

# Boosting and Classification of Electronic Nose Data

Francesco Masulli<sup>1,2</sup>, Matteo Pardo<sup>3</sup>, Giorgio Sberveglieri<sup>3</sup>, and  
Giorgio Valentini<sup>1,4</sup>

<sup>1</sup> INFN, Istituto Nazionale per la Fisica della Materia  
Via Dodecaneso 33, 16146 Genova, Italy

<sup>2</sup> Dipartimento di Informatica, Università di Pisa  
Corso Italia 40, 56125, Pisa, Italy  
masulli@di.unipi.it

<sup>3</sup> INFN and Dipartimento di Chimica e Fisica  
Via Valotti 9-25123 Brescia, Italy

pardo@tflab.ing.unibs.it, sbervegl@tflab.ing.unibs.it

<sup>4</sup> DISI - Dipartimento di Informatica e Scienze dell'Informazione  
Università di Genova, Via Dodecaneso 35, 16146 Genova, Italy  
valenti@disi.unige.it

**Abstract.** Boosting methods are known to improve generalization performances of learning algorithms reducing both bias and variance or enlarging the margin of the resulting multi-classifier system. In this contribution we applied Adaboost to the discrimination of different types of coffee using data produced with an Electronic Nose. Two groups of coffees (blends and monovarieties), consisting of seven classes each, have been analyzed. The boosted ensemble of Multi-Layer Perceptrons was able to halve the classification error for the blends data and to diminish it from 21% to 18% for the more difficult monovarieties data set.

## 1 Introduction

Boosting methods have been successfully applied to many domains, ranging from text filtering, to natural language processing, classification of multisource remote sensing data, Geographical Information Systems, diagnosis of tumors and others [14, 15, 9, 7, 11, 1].

In this paper we present a novel application of boosting to electronic noses data. Electronic Noses (EN), in the broadest meaning, are instruments that analyze gaseous mixtures for discriminating between different (but similar) mixtures and, in the case of simple mixtures, quantify the concentration of the constituents.

Electronic nose data show typically a relatively small cardinality, as the experimental techniques are complex and time-consuming, and from this point of view resampling methods can help to improve the generalization capabilities of classifiers with small data sets.

In fact bagging and boosting are known to reduce variance [6]. Taking a weighted majority over many hypotheses, trained on different examples drawn

from the same data set, has the effect of reducing the random variability of the combined hypotheses: as a consequence, the variance is reduced. Boosting achieves bias reduction generating distributions that concentrate on harder examples, trying to generate diverse base learners specialized in different resampled data sets [5].

The generalization capabilities of boosting can also be explained in the framework of large margin classifiers theory, as it had been shown that boosting enlarges the margins of the training examples [17]. Recently Pedro Domingos showed the equivalence between the bias-variance interpretation and the margin-based explanation of the error reduction induced by boosting methods [4].

We applied boosted ensemble of Multi-Layer Perceptrons (MLP) to the classification of commercial coffees using electronic nose data. Commercial coffees are blends, which, for economic reasons, contain monovarietal coffees of various origins. For the producers the availability of analysis and control techniques is of great importance. There exists a rich literature on the characterization of coffee using the chemical profile of one of its fractions, such as the vapor of green or roasted beans or the phenolic fraction.

The usage of a single neural network (normally a Multi-Layer Perceptron, but Radial Basis Function Networks had also been investigated) as a classifier is a common solution to pattern recognition problems in EN odor analysis [8, 12]. We tried to improve the classification capabilities of single MLP using boosting methods.

The paper is organized as follows: the next section briefly introduces electronic noses and some pre-processing problems that characterize the applicative domain, then the experimental setup, results and discussion on the application of boosted MLP ensemble to electronic nose data are presented.

## 2 Electronic noses

Electronic Noses (EN) consist of a sampling system (for a reproducible collection of the mixture), an array of sensors (which is the heart of the system), electronic circuitry and data analysis software [8]. EN using arrays of chemical sensors can be divided into three categories according to the type of sensitive material used: inorganic crystalline materials (e.g. semiconductors, as in MOS-FET structures, and metal oxides); organic materials and polymers; biologically derived materials. Comparatively to classical techniques (e.g. the combination of gas chromatography and mass spectroscopy (GC-MS)), ENs are simpler, cheaper devices. They recognize a fingerprint, that is a global information, of the samples to be classified.

In particular for our experimentation we used the Pico-1 EN developed at the Gas Sensory Lab. in Brescia. The Pico-1 EN makes use of six thin film semiconductor sensors. For this experiment three  $SnO_2$ -based (one catalyzed with gold, one with palladium and one with platinum) and three  $Ti-Fe$  sensors were employed. All of them were grown by sputtering with the RGTO technique [13].

The odor to be analyzed can be sampled either in a static way with a programmable autosampler comprising a syringe, or in a dynamic way letting the carrier flush through the headspace, or from stained steel canisters or nalophan bags through a pump. For this application the possibility of easily preparing the sample suggested the adoption of the more reproducible static headspace extraction with the autosampler.

Pico-1 precisely controls the sensor temperature via a feedback loop. Further, there is the possibility to steer the EN remotely via the TCP-IP interface. A simple user interface for the preliminary analysis of data (graphs of sensor responses, time development of extracted features, PCA score and loading plots) has also been implemented in *Matlab*.

The typical measurement consists of the exposure of the sensors to a concentration step, that is a change of odor concentration from zero to  $c$  (each component of the vector stands for a gas component) and back to zero again, and of the recording of the subsequent change in resistance. The classical feature extracted from the response curve is the relative change in resistance.

### 3 Experimental Setup

We used the the Pico-1 EN to analyze blended coffees (7 different blends) and 6 single varieties plus the Italian Certified Espresso (ICE). The *blended coffees* data set is composed by 187 samples and the *monovariety coffees* data set is composed by 225 samples. The data are six dimensional corresponding to the relative change in resistance achieved from the data registered by the six sensors.

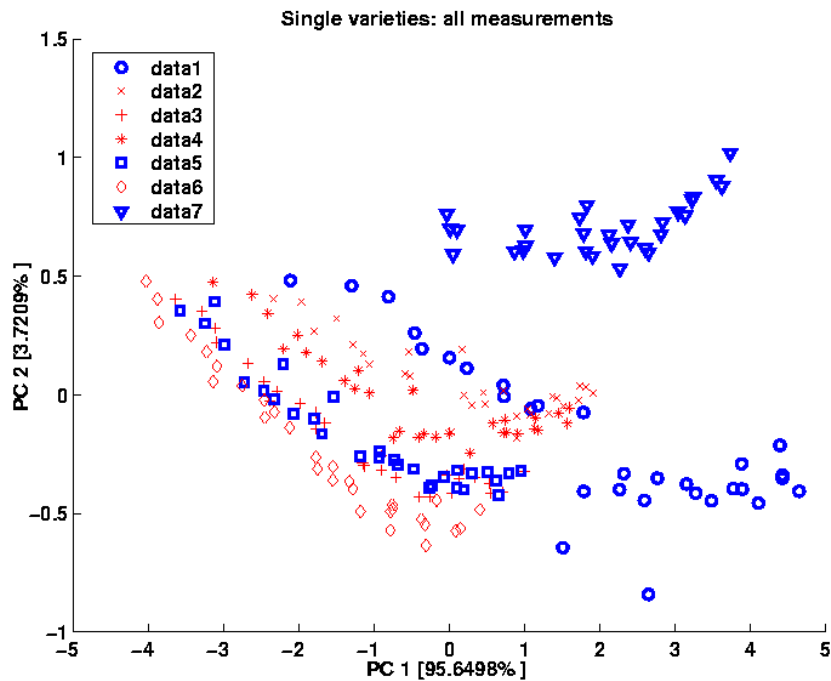
For these experiments, 2 g of ground coffee are introduced into a vial with a volume of 20 cm<sup>3</sup> which is crimped with seal and septa. The vial is then left in an incubation oven at 50 C for 30 minutes in order to generate the aroma. Ten vials for every coffee type of the monovariety group and 12 vials for every coffee type of the blend group were prepared. Three successive extractions were performed from the same vial. While the data set is not big for machine learning standards, this is a considerable dataset to be collected with an E-Nose.

The first part of data analysis deals with signal processing (e.g. removal of spikes, noise filtering), the choice of the features to be considered in the subsequent analysis and data visualization stages. Drift correction is also considered as part of this processing of the data. This part of data is crucial for the quality of the final results, and requires a constant exchange with the experimental process, mainly to establish a sufficiently good and reliable measurement protocol.

The second part of the data analysis deals with inferring the relationships between the EN data and the corresponding class labels. We compared the results of a single Multi-Layer Perceptron (MLP) with the corresponding boosted MLP ensemble, using the *Adaboost.M1* algorithm introduced by Freund and Schapire [5, 6]. This boosting algorithm was originally designed for two-class classification problems, but it has been demonstrated that it is effective also for multiclass classification problems when the base learner is strong enough to achieve reasonably high accuracy [5]. In our implementation we used boosting

by resampling, i.e. we chose a set of examples from the training set at random with replacement, according to the current probability distribution of the data. We randomly split the data in a training (70% of the data) and in a testing set (30%) and we repeated training of each learning machine six times using different pseudorandom initialization weights. The same series of pseudorandom initialization weights have been used both for single MLPs and boosted MLPs. In both cases we exploited MLPs with one hidden layer, using 5, 7 and 9 hidden units for the *blended coffees* and 20, 30 and 40 hidden units for the *monovariety coffees* data sets. All the experimentations have been performed developing applications using the C++ *NEUROjects*<sup>5</sup> library [19].

A global impression of the measurements can be gained from the PCA plots in Fig. 1. We note that:



**Fig. 1.** PCA plot of the monovariety group of coffees. The values in square brackets refer to the percentage of the total variance for respectively the first and the second principal components.

1. Drift is present and it affects the first PC. Nonetheless the visual separation obtained with the 2nd and 3rd PCs wasn't seen to be any better.

<sup>5</sup> Download web site: <http://www.disi.unige.it/person/ValentiniG/NEUROjects>

2. Coffee #7 forms a distinct cluster: in fact this is a very roasted coffee and it is considered to be of the worst quality.
3. The ICE points (class #1) form two somewhat distinct clusters. The measurements of the cluster on the bottom right were made at different times: it is possible that the blend's headspace evolved differently with time with respect to that of the SVs (single varieties).
4. A third fact about the SV is that coffees #3 and #5 are mixed. This makes sense since the coffees are of the same type, both being washed arabic beans of good quality.

## 4 Results and Discussion

The results of our experimentation are summarized in Tab. 1. The table represents the results on the test sets of blended and monovariety coffee data sets, using boosted ensembles of MLP and single MLP trained with a classical back-propagation algorithm. Each row of the table shows results relative to MLP or

**Table 1.** MLP and boosted MLP results on blended coffees and monovariety coffees data sets.

<b>MLP results on blended coffees data set</b>									
hidden #	Percent error rate on different runs						Best	Mean	Stdev
5	20.97	19.35	16.13	19.35	20.97	16.13	16.13	18.82	2.01
7	11.29	14.52	14.52	11.29	19.35	19.35	11.29	15.05	3.31
9	16.13	17.74	16.13	17.74	17.74	17.74	16.13	17.20	0.76
<b>MLP results on monovariety coffees data set</b>									
hidden #	Percent error rate on different runs						Best	Mean	Stdev
20	21.43	23.21	21.43	21.43	21.43	23.21	21.43	22.02	0.84
30	23.21	23.21	21.43	23.21	21.43	23.21	21.43	22.62	0.84
40	25.00	25.00	23.21	23.21	23.21	23.21	23.21	23.81	0.84
<b>Boosted MLP results on blended coffees data set</b>									
hidden #	Percent error rate on different runs						Best	Mean	Stdev
5	9.68	11.29	9.68	11.29	11.29	11.29	9.68	10.75	0.83
7	6.45	9.68	9.68	9.68	6.45	9.68	6.45	8.60	1.67
9	11.29	9.68	6.45	9.68	6.45	11.29	6.45	9.14	2.20
<b>Boosted MLP results on monovariety coffees data set</b>									
hidden #	Percent error rate on different runs						Best	Mean	Stdev
20	21.43	21.43	19.64	21.43	19.64	21.43	19.64	20.83	0.92
30	21.43	17.86	19.64	17.86	19.64	21.43	17.86	19.64	1.60
40	23.21	17.86	19.64	19.64	23.21	17.86	17.86	20.24	2.44

boosted MLP with a predefined number of hidden units. The first column of the

table refers to the number of hidden units of a single MLP or of a single MLP base learner of the boosted ensemble. The next 6 columns correspond to percent error rates obtained by different pseudorandom initialization of the weights of the MLP. The 8<sup>th</sup> column show the minimum error achieved (Best), while the next corresponds to the average error (Mean) and the last shows the standard deviation of the percent error rate (Stdev).

Comparing the overall results on the blended coffee data set between MLP and boosted MLP (Tab. 1), we can remark that the average error (Mean) is halved using boosted MLP ensembles: The percent error rate on the test set drops down from 15.05 to 8.60, using MLP with 7 hidden units as base learners, and similar results are obtained also using MLP with 5 and 9 hidden units. The minimum error, also, is reduced in a similar way, from 11.29 to 6.45 %.

A reduction of the percent error rate, both for the average and the minimum error can be observed also on the monovariety coffee data set, but with a remarkably lower decrement. In this case the average error decreases only from 22.02 to 19.64 and the minimum error from 21.43 to 17.86.

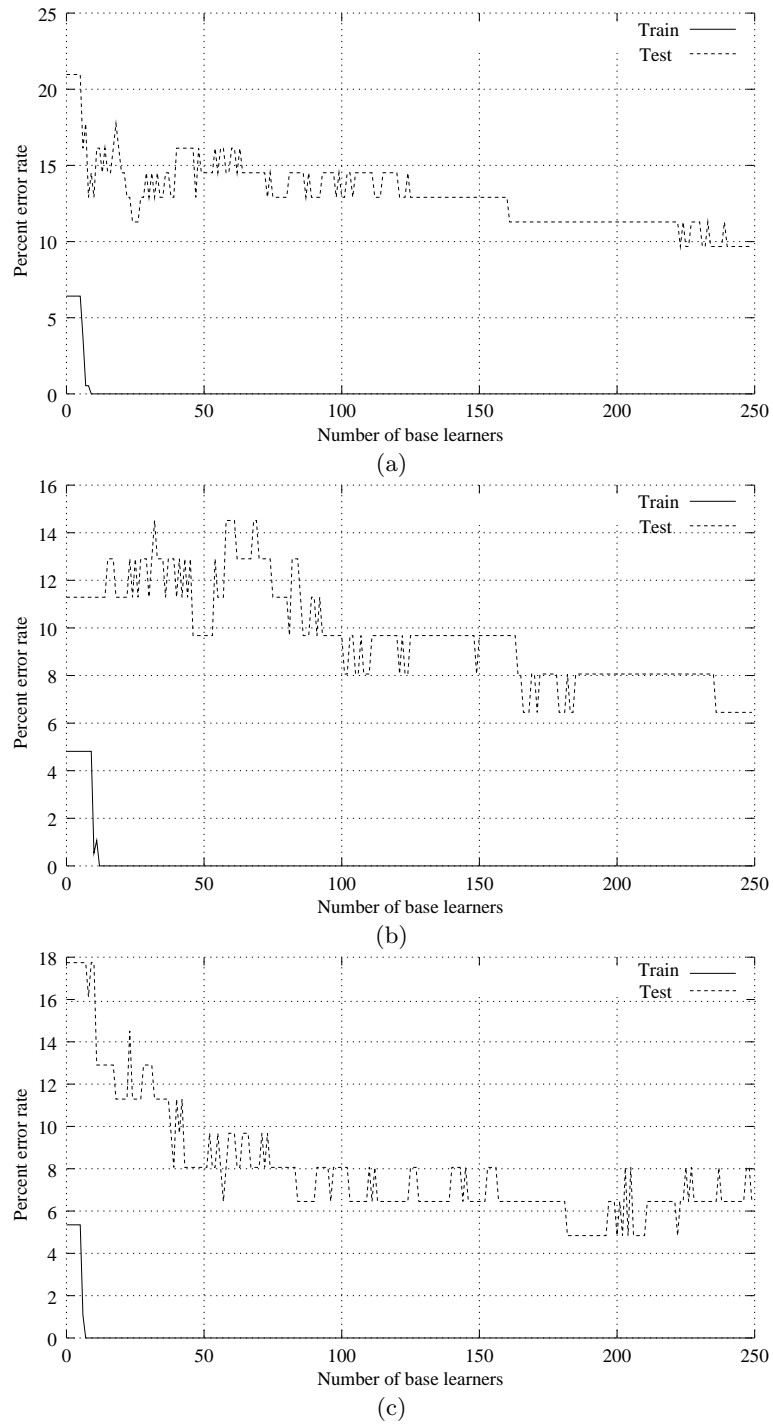
Fig. 2 and 3 show the error rates of the boosted MLP as a function of the number of base learners of the ensemble. The error rate on the training set drops to 0 after about 10 rounds of boosting on the blended coffee data set (Fig. 2), and after about 150 rounds on the monovariety coffee data set (Fig. 3). In both cases an exponential decrement of the error can be observed, according to Freund and Schapire’s theorem stating that the training error exponentially falls to zero incrementing the number of base learners, given that the weighted error of each base learner is less than 0.5 [5].

The test error on the blended data set continues to decrease, even after the training error reaches zero. A similar trend can also be noted in the monovariety data set, even if the test error lowers more slowly. This fact has been observed also in other works [16, 18, 17] and has been explained in the framework of large margin classifiers, interpreting boosting as an algorithm that enlarges the margins of the training examples [17]: even if the training error reaches zero the boosting algorithm continues to enhance the margins, focusing on the hardest examples. As a consequence, the generalization capabilities of the boosted ensemble are improved [16].

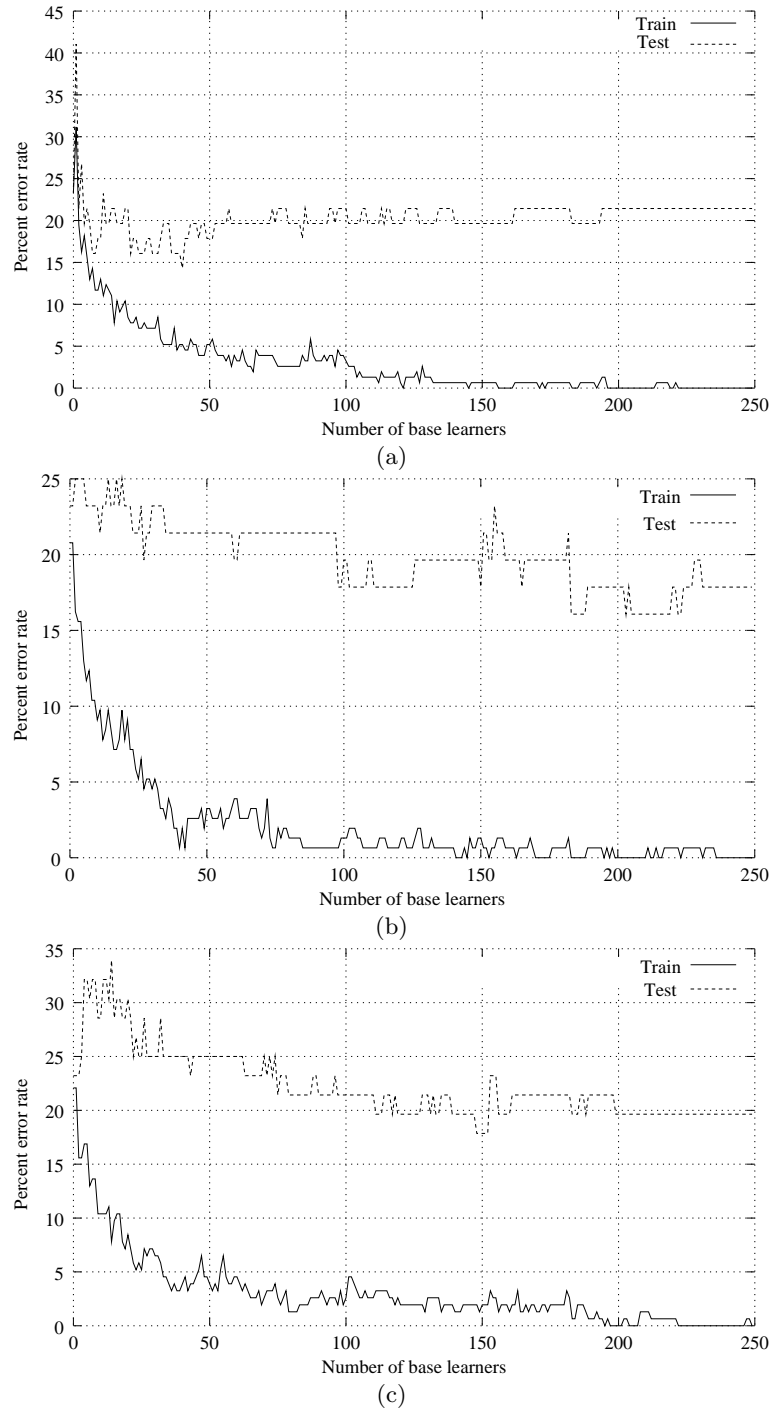
Note that the spiky curves in Fig. 2 and 3 are due to the relative small number of examples in the testing set.

The test error on the monovariety data set decreases slowly compared with the blended data set and using a less complex MLP as base learner the error remains unchanged at about 20% (Fig. 3 (a)). Moreover, the training error drops to zero only after more than 100 rounds of boosting. These results on the monovariety coffee data set can be explained considering three different but correlated items.

First, learning monovariety data could be an hard classification task: for instance, our results obtained using other ensemble methods such as Output Coding decomposition [2] show that it is difficult to shrink the error rate below 20% [10].



**Fig. 2.** Error curves for boosting MLP on the blended coffees data set. The base classifiers are MLP with 5 (a), 7 (b) and 9 (c) hidden units.



**Fig. 3.** Error curves for boosting MLP on the monovariety coffees data set. The base classifiers are MLP with 20 (a), 30 (b) and 40 (c) hidden units.



Second, outliers could be present in the data: the high values of the weights registered in subsets of the data could suggest that some data are difficult to learn, or that they are candidates to be outliers [6]. In fact the PCA plot in Fig. 1 shows that, for class one, a subset of the data is distinctly separated from the others, and this could be related to the fact that, for each class, three carousels of vials were analyzed: it is likely that for one of these carousels the autosampler settings have been changed.

Third, effectiveness of boosting fails when data are corrupted by noise [3]: During the sensitive analytic procedures involved in measurements performed through electronic noses, some noise can be added. The PCA plots in Fig. 1 show that drift phenomena are present.

Boosting enhances classification performances, but it requires training of ensembles of learning machines, with increasing computational costs. However, we need to perform an accurate model selection to achieve good results with a single MLP, and this requires an accurate and time consuming planning of the experimentation, while with boosting also a weak learner not accurately tuned for a particular problem can achieve good generalization results. For instance, in the presented experimentation, the worst boosted MLP achieves better result than the best single MLP, both for blended and monovariety data sets (Tab. 1). Moreover, we can also note that sometimes a remarkable reduction in the test error is reached even after few iterations of the boosting algorithm (Fig. 2), reducing in such a way the computational cost.

## 5 Conclusion

The Electronic Nose represents a new technique for the analysis of odorous mixtures, such as that emitted from coffee. Boosting improved classification performances of electronic noses, reducing in a significant way both the minimum and the average testing error on multiple runs of the boosted ensemble of MLP. Moreover a remarkable reduction of the error is reached even after few iterations of boosting. Even if boosting achieves its best performances with complex algorithms such as C4.5 or backpropagation when there is a reasonably large amount of data available [6], we have halved the testing error on the blended coffee data set, with only 187 training examples. On the other hand the moderate reduction of the test error achieved on the monovariety coffee data set can be explained considering that boosting is especially susceptible to noise and outliers.

## Acknowledgments

This work has been partially funded by *Progetto Finalizzato* CNR-MADESS II, INFN and University of Genova. We thank Davide Della Casa for the implementation of the software applications used in this work.

## References

- [1] G.J. Briem, J.A. Benediktsson, and J.R. Sveinsson. Boosting. Bagging and Consensus Based Classification of Multisource Remote Sensing Data. In *MCS 2001, Cambridge, UK*, volume 2096 of *LNCS*, pages 279–288. Springer-Verlag, 2001.
- [2] T.G. Dietterich. Ensemble methods in machine learning. In *MCS 2000, Cagliari, Italy*, volume 1857 of *LNCS*, pages 1–15. Springer-Verlag, 2000.
- [3] T.G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization. *Machine Learning*, 40(2):139–158, 2000.
- [4] P. Domingos. A Unified Bias-Variance Decomposition and its Applications. In *Proceedings of the 17<sup>th</sup> ICML*, pages 231–238, Stanford, CA, 2000. Morgan Kaufmann.
- [5] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and Systems Sciences*, 55(1):119–139, 1997.
- [6] Y. Freund and R.E. Schapire. Experiments with a new boosting algorithm. In *Proceedings of the 13<sup>th</sup> ICML*, pages 148–156. Morgan Kaufman, 1996.
- [7] C. Furlanello and S. Merler. Boosting of Tree-based Classifiers for Predictive Risk Modeling in GIS. In *MCS 2000, Cagliari, Italy*, volume 1857 of *LNCS*, pages 220–229. Springer-Verlag, 2000.
- [8] Gardner and Bartlett. *Electronic noses*. Oxford University Press, 1999.
- [9] M. Haruno, S. Shirai, and Y. Ooyama. Using decision trees to construct a practical parser. *Machine Learning*, 34:131–149, 1999.
- [10] F. Masulli, G. Valentini, M. Pardo, and G. Sberveglieri. Classification of sensor array data by Output Coding decomposition methods. In *International Workshop MATCHEMS 2001*, pages 169–172, Brescia, Italy, 2001.
- [11] S. Merler, C. Furlanello, B. Larcher, and A. Sboner. Tuning Cost-Sensitive Boosting and its Application to Melanoma Diagnosis. In *MCS 2001, Cambridge, UK*, volume 2096 of *LNCS*, pages 32–42. Springer-Verlag, 2001.
- [12] M. Pardo, G. Niederjaufner, G. Benussi, E. Comini, G. Faglia, G. Sberveglieri, M. Holmberg, and I. Lundstrom. Data preprocessing enhances the classification of different brands of espresso coffee with an electronic nose. *Sensors and Actuators B*, 69, 2000.
- [13] G. Sberveglieri. *Sensors and Actuators B*, 6, 1992.
- [14] R. Schapire and Y. Singer. Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.
- [15] R. Schapire, Y. Singer, and A. Singhal. Boosting and Rocchio applied to text filtering. In *Proceedings of the 11<sup>th</sup> International Conference on Research and Development in Information Retrieval*, 1998.
- [16] R.E. Schapire. A brief introduction to boosting. In *16<sup>th</sup> IJCAI*, pages 1401–1406. Morgan Kaufman, 1999.
- [17] R.E. Schapire, Y. Freund, P. Bartlett, and W. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26(5):1651–1686, 1998.
- [18] H. Schwenk and Y. Bengio. Training methods for adaptive boosting of neural networks. In *Advances in Neural Information Processing Systems*, volume 10, pages 647–653. 1998.
- [19] G. Valentini and F. Masulli. NEUROObjects: an object-oriented library for neural network development. *Neurocomputing*. (in press).