# Online Ensemble Learning

**Nikunj C. Oza**
Computer Science Division
University of California
Berkeley, California 94720–1776
oza@cs.berkeley.edu

Ensemble learning methods train combinations of base models, which may be decision trees, neural networks, or others traditionally used in supervised learning. Ensemble methods have gained popularity because many researchers have demonstrated their superior prediction performance relative to single models on a variety of problems especially when the correlations of the errors made by the base models are low (e.g., (Freund & Schapire 1996; Tumer & Oza 1999)). However, these learning methods have largely operated in batch mode—that is, they repeatedly process the entire set of training examples as a whole. These methods typically require at least one pass through the data for each base model in the ensemble. We would instead prefer to learn the *entire* ensemble in an *online* fashion, i.e., using only one pass through the entire dataset. This would make ensemble methods practical when data is being generated continuously so that storing data for batch learning is impractical, or in data mining tasks where the datasets are large enough that multiple passes would require a prohibitively large training time.

We have so far developed online versions of the popular bagging (Breiman 1994) and boosting (Freund & Schapire 1996) algorithms. We have shown empirically that both online algorithms converge to the same prediction performance as the batch versions and proved this convergence for online bagging (Oza 2000). However, significant empirical and theoretical work remains to be done. There are several traditional ensemble learning issues that remain in our online ensemble learning framework such as the number and types of base models to use, the combining method to use, and how to maintain diversity among the base models.

When learning large datasets, we may hope to avoid using all of the training examples and/or input features. We have developed input decimation (Tumer & Oza 1999), a technique that uses different subsets of the input features in different base models. We have shown that this method performs better than combinations of base models that use all the input features because of two characteristics of our base models: they overfit less by using only a small number of highly-relevant input features, and they have lower correlations in their errors because they use different input feature subsets. However, our method of selecting input features currently examines the entire training set at once, which makes it unsuitable for our online ensemble framework. We are working on extending input decimation to select appropriate feature subsets online. We may also be able to select a small subset of the training examples without a significant degradation in generalization performance. There are many possible reasons for this, such as: the available data may be very dense in the space of possible points, the user may explicitly choose to learn using a subset of the examples (e.g., examples with a particular attribute value), or the user may only need a relatively low-quality solution. For example, the last possibility is often true when learning ensemble models—we often do not need high-performing base models because, if the correlations of their errors are low, then the ensemble will still perform well. We would also like to devise confidence measures for the base models or entire ensemble to help us determine when we have reached a suitable accuracy and; therefore, reduce the number of data points that we have to learn.

We need to formally characterize the performance of our online ensembles relative to batch ensembles and the best base models. We would also like to extend our work to learn time-series data which may have changing underlying statistical properties (e.g., in a factory, there may be 24-hour cycles in the data). This requires that our learning algorithms detect the different regimes in the data and devise base models for each of these regimes online.



## References

Breiman, L. 1994. Bagging predictors. Technical Report 421, Department of Statistics, University of California, Berkeley.

Freund, Y., and Schapire, R. 1996. Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on Machine Learning*, 148–156. Bari, Italy: Morgan Kaufmann.

Oza, N. C. 2000. Learning bagged and boosted ensembles online. Submitted for Publication.

Tumer, K., and Oza, N. C. 1999. Decimated input ensembles for improved generalization. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN-99)*.