ is known or can be approximated. The traditional procedure is to run $\mathcal{L}$ for N times on (possibly perturbed versions of) the training set T at hand (e.g., in the case of Bagging, on bootstrap replicates of the original training set), or on a modified version of $\mathcal{L}$ [4]. However, for given R and $\mathcal{L}$ the corresponding $\mathbb{P}_{\mathrm{R},\mathcal{L}}[\Psi]$ could be either analytically derived or empirically modelled *beforehand*, as a function of T, as we shall show in Sect. 4. Accordingly, the alternative implementation we propose is to start from the model $\mathbb{P}_{\mathrm{R},\mathcal{L}}[\Psi]$, computed as a function of the training set T at hand, and then independently draw N i.i.d. realizations $\psi_1, \ldots, \psi_N$ of the classifier parameters by directly sampling from $\mathbb{P}_{\mathrm{R},\mathcal{L}}[\Psi]$, i.e., *without* manipulating T nor running $\mathcal{L}$. In practice, the distribution $\mathbb{P}_{\mathrm{R},\mathcal{L}}[\Psi]$ "bypasses" the manipulation of T and the need of running $\mathcal{L}$ to obtain the ensemble members, as it directly models the effects of these procedures on the classifier parameters. The above approach can be seen as modelling and reproducing the variance reduction effect of traditional randomization techniques on the loss function of base classifiers, by "reverse engineering" their mechanism on the classifier parameters. In other words, it learns the parameter distribution that produces such a variance reduction effect on the loss function, to reproduce it through sampling the classifiers from the learned distribution, instead of learning the classifiers on manipulated versions of the training data. Note that our approach does not necessarily reduce the variance of the classifier parameters (which is not its rationale), exactly as traditional randomization techniques do not necessarily reduce it. A possible advantage of re-implementing existing randomization techniques using the above approach is a lower processing time for ensemble construction.

More interestingly, the above formalization suggests a different approach for developing *novel* randomization techniques, alternative to the manipulation of the training data or the learning algorithm, followed by N runs of $\mathcal{L}$. This approach consists in directly defining a suitable distribution $\mathbb{P}_{\mathcal{L}}[\Psi]$, not derived from any actual procedure R, and then in sampling from it the parameter values of the N ensemble members. This translates the requirement of defining an effective procedure for manipulating the training data or the learning algorithm, into defining a suitable distribution $\mathbb{P}_{\mathcal{L}}[\Psi]$, in terms of improving the ensemble performance by reducing the variance of the loss function. This problem is challenging, for several reasons. One reason is that different distributions should be defined for different base classifiers, that are characterized by different parameters; for instance, the distribution of the coefficients of the nearest mean classifier induced by Bagging is likely to be different from the one of the connection weights of a neural network induced by the same Bagging randomization technique. Another reason is that understanding and modelling how the parameters of a given classifier *jointly* affect the variance of its loss function can be very difficult.

Accordingly, in this work we take a first step toward a practical implementation of our approach, by analyzing and modelling the joint parameter distribution induced by *existing* randomization techniques on some base classifiers. To this aim, we focus on the popular Bagging technique, and consider three base classifiers (summarized in Sect. 4.1) that can be dealt with analytically. The results of such an analysis can provide useful insights on the characteristics that parameter distributions should exhibit to reproduce the variance reduction effect of existing randomization techniques on the loss function of base classifiers.

4. Joint parameter distribution of "bagged" classifiers

For the sake of simplicity, and with no loss of generality, in the following we consider two-class problems. All the results of this section can be easily extended to multi-class problems, as explained in Sect. 4.1.

Let $\mathcal{Y} = \{+1, -1\}$ denote the class labels, and n_1 and n_2 (with $n = n_1 + n_2$) the number of training instances from the two classes, *i.e.*: $T = \{(\mathbf{x}_i, +1)\}_{i=1}^{n_1} \bigcup \{(\mathbf{x}_i, -1)\}_{i=n_1+1}^{n_1+n_2}$. We make the usual assumption of i.i.d. training instances. To make analytical derivations possible, we consider Gaussian class-conditional distributions. The case of non-Gaussian, and even unknown distributions will be discussed in Sect. 4.6.

We denote by $\mathbf{m}_k = (m_{k,1}, \ldots, m_{k,d})^\top$ and by $\mathbf{S}_k$, for $k = 1, 2$, the maximum-likelihood estimates of the mean μ_k and covariance matrix Σ_k of class $+1$ and -1, respectively, *i.e.*:

$$\mathbf{m}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \mathbf{x}_i, \qquad \mathbf{S}_k = \frac{1}{n_k - 1} \sum_{i=1}^{n_k} (\mathbf{x}_i - \mathbf{m}_k)(\mathbf{x}_i - \mathbf{m}_k)^\top, \qquad k = 1, 2. \tag{2}$$

In our derivations four random variables play the main role: the sample mean $\mathbf{m}_k^*$ and the sample covariance matrix $\mathbf{S}_k^*$ of the bootstrap replicates of T. Since the sample mean of n i.i.d. realizations of a multivariate Gaussian $\mathcal{N}(\mu, \Sigma)$ follows the distribution $\mathcal{N}(\mu, \frac{1}{n}\Sigma)$, according to [15] we approximate the distribution of $\mathbf{m}_k^*$, $k = 1, 2$, with independent multivariate Gaussians:

$$\mathcal{N}\left(\mu_k, \frac{1}{n_k}\Sigma_k\right), \ k = 1, 2. \tag{3}$$

The above approximation is accurate even when the data distribution is non-Gaussian, provided that n_k large enough, in virtue of the Central Limit Theorem (CLT). According to a *heuristic* rule, for $n_k \geq 30$ the application of the CLT is well justified. An even smaller value of n_k could be enough, as shown in Sect. 5.1.

Beside $\mathbf{m}_k^*$, the sample covariance matrix $\mathbf{S}_k^*$ is a random variable, too. However, considering both mean and covariance matrix as random variables makes the analytical derivation very difficult (especially in the computation of the inverse of the covariance matrix), because, as a consequence of Eq. (2), we should consider two dependent random variables $\mathbf{m}_k^*$ and $\mathbf{S}_k^*$. Therefore, to further simplify our analysis we shall approximate the sample covariance matrices of *any* bootstrap replicate $T^*(j)$ with the corresponding (constant) covariance matrix

of the data distribution:

$$\mathbf{S}_k^*(j) \simeq \Sigma_k,\ j = 1, \ldots, N\ . \tag{4}$$

We shall evaluate by numerical simulations the accuracy of this approximation in Sect. 5. Accordingly, in our analysis only the $\mathbf{m}_k^*$'s are random variables.

Based on the above assumptions and results, in the following we derive the joint parameter distributions of the base classifiers mentioned above, and summarized in Sect. 4.1, under the assumption of Gaussian class-conditional distributions, and under different forms of the covariance matrices Σ_1 and Σ_2:

- Case 1: identical covariance matrices, proportional to the identity matrix: $\Sigma_1 = \Sigma_2 = \sigma^2\mathbf{I}$.
- Case 2: identical covariance matrices, proportional to the identity matrix but with different diagonal values: $\Sigma_1 = \Sigma_2 = \Sigma = \vec{\sigma}^2\mathbf{I}$, where $\vec{\sigma} = (\sigma_1^2, \ldots, \sigma_d^2)$.
- Case 3: identical covariance matrices having a general form: $\Sigma_1 = \Sigma_2 = \Sigma$.
- Case 4: diagonal covariance matrices, different from each other: $\Sigma_1 = \vec{\sigma}_1^2 I$, $\Sigma_2 = \vec{\sigma}_2^2 I$ such that $\vec{\sigma}_1 \neq \vec{\sigma}_2$, $\vec{\sigma}_1^2 = (\sigma_{1,1}^2, \ldots, \sigma_{1,d}^2)$, $\vec{\sigma}_2^2 = (\sigma_{2,1}^2, \ldots, \sigma_{2,d}^2)$.

We finally consider in Sect. 4.6 the most general case of non-Gaussian or unknown data distribution.

4.1. Base classifiers

Here we summarize the three base classifiers considered in this work, considering their "ideal" discriminant function, *i.e.*, written in terms of the true parameters of the underlying data distribution. All such classifiers provide the optimal discriminant function (either asymptotically with respect to the training set size, or in the ideal case when the data distribution is known) when the class-conditional distributions are Gaussian, under different forms of the covariance matrices of the two classes. This fact allows to analytically derive the joint parameter distributions of the corresponding "bagged" classifiers.

To extend the following results to problems with more than two classes, it suffices to carry out the same derivations for the discriminant function of each class, which has the same form as in the two-class case.

Nearest-mean classifier (NMC). This is a linear classifier whose discriminant function is defined as:

$$g(\mathbf{x}) = \mathbf{w}^\top \cdot (\mathbf{x} - \mathbf{x}_0) \ , \tag{5}$$

where

$$\mathbf{w} = \mu_1 - \mu_2, \quad \mathbf{x}_0 = \frac{1}{2}(\mu_1 + \mu_2) \ . \tag{6}$$

This is the optimal classifier, if the class-conditional distributions are Gaussian and $\Sigma_1 = \Sigma_2 = \sigma^2\mathbf{I}$.

Linear Discriminant Classifier (LDC). The LDC is another well-known linear classifier. Its discriminant function is given by (5), where:

$$\mathbf{w} = \Sigma^{-1}(\mu_1 - \mu_2), \quad \mathbf{x}_0 = \frac{1}{2}(\mu_1 + \mu_2) \ , \tag{7}$$

and $\Sigma = \frac{1}{2}(\Sigma_1 + \Sigma_2)$. The LDC is the optimal classifier if the class-conditional distributions are Gaussian with identical covariance matrices (of any form): $\Sigma_1 = \Sigma_2 = \Sigma$.

Quadratic Discriminant Classifier (QDC). This classifier produces a quadratic discriminant function:

$$g(\mathbf{x}) = \mathbf{x}^\top \mathbf{W}\mathbf{x} + \mathbf{w}^\top \mathbf{x} + w_0 \ , \tag{8}$$

where

$$\begin{aligned} \mathbf{W} &= \frac{1}{2}(\Sigma_2^{-1} - \Sigma_1^{-1}) \ , \\ \mathbf{w} &= (\Sigma_1^{-1}\mu_1 - \Sigma_2^{-1}\mu_2) \ , \\ w_0 &= \frac{1}{2}(\mu_2^\top \Sigma_2^{-1}\mu_2 - \mu_1^\top \Sigma_1^{-1}\mu_1) \ . \end{aligned} \tag{9}$$

The QDC is the optimal classifier when the class-conditional distributions are Gaussian, without constraints on the form of covariance matrices.

4.2. Case 1: identical covariance matrices proportional to the identity matrix

In this section we assume

$$\Sigma_1 = \Sigma_2 = \sigma^2 \mathbf{I} = \Sigma \ , \tag{10}$$

for some value of $\sigma \in \mathbb{R}$, where Σ denotes the common covariance matrix.

4.2.1. Joint parameter distribution

Due to our approximation (4), under assumption (10) the LDC and QDC classifiers coincide with the NMC (see Sect. 4.1). Their discriminant function is given by Eq. (5), where $\mathbf{w} = (w_1, \ldots, w_d)^\top = \mu_1 - \mu_2$, and $\mathbf{x}_0 = (x_{01}, \ldots, x_{0d})^\top = \frac{1}{2}(\mu_1 + \mu_2)$. Such a function is a hyperplane orthogonal to the line joining μ_1 and μ_2, and it is independent on Σ. A single classifier is therefore described by means of $d+1$ independent parameters, *i.e.* $\mathbf{w}$ (a d-dimensional vector) and $w_0 = \mathbf{w}^\top \cdot \mathbf{x}_0$ (a scalar value). Consequently, the parameter vector is $\Psi = (\mathbf{w}, w_0) \in \mathbb{R}^{d+1}$.

According to approximation (4), also the "bagged" QDC and LDC coincide with the "bagged" NMC, which is defined by $\mathbf{w}^* = \mathbf{m}_1^* - \mathbf{m}_2^*$ and $\mathbf{x}_0^* = \frac{1}{2}(\mathbf{m}_1^* + \mathbf{m}_2^*)$, where both quantities are independent on Σ.

Our goal is to derive the joint distribution of the corresponding parameter vector $\Psi^* = (\mathbf{w}^*, w_0^*) \in \mathbb{R}^{d+1}$. However, whereas the distribution of $\mathbf{w}^*$ is Gaussian, the one of $w_0^* = (\mathbf{w}^*)^\top \cdot \mathbf{x}_0^* = \frac{1}{2}((\mathbf{m}_1^*)^\top \cdot \mathbf{m}_1^* - (\mathbf{m}_2^*)^\top \cdot \mathbf{m}_2^*)$ is not, and involves non-central Chi-Squared distributions which are more difficult to treat. For this reason we consider the following, redundant parameter vector:

$$\Psi^* = (\mathbf{w}^*, \mathbf{x}_0^*) = (w_1^*, \ldots, w_d^*, x_{01}^*, \ldots, x_{0d}^*) \in \mathbb{R}^{2d} \ , \tag{11}$$

since also the distribution of $\mathbf{x}_0^*$ is Gaussian. From the above discussion, it follows that the distribution of Ψ^* can be approximated by a Gaussian:

$$\mathcal{N}(\xi, \Sigma_\xi) \ , \tag{12}$$

where the expected value $\xi \in \mathbb{R}^{2d}$ and the $2d \times 2d$ covariance matrix Σ_ξ are:

$$\xi = (w_1, \ldots, w_d, x_{01}, \ldots, x_{0d}), \qquad \Sigma_\xi = \begin{pmatrix} \Sigma_{\mathbf{w}^*} & \Sigma_{\mathbf{w}^*, \mathbf{x}_0^*} \\ \Sigma_{\mathbf{w}^*, \mathbf{x}_0^*} & \Sigma_{\mathbf{x}_0^*} \end{pmatrix} \ , \tag{13}$$

and $\Sigma_{\mathbf{w}^*,\mathbf{x}_0^*}$ is a $d \times d$ matrix whose components are the covariances among all the $\mathbf{w}^*$ and $\mathbf{x}_0^*$ components:

$$\Sigma_{\mathbf{w}^*,\mathbf{x}_0^*} = \{\mathrm{cov}(w_i^*, x_{0j}^*)\}_{i,j=1,\ldots,d} \ . \tag{14}$$

According to assumption (10), denoting $\left(\frac{1}{n_1} + \frac{1}{n_2}\right)$ by $\overline{n}$, we have:

$$\Sigma_{\mathbf{w}^*} = \sigma^2 \overline{n} \mathbf{I}_d, \quad \Sigma_{\mathbf{x}_0^*} = \frac{\sigma^2}{4} \overline{n} \mathbf{I}_d, \quad \Sigma_{\mathbf{w}^*,\mathbf{x}_0^*} = \frac{\sigma^2}{2} \overline{n} \mathbf{I}_d \ . \tag{15}$$

Note that the above results follow from the following properties: *i)* $\mathbf{m}_1^*$ and $\mathbf{m}_2^*$ are independent random variables; *ii)* the components of the random vectors $\mathbf{m}_1^*$ and $\mathbf{m}_2^*$ are independent on each other, since the features are uncorrelated according to assumption (10); *iii)* the Normal distribution belongs to the *Lévy alpha-stable distribution family* [19], *i.e.* linear combination of independent Normal variables is a Normal variable. We also point out that the off-diagonal terms of the sub-matrix $\Sigma_{\mathbf{w}^*,\mathbf{x}_0^*}$ are equal to zero because the random variables x_{0i}^* and w_j^* are independent for $i \neq j$. The situation is different for the diagonal terms $\mathrm{cov}(x_{0i}^*, w_i^*)$ because the random variables $x_{0i}^* = (\mu_{1,i}^* - \mu_{2,i}^*)/2$ and $w_i^* = \mu_{1,i}^* - \mu_{2,i}^*$ are dependent; the only exception is when both classes have the same number of training instances, in which case also the latter terms are null.

Finally, we point out that, although the discriminant function of a single NMC does not depend on Σ, the covariance matrix Σ_ξ of the corresponding parameter distribution, given by Eq. (12), does depend on Σ.

4.2.2. Confidence regions for the distribution parameters

In this section we derive the confidence regions for the parameters of the distribution derived above. Since we deal with a finite number $n = n_1 + n_2$ of instances, the estimation of the "distance" between the true and the estimated statistic is given by the confidence regions involving the Student's t-distribution [20] for the one-dimensional case, and the Hotelling's T-squared distribution [21] (which is a generalization of the former) used for multivariate tests. We consider this kind of distributions in place of the standard confidence intervals because generally we do not know the covariance matrix of the data

and we use its maximum likelihood estimate (see Eq. (2)). In particular, in the one-dimensional case ($d = 1$) the considered classifiers are defined only by the parameter $x_0 = (\mu_1 + \mu_2)/2$. In this case the confidence interval is given by:

$$\frac{m_1 + m_2}{2} \pm t_{n-1}^{(1-\alpha)} \cdot \frac{s}{\sqrt{n}} \ , \tag{16}$$

where $t_{n-1}^{(1-\alpha)}$ is the $(1-\alpha)$-th percentile of the Student's t-distribution for $n-1$ degrees of freedom, m_1 and m_2 are the sample means, and s is the estimated standard deviation of the data.

In the more general case of $d > 1$, the set of hypotheses we have to test is:

$$\begin{cases} H_0 : \ \xi = \xi^{(0)} \\ H_1 : \xi \neq \xi^{(0)} \quad \text{for some } i \end{cases} \tag{17}$$

where $\xi^{(0)} = (\xi_1^{(0)}, \xi_2^{(0)}, \ldots, \xi_{2d}^{(0)})$ is a known vector.

According to the T^2 Hotelling test (which generalizes the Student's t-test discussed above), the hypothesis H_0 is accepted with probability $1 - \alpha$, if:

$$Pr\{T_{\Psi^*}^2 < T_{2d}^2(n-1, \alpha)\} = 1 - \alpha \ , \tag{18}$$

where $T_{\Psi^*}^2 = (\mathbf{m}_{\Psi^*} - \xi^{(0)})^T \mathbf{S}_{\Psi^*}^{-1} (\mathbf{m}_{\Psi^*} - \xi^{(0)})$ (with $\mathbf{m}_{\Psi^*}$ and $\mathbf{S}_{\Psi^*}$ estimated values of ξ and Σ_ξ respectively) and $T_{2d}^2(n-1, \alpha)$ is the α-th percentile of the $2d$-dimensional T^2 Hotelling distribution with $n-1$ degrees of freedom. Eq. (18) represents the *confidence ellipsoid* centered in $\xi^{(0)}$. Obviously, the hypothesis H_0 is refused (with the same probability), if $T_{\Psi^*}^2 > T_{2d}^2(n-1, \alpha)$.

4.3. Case 2: identical, diagonal covariance matrices

We now discuss the case in which the covariance matrices of the classes are identical and proportional to the identity matrix, but with different diagonal values, *i.e.*:

$$\Sigma_1 = \Sigma_2 = \Sigma = \vec{\sigma}^2 \mathbf{I} \ , \tag{19}$$

where $\vec{\sigma} = (\sigma_1^2, \ldots, \sigma_d^2)$.

The LDC is the optimal classifier under the above assumption, and coincides with the QDC. We analyze first these two classifiers. Their decision function

is the one of Eq. (5), where $\mathbf{x}_0$ and $\mathbf{w}$ are given by Eq. (7), with $\mathbf{w}$ depending on the matrix Σ. Accordingly, the resulting discriminant function is again a hyperplane, but it is not orthogonal to the line joining μ_1 and μ_2.

In order to derive the distribution of the parameters $\Psi^* = (\mathbf{w}^*, \mathbf{x}_0^*)$ of the "bagged" classifier, we recall the following well-known property. If $X \sim \mathcal{N}(\mu, \Sigma)$ is a p-dimensional random variable with a multivariate Normal distribution, and A and b are respectively a non-singular matrix and a vector of proper size, then also $Y = AX + b$ has a multivariate Normal distribution, such that $Y \sim \mathcal{N}(A\mu + b, A\Sigma A^T)$. This implies that Ψ^* has a multivariate Normal distribution $\mathcal{N}(\xi, \Sigma_\xi)$, where:

$$\xi = \left[\frac{\mu_{1,1} - \mu_{2,1}}{\sigma_1^2}, \ldots, \frac{\mu_{1,d} - \mu_{2,d}}{\sigma_d^2}, \frac{\mu_{1,1} + \mu_{2,1}}{2}, \ldots, \frac{\mu_{1,d} + \mu_{2,d}}{2}\right] , \tag{20}$$

and Σ_ξ has the same structure as in Eq. (13), where:

$$\Sigma_{\mathbf{w}^*} = \frac{1}{\vec{\sigma}^2}\overline{n}\mathbf{I}_d, \quad \Sigma_{\mathbf{x}_0^*} = \frac{\vec{\sigma}^2}{4}\overline{n}\mathbf{I}_d, \quad \Sigma_{\mathbf{w}^*,\mathbf{x}_0^*} = \frac{1}{2}\overline{n}\mathbf{I}_d \ . \tag{21}$$

Note that also in this case $\Sigma_{\mathbf{w}^*,\mathbf{x}_0^*}$ is the null matrix only if the classes exhibit identical prior probabilities, otherwise it is a diagonal matrix.

Consider now the NMC, which is suboptimal under assumption (19). In this case, the parameter distribution of the "bagged" NMC turns out to be the one derived in Sect. 4.2.1, given by Eqs. (13) and (15), where $\sigma^2 = \left(\frac{1}{d}\sum_{i=1}^{d}\sigma_i^2\right)$.

Consider finally the confidence regions for the distribution parameters (20) and (21). We obtain results similar to the ones discussed in Sect. 4.2.2 where $\mathbf{m}_{\Psi^*}$ and $\mathbf{S}_{\Psi^*}$, in the $T^2_{\Psi^*}$ formula, are the estimated values of ξ and Σ_ξ respectively, given by Eqs. (20) and (21).

4.4. Case 3: identical covariance matrices

Here we discuss the case of identical, generic covariance matrices:

$$\Sigma_1 = \Sigma_2 = \Sigma \ . \tag{22}$$

In this case the features are correlated, which does not allow us to analytically derive all the elements of the covariance matrix Σ_ξ of the parameter distribution

of the "bagged" classifiers. Nevertheless, by performing an appropriate rotation of the feature space, we obtain the diagonal matrix $A^{-1}\Sigma A$ (where A is the eigenvector matrix) whose elements are the eigenvalues $\lambda_1, \ldots, \lambda_d$ of Σ. This leads us to the case already discussed in Sect. 4.3. The distribution of the parameter Ψ^*, for the different classifiers considered, is therefore the one derived in Sect. 4.3, where σ_i^2 is replaced by λ_i, $i = 1, \ldots, d$, and $\mathbf{w}$ and $\mathbf{x}_0$ refer to the values computed in the rotated feature space. Similarly, the same results presented in Sect. 4.2.2 hold for the corresponding confidence regions.

4.5. Case 4: different, diagonal covariance matrices

Here we present the results when the covariance matrices of the classes are different and have a diagonal form:

$$\Sigma_1 = \vec{\sigma}_1^2\mathbf{I}, \quad \Sigma_2 = \vec{\sigma}_2^2\mathbf{I}, \quad \vec{\sigma}_1 \neq \vec{\sigma}_2 \ , \tag{23}$$

where $\vec{\sigma}_k^2 = (\sigma_{k,1}^2, \ldots, \sigma_{k,d}^2)$, $k = 1, 2$.

4.5.1. Joint parameter distribution

The QDC is the optimal classifier under assumption (23). Its decision function is given by Eqs. (8) and (9). Due to assumption (4), the quantity $\mathbf{W} = \frac{1}{2}(\Sigma_2^{-1} - \Sigma_1^{-1})$ is a constant term. Accordingly, the parameter of the "bagged" QDC classifier whose distribution we have to derive is $\Psi^* = (\mathbf{w}^*, w_0^*)$. First, according to Eq. (9) we have:

$$\mathbf{w}^* = \Sigma_1^{-1}\mathbf{m}_1^* - \Sigma_2^{-1}\mathbf{m}_2^* \ . \tag{24}$$

It is easy to see that $\mathbf{w}^*$ approximately follows a multivariate Normal distribution:

$$\mathcal{N}\left(\Sigma_1^{-1}\mu_1 - \Sigma_2^{-1}\mu_2, \frac{1}{n_1}\Sigma_1^{-1} + \frac{1}{n_2}\Sigma_2^{-1}\right) \ . \tag{25}$$

Next, to derive the distribution of w_0^* it is convenient to multiply it by the number of training instances of each class, n_k, and to rewrite the resulting quantity (see Eq. (9)) as $w_{0,1}^* - w_{0,2}^*$, where:

$$w_{0,k}^* = n_k(\mathbf{m}_k^*)^\top\Sigma_k^{-1}(\mathbf{m}_k^*) = n_k\sum_{i=1}^{d}\left(\frac{m_{k,i}^*}{\sigma_{k,i}}\right)^2, \quad k = 1, 2. \tag{26}$$

Consider now that $\{\mathbf{m}^*_{k,i}\}_{i=1,\ldots,d}$, $k = 1, 2$, are independent random variables, and their distribution is approximately Gaussian; this implies:

$$\frac{\sqrt{n_k}\mathbf{m}^*_{k,i}}{\sigma_{k,i}} \;\sim\; \mathcal{N}(\frac{\sqrt{n_k}\mu_{k,i}}{\sigma_{k,i}}, 1), \quad i = 1, \ldots, d, \quad k = 1, 2\ . \tag{27}$$

It follows that the random variables $w^*_{0,k}$ in Eq. (26) approximately follow non-central Chi Squared distributions with d degrees of freedom [23]:

$$w^*_{0,k} \;\sim\; \chi^2(d, \rho_k), \quad k = 1, 2\ , \tag{28}$$

where ρ_k is given by:

$$\rho_k = n_k \mu_k^\top \Sigma_k^{-1} \mu_k = n_k \sum_{i=1}^{d} \left(\frac{\mu_{k,i}}{\sigma_{k,i}}\right)^2, \quad k = 1, 2. \tag{29}$$

We point out that the components of the random variable $\mathbf{w}^* = (w^*_1, \ldots, w^*_d)$ are independent on each other (due to assumption (23)), but they are not independent on $w^*_{0,1}$ and $w^*_{0,2}$. For instance, the covariance between the first component of $\mathbf{w}^*$ and $w^*_{0,1}$, calculated under the assumption of independent features, is $\mathrm{cov}\left(w^*_1, w^*_{0,1}\right) = \frac{2\mu_{1,1}}{\sigma^2_{1,1}}$. In the same way one obtains the covariance between the other components of $\mathbf{w}^*$ and $w^*_{0,1}$ or $w^*_{0,2}$.

Consider finally the "bagged" LDC and NMC, which are suboptimal under assumption (23). Their parameter distribution is the one we obtained in Sects. 4.2–4.4, where $\Sigma = \frac{1}{2}(\Sigma_1 + \Sigma_2)$, depending on the form of Σ.

4.5.2. Confidence regions for the distribution parameters

The confidence region for the multivariate Gaussian random variable $\mathbf{w}^*$ given by Eq. (24) can be computed by means of the T^2 Hotelling test, as previously discussed. The set of hypotheses that has to be tested is indeed:

$$\begin{cases} H_0 :\ \omega_{1i} = \omega_{0i} \quad \forall i = 1, \ldots, d, \\ H_1 : \omega_{1i} \neq \omega_{0i} \quad \text{for some } i, \end{cases} \tag{30}$$

where $\omega_0 = (\omega_{01}, \ldots, \omega_{0d})$ is a known vector. We accept the hypothesis H_0 with probability $1 - \alpha$, if:

$$Pr\{T^2_{\mathbf{w}^*} < T^2_d(n - 1, \alpha)\} = 1 - \alpha\ , \tag{31}$$

where $T^2_{\mathbf{w}^*} = (\mathbf{m}_{\mathbf{w}^*} - \omega_0)^\top \mathbf{S}^{-1}_{\mathbf{w}^*}(\mathbf{m}_{\mathbf{w}^*} - \omega_0)$, whereas $\mathbf{m}_{\mathbf{w}^*}$ and $\mathbf{S}_{\mathbf{w}^*}$ are the estimates of the distribution parameters of $\mathbf{w}^*$ in Eq. (25). The derivation of the confidence region for the non-central Chi-Squared variables $w^*_{0,1}$ and $w^*_{0,2}$ is more difficult [16], and we omit it for the sake of simplicity.

4.6. General case: non-Gaussian or unknown data distribution

Up to now we derived the distribution of the parameters of "bagged" classifiers under the assumption that the data has a multivariate Gaussian distribution with known class-conditional means μ_k and covariance matrices Σ_k. In practice one has no access to the true values of μ_k and Σ_k. Nevertheless, in the case of Gaussian data distribution, all the above results still hold by further approximating the distribution (3) of the random variable $\mathbf{m}^*_k$ by the following Gaussian distribution, in which the sample means $\mathbf{m}_k$ and covariance matrices $\mathbf{S}_k$ (estimated from T, see Eq. (2)) are used in place of μ_k and Σ_k:[1] *i.e.*:

$$\mathcal{N}\left(\mathbf{m}_k, \frac{1}{n_k}\mathbf{S}_k\right), \; k = 1, 2 \; . \tag{32}$$

Moreover, thanks to the CLT the distribution of $\mathbf{m}^*_k$ is approximated with good accuracy by Eq. (32) even if the underlying data distribution is non-Gaussian, provided that the sample size n_k is sufficiently large as already mentioned above (say, $n_k > 30$, although in practice even a small value of n_k is enough, as we will show in Sect. 5.1). In particular, this allows the above results to be exploited also in the practical case of unknown data distribution.

According to our approach and to the above results, we are now in the position of presenting the procedure for constructing an ensemble of NMC, LDC or QDC classifiers, using our approach to simulate Bagging, in a practical setting with unknown data distribution. Given the decision functions of such classifiers in Eqs. (5)–(9), one has to independently sample N realizations of

[1] Although in a boostrap replicate T^* of size $n = n_1 + n_2$ the number of samples from each class can be different from n_k ($k = 1, 2$), Eq. (32) is a good approximation for a sufficiently large value of n_k, thanks to the CLT.

their parameters, *i.e.*, $\Psi^*(j) = (\mathbf{w}^*(j), \mathbf{x}_0^*(j))$ for NMC and LDC, and $\Psi^*(j) = (\mathbf{w}^*(j), w_0^*(j))$ for the QDC, with $j = 1, \ldots, N$.

The corresponding distributions depend on μ_k and Σ_k, that we approximate with $\mathbf{m}_k$ and $\mathbf{S}_k$, $k = 1, 2$. Note that, generally, the sample covariance matrices $\mathbf{S}_1$ and $\mathbf{S}_2$ are different and non-diagonal. For each base classifier, the distributions are the following ones:

NMC : the distribution of $\Psi^* = (\mathbf{w}^*, \mathbf{x}_0^*)$ is approximated by a multivariate Gaussian $\mathcal{N}(\xi, \Sigma_\xi)$ as in Sect. 4.2; the values of its mean ξ and covariance matrix Σ_ξ are given by Eqs. (13) and (15), where the scalar σ^2 is approximated by the mean value of the diagonal elements of $\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2)$.

LDC : according to Sect. 4.4, the distribution of $\Psi^* = (\mathbf{w}^*, \mathbf{x}_0^*)$ is a multivariate Gaussian $\mathcal{N}(\xi, \Sigma_\xi)$, and the values of ξ and Σ_ξ are given respectively by Eqs. (20) and (21). In practice, an alternative and easier way to obtain the parameters of the "bagged" LDC according to our approach is to sample only the values of the random variables $\mathbf{m}_1^*$ and $\mathbf{m}_2^*$, whose distributions are approximated by Eq. (32), and to plug them into Eq. (7), in which the covariance matrices Σ_k are approximated by the corresponding $\mathbf{S}_k$.

QDC : in Sect. 4.5 we derived the distribution of the parameter $\Psi^* = (\mathbf{w}^*, w_0^*)$. As in the previous case, we can obtain the parameters of the "bagged" QDC by sampling only the values of the random variables $\mathbf{m}_1^*$ and $\mathbf{m}_2^*$, whose distributions are approximated by Eq. (32), and then plugging them into Eq. (9).

5. Experiments

To evaluate the proposed randomization approach we carry out experiments on 27 two-class data sets, using as base classifiers NMC, LDC and QDC. Our first aim is to verify whether and to what extent the parameter distribution of classifiers obtained by Bagging can be approximated by the ones we derived in Sect. 4. Secondly, we compare the classification performance of Bagging

Table 2: Characteristics of the data sets used in our experiments. The number of instances in each class are shown between brackets.

Dataset	Instances	Features
1) Correlated Gaussian	400 (200+200)	30
2) Uncorrelated Gaussian	1000 (500+500)	10
3) Acute Inflammations	120 (70+50)	7
4) Banknote authentication	1372 (762+610)	4
5) Blood-transfusion	748 (570+178)	4
6) Climate Model Simulation Crashes	540 (46+494)	19
7) Connectionist Bench (Sonar, Mines vs. Rocks)	208 (97+111)	60
8) Daphnet Freezing of Gait	1.14 (1.03+0.11) $\cdot 10^6$	9
9) Default of credit card clients	30000 (23364+6636)	23
10) Diabetic Retinopathy Debrecen	1151 (764+387)	19
11) EEG Eye State	14980 (8257+6723)	14
12) Bands	365 (135+230)	19
13) Cancer	699 (458+241)	9
14) German	1000 (700+300)	24
15) Pima	768 (500+268)	8
16) Spectfheart	267 (55+212)	44
17) Fertility	100 (88+12)	9
18) Haberman's Survival	306 (225+81)	3
19) Hill_Valley_with_noise_	1212 (606+606)	100
20) Hill_Valley_without_noise_	1212 (600+612)	100
21) ILPD (Indian Liver Patient Dataset)	583 (416+167)	10
22) Ionosphere	351 (225+126)	34
23) LSVT Voice Rehabilitation	126 (42+84)	310
24) MAGIC Gamma Telescope	19020 (6688+12332)	10
25) Mesothelioma disease	324 (228+96)	34
26) Wisconsin Diagnostic Breast Cancer	569 (357+212)	30
27) Recognition of Handwritten Digits: 0 vs 1	360 (178+182)	15
28) Recognition of Handwritten Digits: 1 vs 2	359 (182+177)	15
29) Recognition of Handwritten Digits: 2 vs 3	360 (177+183)	15
30) Recognition of Handwritten Digits: 3 vs 4	364 (183+181)	15
31) Recognition of Handwritten Digits: 4 vs 5	363 (181+182)	15
32) Recognition of Handwritten Digits: 5 vs 6	363 (182+181)	15
33) Recognition of Handwritten Digits: 6 vs 7	360 (181+179)	15
34) Recognition of Handwritten Digits: 7 vs 8	353 (179+174)	15
35) Recognition of Handwritten Digits: 8 vs 9	354 (174+180)	15

with that of classifier ensembles obtained by our approach using the parameter distributions derived for Bagging. Finally, we show an example of the definition of a novel randomization technique according to our approach, i.e., by directly defining a parameter distribution for the base classifier at hand. To this aim, we modify the parameter distribution we derived for Bagging for the above base classifiers.

The main characteristics of the data sets are reported in Table 2. We used two artificial datasets whose distribution is known (Correlated Gaussian [14]

and Uncorrelated Gaussian) and 25 real-world data sets from the UCI repository.[2] Since Handwritten Digits is a 10-class data set, we considered nine two-class problems which consist in discriminating digits i and $i+1$; we also used only the first 15 out of 64 features, to make the two-class problem more difficult. Both artificial data sets exhibit Gaussian class-conditional distributions; in Uncorrelated Gaussian the covariance matrices are identical and proportional to the identity matrix, which is the setting investigated in Sect. 4.2 (Eq. 10), whereas in Correlated Gaussian they are diagonal matrices with the variance of the second feature equal to 40 and the other ones equal to 1. The Correlated Gaussian data set is also rotated for the first two features using a rotation matrix $\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$.

We randomly subdivided each data set, using stratified sampling, into a training set made up of 80% of the instances and a test set containing the remaining instances. To evaluate the effect of the training set size on our approach, we considered four different training sets of increasing size, $n^{(1)} < n^{(2)} < n^{(3)} < n^{(4)}$ where $n^{(4)}$ corresponds to the original training set; we then set the smallest size $n^{(1)}$ equal to the number of features d, which corresponds to the "instability region" where Bagging was found to particularly effective for the considered classifiers in [14] (see Sect. 2); we then set the intermediate sizes $n^{(2)}$ and $n^{(3)}$ as $n^{(1)} + \frac{1}{3}\left(n^{(4)} - n^{(1)}\right)$ and $n^{(1)} + \frac{2}{3}\left(n^{(4)} - n^{(1)}\right)$. We obtained the training sets of size lower than $n^{(4)}$ by a stratified sampling from the original training set. For each base classifier and training set size we built two ensembles of $N = 31$ classifiers: one using Bagging, and one using our approach, as described in Sect. 4.6. We repeated the above procedure for ten times, and averaged the results.

[2] `http://archive.ics.uci.edu/ml`

5.1. Verification of the Gaussianity of the classifier parameters obtained by Bagging

To evaluate the accuracy of the approximation of the distribution of classifier parameters we derived in Sect. 4, we focused on two data sets: the artificial Uncorrelated Gaussian, and the real-world Breast Cancer. The former exhibits Gaussian class-conditional covariance matrices, whereas the distribution of the latter is unknown and its features are correlated. In particular, we used the well-known *Jarque-Bera* gaussianity test [17] to evaluate whether the distributions of the vectors $\mathbf{w}^*$ and $\mathbf{x}_0^*$ obtained by Bagging using the NMC and LDC are well approximated by the derived multivariate Normal distributions. The *Jarque-Bera* test is commonly used for verifying if the data comes from a Normal distribution with unknown parameters, corresponding to the null hypothesis. When the *p-value* [18] is smaller than 0.05, the test rejects the null hypothesis at the default 5% significance level (which means that the distribution is not Gaussian), otherwise the null hypothesis is accepted (*i.e.*, the distribution is considered Gaussian). We performed the test for each $\mathbf{w}^*$ and $\mathbf{x}_0^*$ component separately because, if a random vector follows a Gaussian distribution, its individual components are Gaussian random variables too. We also performed the test for two different training set sizes, one corresponding to the instability region (we chose $n^{(1)} = 10$ for both data sets, since the number of features is 10 for Uncorrelated Gaussian and 9 for Breast Cancer), and one for a larger training set size of $n^{(2)} = 300$ and $n^{(3)} = 300$, respectively. We did not perform the test for the QDC, for which a different approach was used (see Sect. 4.6), which cannot lead to Gaussian distributions.

Results are shown in Tables 3 and 4 for the Uncorrelated Gaussian and Breast Cancer data set, respectively, obtained using NMC as the base classifier, where $\bar{\mathbf{x}}_0^{(s)}$ and $\sigma^2_{\bar{\mathbf{x}}_0^{(s)}}$ denote mean and covariance of each $\mathbf{x}_0$ component obtained by implementing Bagging using our approach (columns 1 and 2), and $\bar{\mathbf{x}}_0^*$ and $\sigma^2_{\bar{\mathbf{x}}_0^*}$ denote the same quantities for the components of $\mathbf{x}_0$ obtained by the original Bagging (columns 3 and 4). We point out that such means and variances were computed over 310 values, given by 31 classifiers $\times$ 10 runs of

the experiments. Similarly, $\bar{\mathbf{w}}^*$ and $\sigma^2_{\bar{\mathbf{w}}^*}$ denote the same quantities for the components of $\mathbf{w}$ obtained by our implementation of Bagging (columns 6 and 7) and by the original version (columns 8 and 9). Columns 5 and 10 show the *p-value* related to each $\mathbf{x}_0$ and $\mathbf{w}$ component, respectively.

According to the test, the random variables $\mathbf{x}_0^*$ and $\mathbf{w}^*$ obtained from Bagging follow Gaussian distributions for both data sets, and for both training set sizes. Indeed, the p-value is always greater than the default value 0.05 except for sporadic cases in the instability region. In particular, the p-value increases in both cases as the training set size increases, which is in agreement with the CLT.

We point out that, for the two data sets above, in the instability region the training set size $n^{(1)} = 10$ is lower than 30, which, according to the heuristic rule mentioned in Sect. 4, is the minimum value which justifies the application of the CLT. This fact provides evidence that, as we mentioned in Sect. 4, the distribution of the classifier parameters obtained by Bagging can be well approximated by a Gaussian also for a training set size lower than 30, even if the original data distribution is not Gaussian.

We finally point out that the average parameter values obtained by our implementation of Bagging are very close to the ones of the original Bagging.

Table 5 shows the results obtained using the LDC as base classifier (the vector $\mathbf{x}_0$ is omitted, as it is identical to the one of NMC). For this classifier it was not possible to compute the parameter $\mathbf{w}^*$ for a training set size equal to $n^{(1)}$, since in the instability region the covariance matrix of the data is ill-conditioned. On the other hand, for a training set size $n = 300$, the random variable $\mathbf{w}^*$ (as well as $\mathbf{x}_0$, see above) is well approximated by a Gaussian distribution for both data sets, as in the case of the NMC.

The above results provide evidence that Bagging can be effectively re-implemented by our approach for the NMC and LDC classifiers using the parameter distributions we derived in Sect. 4, also in the practical cases of data sets with unknown distribution.

Table 3: NMC base classifier, Uncorrelated Gaussian data set. Comparison between mean value and variance of the $\mathbf{x}_0^*$ and $\mathbf{w}^*$ components of the classifier parameter obtained by our approach (columns 1-4) and by Bagging (columns 6-9), for training set sizes $n^{(1)} = 10$ and $n^{(2)} = 300$. The p-value for the Normality test (see text) is shown for all the elements of the above vectors obtained by Bagging: a value higher than 0.05 means that the corresponding random variable has a Normal distribution, at the default 5% significance level.

$n = 10$

Our approach		**Bagging**			**Our approach**		**Bagging**		
$\bar{\mathbf{x}}_0^{(s)}$	$\sigma^2_{\bar{\mathbf{x}}_0^{(s)}}$	$\bar{\mathbf{x}}_0^*$	$\sigma^2_{\bar{\mathbf{x}}_0^*}$	**p-value**	$\bar{\mathbf{w}}^{(s)}$	$\sigma^2_{\bar{\mathbf{w}}^{(s)}}$	$\bar{\mathbf{w}}^*$	$\sigma^2_{\bar{\mathbf{w}}^*}$	**p-value**
0.4403	0.0931	0.4400	0.0990	0.1469	0.5878	0.3842	0.6489	0.3935	0.3455
0.3857	0.0892	0.3373	0.0990	0.0266	-0.7716	0.4037	-0.6758	0.3935	0.4728
0.7065	0.0823	0.6572	0.0929	0.0834	0.4011	0.3615	0.3153	0.3331	0.0733
0.6414	0.1020	0.6813	0.0868	0.0187	0.1434	0.4264	0.0212	0.4065	0.5000
0.4841	0.1027	0.5468	0.1074	0.5000	-0.4147	0.3768	-0.2477	0.4113	0.0015
0.8863	0.1102	0.8477	0.0848	0.5000	0.4958	0.3852	0.3079	0.3650	0.5000
0.4627	0.0948	0.4830	0.0976	0.5000	-0.3146	0.3392	-0.2643	0.4209	0.2741
0.5203	0.1109	0.4764	0.1172	0.5000	0.4820	0.3945	0.4082	0.4561	0.2204
0.7066	0.1143	0.7031	0.0962	0.0867	-0.2706	0.3986	-0.4562	0.3594	0.5000
0.7450	0.0765	0.7418	0.0803	0.2839	-0.0418	0.3623	-0.0799	0.3311	0.5000

$n = 300$

Our approach		**Bagging**			**Our approach**		**Bagging**		
$\bar{\mathbf{x}}_0^{(s)}$	$\sigma^2_{\bar{\mathbf{x}}_0^{(s)}}$	$\bar{\mathbf{x}}_0^*$	$\sigma^2_{\bar{\mathbf{x}}_0^*}$	**p-value**	$\bar{\mathbf{w}}^{(s)}$	$\sigma^2_{\bar{\mathbf{w}}^{(s)}}$	$\bar{\mathbf{w}}^*$	$\sigma^2_{\bar{\mathbf{w}}^*}$	**p-value**
0.4644	0.0069	0.4702	0.0056	0.5000	0.5435	0.0222	0.5315	0.0174	0.5000
0.3980	0.0063	0.4011	0.0056	0.5000	-0.6715	0.0252	-0.6746	0.0209	0.1695
0.6647	0.0058	0.6597	0.0052	0.3223	0.3376	0.0201	0.3591	0.0209	0.5000
0.6613	0.0051	0.6582	0.0062	0.5000	0.0868	0.0231	0.0724	0.0207	0.5000
0.5131	0.0053	0.5037	0.0070	0.3422	-0.3748	0.0244	-0.3831	0.0214	0.5000
0.8822	0.0053	0.8794	0.0057	0.1412	0.4639	0.0221	0.4495	0.0241	0.5000
0.4401	0.0053	0.4305	0.0047	0.1244	-0.3001	0.0235	-0.3033	0.0244	0.3213
0.4592	0.0070	0.4649	0.0057	0.5000	0.4066	0.0202	0.4100	0.0235	0.2833
0.7219	0.0054	0.7212	0.0057	0.5000	-0.3173	0.0276	-0.3455	0.0243	0.5000
0.7084	0.0054	0.7079	0.0048	0.4572	-0.0302	0.0206	0.0038	0.0215	0.4309

5.2. Performance comparison

In this section we compare the classification performance of the original Bagging with the one of its implementation based on our approach (according to Sect. 4.6).

For each base classifier and training set size we report in Tables 6–8, respectively for the NMC, LDC and QDC, the average accuracy of the original Bagging and the difference ΔA between the accuracy of our approach and the one of the original Bagging, over the ten runs of our experiments. We did not report the variance, since it was always very small: it ranged from 0 to 0.06 over all classifiers and data sets, with an average value of about $4 \cdot 10^{-3}$. In some cases (denoted by "–" in the tables) it was not possible to compute the bagged

Table 4: NMC base classifier, Breast Cancer data set. See caption of Table 3 for the details.

$n = 10$

Our approach		**Bagging**			**Our approach**		**Bagging**		
$\bar{\mathbf{x}}_0^{(s)}$	$\sigma^2_{\bar{\mathbf{x}}_0^{(s)}}$	$\bar{\mathbf{x}}_0^*$	$\sigma^2_{\bar{\mathbf{x}}_0^*}$	**p-value**	$\bar{\mathbf{w}}^{(s)}$	$\sigma^2_{\bar{\mathbf{w}}^{(s)}}$	$\bar{\mathbf{w}}^*$	$\sigma^2_{\bar{\mathbf{w}}^*}$	**p-value**
0.5173	0.0046	0.5028	0.0036	0.0769	-0.4322	0.0166	-0.4295	0.0143	0.1455
0.3976	0.0034	0.3973	0.0036	0.0607	-0.5166	0.0149	-0.5266	0.0143	0.0014
0.3925	0.0035	0.4013	0.0030	0.5000	-0.5028	0.0135	-0.5095	0.0162	0.0874
0.3381	0.0042	0.3445	0.0050	0.3038	-0.3950	0.0169	-0.4313	0.0222	0.5000
0.3659	0.0035	0.3638	0.0033	0.0217	-0.2958	0.0125	-0.3086	0.0131	0.0124
0.4575	0.0051	0.4572	0.0054	0.0814	-0.6217	0.0206	-0.6151	0.0208	0.0011
0.3992	0.0041	0.4057	0.0031	0.2081	-0.3760	0.0155	-0.4008	0.0127	0.5000
0.3297	0.0048	0.3674	0.0055	0.0662	-0.4672	0.0229	-0.4702	0.0232	0.5000
0.1789	0.0035	0.1882	0.0034	0.0010	-0.1596	0.0140	-0.1542	0.0134	0.0121

$n = 300$

Our approach		**Bagging**			**Our approach**		**Bagging**		
$\bar{\mathbf{x}}_0^{(s)}$	$\sigma^2_{\bar{\mathbf{x}}_0^{(s)}}(10^{-3})$	$\bar{\mathbf{x}}_0^*$	$\sigma^2_{\bar{\mathbf{x}}_0^*}(10^{-3})$	**p-value**	$\bar{\mathbf{w}}^{(s)}$	$\sigma^2_{\bar{\mathbf{w}}^{(s)}}(10^{-3})$	$\bar{\mathbf{w}}^*$	$\sigma^2_{\bar{\mathbf{w}}^*}(10^{-3})$	**p-value**
0.5095	0.1989	0.5061	0.2030	0.5000	-0.4291	0.7989	-0.4219	0.7421	0.5000
0.3940	0.2077	0.3944	0.2030	0.3722	-0.5236	0.8155	-0.5226	0.7421	0.4137
0.4003	0.2051	0.3999	0.1650	0.4124	-0.5106	0.7703	-0.5091	0.7163	0.0411
0.3469	0.2295	0.3467	0.2569	0.5000	-0.4188	0.8968	-0.4147	0.9767	0.1936
0.3698	0.1949	0.3708	0.1782	0.5000	-0.3192	0.7778	-0.3165	0.6256	0.2785
0.4513	0.2242	0.4508	0.2821	0.4785	-0.6207	0.8866	-0.6180	0.9973	0.4089
0.4043	0.1995	0.4035	0.1798	0.1910	-0.3879	0.7424	-0.3869	0.6274	0.4889
0.3572	0.2034	0.3563	0.3057	0.4836	-0.4524	0.8493	-0.4563	1.1133	0.5000
0.1808	0.1935	0.1810	0.1453	0.5000	-0.1527	0.7895	-0.1524	0.6306	0.5000

classifier, due to the very small training set size.

Table 6 shows that our approach provided a classification performance very close to Bagging when the NMC was used as the base classifier. ΔA is always less than 0.05, and in most cases it equals zero or is very small. We further checked whether these differences are statistically significant at level $\alpha = 0.05$, that is, if it is unlikely to observe them by chance. To this aim we run the paired Wilcoxon test, as suggested in [22] for comparisons over multipler data sets. The test gave a p-value $p \gg \alpha$ (p was between 0.21 and 0.58, depending on the training set size). Accordingly, the difference in performance between Bagging and its implementation using our approach is not statistically significant.

For the LDC, Table 7 shows only a partial agreement between the original Bagging and our approach. For 26 out of 35 datasets ΔA is lower than 0.05. In general, the original Bagging seems to perform better. The Wilcoxon test gave a p-value $p \ll \alpha$ (p was between 10^{-5} and $4 \cdot 10^{-3}$, depending on the training set size): accordingly, the null hypothesis that the original Bagging and our

Table 5: LDC base classifier, Uncorrelated Gaussian and Breast Cancer datasets. Comparison between the mean value and variance of the $\mathbf{w}^*$ component of the classifier parameter obtained by our approach and by Bagging (columns 1–2 and 3–4, respectively), for a training set size $n = 300$, and p-value for the Normality test (see caption of Table 3 for the details).

Uncorrelated Gaussian ($n = 300$)				
Our approach		**Bagging**		
$\bar{\mathbf{w}}^{(s)}$	$\sigma^2_{\bar{\mathbf{w}}^{(s)}}$	$\bar{\mathbf{w}}^*$	$\sigma^2_{\bar{\mathbf{w}}^*}$	**p-value**
0.5672	0.0207	0.5795	0.0276	0.5000
-0.6781	0.0300	-0.6618	0.0276	0.5000
0.4049	0.0391	0.3656	0.0360	0.5000
0.0070	0.0262	0.0092	0.0340	0.5000
-0.4258	0.0249	-0.3602	0.0266	0.5000
0.4622	0.0188	0.4743	0.0400	0.5000
-0.4187	0.0330	-0.4022	0.0363	0.5000
0.4642	0.0186	0.4688	0.0346	0.5000
-0.3295	0.0374	-0.3102	0.0467	0.0665
-0.0178	0.0211	-0.0264	0.0273	0.5000
Breast Cancer ($n = 300$)				
Our approach		**Bagging**		
$\bar{\mathbf{w}}^{(s)}$	$\sigma^2_{\bar{\mathbf{w}}^{(s)}}$	$\bar{\mathbf{w}}^*$	$\sigma^2_{\bar{\mathbf{w}}^*}$	**p-value**
-9.9882	10.8395	-8.7553	9.0166	0.5000
-5.7018	10.8395	-5.4385	9.0166	0.0208
-4.2521	9.5173	-3.4918	6.9907	0.5000
-1.7440	4.5151	-1.4015	4.9878	0.0277
-2.2664	7.1830	-2.2900	8.9879	0.5000
-11.3144	6.1746	-11.6187	7.0698	0.0164
-3.6291	5.9336	-4.4045	7.0778	0.0910
-4.3743	3.8106	-3.8320	4.8822	0.5000
1.2484	7.9784	0.3680	8.1049	0.5000

approach are equivalent can be rejected.

The results for the QDC, shown in Table 8, are similar. The Wilcoxon test gave a p-value $p \gg \alpha$ (p was between 0.098 and 0.59, depending on the training set size), which means that also in this case the difference in performance is not statistically significant.

5.3. Defining new randomization techniques: an example

The above experiments provided evidence that, at least for the considered base classifiers, Bagging can also be implemented using to our approach. Based on these results, we give now an example of how a *novel* randomization technique can be defined according to our approach, i.e., by directly defining the parameter distribution of a given base classifier, exploiting knowledge of parameter distributions induced by existing techniques. In this example we modify the

distribution induced by Bagging for the NMC, derived in Sect. 4. In particular, we modify the covariance matrix Σ_ξ of Eq. (15) into a new covariance matrix Σ'_ξ such that:

$$\Sigma_{\mathbf{x}_0^*} = \frac{\Sigma}{2}, \quad \Sigma_{\mathbf{w}^*} = 2\Sigma, \quad \Sigma_{\mathbf{x}_0^*,\mathbf{w}^*} = \mathbf{0}_{d\times d} \ . \tag{33}$$

In this simple example related to a linear classifier we attempt to increase diversity by increasing the variance of the parameter distribution, to shift the accuracy-diversity trade-off in favour of a higher diversity. The results are reported in Table 9. For comparison we also report the performance of the original Bagging.

It can be seen that our randomization technique attained a reasonable performance, in the sense that it is close to the one of an existing, traditional randomization technique like Bagging. In particular, in our previous experiments the covariance matrix of Eq. (15) (obtained by modelling the original Bagging) lead to $|\Delta\mathrm{A}| \leq 0.05$ for all datasets and for all training set sizes. Using the covariance matrix of Eq. (33), the performance increases for some datasets and decreases for others, which can be interpreted as the effect of an increased diversity between the individual classifiers.

These preliminary results provide some evidence of the viability of our alternative approach to the implementation of randomization techniques, which motivates further investigation on the definition of suitable distributions of classifier parameters.

6. Discussion and conclusions

We proposed a novel approach for defining and implementing randomization techniques for classifier ensemble construction. It is based on modelling the joint probability distribution of the parameters of a given base classifier, and then obtaining the ensemble members by directly sampling from such a distribution their parameter values, instead of manipulating the training data and running the learning algorithm for each of them. This approach can also be exploited

as an alternative implementation of existing randomization techniques, if the parameter distribution they induce on a base classifier can be derived or approximated; in this case, a practical advantage is the reduction of processing cost in the ensemble construction stage, as no manipulation of training data is required, nor running the learning algorithm. We point out that our approach can be used also for one-class classifiers, for which the use of ensembles constructed by Bagging has already been proposed by several authors.

To define new randomization techniques based on our approach, a crucial issue is to define a joint parameter distribution capable of providing an advantageous trade-off between accuracy and diversity of the resulting classifiers, in terms of reducing the variance of their loss function and thus improving ensemble performance, analogously to existing techniques. A useful way to obtain insights on the characteristics of such a distribution is to analyze the one induced by existing randomization techniques. In this paper we took a first step in this direction, focusing on Bagging and on three well-known base classifiers that can be dealt with analytically. We then provided a preliminary example of the definition of a new technique, by modifying the joint parameter distribution induced by Bagging on the same classifiers.

We finally summarize the assumptions and the main limitations of our approach. To model the joint parameter distribution induced by a given randomization technique on a given base classifier, some specific assumptions and approximations may be necessary to allow or simplify analytical derivations. In this work we made two assumptions specific to Bagging and to the considered nearest mean, linear and quadratic discriminant classifiers: the class-conditional distributions are Gaussian, and the class-covariance matrices of bootstrap replicates are identical to the ones of the original training set. The first assumption turned out to be not a limitation: we empirically found that the resulting approximation of the parameter distribution can be accurate also for very small training set sizes and for non-Gaussian class-conditional distributions. We found instead that the second assumption can be not accurate enough in some cases; it is, however, possible to compute also the distribution of the sample covari-

ance matrix of bootstrap replicates, at the expense of more complex derivations. Nevertheless, we point out that the main goal of such an analytical study is not to accurately approximate the joint parameter distributions of randomization techniques defined as a procedure for manipulating training data or the learning algorithm (which can be useful to provide an alternative implementation of such techniques), but is rather to obtain insights on the parameter distributions they induce on base classifiers, and thus guidelines for the definition of novel techniques based on our approach, which avoids explicit manipulation of training data and of learning algorithms.

Another characteristic of our approach is that the joint parameter distribution of a base classifier depends on the training data at hand. This is reasonable, as it reflects the fact that in traditional randomization techniques the individual classifiers are learned on manipulated versions of the training data. For instance, in the model derived in this paper, the joint distributions of the considered classifiers depend on the sample means and covariance matrices of training data. However, if the training set at hand is small, or if it contains outliers, statistics estimated from it could be not accurate. This can be a problem if our approach is used to re-implement a traditional randomization technique, since its approximation may in turn be not accurate. To address this issue, our approach could be implemented using robust statistics [5], e.g., computing the median or the trimmed mean instead of the simple mean of training data in feature space.

The results of this paper are limited to Bagging and to the base classifiers mentioned above. This choice was made to allow analytical derivations of the joint parameter distribution. For a more thorough understanding it is however desirable to extend our analysis to other randomization techniques and other base classifiers. The main difficulty under this viewpoint is that developing analytical models of the joint parameter distribution can be challenging for some kinds of classifiers. For instance, this may be the case of a non-parametric classifier like neural networks, where the number of parameters (connection weights) can be very high, and at the same time they are not related to statistics of training data (e.g., sample means and covariance matrices), contrary to the

parametric base classifiers considered in this work. Another challenging example are decision trees, whose decision function is a *structured* one; for the same reason, the Random Forest ensemble technique may be difficult to analyze, as it uses decision trees as base classifiers. We believe that extending our analysis and addressing the above issue are the main directions for future work.

Acknowledgement

This work has been partly supported by the project "Computational quantum structures at the service of pattern recognition: modeling uncertainty" [CRP-59872] funded by Regione Autonoma della Sardegna, L.R. 7/2007, Bando 2012.

References

[1] L.I. Kuncheva, Combining Pattern Classifiers, Second Edition, John Wiley & Sons, Inc, Hoboken, NJ, USA, 2014.

[2] Z.-H. Zhou, Ensemble Methods: Foundations and Algorithms, Chapman & Hall/CRC, 2012.

[3] T.K. Ho, The random subspace method for constructing decision forests, IEEE Trans. Patt. Anal. Mach. Intell. 20 (1998) 832–844.

[4] L. Breiman, Random Forests, Machine Learning. 45 (2001) 5–32.

[5] P.J. Huber, Robust Statistics, John Wiley & Sons, 1981.

[6] L. Breiman, Bagging Predictors, Machine Learning. 24 (1996) 123–140.

[7] J.J. Rodriguez, L.I. Kuncheva, C.J. Alonso, Rotation Forest: A New Classifier Ensemble Method, IEEE Trans. Patt. Anal. Mach. Intell. 28 (2006) 1619–1630.

[8] Y. Freund, R.E. Schapire, Experiments with a New Boosting Algorithm, In Int. Conf. on Machine Learning (1996), pp. 148–156.

[9] M. Skurichina, R.P.W. Duin, Bagging for linear classifiers, Pattern Recognition. 31 (1998) 909–930.

[10] G. Fumera, F. Roli, A. Serrau, A Theoretical Analysis of Bagging as a Linear Combination of Classifiers, IEEE Trans. Pattern Anal. Machine Intell. 30, 1293–1299.

[11] R. Tibshirani, Bias, Variance and Prediction Error for Classification Rules, Tech. Rep. 9602, University of Toronto (1996).

[12] D.H. Wolpert, W.G. Macready, An Efficient Method To Estimate Bagging's Generalization Error, Machine Learning 35 (1999) 41–55.

[13] Y. Grandvalet, Bagging Equalizes Influence, Machine Learning 55 (2004) 251–270.

[14] M. Skurichina, R.P.W. Duin, Bagging, Boosting and the Random Subspace Method for Linear Classifiers, Patt. Anal. Appl. 5 (2002) 121–135.

[15] B. Efron, R.J. Tibshirani, An Introduction to the Bootstrap, Chapman & Hall/CRC, 1993.

[16] J.T. Kent, T.J. Hainsworth, Confidence intervals for the noncentral chi-squared distribution, J. of Stat. Planning and Inference. 46 (1995) 147–159.

[17] C.M. Jarque, A.K. Bera, A Test for Normality of Observations and Regression Residuals, Int. Stat. Rev. 55 (1987) 163.

[18] R.A. Fisher, Statistical Methods for Research Workers, (1925), Oliver & Boyd, Edinburgh.

[19] B. Mandelbrot, The Pareto-Levy Law and the Distribution of Income, International Economic Review 1 (1960) 79.

[20] R.V. Hogg and A. T. Craig: Introduction to Mathematical Statistics. The Macmillan Company, New York (1978).

[21] H. Hotelling, The Generalization of Student's Ratio, in: Breakthroughs in Statistics, Springer, 1992, pp. 54–65.

[22] J. Demsar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine learning research 7 (2006) 1–30.

[23] S. Kotz, N. Balakrishnan, N.L. Johnson, Continuous Multivariate Distributions, John Wiley & Sons, 2005.

Table 6: NMC base classifier: accuracy of the original Bagging, and difference (Δ) between the accuracy of its implementation using our approach and that of the original Bagging, for different training set sizes.

	Bagging				Δ			
Dataset	$n^{(1)}$	$n^{(2)}$	$n^{(3)}$	$n^{(4)}$	$n^{(1)}$	$n^{(2)}$	$n^{(3)}$	$n^{(4)}$
1)	0.55	0.57	0.60	0.60	0.01	0.00	0.00	0.00
2)	0.61	0.69	0.70	0.70	0.00	0.00	0.00	0.00
3)	0.93	0.98	1.00	1.00	-0.05	0.00	0.00	0.00
4)	0.78	0.86	0.86	0.86	0.01	0.00	0.00	0.00
5)	-	0.66	0.68	0.69	-	0.01	0.00	0.00
6)	-	1.00	1.00	1.00	-	0.00	0.00	0.00
7)	0.69	0.68	0.69	0.70	-0.01	0.00	0.01	0.00
8)	-	0.55	0.55	0.55	-	0.00	0.00	0.00
9)	0.67	0.65	0.66	0.66	0.02	0.00	0.00	0.00
10)	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00
11)	0.52	0.56	0.56	0.56	0.01	0.00	0.00	0.00
12)	0.59	0.61	0.62	0.63	-0.02	-0.01	0.01	-0.01
13)	0.94	0.95	0.96	0.96	0.00	0.00	0.00	0.00
14)	0.69	0.70	0.69	0.69	-0.01	0.00	0.00	0.00
15)	0.71	0.74	0.73	0.74	0.00	0.00	0.00	-0.01
16)	0.69	0.68	0.69	0.69	0.03	0.03	0.00	0.01
17)	-	0.75	0.73	0.71	-	0.00	0.04	0.00
18)	-	0.64	0.67	0.68	-	-0.03	-0.01	-0.02
19)	0.49	0.49	0.50	0.51	0.01	0.00	0.00	0.00
20)	0.51	0.51	0.52	0.52	0.01	0.00	0.01	0.01
21)	0.61	0.63	0.64	0.65	0.04	0.00	0.00	0.00
22)	0.78	0.78	0.75	0.75	-0.03	-0.01	-0.01	0.00
23)	0.78	0.79	0.76	0.78	0.02	0.01	0.04	0.03
24)	0.69	0.77	0.77	0.77	0.00	0.00	0.00	0.00
25)	0.95	1.00	1.00	1.00	-0.01	0.00	0.00	0.00
26)	0.93	0.94	0.94	0.93	0.00	0.00	0.00	0.00
27)	0.79	0.85	0.85	0.85	0.02	0.00	0.00	0.00
28)	0.82	0.84	0.85	0.85	0.00	0.00	0.00	0.00
29)	0.78	0.80	0.80	0.80	0.00	0.00	0.00	0.00
30)	0.96	0.97	0.97	0.97	0.00	0.00	0.00	0.00
31)	0.97	0.97	0.97	0.97	0.00	0.00	0.00	0.00
32)	0.97	0.97	0.97	0.97	0.00	0.00	0.00	0.00
33)	0.93	0.94	0.94	0.95	0.00	0.00	0.00	0.00
34)	0.74	0.76	0.76	0.75	0.00	-0.01	0.01	0.01
35)	0.54	0.60	0.63	0.62	0.00	0.01	-0.01	0.01

Table 7: LDC base classifier: accuracy of the original Bagging, and difference (Δ) between the accuracy of its implementation using our approach and that of the original Bagging, for different training set sizes.

	Bagging				Δ			
Dataset	$n^{(1)}$	$n^{(2)}$	$n^{(3)}$	$n^{(4)}$	$n^{(1)}$	$n^{(2)}$	$n^{(3)}$	$n^{(4)}$
1)	0.54	0.90	0.93	0.93	0.09	0.00	0.00	0.00
2)	0.64	0.74	0.73	0.73	-0.13	0.00	0.00	0.00
3)	0.90	0.99	1.00	1.00	0.00	0.00	0.00	0.00
4)	0.77	0.98	0.98	0.98	0.13	0.00	0.00	0.00
5)	-	0.78	0.77	0.78	-	-0.16	-0.14	-0.14
6)	0.90	0.93	0.94	0.94	-0.29	-0.08	-0.10	-0.12
7)	0.74	0.63	0.72	0.74	-0.25	0.03	-0.02	0.02
8)	-	0.90	0.90	0.90	-	-0.38	-0.38	-0.37
9)	0.71	0.81	0.81	0.81	-0.16	-0.11	-0.12	-0.12
10)	0.67	0.73	0.73	0.73	-0.10	0.01	0.01	0.02
11)	0.56	0.64	0.64	0.64	-0.06	-0.02	-0.01	-0.01
12)	0.57	0.61	0.65	0.66	-0.04	-0.02	-0.05	-0.06
13)	0.90	0.96	0.96	0.96	-0.04	0.01	0.01	0.00
14)	0.68	0.76	0.77	0.77	-0.11	-0.05	-0.07	-0.06
15)	0.67	0.76	0.76	0.77	-0.16	-0.01	-0.01	-0.01
16)	0.70	0.66	0.74	0.76	-0.16	-0.03	-0.08	-0.05
17)	0.89	0.82	0.86	0.93	-0.37	-0.14	-0.17	-0.25
18)	-	0.73	0.75	0.74	-	-0.05	-0.02	-0.01
19)	0.63	0.64	0.64	0.66	-0.09	-0.05	-0.03	-0.01
20)	0.68	0.68	0.68	0.69	-0.07	-0.03	0.00	-0.03
21)	0.66	0.71	0.71	0.71	-0.13	-0.09	-0.07	-0.07
22)	0.75	0.80	0.83	0.84	-0.02	-0.01	-0.01	-0.01
23)	0.81	0.81	0.80	0.81	-0.04	-0.05	-0.05	-0.04
24)	0.67	0.78	0.78	0.78	-0.08	0.01	0.01	0.01
25)	0.66	0.65	0.68	0.70	-0.09	-0.08	-0.07	-0.07
26)	0.92	0.94	0.95	0.95	-0.21	0.01	0.01	0.02
27)	0.83	0.92	0.93	0.94	-0.05	-0.01	0.00	0.00
28)	0.84	0.91	0.92	0.92	-0.09	0.00	0.00	0.00
29)	0.66	0.75	0.76	0.77	-0.06	0.00	0.01	-0.01
30)	0.93	0.96	0.96	0.96	-0.09	0.00	0.00	0.00
31)	0.95	0.98	0.98	0.98	-0.10	0.00	0.00	0.00
32)	0.94	0.96	0.97	0.97	-0.09	0.00	0.00	0.00
33)	0.95	0.98	0.98	0.99	-0.03	0.00	0.00	0.00
34)	0.72	0.83	0.86	0.86	-0.07	0.00	-0.01	0.00
35)	0.53	0.63	0.67	0.68	0.00	0.00	0.00	0.01

Table 8: QDC base classifier: accuracy of the original Bagging, and difference (Δ) between the accuracy of its implementation using our approach and that of the original Bagging, for different training set sizes.

	Bagging				Δ			
Dataset	$n^{(1)}$	$n^{(2)}$	$n^{(3)}$	$n^{(4)}$	$n^{(1)}$	$n^{(2)}$	$n^{(3)}$	$n^{(4)}$
1)	0.52	0.71	0.82	0.87	0.00	0.04	0.00	0.00
2)	0.52	0.67	0.69	0.70	0.04	0.01	0.00	0.00
3)	0.78	1.00	1.00	1.00	0.05	-0.04	-0.03	-0.02
4)	-	0.99	0.99	0.99	-	0.00	0.00	0.00
5)	-	0.75	0.76	0.76	-	-0.08	-0.09	-0.08
6)	-	0.92	0.08	0.08	-	-0.17	0.83	0.84
7)	0.68	0.74	0.66	0.59	-0.02	-0.12	-0.02	0.13
8)	-	0.81	0.82	0.82	-	-0.06	-0.06	-0.07
9)	0.24	0.50	0.46	0.47	-0.02	0.15	0.18	0.17
10)	0.47	0.67	0.67	0.67	-0.01	0.03	0.04	0.03
11)	0.51	0.66	0.59	0.56	-0.01	0.11	0.16	0.17
12)	0.46	0.61	0.63	0.60	-0.09	0.04	0.05	0.06
13)	0.70	0.95	0.95	0.95	-0.15	-0.06	-0.04	-0.04
14)	0.31	0.69	0.72	0.73	0.00	0.00	-0.03	-0.05
15)	-	0.72	0.73	0.72	-	-0.02	-0.04	-0.05
16)	0.80	0.80	0.80	0.80	-0.61	-0.34	0.00	0.00
17)	-	0.94	0.94	0.94	-	-0.70	-0.11	-0.01
18)	-	0.74	0.75	0.75	-	-0.13	-0.10	-0.08
19)	0.56	0.53	0.53	0.54	-0.02	-0.02	0.02	0.00
20)	0.54	0.52	0.53	0.55	-0.02	0.01	0.02	0.02
21)	-	0.63	0.60	0.59	-	0.07	0.09	0.11
22)	0.68	0.70	0.87	0.88	-0.31	0.03	-0.22	-0.24
23)	0.69	0.75	0.71	0.76	-0.35	-0.42	-0.37	-0.42
24)	0.41	0.79	0.79	0.79	0.02	-0.23	-0.23	-0.23
25)	0.71	0.99	1.00	1.00	-0.41	-0.29	-0.28	-0.27
26)	0.50	0.95	0.95	0.96	-0.06	-0.05	-0.08	-0.09
27)	0.61	0.46	0.46	0.46	0.09	0.47	0.47	0.48
28)	0.65	0.49	0.49	0.49	0.09	0.42	0.42	0.41
29)	0.57	0.53	0.53	0.53	0.08	0.22	0.26	0.27
30)	0.76	0.50	0.50	0.50	0.04	0.48	0.48	0.48
31)	0.77	0.53	0.53	0.53	0.04	0.46	0.46	0.46
32)	0.64	0.50	0.50	0.50	0.05	0.46	0.47	0.47
33)	0.72	0.51	0.51	0.51	0.00	0.44	0.45	0.45
34)	0.55	0.50	0.50	0.50	0.09	0.36	0.36	0.36
35)	0.54	0.49	0.49	0.49	0.00	0.14	0.17	0.18

Table 9: Comparison between Bagging and our synthetic randomization technique using NMC as base classifier and an "alternative" covariance matrix (the one given by Eq. (33)): accuracy of Bagging and difference between Synthetic and Bagging accuracy (Δ) for different training set sizes.

	Bagging				Δ			
Dataset	$n^{(1)}$	$n^{(2)}$	$n^{(3)}$	$n^{(4)}$	$n^{(1)}$	$n^{(2)}$	$n^{(3)}$	$n^{(4)}$
1)	0.57	0.61	0.62	0.63	-0.05	-0.06	-0.09	-0.10
2)	0.60	0.73	0.74	0.74	0.02	-0.09	-0.11	-0.11
3)	0.68	0.70	0.70	0.70	0.03	0.01	0.01	0.01
4)	0.66	0.74	0.74	0.74	0.00	-0.07	-0.08	-0.07
5)	0.95	0.95	0.95	0.95	-0.01	-0.02	-0.01	-0.01
6)	-	0.66	0.67	0.69	-	0.10	0.09	0.08
7)	0.52	0.63	0.64	0.65	0.16	0.09	0.08	0.07
8)	0.55	0.59	0.62	0.63	0.10	0.06	0.03	0.02
9)	0.78	0.79	0.78	0.78	-0.13	-0.14	-0.14	-0.13
10)	0.72	0.71	0.70	0.71	0.04	0.05	0.06	0.05
11)	0.80	0.98	1.00	1.00	-0.05	-0.18	-0.14	-0.20
12)	0.73	0.85	0.85	0.85	0.01	-0.05	-0.09	-0.06
13)	-	1.00	1.00	1.00	-	-0.09	-0.09	-0.09
14)	0.63	0.65	0.68	0.67	-0.08	-0.09	-0.11	-0.14
15)	-	0.55	0.55	0.55	-	0.35	0.35	0.35
16)	0.61	0.66	0.66	0.66	0.17	0.12	0.12	0.12
17)	1.00	1.00	1.00	1.00	0.00	-0.01	-0.01	-0.01
18)	0.54	0.57	0.56	0.57	0.02	-0.02	-0.01	-0.02
19)	0.92	0.93	0.93	0.93	-0.03	-0.03	-0.04	-0.04
20)	-	0.79	0.78	0.74	-	0.10	0.11	0.15
21)	-	0.61	0.64	0.61	-	0.10	0.08	0.11
22)	0.49	0.51	0.50	0.50	0.00	-0.01	0.01	0.00
23)	0.51	0.52	0.52	0.52	0.00	0.00	0.00	-0.02
24)	0.75	0.74	0.77	0.75	-0.12	-0.11	-0.13	-0.11
25)	0.69	0.76	0.76	0.76	0.00	-0.06	-0.07	-0.06
26)	0.94	1.00	1.00	1.00	-0.24	-0.30	-0.30	-0.30
27)	0.82	0.84	0.84	0.83	-0.02	-0.02	-0.01	-0.02
28)	0.85	0.87	0.86	0.86	-0.02	0.00	0.00	0.00
29)	0.73	0.75	0.75	0.76	0.00	-0.01	-0.02	-0.02
30)	0.97	0.98	0.98	0.98	-0.01	0.00	-0.01	-0.01
31)	0.97	0.98	0.98	0.98	-0.01	0.00	0.00	-0.01
32)	0.97	0.98	0.98	0.98	0.00	0.00	-0.01	-0.01
33)	0.94	0.94	0.94	0.95	-0.01	0.01	0.00	0.00
34)	0.69	0.76	0.75	0.76	-0.03	-0.07	-0.05	-0.04
35)	0.55	0.60	0.60	0.60	-0.03	-0.10	-0.08	-0.06