The Effect of Distance Metrics on Boosting with Dynamic Weighting Schemes

Xinzhu Yang, Bo Yuan and Wenhuang Liu
Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, P.R. China
yangxz03@mails.tsinghua.edu.cn, {yuanb, liuwh}@sz.tsinghua.edu.cn

Abstract—**This paper presents some preliminary experimental results on RegionBoost, which is a typical example of a class of Boosting algorithms based on dynamic weighting schemes. It is shown that the performance of RegionBoost with the *k*-Nearest Neighbor (kNN) algorithm as the competency predictor of its basic classifiers can be significantly improved on a variety of standard UCI benchmark datasets by using non-Euclidean distance metrics.**

Keywords-boosting; dynamic weighting; RegionBoost; kNN; fractional distance metircs

I. INTRODUCTION

Since ensemble learning can help improve the accuracy of a single learning model, it has been an active topic in supervised learning for more than two decades. An ensemble of classifiers refers to a set of classifiers whose individual decisions are combined in some way to determine the class labels of unknown samples[1]. As a result, how to combine the individual classifiers is a key question in ensemble learning. In the past, many strategies have been developed such as unweighted voting in Bagging[2] and Random Forests[3], weighted voting in Boosting[4], learning a combiner function[5] and stacking[6].

Boosting[4] encompasses a family of successful ensemble mechanisms by sequentially producing a series of classifiers and combining them using weighted voting. The training set used for each classifier is chosen based on the performance of the earlier classifier(s) in the series[7]. AdaBoost[8] is a commonly used version of Boosting algorithms. In the training process, AdaBoost decreases the weights of training samples classified correctly by the former classifier and increases the weights of those classified incorrectly to create the new training set.

When combining the decisions, AdaBoost assigns a single value weight α_t to each basic classifier $h_t(x)$ based on its accuracy on the entire training set at iteration t, which means that the weights of classifiers are constant and will not change for different new samples. However, this approach ignores the inherent performance variability of classifiers on new samples with different features. In fact, the same basic classifier may not be as effective at classifying samples in some areas of the feature space as in other areas.

This issue has been addressed by several improved Boosting algorithms independently, which replace the fixed weighting methods with dynamic weighting schemes, including RegionBoost[7], DynaBoost[9], *i*Boost[10], WeightBoost[11] and local Boost[12]. A crucial factor of these schemes is how to dynamically adjust the weights with regard to the input samples.

Apart from some limited progresses, theoretical analysis on the ensemble mechanisms has shown to be a very challenging task. For the dynamic weighting schemes, both theoretical analysis and comprehensive empirical studies have been rare in the literature. In this paper, an empirical study focusing on Boosting with dynamic weighting schemes, especially on how the distance metrics may affect the performance of the Boosting algorithms is presented.

In the next Section, a brief literature review on Boosting with dynamic weighting schemes is given, with focus on a typical example called RegionBoost. In Section III, the behaviors of the L_p norm distance metric and its extension fractional distance metrics are discussed. Section IV presents the experimental results. This paper is concluded in Section V with a list of directions for future work.

II. DYNAMIC WEIGHTING SCHEMES

A. Framework

One of the major advantages of ensemble learning over a single model is that an ensemble approach allows each basic model to focus on a different aspect of the dataset. When combined together, they are able to explain the whole dataset thoroughly. Therefore, in order to take the full strength of ensemble learning, a good combination strategy should be able to examine the input pattern and only invoke the basic models that are appropriate for the specific input pattern[11].

In order to overcome the limitation of the constant weighting strategies, a number of Boosting methods with dynamic weighting schemes have been proposed independently with proven superior performance over standard Boosting algorithms. However, the essential idea is very similar: the basic classifiers should be weighted based on not only the overall training accuracy but also the sample to be classified. In this paper, we use the term “Boosting with dynamic weighting schemes” to summarize this class of new Boosting algorithms.

Although the implementation details are more or less different, the framework of Boosting with dynamic weighting schemes can be illustrated as Fig. 1. An extra learner $\alpha_t(x)$ is introduced for every classifier $h_t(x)$ as a competency predictor in order to evaluate the dynamic input-dependent weights for each classifier.

For example, in WeightBoost, the weights are determined by an exponential function of the combination of all the former classifiers $\alpha_t(x) = \alpha_t e^{-|\beta H_{t-1}(x)|}$.

More generally, an extra learner is trained to simulate the performance of each basic classifier on the sample space. In other words, this extra learner is used to indicate whether a certain basic classifier is likely to yield accurate results given an unknown sample. Most of the common classification methods have been employed as the competency predictors such as kNN (RegionBoost), Neural Networks (RegionBoost) and Decision Trees (*i*Boost).

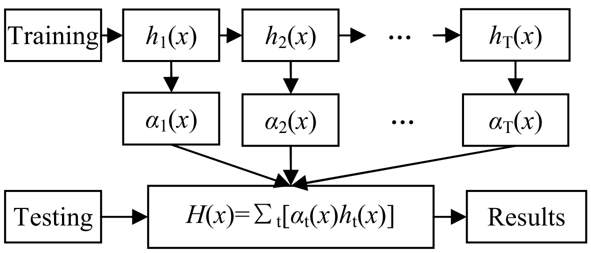


Figure 1. The framework of Boosting with dynamic weighting schemes.

It is clear that one of the key issues in Boosting with dynamic weighting schemes is how to construct the competency predictors in order to appropriately determine the weights, which can significantly affect the performance of the combined classifiers.

B. *RegionBoost*

In this paper, we used RegionBoost as the representative example of Boosting with dynamic weighting schemes in the experimental studies. The main idea behind RegionBoost is to build an extra model upon each basic classifier based on its training results (i.e., whether a training sample is classified correctly or not). By doing so, this new model is able to estimate the accuracy or the competency of the classifier for each new sample.

One of the intuitive approaches to estimating model competency is to find the k points in the training set nearest to the new sample to be classified and then use the performance of each classifier on these k points as the measurement[7]. More specifically, the weight of each classifier is determined by the percentage of points (out of k points) correctly classified by this classifier [7].

As a result, how well this kNN model can represent the performance of a certain classifier on the new example is vital to the success of this dynamic weighting approach. Since kNN is a technique explicitly relies on the distance measure defined over the sample space, the choice of different distance metrics may significantly affect the performance of kNN, and subsequently the accuracy of RegionBoost. In the original RegionBoost, the Euclidean distance is used as a natural choice.

III. Distance Metrics

In this paper, we focused on the behavior of RegionBoost with the L_p norm and its extensions. The definition of the L_p norm is:

$$L_p(x,y)=\sum_{i=1}^{d}(\|x^i-y^i\|^p)^{1/p},\quad x,y\in R^d,\ p\in Z. \tag{1}$$

It is easy to see that the L_2 norm is the most commonly used Euclidean distance metric while the L_1 norm is the Manhattan distance metric.

Recent research shows that, in high dimensional spaces, the validity of the L_p norm in measuring the similarity between data points is sensitive to the value of p. For example, the Manhattan distance metric (L_1 norm) is consistently more preferable than the Euclidean distance metric (L_2 norm) for high dimensional data mining applications[13]. Furthermore, a natural extension of the L_p norm to fractional distance metrics is introduced and examined from both the theoretical and empirical perspectives[13].

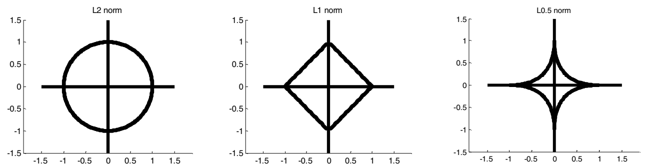
The fractional distance metric is defined as:

$$L_f(x,y)=\sum_{i=1}^{d}[(x^i-y^i)^f]^{1/f},\quad x,y\in R^d,\ f\in(0,1). \tag{2}$$

In order to give a better image of the behavior of the fractional distance metrics, Fig. 2 shows the unit spheres for different distance metrics in the 2D space.

From the theoretical perspective, fractional distance metrics provide better divergence between the maximum and minimum distances to a given query point than integral distance metrics. This feature makes a proximity query more meaningful and stable. Empirical studies also demonstrate that fractional distance metrics can significantly improve the effectiveness of some standard classification and clustering algorithms such as kNN and k-Means on high dimensional datasets[13]. In the meantime, fractional distance measures have been applied to content-based image retrieval and the experiments also show that retrieval performances of these measures consistently outperform the Manhattan and Euclidean distance metrics when used with a wide range of high-dimensional visual features[14].

Since the fractional distance metrics can enhance the performance of kNN as a classifier, it is straightforward to raise a question: whether it can also improve the accuracy of Boosting algorithms such as RegionBoost that use kNN as the competency predictor. This issue will be examined in details in the next section.



L_2 norm L_1 norm $L_{0.5}$ norm

Figure 2. Unit spheres for L_2, L_1 and $L_{0.5}$ norm distance metrics (2D).

IV. EXPERIMENTS

This section presents the empirical studies on RegionBoost with different distance metrics. Unlike the original RegionBoost, we used Decision Trees as the basic classifiers instead of Neural Networks used in the original RegionBoost in order to reduce the computational complexity. In each iteration, the resampling mechanism was employed to change the distribution of samples in the training set.

Ten datasets from the UCI machine learning repository[15] were used in the experiments, including both high and low dimensional ones. The details of these datasets are shown in TABLE I.

In the experiments, all datasets were normalized to the range [0, 1] in each dimension and the reported results were averaged over five independent 5-fold cross validation trials. For each experiment, an ensemble of 100 trees was created and kNN (k=5) was used to determine the dynamic weights for each basic classifier. Note that what the fractional distance metrics actually changed was the nearest neighborhood of each query point.

TABLE I. A SUMMARY OF DATASETS

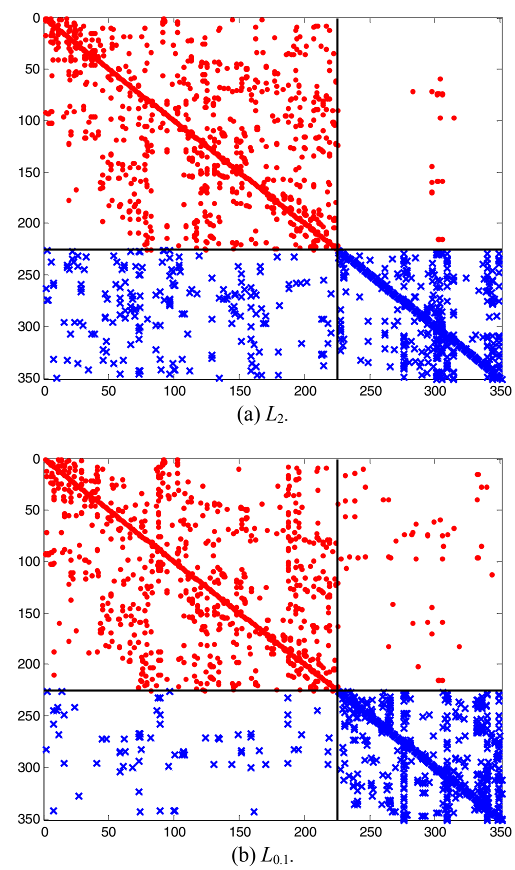
Dataset	Attributes	Samples	Class
Musk	168	476	2
Sonar	60	208	2
Ionosphere	34	351	2
Breast	30	569	2
Credit Ger	24	1000	2
Segmentation	19	2310	7
Credit Aus	14	690	2
Heart	13	270	2
Pima	8	768	2
Bupa	6	345	2

The error rates of the standard AdaBoost (the same as the RegionBoost except the weighting scheme), RegionBoost with different distance metrics and kNN (used as the benchmark) are shown in TABLE II. For RegionBoost, results with L_2, L_1, $L_{0.5}$, $L_{0.25}$, and $L_{0.1}$ are presented. For kNN, only the best results obtained from the five different distance metrics are presented. Several interesting phenomena can be observed from these experimental results.

Firstly, as far as the classification accuracy is concerned, after 100 iterations, RegionBoost outperformed both AdaBoost and kNN on 7 out of 10 data sets. More importantly, in 5 out of these 7 occasions, RegionBoost was employed with fractional distance metrics. This evidence shows clearly that fractional distance metrics may change the performance of kNN as a competency predictor of RegionBoost and consequently help improve the accuracy of RegionBoost.

In order to illustrate the effect of fractional distance metrics on kNN, we applied one of the common visualization techniques, shaded similarity matrix [16], which has long been used in hierarchical cluster analysis, and has also been successfully used for visualizing the nearest neighbor classification method recently.

In a similarity matrix, given the k nearest neighbors of the i^{th} sample j_1, …, j_k, only the cells (i, j_1), …, (i, j_k) are displayed[16]. Fig. 3 shows an example based on the Ionosphere dataset where '.' and 'x' represent the neighbors of the two classes of samples respectively (samples were already sorted based on their class labels). Ideally, all the neighbors of a sample should be within the same block. In Fig. 3a (Euclidean distance), there are 213 outliers, compared to 125 outliers in Fig. 3b (fractional distance). The corresponding kNN error rates were 0.1561 vs. 0.0969. These results give the evidence that fractional distance metrics are able to change the neighborhood of query samples, and improve the weighting accuracy of kNN as well as the performance of RegionBoost.

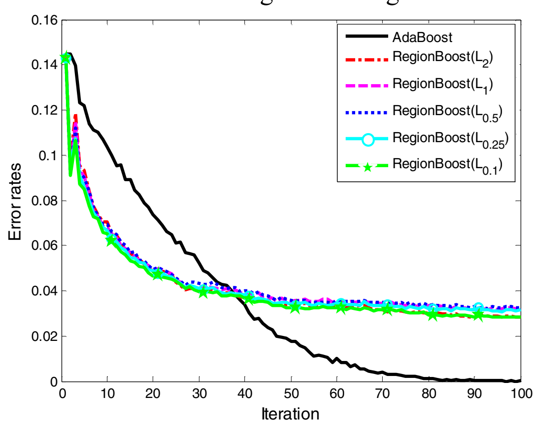


(a) L_2.

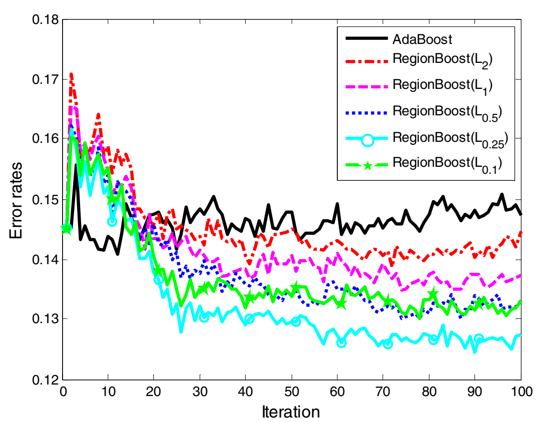
(b) $L_{0.1}$.

Figure 3. Shaded similarity matirx on Ionosphere.

Secondly, by comparing the training error rates between AdaBoost and RegionBoost, it is clear that, in AdaBoost, the training error can often keep decreasing during iterations and finally reach a level close to zero, which is consistent with the theoretical analysis. On the contrary, the training error of RegionBoost may reach a stable level well above zero without any further significant improvement. However, the generalization error rates of AdaBoost and RegionBoost had almost the same trends and RegionBoost behaved even better on most datasets. An example on the Credit_Aus dataset is shown below in Fig. 4a and Fig. 4b.



(a) Training error rates on Credit_Aus



(b) Test error rates on Credit_Aus

Figure 4. Error rates of Adaboost and RegionBoost on Credit_Aus.

Thirdly, RegionBoost has an extra parameter k whose choice is mostly based on experiments. Although the particular value (k=5) used in this paper worked reasonably well, additional experiments may need to be conducted to further investigate its influence on the performance of RegionBoost.

Finally, although the motivation of fractional distance metrics is to tackle the issue associated with high dimensional datasets, in our experiments, there has not been a clear pattern between the dimensionality of datasets and the performance of RegionBoost. For example, The Musk dataset has the highest dimensionality 168 but RegionBoost achieved its best performance with L_1. By contrast, RegionBoost worked best with $L_{0.25}$ on the Heart dataset, which has only 13 features.

V. Conclusions and Future Work

In this paper, we empirically investigated the performance of a typical Boosting algorithm with dynamic weighting schemes called RegionBoost, which employs kNN as the competency predictor of its basic classifiers. The major motivation was to examine the influence of different distance metrics on kNN as well as RegionBoost.

Experimental results showed that the performance of RegionBoost was improved with non-Euclidean distance metrics on several standard UCI datasets. This finding serves as evidence that the choice of distance metrics may play an important role in RegionBoost and potentially many other Boosting algorithms with dynamic weighting schemes.

In addition to the preliminary results presented here, there is still plenty of room for future research. For example, the interaction between the properties of the dataset and the dynamic weighting schemes needs better understanding, which may help discover why RegionBoost may not perform as well as (if not better than) the standard AdaBoost on some other datasets. In the meantime, the influence of distance metrics on RegionBoost with competency indicators other than kNN is also worth investigation.

References

[1] T. G. Dietterich, “Machine Learning Research: Four Current Directions,” AI Magazine, vol. 18, 1997, pp. 97-136.

[2] L. Breiman, “Bagging Predictors,” Machine Learning, vol. 24, 1996, pp. 123-140.

[3] L. Breiman, “Random Forests,” Machine Learning, vol. 45, 2001, pp. 5-32.

[4] R. E. Schapire, “The Strength of Weak Learnability,” Machine Learning, vol. 5, 1990, pp. 197-227.

[5] M. I. Jordan and R. A. Jacobs, “Hierarchical Mixtures of Experts and the EM Algorithm,” Neural Computation, vol. 6, 1994, pp. 181-214.

[6] K. M. Ting and I. H. Witten, “Issues in Stacked Generalization,” Journal of Artificial Intelligence Research, vol. 10, 1999, pp. 271-289.

[7] R. Maclin, “Boosting Classifiers Regionally,” Proc. The 15th National Conference on Artificial Intelligence, Madison, WI, 1998, pp. 700-705.

[8] Y. Freund and R. E. Schapire, “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting,” Journal of computer and system sciences, vol. 55, 1997, pp. 119-139.

[9] P. Moerland and E. Mayoraz, “DynaBoost: Combining Boosted Hypotheses in a Dynamic Way,” Technical Report, IDIAP-RR, Switzerland, 1999.

[10] S. Kwek and C. Nguyen, “iBoost: Boosting Using an Instance-Based Exponential Weighting Scheme,” Proc. The 13th European Conference on Machine Learning, Helsinki, Finland, 2002, pp. 245-257.

[11] R. Jin, Y. Liu, L. Si, J. Carbonell and A. G. Hauptmann, “A New Boosting Algorithm Using Input-Dependent Regularizer,” Proc. the 20th International Conference on Machine Learning, Washington, DC, 2003.

[12] C.-X. Zhang and J.-S. Zhang, “A Local Boosting Algorithm for Solving Classification Problems,” Computational Statistics & Data Analysis, vol. 52, 2008, pp. 1928-1941.

[13] C. C. Aggarwal, A. Hinneburg and D. A. Keim, “On the Surprising Behavior of Distance Metrics in High Dimensional Space,” Proc. the

8th International Conference on Database Theory, London, UK, 2001.

[14] P. Howarth and S. Rüger, “Fractional Distance Measures for Content-Based Image Retrieval,” Proc. the 27th European Conference on Information Retrieval, Santiago de Compostela, Spain, 2005, pp. 447-456.

[15] A. Asuncion and D. J. Newman, “UCI Machine Learning Repository [http://www.ics.uci.edu/~mlearn/ MLRepository.html].” Irvine, CA: University of California, School of Information and Computer Science, 2007.

[16] J. Wang, B. Yu and L. Gasser, “Classification Visualization with Shaded Similarity Matrix,” Technical Report, GSLIS University of Illinois at Urbana-Champaign, 2002.

TABLE II. RESULTS OF EXPERIMENTS OF ADABOOST, REGIONBOOST AND KNN WITH DIFFERENT DISTANCE METRICS

Dataset	**Iteration**	**AdaBoost**	**RegionBoost**					**kNN (*k*=5)**
			L_2	L_1	$L_{0.5}$	$L_{0.25}$	$L_{0.1}$	
Musk	50	**0.0937**	0.0939	0.0954	0.1006	0.1038	0.1021	0.1664 ($L_{0.1}$)
	100	0.0832	0.0878	**0.0826**	0.0912	0.1009	0.1021	
Sonar	50	0.1768	0.1364	**0.1354**	0.1365	0.1433	0.1414	0.1709 ($L_{0.1}$)
	100	0.1500	0.1279	0.1250	0.1212	**0.1202**	**0.1202**	
Ionosphere	50	0.0707	0.0798	0.0758	0.0712	0.0701	**0.0673**	0.0969 ($L_{0.1}$)
	100	0.0672	0.0781	0.0775	0.0707	0.0690	**0.0667**	
Breast	50	0.0593	0.0564	0.0545	0.0545	0.0560	0.0574	**0.0535 (L_2)**
	100	0.0613	0.0545	0.0564	0.0574	0.0536	0.0555	
Credit Ger	50	0.2864	0.2665	0.2705	0.2621	**0.2620**	0.2625	0.2772 ($L_{0.5}$)
	100	0.2778	0.2734	0.2773	0.2713	**0.2679**	0.2709	
Segmentation	50	0.1243	0.1029	0.0981	0.1029	0.1033	0.1043	**0.0938 (L_1)**
	100	0.1386	0.1048	0.1048	0.1100	0.1081	0.1105	
Credit Aus	50	0.1478	0.1451	0.1406	0.1339	**0.1296**	0.1338	0.1301 ($L_{0.25}$)
	100	0.1472	0.1446	0.1372	0.1330	**0.1275**	0.1330	
Heart	50	0.2104	0.1815	0.1852	0.1815	0.1793	**0.1785**	0.1837 ($L_{0.25}$)
	100	0.2193	0.1881	0.1896	0.1837	**0.1830**	0.1844	
Pima	50	0.2596	**0.2545**	0.2631	0.2616	0.2733	0.2767	0.2628 (L_1)
	100	0.2622	**0.2562**	0.2616	0.2605	0.2740	0.2742	
Bupa	50	0.2864	0.2664	0.2687	0.2626	0.2571	0.2594	**0.2481 ($L_{0.25}$)**
	100	0.2991	0.2649	0.2620	0.2672	0.2510	0.2568	