Experimental Comparison between Bagging and Monte Carlo Ensemble Classification

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

Lorenza Saitta SAITTA@MFN.UNIPMN.IT
Dipartimento di Informatica, Università del Piemonte Orientale, 15100 Alessandria, Italy

Abstract

Properties of ensemble classification can be studied using the framework of Monte Carlo stochastic algorithms. Within this framework it is also possible to define a new ensemble classifier, whose accuracy probability distribution can be computed exactly. This paper has two goals: first, an experimental comparison between the theoretical predictions and experimental results; second, a systematic comparison between bagging and Monte Carlo ensemble classification.

1. Introduction

Recently, the theory of Monte Carlo algorithms has been proposed as a framework to investigate properties of ensemble classification and learning (Esposito & Saitta, 2004). A new ensemble classifier, $Amp\mathcal{MC}$, was introduced as a by-product of the analysis. Starting from that work, the first goal of this paper is to experimentally verify the validity of the theoretical predictions about classification accuracy. Another goal is to assess the explicative power of the Monte Carlo framework; as it was shown by Esposito and Saitta (2004), this framework allowed some interesting properties of ensemble classification to be discovered/explained. Then, a pertinent question would be: is the framework only suitabe to study $Amp\mathcal{MC}$'s properties or is it useful to also assess features of other ensemble learners? If this question could be answered positively, then we would have at our disposal a powerful and well assessed tool for studying ensemble learners in general. In this paper we actually provide a positive answer to the question with respect to Bagging (Breiman, 1996).

Starting from the above perspective, the comparison between $Amp\mathcal{MC}$ and Bagging is not oriented toward providing evidence of any supposed "superiority" of one of the two algorithms over the other, but it simply tries to establish conditions under which transfer of properties is sound. Obviously, both algorithms have advantages and drawbacks, which we also try to assess.

In order to make this paper self-consistent, we recall in Section 2 some definitions about Monte Carlo algorithms and summarize some previous results. In Section 3 a theoretical comparison between $Amp\mathcal{MC}$ and Bagging is provided. Section 4 describes the experimental environment, while Section 5 reports results on natural datasets. Section 6 describes experiments performed on artificial datasets and some conclusions are reported in Section 7.

2. Background

In this section we briefy recall some definitions about Monte Carlo algorithms, and a summary of Esposito and Saitta (2004)'s results, for the sake of self-consistency.

2.1. Monte Carlo stochastic algorithms

According to Brassard and Bratley (1988), a stochastic algorithm $\mathcal{MC}$ is *Monte Carlo* when, applied to any instance x of a class X of problems, it always returns an answer (belonging to a predefined discrete set Y), but this answer may occasionally be incorrect.

A Monte Carlo algorithm is *consistent* if it never returns two different, both *correct*, answers to the same problem. Finally, if the probability that $\mathcal{MC}$ returns a correct answer for any problem instance x is at least p, $\mathcal{MC}$ is said to be *p-correct*.

In the context of classification tasks, the set X of examples corresponds to the set of problems, the answers

Appearing in *Proceedings of the 22^{nd} International Conference on Machine Learning*, Bonn, Germany, 2005. Copyright 2005 by the author(s)/owner(s).

to the problems to the class labels to be assigned, and the classifier to $\mathcal{MC}$. If a classification problem has a Bayes error equal to zero, then $\mathcal{MC}$ is certainly consistent.

Let $\omega(x)$ be the correct class of x. As $\mathcal{MC}$ has no memory, not only each problem x is to be handled independently from the others, but also multiple occurrences of the same problem x are to be handled independently as well. As a consequence, one occurrence of x may receive an answer, and another occurrence of x may receive a different one.

The great interest of Monte Carlo algorithms resides in their *amplification* ability: let X be a class of problems, and let $\mathcal{MC}$ be a Monte Carlo algorithm both consistent and p-correct over X. By running $\mathcal{MC}$ T times on a fixed instance $x \in X$ and taking the majority answer as the result, the probability that x is correctly classified approaches 1 exponentially fast with T, provided that p is greater than 1/2.

In this paper we only consider binary classification problems, i.e., $Y = \{+1, -1\}$. Let, in the following, $|X| = N$, and let $D = \{d_k | 1 \leq k \leq N\}$ be a fixed probability distribution over X. As we are not interested in the description language used to represent hypotheses, these last are only considered from an *extensional* point of view, i.e., we only take notice of the way they partition examples into positive and negatives ones. Hence, for a discrete example space, there are only finitely many different hypotheses, each one corresponding to a different subset of X (positive examples).

2.2. Problem setting

Given the above premises, let us define the Monte Carlo matrix M reported in Table 1. The matrix has a number of rows equal to N, and a number R of columns equal to the number of available hypotheses, each one with a probability associated to it: these hypotheses may have been provided by an oracle, or have been previously learned from a given set of learning sets. Let Φ be the set of the hypotheses, and q the associated probability distribution. Then, we abstract away from the working of the learner, and learning can be simulated by extracting hypotheses from Φ according to q.

Given a hypothesis φ_j and an example x_k, the classification $\varphi_j(x_k) \in Y$ 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 $\varphi_j(x_k) = \omega(x_k)$. The value $p_j(x_k)$ is the entry in row k and column j of matrix M.

Table 1. Theoretical setting: Matrix M

		φ_1		φ_j		φ_R	
		q_1		q_j		q_R	
x_1	d_1	$p_1(x_1)$		$p_j(x_1)$		$p_R(x_1)$	$p(x_1)$
$\dots$							$\dots$
x_k	d_k	$p_1(x_k)$		$p_j(x_k)$		$p_R(x_k)$	$p(x_k)$
$\dots$							$\dots$
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

Let $p(x_k)$ be the average of the entries of row k in matrix M:

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

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

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

The lowest, right most element of the matrix M is the average taken on all matrix elements:

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

The r_j and $p(x_k)$ values are not totally independent, as they are related by (1).

2.3. Summary of previous results

In order to exploit the amplification ability, the following Monte Carlo algorithm $\mathcal{MC}(x_k|\Phi, q)$ can be applied to a problem x_k, extracted from X according to D:

```
$\mathcal{MC}(x_k|\Phi, q)$
  Extract $\varphi_j$ from $\Phi$ according to $q$
  Classify $x_k$ with $\varphi_j$
  Return $\varphi_j(x_k)$
```

As $p(x_k)$ is the probability of extracting a hypothesis φ_j that correctly classifies x_k, $\mathcal{MC}$ is $p(x_k)$-correct on x_k. If we make T calls to $\mathcal{MC}$ and accept the majority answer, we can let the probability of success of $\mathcal{MC}$ become as close to 1 as desired, provided that $\mathcal{MC}$ is consistent, $p(x_k) > 1/2$, and that the calls to $\mathcal{MC}$ are independent.

Repeating $\mathcal{MC}$ T times on the same x_k and taking the majority answer amounts to apply the algorithm $Amp\mathcal{MC}$.

By analysing *AmpMC*, Esposito and Saitta (2004) have reported a number of results; some of them, relevant to this work, are the following ones:

- In the limit of combining all available hypotheses, the classification accuracy ρ_∞ is given by the sums of the probabilities d_k of the examples that have $p(x_k) > 1/2$.
- The probability $\pi_t(x_k)$ of classifying correctly example k using T hypotheses extracted randomly from Φ accordingly to q is $\pi_t(x_k) = \sum_{t=\lfloor \frac{T}{2}+1 \rfloor}^{T} \binom{T}{t} p(x_k)^t (1-p(x_k))^{T-t}$.
- Let ρ_T denote the accuracy when T hypotheses are combined. The exact probability distribution of ρ_T can be computed.

3. Monte Carlo amplification vs. Bagging

A theoretical analysis of Bagging was provided by Breiman (1996) in a setting analogous to the one considered by Esposito and Saitta (2004), i.e., the example set X is known (together with a continuous probability distribution $P_X(x)$ over the examples), as well as the set of all hypotheses generated by a weak learner from bootstrap replicates of a given dataset.

Let us first establish the correspondence between the notations in the two approaches. Esposito and Saitta (2003a; 2003b; 2004) consider a discrete set $X = \{x_k | 1 \leq k \leq N\}$ of examples, with an associated probability distribution $D = \{d_k | 1 \leq k \leq N\}$; then, Breiman's integration over x corresponds to summation over k. Moreover, Breiman allows the Bayes error to be different from zero, as the probability $P(j|x)$ that an example x is a-priori labelled as class j may be different from 0 and 1. We only consider Bayes error zero, i.e.:

$$P(j|x) = \begin{cases} 1 & \text{if } \omega(x) = j \\ 0 & \text{if } \omega(x) \neq j \end{cases} \qquad (2)$$

In (2), $\omega(x)$ denotes the correct class of x. In an analogous way, we can rewrite Breiman's probability $Q(j|x)$ that a random hypothesis assigns class j to example x in our terms as follows:

$$Q(j|x) = \begin{cases} p(x) & \text{if } \omega(x) = j \\ 1 - p(x) & \text{if } \omega(x) \neq j \end{cases}$$

Breiman introduces the notion of order-correctness of a hypothesis w.r.t an example, i.e., a hypothesis φ is *order-correct* w.r.t. x, if:

$$\arg\max_j Q(j|x) = \arg\max_j P(j|x) \qquad (3)$$

In our framework, relation (3) is satisfied for x_k iff the Monte Carlo probability of p(x_k) is greater than 0.5. In other words, Breiman's notion of order-correctness coincides with Monte Carlo's amplifiability.

Afterwards, Breiman computes the accuracy r_A of the aggregated classifier ϕ_A, as the sum of two terms, one over the examples for which ϕ_A is order-correct, and one over the remaining examples. Applying the Monte Carlo framework, the first term is equal to the asymptotic accuracy ρ_∞, whereas the second one is zero. As a conclusion, Breiman's formulas can be derived as special cases from Monte Carlo theory.

It is worth noticing that Breiman's experimental observation that "poor predictors can be transformed into worse ones" has a double justification in the Monte Carlo theory: first, when the Bayes error is different from zero, Bagging is inconsistent (according to Monte Carlo theory), and, second, low hypothesis accuracies may let the number of order-correct examples go to zero (Esposito & Saitta, 2004).

Since Bagging has its own hypothesis extraction mechanism (learning strategy) and its own way of classifying the examples, we shall model, first of all, how Bagging relates to the matrix M. All the parameters introduced so far are still valid. As explained in Section 3, the Monte Carlo ensemble classifier *AmpMC* works by extracting from Φ, according to q, T hypotheses for each occurrence of each example x_k. Hence, the sets of hypotheses used for each example x_k are normally different. Then, we could say that *AmpMC* works *by rows*. On the contrary, Bagging only uses T hypotheses, and applies them to all the examples. We could say that Bagging works *by columns*.

The relevant question now is how these two ways of working impact classification results. An answer to this question is provided by the following theorem:

Theorem. *The expected error of the Bagging algorithm with T hypotheses is equal to the expected error of the Monte Carlo classifier with T hypotheses.*

Proof. Let us consider the Monte Carlo matrix M, and let $H_T = \langle \varphi_{c_1}, \varphi_{c_2}, \ldots, \varphi_{c_T} \rangle$ be a selection (repetitions are allowed) of cardinality T of its columns. The accuracy of selection H_T, which corresponds to a bagged hypothesis, is given by the total weight of the rows in M for which the sum of the 1's corresponding to the columns in H_T is larger than $\lfloor T/2 \rfloor$. In formulae:

$$\rho_T^{\mathsf{Bag}}(H_T) = \sum_k d_k I_{\sum_{j=1}^{T} \varphi_{c_j}(x_k) > \lfloor \frac{T}{2} \rfloor}$$

The expectation of ρ_T^{Bag} can be evaluated by considering all possible selections of T columns:

$$E\left[\rho_T^{\mathsf{Bag}}\right] = E_{H_T}\left[\rho_T^{\mathsf{Bag}}(H_T)\right] = \sum_{H_T} \Pr\{H_T\} \cdot \sum_k d_k I_{\sum_{j=1}^T \varphi_{c_j}(x_k) > \lfloor \frac{T}{2} \rfloor}$$

We need to prove that the above formula corresponds to the expected accuracy of the Monte Carlo classifier. Let us start by rearranging a little bit the summations:

$$\sum_{H_T}\left(\Pr\{H_T\} \cdot \sum_k d_k I_{\sum_{j=1}^T \varphi_{c_j}(x_k) > \lfloor \frac{T}{2} \rfloor}\right) = \sum_{H_T}\sum_k \Pr\{H_T\} d_k I_{\sum_{j=1}^T \varphi_{c_j}(x_k) > \lfloor \frac{T}{2} \rfloor} = \sum_k d_k \sum_{H_T} \Pr\{H_T\} I_{\sum_{j=1}^T \varphi_{c_j}(x_k) > \lfloor \frac{T}{2} \rfloor}$$

By noticing that the inner summation is the probability of classifying correctly example k using T hypotheses extracted randomly and accordingly to q from the Monte Carlo matrix, we obtain:

$$E[\rho_T^{\mathsf{Bag}}] = \sum_k d_k \sum_{H_T} \Pr\{H_T\} I_{\sum_{j=1}^T \varphi_{c_j}(x_k) > \lfloor \frac{T}{2} \rfloor} = \sum_k d_k \pi_T(x_k) = E[\rho_T^{\mathsf{MC}}]$$

Which completes the proof. □

On the contrary, nothing can be said, *a priori*, about the error variances in the two approaches: either one can, in fact, be lower. In Figure 1 two examples of M matrices, in which the variances differ, are reported. Based on the above findings, it seems plausible to claim Bagging to be a cheaper approximation of *AmpMC*: in fact, Bagging only learn T hypotheses for classifying N examples, whereas *AmpMC* learn $N \cdot T$ hypotheses for the same task. The expected error is the same, and Bagging's variance may be either higher or lower then *AmpMC*'s. Actually, we can say a little more about the variance; in (Esposito & Saitta, 2004) it was shown that by using *AmpMC* and considering independently each occurrence of the same example x_k, a provably lower variance is obtained w.r.t. the case in which the same instance is always classified in the same way. As Bagging always classifies in the same way any occurrence of the same example, it is reasonable to expect that its variance be often greater than *AmpMC*' variance.

$$\begin{pmatrix} 1&1&1&1&1&1&0&0&0&0\\ 1&1&1&1&1&1&0&0&0&0\\ 1&1&1&1&1&1&0&0&0&0\\ 1&1&1&1&1&1&0&0&0&0\\ 1&1&1&1&1&1&0&0&0&0\\ 1&1&1&1&1&1&0&0&0&0\\ 1&1&1&1&1&1&0&0&0&0 \end{pmatrix}$$

(a) A Monte Carlo matrix for which the error variance of *AmpMC* is lower than Bagging's

$$\begin{pmatrix} 0&1&0&1&0&1&0&1&0&1\\ 1&0&1&0&1&0&1&0&1&0\\ 0&1&0&1&0&1&0&1&0&1\\ 1&0&1&0&1&0&1&0&1&0\\ 0&1&0&1&0&1&0&1&0&1\\ 1&0&1&0&1&0&1&0&1&0\\ 0&1&0&1&0&1&0&1&0&1 \end{pmatrix}$$

(b) A Monte Carlo matrix for which the error variance of Bagging is lower than *AmpMC*'s

Figure 1.

4. Empirical Settings

As mentioned in Section 1, the aim of the experimentation is twofold: on the one hand we want to check the correctness of the theoretical predictions of Monte Carlo theory about the classification accuracy of *AmpMC*, and about the relations between *AmpMC* and Bagging. In order to explore this issue, we have used twelve datasets from Irvine's repository (Blake & Merz, 1998). On the other hand, in order to investigate more closely the relations between the variances, we have designed an artificial suite of problems, whose characteristics vary with continuity in such a way that, according to the theory, the two variances are bound to differ.

In all experiments the number T of component hypotheses vary in $\{0, \ldots, 101\}$. Only odd values of T have been considered, as the accuracy graph versus T presents strong oscillations, due to the parity. For each T, classification has been repeated 60 times in order to estimate the statistics of interest. Then, the statistics about the two algorithms have been compared with the theoretical predictions and between each other.

4.1. Natural Datasets Description

In the first set of experiments, we considered 12 binary classification tasks from Irvine's repository (Blake & Merz, 1998). For the sake of simplicity, we selected tasks involving numerical attributes only. The selected datasets are described in Table 2. No tuning of the learning algorithm's parameters has been done.

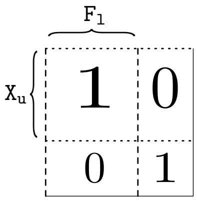
The Monte Carlo matrices used in the experiments have been built starting from a learning set $\mathcal{L}$ and a test set $\mathcal{T}$ in the following way. Five thousand classifiers have been acquired using a CART-like algorithm run on bootstrap replicates (Efron & Tibshirani, 1993) of $\mathcal{L}$. Then, the classifiers have been used to fill the entries in the Monte Carlo matrix M. More precisely, for each $k \in \{1, \ldots, |\mathcal{L}| + |\mathcal{T}|\}$ and $j \in \{1, \ldots, 5000\}$ $M(k, j)$ has been set to 1 if the j-th hypothesis classified correctly example number k, and to 0 otherwise.

Table 2. Description of natural datasets. Nick is the abbreviate name we use in reporting the results, $|A|$ is the number of attributes. $|\mathcal{L}|$ and $|\mathcal{T}|$ are the cardinalities of the training and test sets, respectively.

Dataset	Nick	$\lvert A\rvert$	$\lvert\mathcal{L}\rvert$	$\lvert\mathcal{T}\rvert$
Echocardiogram	echo	9	63	56
Heart Desease Cleveland	hdc	13	144	120
Heart Desease Hungarian	hdh	13	138	119
Heart Desease Switzerland	hds	13	60	54
Heart Desease VA	hdv	13	90	74
Hepatitis	hepa	19	72	62
Ionosphere	iono	34	184	167
Pima	pima	8	408	360
Waveform	wav	21	2505	2495
Waveform (noisy)	wns	40	2505	2495
Wisconsin Diagnostic Breast Cancer	wdbc	30	303	266
Wine	wine	13	92	86

4.2. Artificial Monte Carlo Matrices

The theory points out that differences should become apparent when the "shape" of the Monte Carlo matrix varies. For instance, Figure 1 offers an example. Interesting enough, the two matrices can be rewritten as particular instantiations of the matrix template reported on the left. The picture represents a family of matrices $M_{\mathtt{F_1},\mathtt{X_u}}$, which show four distinct regions arranged in a xor-like pattern. The 1's denote regions of the matrix containing mostly 1's, whereas the 0's denote regions of the matrix containing mostly 0's, Each matrix in the family contains 100 rows and 100 columns. $\mathtt{F_1}$ and $\mathtt{X_u}$ represents respectively the number of hypotheses which stand on the left side of the matrix and the number of examples which occupy its upper part. As it is easy to see $M_{51,100}$ corresponds to a situation similar to the one reported in Figure 1(a), while the matrix $M_{50,50}$ corresponds to Figure 1(b) (we notice that rows and columns of the matrix reported in the figure should be reordered properly for the correspondence to become apparent; this only amounts to renaming both examples and hypotheses).



5. Experiments on Natural Datasets

In this section we describe the experiments aimed at comparing theoretical predictions, experimental observations and links between Bagging and $\mathit{Amp}\mathcal{MC}$.

For each dataset, classification experiments with $\mathit{Amp}\mathcal{MC}$ and Bagging have been performed for 51 values of T, from 1 to 101. For each T value the entire process has been repeated 60 times. The results for pima dataset are presented in Figure 2, and consist of two parts: the left one reports the results for Bagging, i.e., the average (over the 60 repetitions) experimental error and the experimental error ± one standard deviation, for both the learning and the test set. The right part reports the same for $\mathit{Amp}\mathcal{MC}$. Moreover, the theoretical values for the error and its variance, predicted by the Monte Carlo model, are reported as well. The figures corresponding to the other 11 datasets show a similar structure.

In order to quantitatively evaluate the results, a number of statistical tests have been performed. First of all, the empirical values of the error have been compared, via a t-Test, to the theoretical ones, for each T value, for the learning and test sets, for Bagging and $\mathit{Amp}\mathcal{MC}$ (in all, 2448 tests). These tests have supported "equality" between theory and experiments with p > 0.99 in all cases. Then, the experimental variance has been compared to the theoretical one via an F-Test (with degree of freadom $v_1 = 59$), again for each T value, for the learning and test sets, for Bagging and $\mathit{Amp}\mathcal{MC}$. Whereas the tests referring to $\mathit{Amp}\mathcal{MC}$ almost always support "equality" between theory and experiments, with p > 0.99, the same does not occur for Bagging, providing so a first indication of the difference between the two methods. As there is no space to report all the results in details, we concentrate on the most interesting question, namely the difference in variance between Bagging and $\mathit{Amp}\mathcal{MC}$.

In order to compare the two variances, we have performed two series of tests. First of all, we have performed an F-Test for each T value, for the learning and test sets. We have used the statistic $F = \mathsf{var}(\rho_T^{\mathsf{Bag}})/\mathsf{var}(\rho_T^{\mathit{Amp}\mathcal{MC}})$ with degrees of freedom $v_1 = 59$ and $v_2 = 59$. We have used as null hypothesis $H_0 = \mathsf{var}(\rho_T^{\mathsf{Bag}}) > \mathsf{var}(\rho_T^{\mathit{Amp}\mathcal{MC}})$ and performed a one-tail test. The results are summarized in the fourth and fifth column of Table 3. Each column shows (separately for the learning and the test sets) the number of times the test succeeded (indeed, $\mathsf{var}(\rho_T^{\mathsf{Bag}}) > \mathsf{var}(\rho_T^{\mathit{Amp}\mathcal{MC}})$) with a significance level p > 0.95.

Actually, the above F-Test is too local to convey really significant information, because, for the same graphs (for instance, Learning set of Pima with Bagging vs. Learning set of Pima with $Amp\mathcal{MC}$), the test is sometimes positive an sometimes negative. Moreover, the test is unable to capture global features; for instance, it is possible that $\mathsf{var}(\rho_T^{\mathsf{Bag}})$ is always greater than $\mathsf{var}(\rho_T^{Amp\mathcal{MC}})$, but not sufficiently to let the pointwise test succeed. What we need, then, is a more global test, which is able to capture the overall relationship between the two variances over the entire T span. To this purpose, we can use the non-parametric Wilcoxon rank test. We test again the null hypothesis $H_0 = \mathsf{var}(\rho_T^{\mathsf{Bag}}) > \mathsf{var}(\rho_T^{Amp\mathcal{MC}})$ and perform a one-tail test with a stricter significance level, p > 0.99. In this case, when the test fails, we tested also the null hypothesis $H'_0 = \mathsf{var}(\rho_T^{\mathsf{Bag}}) < \mathsf{var}(\rho_T^{Amp\mathcal{MC}})$. The results are reported in the second and third column of Table 3. In these columns, an entry "Bag" denotes that $\mathsf{var}(\rho_T^{\mathsf{Bag}}) > \mathsf{var}(\rho_T^{Amp\mathcal{MC}})$, an entry $\mathcal{MC}$ denotes that $\mathsf{var}(\rho_T^{\mathsf{Bag}}) < \mathsf{var}(\rho_T^{Amp\mathcal{MC}})$, whereas an entry "NO" denotes that the two variances are statistically equal. For the Hepatitis and Wine datasets, only the Wilcoxon test result is reported, because the variances of both $Amp\mathcal{MC}$ and Bagging become identically zero for $T \geq 13$ and $T \geq 15$, respectively.

From Table 3, we notice that the variance of Bagging is greater in 15 out of 24 cases, the variance of $Amp\mathcal{MC}$ is greater in 4 out of 24 cases, whereas in the remaining 5 cases there is no statistically significant difference between the two variances.

Two considerations are in order here. First of all, we may notice that in some cases the results of the Wilcoxon and F tests are clearly in agreement; for example, in the HD (Cleveland), the F test gives a clear predominance of successes. In other cases, such as Waveform, the results from the Wilcoxon and F tests seem to be contradictory: The Wilcoxon test says that for the learning set Bagging has the greater variance, but the F test says that only in 4 cases out of 51 (T values) this is the case. The reason is that Bagging has almost for all T values a greater variance, but not sufficient to let the F test be positive (remember that the ratio between the two variances must be at least 1.54 in order to reach the 0.95 significance level). The Wilcoxon test is able to take into account this situation, which, globally, corresponds to a Bagging's greater variance.

The second observation is that the considered datasets do not contain duplicate examples. According to the theory, Bagging's variance is predicted to increase in datasets with many duplications. Then, the experimentation is favorably biased toward Bagging.

Table 3. Results of Wilcoxon and F tests on natural datasets.

Dataset	Wilcoxon		Test F	
	Learn	Test	Learn	Test
echo	Bag	Bag	40	27
hdc	Bag	Bag	42	37
hdh	NO	$\mathcal{MC}$	11	4
hds	Bag	$\mathcal{MC}$	11	2
hdv	NO	Bag	3	15
hepa	Bag	NO	/	/
iono	Bag	Bag	29	46
pima	NO	Bag	16	32
wav	Bag	$\mathcal{MC}$	4	1
wns	Bag	Bag	24	36
wdbc	Bag	Bag	51	51
wine	$\mathcal{MC}$	NO	/	/

6. Experiments on the Artificial Datasets

In this section we report a second set of experiments which aim to investigate how the behavior of the two algorithms changes with different problem settings, corresponding to different structures of the Monte Carlo matrices. In particular, the matrices we produced were different instantiations of the $M_{\mathtt{F_1},\mathtt{X_u}}$ matrix template.

The results are reported in Figure 3. The figure consists of a matrix of six rows and two columns. The left column contains experiments where $\mathtt{F_1}$ has been set to 53 and $\mathtt{X_u}$ assumes values in $\{55, 70, 95\}$. Similarly, the right column contains experiments having $\mathtt{X_u} = 53$ and $\mathtt{F_1}$ in $\{55, 70, 95\}$. In each plot, we reported with solid lines the theoretical predictions, with dotted lines the observations about $Amp\mathcal{MC}$, and with dotted-dashed lines the observations about Bagging (as in the other figures, $E[\rho_T]$ and $E[\rho_T] \pm \sigma(\rho_T)$ have been plotted for each classifier and for the theoretical predictions).

From the results, we can observe that the variance of Bagging tends to increase as $\mathtt{X_u}$ increase and $\mathtt{F_1}$ is low, but tends to be better than Monte Carlo's one when the opposite happens. Interestingly, we can also notice that the experiments suggest that, being free to choose, one should adopt the $Amp\mathcal{MC}$ way of combining hypotheses. In fact, we can observe that while the variance of Bagging can become very large, this does not happen in the case of $Amp\mathcal{MC}$. The results reported in the second column, in fact, show that even if the variance of $Amp\mathcal{MC}$ is larger than the one of Bagging, this happens because Bagging's variance is

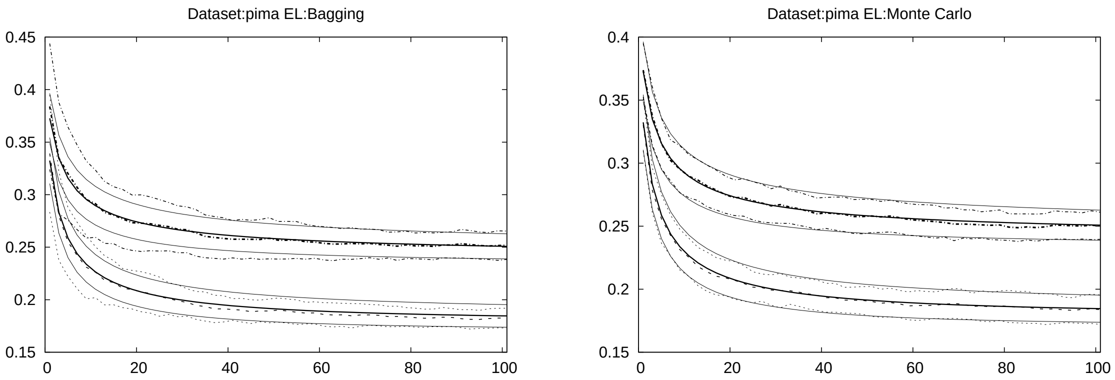


Figure 2. Experiments on Pima dataset. All pictures report the expected error ± one standard deviation for the given ensemble algorithm and increasing values of T. Both theoretical predictions (solid lines) and observed errors (dashed-dotted and dotted lines) are reported. The statistics are separately measured on the training set (dotted lines) and on the test set (dashed-dotted lines).

very small. In other words, even when Bagging has a better error variance, $Amp\mathcal{MC}$ one is “not too bad”.

Unfortunately, in practical situations the choice is not so easy. In fact, the Monte Carlo classification strategy is a costly one since one have no choice but to learn T different hypotheses from scratch each time a novel example is presented. This is clearly unfeasible in contexts where speed is important and data abound.

7. Discussion

In this paper, we extended Esposito and Saitta (2004) work with an investigation of the links between Monte Carlo theory and Bagging, and performed an experimentation, with both natural and artificial datasets, to empirically validate the theory prediction.

A first conclusion that can be drawn is that the experiments show an amazing match with the theory predictions about the error and error variance in all analysed cases.

For what concerns the relation between $Amp\mathcal{MC}$ and Bagging, we have shown, both theoretically and empirically, that the two procedures have the same average error. This result allows some of the found properties from Monte Carlo theory to be applied to Bagging; for instance, the fact that the behaviour of the average error may be non monotone with T, property that was unknown before the introduction of the Monte Carlo framework.

The relationship between the variances is more complex, but we think it is safe to say that Bagging tends to have a larger variance than $Amp\mathcal{MC}$. On the other hand, Bagging can be seen as a ”low cost” $Amp\mathcal{MC}$ algorithm, useful when resources are scarce.

From the experiments of the artificial datasets, we can conclude that $Amp\mathcal{MC}$ is favoured w.r.t. the variance when $\mathtt{X_u}$ is large and $\mathtt{F_1}$ is low. On the other hand Bagging is favoured w.r.t. the variance, when $\mathtt{X_u}$ is low and $\mathtt{F_1}$ is large.

References

Blake, C., & Merz, C. (1998). UCI repository of machine learning databases. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

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

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

Efron, B., & Tibshirani, R. (1993). *An introduction to the bootstrap.* Chapman and Hall.

Esposito, R., & Saitta, L. (2003a). Explaining Bagging with Monte Carlo Theory. *Lecture Notes in Artificial Intelligence* (pp. 189–200). Springer.

Esposito, R., & Saitta, L. (2003b). 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.

Esposito, R., & Saitta, L. (2004). A Monte Carlo Analysis of Ensemble Classification. *Proceedings of the twenty-first International Conference on Machine Learning* (pp. 265–272). Banff, Canada: ACM Press, New York, NY.

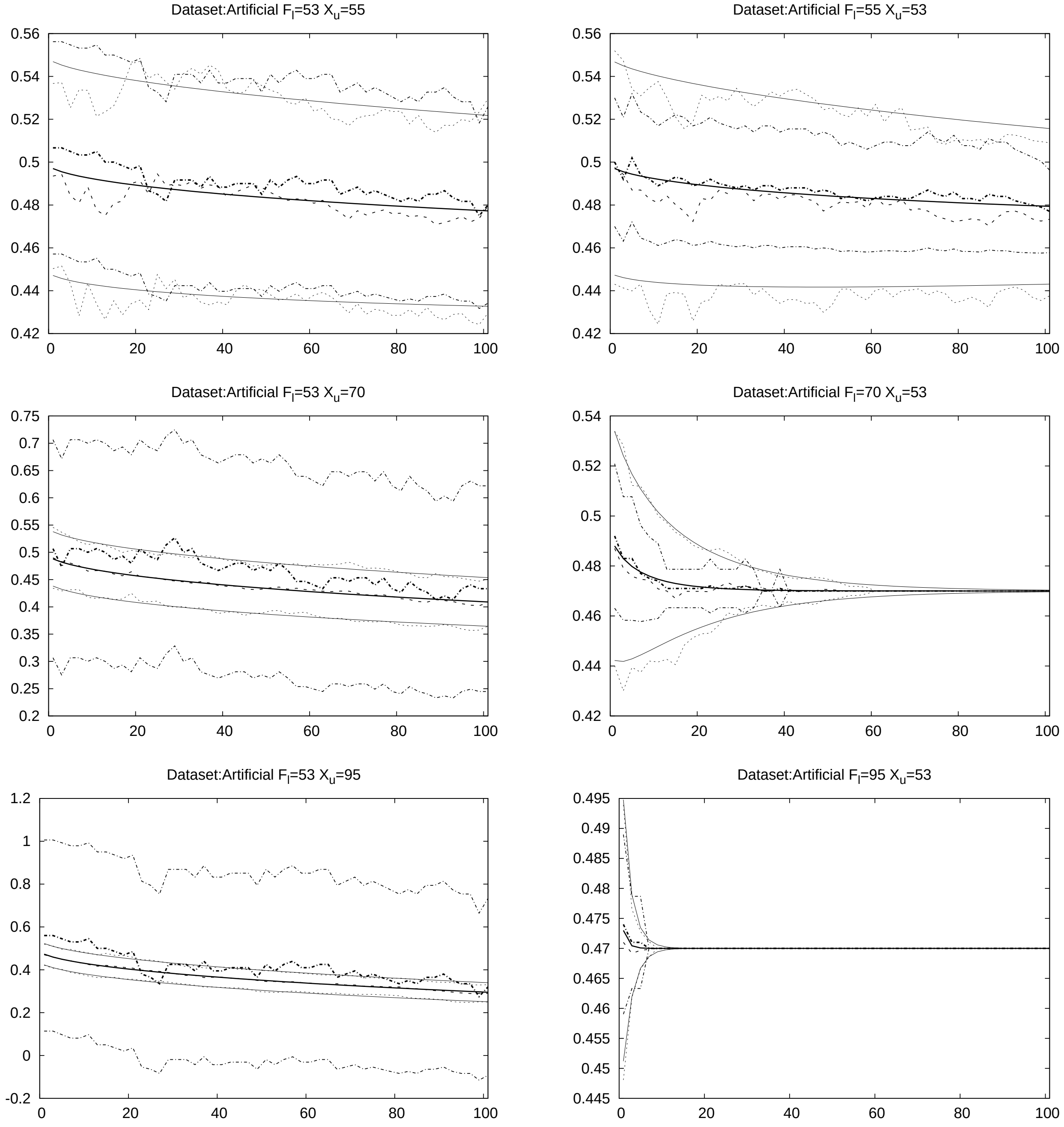


Figure 3. Experiments on artificial datasets. All pictures report the expected error ± one standard deviation of *Amp*$\mathcal{MC}$ and Bagging for increasing values of T. The solid lines reports the theoretical predictions, the dotted lines reports the statistics for *Amp*$\mathcal{MC}$, and the dashed-dotted lines report the statistics for Bagging. Notice that, in order to improve readability, different scales have been used.