A Monte Carlo Analysis of Ensemble Classification

Roberto Esposito ESPOSITO@DI.UNITO.IT
Dipartimento di Informatica, Università di Torino, C.so Svizzera 185, 10149 Torino, Italy

Lorenza Saitta SAITTA@MFN.UNIPMN.IT
Dipartimento di Informatica, Università di Torino, Spalto marengo 33, 13100 Alessandria, Italy

Abstract

In this paper we extend previous results providing a theoretical analysis of a new Monte Carlo ensemble classifier. The framework allows us to characterize the conditions under which the ensemble approach can be expected to outperform the single hypothesis classifier. Moreover, we provide a closed form expression for the distribution of the true ensemble accuracy, as well as of its mean and variance. We then exploit this result in order to analyze the expected error behavior in a particularly interesting case.

1. Introduction

Ensemble learning has been brought to the attention of the Machine Learning (ML) community by Schapire (1990), who proved that the notions of *strong learnability* and *weak learnability* (Kearns & Valiant, 1988) are equivalent. Since then, ensemble learning has been actively investigated (Freund & Schapire, 1996b; Freund & Schapire, 1996a; Breiman, 1996; Jiang, 2001; Kuncheva & Whitaker, 2003).

From the empirical point of view, ensemble learning shows an amazing effectiveness and robustness to overfitting. In the attempt to explain this appealing behavior, many theoretical and empirical works have tried to relate ensemble learning to results from other fields (Freund & Schapire, 1996a; Collins et al., 2000; Rätsch et al., 2000; Schapire, 1999).

On the other hand, Monte Carlo theory is well established and its results have been applied in several fields, notably Physics and Engineering. According to Brassard and Bratley (1988), a Monte Carlo algorithm is a probabilistic algorithm that, applied to an instance of a class of problems, always provides an answer, but this answer may occasionally be incorrect. In Section 2 more details will be provided.

Following previous work (Esposito & Saitta, 2003), we propose to use a Monte Carlo algorithm to design a new ensemble learner. The aim of this proposal is not to obtain a "superior" algorithm, but to exploit it to enlighten some basic phenomena underlying ensemble classification. The approach suggests a way to look at inductive learning from an entirely new perspective. In fact, it turns out that, in order to understand ensemble classification, the error rate of the hypotheses belonging to the hypothesis space are not so important as one may expect, but rather their collective behavior on each *single* example is what matters. Moreover, Monte Carlo theory acts as a unifying framework, where parameters introduced independently in ensemble learning can be easily embedded, and their role explained in a principled way. For instance, the concept of "margin" has a precise counterpart in this theory, as well as the notion of "order correctness" (Breiman, 1996). Also a decomposition of the error into a bias and a variance part naturally arises. Finally, a closed-form, exact probability distribution of the "true" ensemble accuracy can be computed, as well as its mean and variance.

Even though the paper is biased toward theoretical understanding, results on two Irvine datasets are briefly described, in order to show the agreement between theoretical predictions and experimental findings.

2. Monte Carlo Algorithms

For the sake of self-consistency we report here the basic concepts about Monte Carlo algorithms provided by Brassard and Bratley (1988).

Let $\mathcal{S}$ be a class of problems, $\mathcal{Y}$ a finite set of answers for the problems in $\mathcal{S}$, and $\omega : \mathcal{S} \rightarrow 2^{\mathcal{Y}}$ be a

Appearing in *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada, 2004. Copyright 2004 by the authors.

function that maps each problem into the set of its correct answers. A stochastic algorithm *MC* mapping $\mathcal{S}$ into $\mathcal{Y}$ is said to be a *Monte Carlo algorithm* if it always terminates returning an answer y, which may occasionally be incorrect. A Monte Carlo algorithm is *consistent* if it never outputs two distinct *correct* answers for the same problem s; it is said to be p-correct if the probability that it gives a correct answer to a problem instance s is at least p, *independently* of the specific problem instance s considered. The *advantage* of the algorithm is defined as $\gamma = p - 0.5$.

Let *MC* be a consistent, p-correct Monte Carlo algorithm; let the advantage γ of the algorithm be strictly positive, and let us iterate T times the algorithm over a fixed problem instance s. The number t of times that the algorithm is correct on s is easily recognized to follow a binomial distribution with parameters p and T, i.e., the probability that exactly t successes (correct answers) occur is given by:

$$\Pr\{t\} \geq \binom{T}{t} p^t (1-p)^{T-t}$$

where equality holds for those instances for which the probability of success is exactly p. Let us take the majority answer as the answer of the iterative procedure. Then, this answer will be correct when *MC* is correct more than half the times:

$$\pi = \Pr\left\{t > \frac{T}{2}\right\} = \sum_{t=\left\lfloor \frac{T}{2} \right\rfloor + 1}^{T} \binom{T}{t} p^t (1-p)^{T-t} \qquad (1)$$

The above relation can be used to prove the main theorem in Monte Carlo theory, which states that: "The advantage of a p-correct Monte Carlo algorithm *MC* can be *amplified* as much as desired by increasing the number of iterations, provided that (a) *MC* is consistent, (b) $p > \frac{1}{2}$, and (c) different runs of *MC* on s are independent".

3. Monte Carlo Ensemble Learning

In this section we present a new ensemble learning algorithm based on Monte Carlo theory. As already mentioned, the interest of the algorithm is not in its alleged superiority over existing ones (even though it shows some nice properties) but in its ability to shed a light on some basic phenomena underlying ensemble classification.

3.1. Complete Information

As we are not interested, for the moment, in the details of a specific weak learner, let us abstract away the learning process, and consider as given by an oracle a set Φ of hypotheses and the associated probability distribution $\mathbf{q}$. In a real learning setting, Φ can be thought of as the set of all learnable hypotheses, derived beforehand starting from all the allowed learning sets. Then, learning can be simulated by extracting with replacement from Φ hypotheses according to $\mathbf{q}$. In the following we will also consider as given the whole set $\mathcal{X}$ of examples, together with its probability distribution $\mathbf{d}$. This is clearly an ideal case in which complete information is provided to the "learner". All the computed entities (accuracy, margin, ...) are hence the "true" ones. Let us represent the available information by means of the matrix $\mathcal{M}$ reported in Table 1. In $\mathcal{M}$, each row corresponds to an example x_k in $\mathcal{X}$, and each column to a (extensional) hypothesis φ_j in Φ.

Table 1. Theoretical setting

		φ_1		φ_j		φ_R	
		q_1		q_j		q_R	
x_1	d_1	$p_1(x_1)$		$p_j(x_1)$		$p_R(x_1)$	$p(x_1)$
...							
x_k	d_k	$p_1(x_k)$		$p_j(x_k)$		$p_R(x_k)$	$p(x_k)$
...							
x_N	d_N	$p_1(x_N)$		$p_j(x_N)$		$p_R(x_N)$	$p(x_N)$
		r_1		r_j		r_R	r

Given a hypothesis φ_j and an example x_k, the classification $\varphi_j(x_k) \in \mathcal{Y} = \{+1, -1\}$, assigned by φ_j to x_k may be either correct or incorrect. Let $p_j(x_k) \in \{1, 0\}$ be the probability that hypothesis φ_j correctly classifies example x_k, i.e., that $\varphi_j(x_k) = \omega(x_k)$, being ω the target concept[1]. Let $p(x_k)$ be the average of such probabilities for x_k:

$$p(x_k) = \sum_{j=1}^{R} q_j p_j(x_k) \qquad (2)$$

Moreover, let us take the average of the $p_j(x_k)$'s over the columns; we obtain, for each φ_j, its "true" accuracy, r_j:

$$r_j = \sum_{k=1}^{\mathrm{N}} d_k p_j(x_k) \qquad (3)$$

The r_j values and the $p(x_k)$ values are not totally independent, as the following relation holds:

$$r = \sum_{k=1}^{N} \sum_{j=1}^{R} d_k q_j p_j(x_k) = \sum_{k=1}^{\mathrm{N}} d_k p(x_k) = \sum_{j=1}^{R} q_j r_j \qquad (4)$$

[1] The assumption $p_j(x_k) \in \{0, 1\}$ implies that the Bayes error is zero.

In order to build up an ensemble learner with a Monte Carlo algorithm, we note that the set $\mathcal{S}$ of problems coincides with $\mathcal{X}$; in fact, each example $x_k \in \mathcal{X}$ is a problem to solve. Moreover, $\mathcal{Y} = \{+1, -1\}$ is the set of answers.

Let $MC(x_k|\Phi, \mathbf{q})$ be a Monte Carlo algorithm that extracts a hypothesis φ_j from Φ accordingly to $\mathbf{q}$, and returns $\varphi_j(x_k) = y\,(x_k) \in \mathcal{Y}$. As $p(x_k)$ is the probability of extracting a hypothesis φ_j that correctly classify x_k, MC is $p(x_k)$-correct on x_k. Let $\gamma_k = p(x_k) - \frac{1}{2}$ be the advantage of MC on x_k. If we make T calls to MC and take the majority answer, we can amplify its advantage as mentioned earlier.

Let us now analyze the three required conditions. The consistency condition is verified, as long as the Bayes error is zero, because, in this case, x_k has only one correct label and hence MC cannot be but consistent. We will assume this condition to be true in the rest of the paper. The second condition ($p > 0.5$) can be checked for by computing $p(x_k)$. The last condition (hypothesis independence) is true as long as we draw independently the hypotheses from Φ with replacement.

Given Φ and $\mathcal{X}$, we can classify the examples in two ways: *selection* (single hypothesis classification) or *combination* (ensemble classification). By considering the selection strategy, let us define:

$$\tau_\infty = \max_{1 \leq j \leq R} r_j$$

Notice that $\tau_\infty = 1$ when the set Φ contains the true concept ω. Then, the best selection strategy consists in choosing a hypothesis $\varphi_{j^*}(\cdot)$ that has $r_{j^*} = \tau_\infty$.

The ensemble classification strategy consists in taking a strict majority voting of all the φ_j's on x_k. As $p(x_k) > \frac{1}{2}$ iff the number of 1's in the row is greater than the number of 0's, the examples correctly classified in the limit are all and only those that have $p(x_k) > \frac{1}{2}$. Let $\mathcal{X}_\mathrm{A} = \{x_k | p(x_k) > \frac{1}{2}\}$ be the subset of $\mathcal{X}$ containing the amplifiable examples, i.e., those that have a probability of being correctly classified greater than $\frac{1}{2}$, and let $|\mathcal{X}_\mathrm{A}| = S$. The *asymptotic accuracy* of the ensemble classifier will be:

$$\rho_\infty = \|\mathcal{X}_\mathrm{A}\| = \sum_{k=1}^{S} d_k \qquad (5)$$

Let us notice that (5) gives a pessimistic value of ρ_∞, because all the x_k's with $p(x_k) = \frac{1}{2}$ are considered misclassified. Actually, they could be classified according to the strategy of assigning to them the majority class, reducing thus the number of those which are actually misclassified.

It is clear that the case of availability of complete information is never realized in practice. However, its analysis helps understanding more realistic learning setting, and identifies the important parameters to be estimated.

When all the hypotheses are at hand, the function $H_\infty(x)$ produced by the ensemble classifier, is the following one:

$$H_\infty(x) = \sum_{j=1}^{R} q_j \varphi_j(x)$$

The class assigned to x will be $y\,(x) = \mathrm{sign}\,(H_\infty(x))$. We can prove the following theorems:

Theorem 1 *For each x, the product $H_\infty(x) \cdot \omega(x)$ is greater, equal or less than 0 iff $p(x)$ is greater, equal or less than $\frac{1}{2}$, respectively.*

Theorem 2 *The asymptotic accuracy ρ_∞ is always included in the interval : $2r - 1 \leq \rho_\infty < 2r$.*

In the literature it is often said that bagging (or, in general, ensemble learning) works only when the combined hypotheses have singularly an accuracy greater than $\frac{1}{2}$. Actually, this is not exactly the case. In fact, even with single hypotheses with accuracy less then $\frac{1}{2}$ it is possible to obtain higher accuracies by Monte Carlo amplification. However, in order to reach $\rho_\infty = 1$, the value of r must be at least $\frac{1}{2}$. Even in this case, some of the combined hypotheses are allowed to have an accuracy less than $\frac{1}{2}$, without hindering amplification, provided that r does not go below $\frac{1}{2}$.

The learning problem is completely determined when the $p(x_k)$'s and the r_j's are given. As a consequence, also r is known. Extreme (and interesting) cases occur when either $r \approx 1$ or $r \approx \frac{1}{2}$. In order to investigate the relations between single hypothesis classification and ensemble classification, we can analyze the relations between ρ_∞ and τ_∞, which are graphically illustrated in Figure 1.

According to (4), if $r \approx 1$, most $p(x_k)$'s and most r_j's must be close to 1. Then, τ_∞ and ρ_∞ are close to 1 as well. This situation corresponds to point A in Figure 1. In this case, all hypotheses are extremely good, the Monte Carlo ensemble learner cannot do better than single hypothesis classification.

More in general, the distributions of the $p(x_k)$'s and the r_j's can assume, among others, any of the forms reported in Figure 2. Hence, there are, in principle, 16 extreme cases. Let us analyze some of the combinations.

When both the $p(x_k)$'s and the r_j's follow distribu-

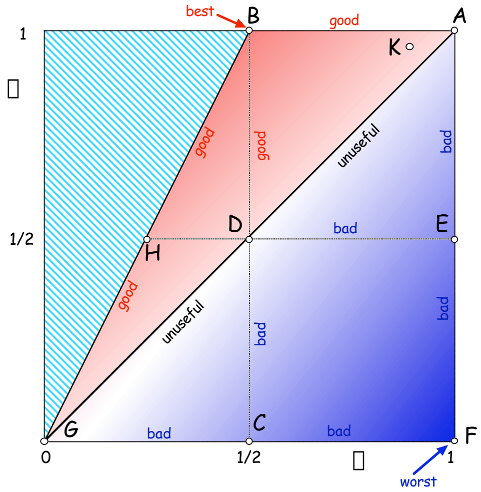
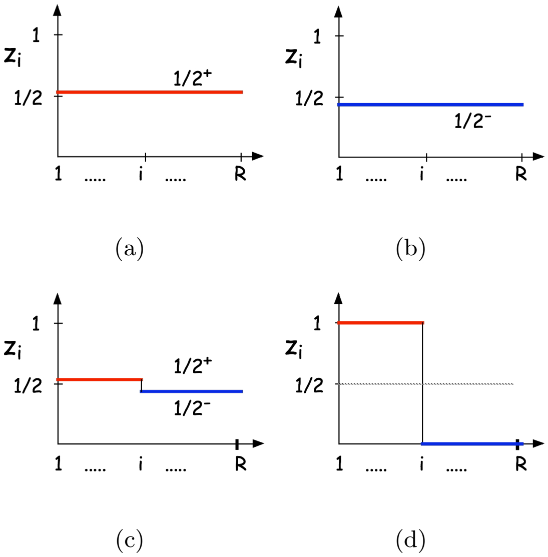


Figure 1. Relationships between the values of ρ_∞ and τ_∞ and the suitability of using ensemble classification rather than single hypothesis classification.



(a)

(b)

(c)

(d)

Figure 2. Extreme probability distributions of a stochastic variable $\mathcal{Z}$ with mean close to 1/2. The values of the abscissas correspond to the ordinal indices of the values, whereas the ordinates are the actual values of $\mathcal{Z}$, decreasing with increasing i (from left to right). The notation α^+ (α^-) means that the considered value is a little more (less) than α.

tion (2(a)), the best single hypothesis has accuracy $\tau_\infty \approx \frac{1}{2}$. On the other hand, by definition, $\mathcal{X}_\mathrm{A} = \mathcal{X}$ and, hence, the ensemble classifier has accuracy $\rho_\infty = 1$. This is the ideal case for ensemble learning: with hypotheses only a little better than random guess a perfect ensemble classifier is obtained. This situation corresponds to point B in Figure 1.

When both the $p(x_k)$'s and the r_j's follow distribution (2(b)), we recognize a bad case for ensemble learning: with base hypotheses with accuracy close to random guess, the ensemble classifier is always wrong. This situation corresponds to point C in Figure 1.

When both the $p(x_k)$'s and the r_j's follow distribution (2(c)), we have an useless case for ensemble learning: it cannot achieve an accuracy greater than that of the best single hypothesis. This situation corresponds to point D in Figure 1.

By combining in various ways the distribution of the $p(x_k)$'s with the distribution of the r_j's, also points F, G and H in Figure 1 can be reached. Actually, any point inside the polygon GFABG is reachable. For example, point F is reached when the distribution of the r_j's is of the type (2(d)), i.e., it includes the correct concept, with accuracy 1. If the distribution of the $p(x_k)$'s is of type (2(b)), again $\mathcal{X}_\mathrm{A} = \emptyset$, and, hence, the ensemble classifier has accuracy $\rho_\infty = 0$. Point F represents the worst possible condition for ensemble learning.

As it must be $\rho_\infty < 2r \leq 2\tau_\infty$, the straight line GB corresponds to a boundary that cannot be crossed. On this line, $\rho_\infty = 2\tau_\infty$, and ensemble classification is convenient. Segment DB is good for ensemble classification, as $\rho_\infty > \tau_\infty$ on this segment. On the contrary, segment DC is a bad one, because $\rho_\infty < \tau_\infty$. Other bad segments are DE, EF, and AF, whereas a good one is BA. Finally, the diagonal GA is simply useless as $\rho_\infty = \tau_\infty$ on it.

Another important consequence can be drawn from the analysis. Even though the distributions of the $p(x_k)$'s and r_j's are related by condition (4), this condition is very weak. Usually Machine Learning takes into consideration the distribution of the r_j's. However, the analysis clearly show that the knowledge of the r_j's does not tell anything definite about the likelihood of success of the ensemble classifier: the important distribution is the one of the $p(x_k)$'s, and the same distribution of the r_j's can correspond to very different distributions of the $p(x_k)$'s. Hence, even if we fix beforehand the r_j, the Monte Carlo process may turn out to be useful, useless or harmful.

Standard machine learning techniques consider T

columns in the matrix $\mathcal{M}$, and then combine the classifications of the examples given by the single φ_j's $(1 \leq j \leq T)$. This approach has the drawback that classifications of different examples are not independent, (they are performed by the same set of T hypotheses) and Monte Carlo theory predicts that amplification may be problematic. On the contrary, the same theory tells how amplification should be obtained: for each occurrence of each example x_k, T hypotheses must be extracted from Φ (or learned) and combined through the majority voting mechanism. Then, the set of T hypotheses used for one example occurrence is in general different from the set used for another one. It is clear that we need, to perform this type of classification on a set of M examples, to extract (learn) a number MT of hypotheses, which is the number required by the leave-one-out method.

Notice that, in classical approaches to Machine Learning (even with ensemble methods), an example is always classified in the same way (when Bayes error is equal to zero). With Monte Carlo ensemble classification, each occurrence of the same example is independent of the others, and the learning/classification cycle is repeated as if the example never appeared before. Counterintuitive as it might be, the independence in the occurrences of the same example generates a much smaller variance of the accuracy than with rigidly coupled classifications. This results can be proved theoretically and has been experimentally verified.

Let us now recall that, for binary classification, the margin $\mu(x_k)$ of an example x_k is defined as the difference between the score of the correct class and the score of the incorrect one (Schapire et al., 1998).

Theorem 3 *(Esposito and Saitta (2003)) For any x_k it holds: $\mu(x_k) = 2p(x_k) - 1$, where $\mu(x_k)$ is the "true" margin of x_k.*

Theorem 3 allows a clear explanation of the role of the margin. Increasing the margin of x_k means to increase the Monte Carlo probability of a correct classification of x_k.

Moreover, Breiman has introduced the concept of "order-correct" classifier, and then uses this concept to prove that "If a predictor is order-correct for most inputs, then aggregation can transform it into a nearly optimal predictor" (Breiman, 1996, p.131).

In the settings assumed here, the following theorem holds.

Theorem 4 *A weak learner WL is order-correct on example x_k, iff $p(x_k) > 0.5$.*

Hence, the $p(x_k)$ values appear to play the most fundamental role. As it is evident by now, it is the most important parameter defined by Monte Carlo theory and, at the same time, it can be related naturally to two among the most fruitful concepts in ensemble learning.

3.2. Partial Information

As mentioned before, the case of complete information is a theoretical one. In this section we take a step toward a more realistic setting, by assuming that not all the information is available from the onset.

More precisely, we want to use only T hypotheses extracted from Φ accordingly to distribution $\mathbf{q}$ instead of the whole Φ. Then, building up an ensemble classifier consists in using the following algorithm:

```
AmpMC $(x_k|\Phi, T)$
Extract $\varphi_{j_1}, \varphi_{j_2} \ldots, \varphi_{j_T}$ from $\Phi$ according
    to $\mathbf{q}$
return the majority answer of
    $\varphi_{j_1}(x_k) \ldots \varphi_{j_T}(x_k)$ on $x_k$.
```

AmpMC returns a hypothesis which performs, for each occurrence of x_k, a sequence of Bernoulli trials, each with probability of success $p(x_k)$, where "success" is a correct classification of x_k by any φ_j. *AmpMC* correctly classifies x_k if more than half of the T extracted hypotheses are correct on it. In the following we will use $H_T(x_k)$ to denote the classifier learned by *AmpMC* on x_k.

Definition 1 *Let $\pi_T(x_k)$ be the probability that $H_T(x_k)$ correctly classifies x_k.*

As the number of successes produced by *AmpMC* is a stochastic variable governed by a Binomial distribution, the probability $\pi_T(x_k)$ of observing more than $T/2$ successes can be computed using formula (1). The relation allows us to prove the following theorem:

Theorem 5 *When the number of trials T goes to infinity, the probability $\pi_T(x_k)$ tends to 1 if $p(x_k) > \frac{1}{2}$, tends to 0 if $p(x_k) < \frac{1}{2}$, and tends to $\frac{1}{2}$ if $p(x_k) = \frac{1}{2}$.*

The preceding theorem allows us to assert that, for each $x_k \in \mathcal{X}_{\mathrm{A}}$, $\lim_{T\to\infty} \pi_T(x_k) = 1$. For each element x_k of $\mathcal{X}_{\mathrm{A}}$ its $\pi_T(x_k)$ value increases with T and tends to 1, whereas, for each element x_k not belonging to $\mathcal{X}_{\mathrm{A}}$ the corresponding $\pi_T(x_k)$ value decreases and tends to 0 (or to $\frac{1}{2}$ when $p(x_k) = \frac{1}{2}$). Moreover, the greater $p(x_k) > \frac{1}{2}$, the faster the increase of $\pi_T(x_k)$, whereas the lower $p(x_k) < \frac{1}{2}$, the faster the decrease of $\pi_T(x_k)$. Hence, all examples in $\mathcal{X}_\mathrm{A}$ will be correctly classified in the long run, i.e.:

$$\forall x_k \in \mathcal{X}_\mathrm{A} : \lim_{T \to \infty} \Pr\{H_T(x_k) = \omega(x_k)\} = 1 \qquad (6)$$

Let ρ_T be the true accuracy over $\mathcal{X}$ when an ensemble classifier $H_T(\cdot)$ is used. We would like to compute the probability distribution of ρ_T. In order to do so, let us notice that, as previously explained, a Monte Carlo algorithm *AmpMC* may label in different ways a single example if it is presented to it more than once. Let us consider a set $\mathcal{X}'$ of examples to be classified and let $\mathcal{X}'$ contain each example $x_k \in \mathcal{X}$ a number of times $n_k = \mathrm{N}' \cdot d_k$, with n_k integer and $\sum_{k=1}^{\mathrm{N}} n_k = \mathrm{N}'$.

Let us consider the indicator function $I_k \equiv \mathcal{I}_{H_T(x_k)=\omega(x_k)}$; the accuracy ρ_T of $H_T(\cdot)$ can be written as:

$$\rho_T = \frac{1}{\mathrm{N}'} \sum_{j=1}^{\mathrm{N}'} I_j \qquad (7)$$

I_j is a stochastic variable that takes value 1 with probability $\pi_T(x_k)$, and value 0 with probability $[1 - \pi_T(x_k)]$. From (7) we can compute directly the expected value of ρ_T:

$$\begin{aligned} \mathrm{E}[\rho_T] &= \sum_{j=1}^{\mathrm{N}'} \frac{\pi_T(x_j)}{\mathrm{N}'} = \sum_{k=1}^{N} \frac{n_k}{\mathrm{N}'} \pi_T(x_k) \\ &= \sum_{k=1}^{N} d_k \pi_T(x_k) \end{aligned} \qquad (8)$$

Analogously, it can be shown that:

$$\mathrm{VAR}[\rho_T] = \frac{1}{\mathrm{N}'} \sum_{k=1}^{N} d_k \pi_T(x_k)(1 - \pi_T(x_k)) \qquad (9)$$

Let us notice that, for $T = 1$, $\mathrm{E}[\rho_1] = r$.

From Theorem 5, we obtain:

$$\lim_{T\to\infty} \mathrm{E}[\rho_T] = \rho_\infty \quad \lim_{T\to\infty} \mathrm{VAR}[\rho_T] = 0$$

The variable ρ_T is an asymptotically correct, consistent, and efficient estimator of ρ_∞.

If we consider, instead, the natural setting in which each x_k (however repeated) is always classified in the same way, we obtain the following results for the accuracy $\rho_T^* = \sum_{k=1}^{\mathrm{N}} d_k I_k$:

$$\begin{aligned} \mathrm{E}[\rho_T^*] &= \sum_{k=1}^{\mathrm{N}} d_k \pi_T(x_k) = \mathrm{E}[\rho_T] \\ \mathrm{VAR}[\rho_T^*] &= \sum_{k=1}^{\mathrm{N}} d_k^2 \pi_T(x_k)(1 - \pi_T(x_k)) \geq \mathrm{VAR}[\rho_T] \end{aligned}$$

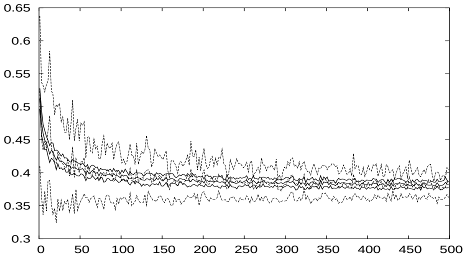


Figure 3. Mean accuracy and variance plot for Bagging (dotted lines) and the Monte Carlo Ensemble Learner (solid lines) on the Ionosphere dataset. The experiment used a weak learner which acquires highly correlated hypotheses (namely it induces a sphere centered on a positive example).

The above equations take into account the dependencies introduced by duplicate examples. We notice, nevertheless, that the issue is more general. In real situations, the primary source of dependencies among the classifications of examples is the weak learner itself. The analysis, in this case, is more difficult and is probably uninteresting for the present discussion. However, for the sake of illustration, we report in Figure 3 the plot of the mean and the variance of the accuracies of the two algorithms in a situation in which the phenomenon is evident. The weak learner used in the experiment produces highly dependent hypotheses (in particular most hypotheses agree on the classification of negative examples). The experiment is also interesting because it shows a situation, based on real data, in which ensemble learning is harmful. In particular, $r_j > 0.5$ for all j, while the ensemble classifier error tends to a value close to 0.4.

The probability distribution of ρ_T is reported in Figure 4, where $\theta_T(\xi) = \Pr\{\rho_T = \xi\}$, and $\mathcal{X}_v$ is a subset of $\mathcal{X}'$ of cardinality v.This probability function can be approximated by $\mathcal{N}(\mathrm{E}[\rho_T], \mathrm{VAR}[\rho_T])$. Empirical evidence shows (and the central limit theorem predicts) that for ($N \gtrsim 100$) the normal approximation is practically indistinguishable from $\theta_T(\xi)$.

An interesting consequence of formula (8) is that we can theoretically evaluate $\mathrm{E}[\rho_T]$ versus T. A particularly interesting case is when the expected accuracy starts below ρ_∞, reaches a point above it and then approaches the limit accuracy from above (see Figure 5). This phenomenon does not appear to have been noticed before, even though some theoretical motivations have been suggested for Adaboost (Jiang, 2001).

Knowing ρ_∞, we know that the ensemble classifier will approach ρ_∞ for increasing T, starting from r. If $\rho_\infty < r$, actually ensemble classification harms accu-

$$\theta_T(\xi) = \sum_{v=\max\{0,\mathrm{N}'\xi-\mathrm{N}'\rho_\infty\}}^{\min\{\mathrm{N}'\xi,(1-\rho_\infty)\mathrm{N}'\}} \left[\sum_{\mathcal{X}'_{\mathrm{N}'\xi-\mathrm{N}'\rho_\infty+v} \subseteq \mathcal{X}'_A} \left(\prod_{x_k \in \mathcal{X}'_{\mathrm{N}'\xi-\mathrm{N}'\rho_\infty+v}} [1-\pi_T(x_k)] \prod_{x_k \in \mathcal{X}'_A - \mathcal{X}'_{\mathrm{N}'\xi-\mathrm{N}'\rho_\infty+v}} \pi_T(x_k) \right) + \sum_{\mathcal{X}'_v \subseteq \mathcal{X}'-\mathcal{X}'_A} \left(\prod_{x_k \in \mathcal{X}'_v} \pi_T(x_k) \prod_{x_k \in \mathcal{X}'-\mathcal{X}'_A-\mathcal{X}'_v} [1-\pi_T(x_k)] \right) \right] \tag{10}$$

Figure 4. Definition of $\theta_T(\xi) = \Pr\{\rho_T = \xi\} = \Pr\{\rho_T = \frac{m_T}{\mathrm{N}'}\}$, where m_T is the number of correct classifications in N'.

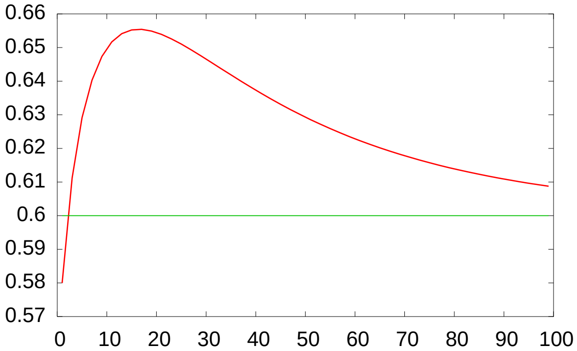


Figure 5. E $[\rho_T]$ versus T, when 60% of the examples have $p(x) = 0.7$ and the remaining 40% have $p(x) = 0.4$.

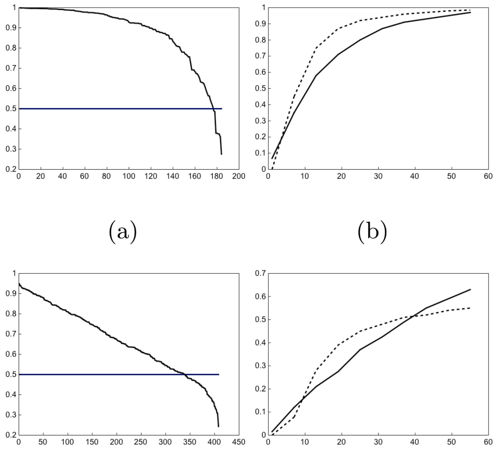
racy, and single hypothesis selection is certainly better. More precisely, let $\Phi_B = \{\varphi_j | r_j \geq \rho_\infty\}$ and $\chi_\infty = \|\Phi_B\|$. The value of χ_∞ is the probability that a single hypothesis with accuracy not less than ρ_∞ is extracted; then we can compute the probability that within T hypotheses at least one of them is better than Monte Carlo ensemble classifier:

$$\eta_T(\rho_\infty) = \Pr\{\tau_T \geq \rho_\infty\} = 1 - b(0, T, \chi_\infty)$$

We can use the following rough rule to compare selection versus combination:

```
**If** $\rho_\infty < r$, **then** *selection*
**else if** $\chi_\infty = 0$ **then** *combination*
**else** choose the strategy that requires the
    lowest $T$ value to arrive with high proba-
    bility close to or higher than $\rho_\infty$.
```

Of course in practice we do not know ρ_∞, χ_∞ and r. However, these are the parameters that have to be estimated. To this aim, a theory of hypothesis testing and of point estimation is needed. A discussion can be found in (Esposito, 2003). Here, we provide, instead, empirical results on two datasets taken from the Irvine repository. The results are reported in Figure 6. The experiments confirm that whether it is convenient to perform single extraction versus combination depends on the $p(x_k)$ distribution.



(a) (b)

(c) (d)

Figure 6. Comparison between Monte Carlo ensemble classification and single hypothesis extraction. The pictures on the left report the $p(x_k)$ distribution associated to a CART-like algorithm for the datasets Ionosphere (top most) and Pima (bottom most). The abscissas report the examples indexes, the ordinates report $p(x_k)$ values. The horizontal lines show $p = 0.5$. The pictures on the right reports the corresponding behavior of the expected error of a Monte Carlo ensemble learner (dashed line) versus the error of a single hypothesis extraction algorithm (full line) as T increases.

4. Discussion

A preliminary analysis of neural network ensembles, along similar lines as those adopted here, can be found in Hansen and Salamon (1990). The authors attempt a theoretical analysis of ensemble classification via majority voting. Even though some of the formulas reported in their work seem similar, syntactically, to some appearing in this paper, there is a deep semantic difference. In fact, their analysis appears to be limited to a very specific case among the ones presented here, i.e., the case in which a set of networks produce independent errors in such a way that the probability of being misclassified by majority voting is the same for each example. According to the analysis reported here, this case may even be impossible to realize. Moreover, the Hansen and Salamon suggest that ensemble classification is always beneficial, which is clearly not the case.

In this paper we adopt the model suggested by Monte Carlo algorithms theory (Brassard & Bratley, 1988) to define a new ensemble classifier. The exploitation of such a model allowed us to compute interesting quantities about a Monte Carlo Ensemble Learner. The distinguishing feature of the Monte Carlo ensemble learner is that it extracts hypotheses at random any time it needs to classify a new example. This makes it a very "atypical" classification tool, but it greatly simplifies its analysis without hindering the opportunity of transferring the results to typical ensemble learners. For instance, Bagging (Breiman, 1996) is an approximation of such a process (Saitta & Esposito, 2004; Esposito, 2003).

References

Brassard, G., & Bratley, P. (1988). *Algorithmics: theory and practice*. Prentice-Hall, Inc.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*, 123–140.

Collins, M., Schapire, R. E., & Singer, Y. (2000). Logistic regression, adaboost and bregman distances. *Computational Learing Theory* (pp. 158–169).

Esposito, R. (2003). *Analyzing ensemble learning in the framework of monte carlo theory*. Doctoral dissertation, Dipartimento di Informatica, Università di Torino, C.so Svizzera 185, 10149 Torino, Italy.

Esposito, R., & Saitta, L. (2003). Monte Carlo Theory as an Explanation of Bagging and Boosting. *Proceeding of the Eighteenth International Joint Conference on Artificial Intelligence* (pp. 499–504). Morgan Kaufman Publishers.

Freund, Y., & Schapire, R. (1996a). Game theory, online prediction and boosting. *Proceedings, 9th Annual Conference on Computational Learning Theory* (pp. 325–332).

Freund, Y., & Schapire, R. E. (1996b). Experiments with a new boosting algorithm. *Proc. 13th International Conference on Machine Learning* (pp. 148–146). Morgan Kaufmann.

Hansen, L. K., & Salamon, P. (1990). Neural networks ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *12*, 993–1001.

Jiang, W. (2001). Some theoretical aspects of boosting in the presence of noisy data. *Proc. 18th International Conf. on Machine Learning* (pp. 234–241). Morgan Kaufmann, San Francisco, CA.

Kearns, M., & Valiant, L. (1988). *Learning boolean formulae or finite automata is as hard as factoring* (Technical Report). Harvard University Aiken Computational Laboratory.

Kuncheva, L. I., & Whitaker, C. J. (2003). Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, *51*, 181–207.

Rätsch, G., Warmuth, M., Mika, S., Onoda, T., Lemm, S., & Müller, K.-R. (2000). Barrier boosting. *Thirteenth Annual Conference on Computational Learning Theory*.

Saitta, L., & Esposito, R. (2004). *Ensemble learning and monte carlo algorithms* (Technical Report TR-04-12). Dipartimento di Informatica, Università del Piemonte Orientale.

Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, *5*, 197–227.

Schapire, R. E. (1999). Drifting games. *Computational Learing Theory* (pp. 114–124).

Schapire, R. E., Freund, Y., Bartlett, P., & Lee, W. S. (1998). Boosting the margin: A new explanation for the effectiveness of voting methods. *Annals of Statistic*, *26*, 1651–1686.