

Learning through Changes: An Empirical Study of Dynamic Behaviors of Probability Estimation Trees

Kun Zhang¹, Zujia Xu², Jing Peng¹, Bill Buckles¹

¹Electrical Engineering and Computer Science Department, Tulane University, New Orleans
zhangk, jp, buckles@eecs.tulane.edu

²Computer Science Department, Dillard University, New Orleans, zxu@dillard.edu

Abstract

In practice, learning from data is often hampered by the limited training examples. In this paper, as the size of training data varies, we empirically investigate several probability estimation tree algorithms over eighteen binary classification problems. Nine metrics are used to evaluate their performances. Our aggregated results show that ensemble trees consistently outperform single trees. Confusion factor trees (CFT) register poor calibration even as training size increases, which shows that CFTs are potentially biased if data sets have small noise. We also provide analysis on the observed performance of the tree algorithms.

1. Introduction

Given sufficient training examples, many classifiers perform quite well if test points are also drawn i.i.d. from the same distribution as the training data. However, in many real-world applications, there may never be enough training examples due to practical factors. Known as a soft classifier, probability estimation trees (PETs) are trained decision trees that generate class membership probabilities for a test point. In cost sensitive learning, PETs are widely used and various algorithms have been proposed for better ranking. In this paper, we systematically compare two recent PETs, i.e. random decision tree (RDT)[3] and confusion factor tree (CFT)[7] with their conventional counterparts, C4.5 [1] and its variations, C4.4 [2], baggedC4.4 [2] and baggedC4.5 on eighteen UCI binary data sets. For each set, we gradually vary the training size, build a tree, and then evaluate the obtained tree on the test examples. Nine evaluation metrics are used to gauge the aggregated behaviors of the above trees: error rate, mean squared error (MSE), cross-entropy, calibration, refinement, resolution, area under ROC curves (AUC), area under lift charts (AULC) and precision. Our motivation for conducting such large-scale experiments is as follows:

1). Intuitively, learning curves of classifiers [6] will always improve as training size increases. But is this valid for different evaluation metrics? Even if this holds, beyond the general trend, what are the relative ranks among these PETs with respect to these metrics?

2). K-fold cross-validation and random splits of training and test sets in terms of 60%:40% ratio are the most common methods for empirical data set generation. A classifier's performance tends to be evaluated vertically at such a split against others. However, it is uninformative about how the classifier itself behaves across different split ratio horizontally, especially with respect to multiple criteria.

3). Some PETs are designed for certain criteria. CFT is designed for optimizing AUC value, while RDT[4] focuses on optimality of probability estimation. How do these trees perform on other standards?

4). Different evaluation metrics have their own merits and weaknesses, and it is not unusual to expect a classifier which is optimal on one metric to be inferior on another. However, to what extent do these metrics agree with each other for a specific PET?

To best of our knowledge, this is the first comprehensive empirical investigation which takes into account above considerations, typically for PETs.

2. Notation and Evaluation Metrics

For a random vector of features x , a classifier is a mapping from x to a class $y \in \{0,1\}$. Given a test point $\{x_i, y_i\}$, $\hat{p}(y_i | x_i)$ represents the predicted probability of the true a-posteriori probability $p(y_i|x_i)$. Class 1 is referred as positive class and class 0 as negative class. We consider here two types PETs, a single tree estimator and an ensemble of single trees. C4.5, C4.4 and CFT are representatives of single PETs. Bagged trees and RDT are ensembles.

Assume all the data sets are completely deterministic and noise free, $p(y_i|x_i)$ is 1 if the label of x_i is y_i and 0 otherwise. We calculate MSE as $1/N \sum_{i=1}^N (p(y_i | x_i) - \hat{p}(y_i | x_i))^2$ and cross entropy as $-1/N \sum_{i=1}^N (p(y_i | x_i) \log \hat{p}(y_i | x_i) + (1 - p(y_i | x_i)) \log(1 - \hat{p}(y_i | x_i)))$. N is the size of test set. Error rate is obtained by thresholding the predicted probabilities at 0.5. Let $\hat{p}_i = \hat{p}(y_i = 1 | x_i)$. Of the N probabilities \hat{p}_i , suppose we estimate k distinct probabilities \hat{p}_j for $j=1, \dots, k$. N_j is the number of predictions having the same value \hat{p}_j , and r_j is the fraction of N_j points whose true labels are positive. Thus, MSE can be decomposed as follows:

$$1/N \left(\sum_{j=1}^k N_j (r_j - \hat{p}_j)^2 + \sum_{j=1}^k N_j r_j (1 - r_j) \right) \quad (1)$$

The first term is known as calibration and the second one is refinement. Please refer [12] for the details and [13] for definition of resolution. To calculate calibration, we use the binning method [4] to discretize the predicted probabilities. The number of bins changes as the test set size reduces gradually. We follow Shapiro et al [10] to compute AULC and Fawcett [11] to calculate AUC. Precision is defined as the fraction of positive predictions that are correctly classified.

3. Experiment Designs

Eighteen UCI binary data sets [9], including adult, mushroom, spam, chess, hypothyroid, sick-euthyroid, tic, pima, breast cancer-wisc, australian, breast cancer-wdbc, housevote, ionosphere, spectf, liver, spect, sonar, and hepatitis, are used as test beds. For each set, we evaluate the following five ratios between training and test data sizes in terms of percentage of the total data points: 10:90, 20:80, 40:60, 60:40 and 80:20. Average results over twenty runs are reported for each PET at each ratio. The minority class is treated as the positive class and the natural class distribution is assumed.

For C4.5s, both the pruned (C4.5P) and unpruned (C4.5UP) trees are investigated. FullC4.5 is built without collapsing and pruning. Four variations of CFTs, pruned (CFT.LC.P) and unpruned (CFT.LC.UP) CFTs with laplace correction, as well as pruned (CFT.FE.P) and unpruned (CFT.FE.UP) CFTs with frequency estimation, are constructed. As in [7], we set the confusion factor to be 0.3. In addition to RDT with half depth (RDTH), we also implement RDT with full depth (RDTF). Each ensemble includes thirty trees as base learners.

4. Experimental Results and Discussion

For a specific metric, we aggregate experiment results over eighteen sets, not just a single domain, to access the overall general trend of each PET. The legends are consistent across all figures.

4.1 MSE, Cross Entropy and Error Rate

Figure 1 shows the learning curves for MSE, cross entropy and error rate. The overall relative ranks and changing behaviors of baggedC4.4, RDTH, C4.4 and CFTs in the MSE plot are quite similar to that in the cross entropy plot. Cross entropy values are unavailable for baggedFullC4.5, RDTF and C4.5s since $\hat{p}(y_i | x_i)$ can be 0. Ensembles are obviously better than single trees. On average, MSE decreases from 14.9% for the poorest tree, CFT.LC.UP, to 8.7% for the best tree, baggedFullC4.5. Cross entropy decreases from 69.1% for the poorest tree, CFT.LC.UP, to 41.5% for the best tree, baggedC4.4. In ensembles, bagged trees outperform RDTs and RDTF is superior to RDTH.

Among single trees, C4.4 performs the best, followed by C4.5s and CFTs. Since the only difference between C4.4 and FullC4.5 is the probability estimation methods, this indicates that laplace correction results in lower MSE than frequency counts [8]. However, once bagging is used, the influence of laplace correction is dramatically reduced. The average MSE difference between these two bagged trees is a mere 0.14%. For error rate, bagged trees once again dominate other methods. RDTF is superior to RDTH, which is consistent with [5]. C4.5P is the best among single trees. The changes in the ranks indicate that error rate, more or less, deviates from MSE and cross entropy.

4.2 Calibration, Refinement and Resolution

Figure 2 presents the learning curves of the PETs for calibration, refinement and resolution. In the calibration plot, CFTs have the largest calibration errors and their curves consistently go upward as the training size increases. The curves of bagged trees and RDTs exhibit a slightly upward trend. C4.4 and C4.5s' curves first decrease, then keep steadily. In the refinement plot, each curve for individual PET decreases monotonically. This indicates that the MSE reductions of these PETs, especially CFTs, are achieved mainly through reduction in refinement. BaggedFullC4.5 has the smallest calibration and refinement error. Although the calibration errors of RDTs are close to C4.4 and C4.5s, their much lower refinement value makes their MSE still better than single trees. C4.4 has the smallest MSE in single trees. This is because its calibration and refinement error are the lowest. C4.5s have the poorest refinement values, but this deficiency is compensated by their much smaller calibration errors compared with CFTs. Therefore, their MSEs are still better than CFTs. If we ignore the absolute scales of refinement and resolution, the resolution plot actually mimics the refinement plot if we project refinement with respect to the x axis.

But why does refinement decrease as the training set grows? And why do the curves in calibration plot assume such different appearances? In the following, we answer the two questions with the notation and definitions specified in section 2.

Recall from equation (1), refinement achieves its minimum value 0 when all of the examples in bin j are either positive or negative. And it reaches its maximum value 0.25 if there is equal number of positive and negative examples. Given total k bins, let \hat{p}_j be the mean probability value of bin j and $0 < \hat{p}_1 < \hat{p}_2 < \dots < \hat{p}_j < \dots < \hat{p}_k < 1$. We have $0 \leq r_1 \leq r_2 \leq \dots \leq (r_{(k+1)/2} \approx 0.5) \leq \dots \leq r_k \leq 1$. Generally, as the training size increases, a classifier can identify more rules with higher confidence. Its predicted probabilities

for each test case will gradually shift to 0 or 1, and the number of probabilities between 0 and 1, especially around the random guess will decrease. During this shifting, r_j will approach 0 for bin_j if \hat{p}_j is close to 0.

For bin_j whose \hat{p}_j is close to 1, r_j will approach 1.

(Note that \hat{p}_j also changes a very small amount). In the extreme case, all of the examples will fall into the bin_1 with $r_1=0$ and bin_k with $r_k=1$ if the classifier can always issue probability which exactly matches the truth. At this point, refinement is 0.

As defined in [7], CFTs estimate probability for a test point by aggregating the contributions of all leaves, not just the target leaf where the test point falls. Since the tree size always increases with the growth of training set, the contributions from the non-target leaves account for a larger proportion while the target leaf contributes less. If the inherent noise of a data set is small or even free, CFTs will affect the probability estimation by taking the contributions of more unrelated leaves into account, thereby causing the calibration to degrade.

Indicated by MSE plot, ensembles can issue more accurate predictions than single trees. With the expansion of training set, these better classifiers can produce quite good predictions for test cases. This increases the fraction of the number of cases in bin_1 and bin_k while decreasing that of bins in between. Since the majority of the test cases are in bin_1 and bin_k , we can show that, as the training size increases, the calibration error of bin_1 , $(\hat{p}_1 - r_1)^2$, will become larger. Here, we use $(\cdot)_0$ to denote the variables at step 0 when the training size is small, and $(\cdot)_1$ represents variables at step 1 at which training size is increased. Since the classifiers improve predictions by moving the positive examples out of bin_1 or by moving negative cases into bin_1 , we will have $(\hat{p}_1)_0 \approx (\hat{p}_1)_1 = \hat{p}_1$, $(r_1)_0 > (r_1)_1$, and $\hat{p}_1 > (r_1)_1$. As $(r_1)_0 \approx \hat{p}_1$, we have $(r_1)_1 < (r_1)_0 < 2\hat{p}_1 - (r_1)_1$, thus $(\hat{p}_1 - (r_1)_0)^2 < (\hat{p}_1 - (r_1)_1)^2 + (\hat{p}_1 - r_k)^2$, it can be shown, also increases in the same way. The similar explanation can be applied to C4.4 and C4.5s.

4.3 AUC, AULC and Precision

Figure 3 shows the learning curves for AUC, AULC and precision. In the AUC plot, ensembles dominate the space, followed by CFTs, C4.4 and C4.5s. The curves of bagged trees and RDTs are almost identical. In CFTs, unpruned trees perform better than pruned trees and laplace correction results in larger AUC than frequency estimation. This observation is in accord with [2], but slightly differs with [7]. C4.5s's AUC values are quite poor. The average AUC increment

from 0.839 for C4.5UP to 0.912 for baggedC4.4 is 9.6%. In the AULC plot, baggedC4.4 is still the best. For the single trees, CFT.LC.UP achieves the largest AULC value. In CFTs, unpruned trees beat pruned tree and laplace correction leads to a larger AULC. RDTF is superior to other PETs with overall precision value of 0.831. The average precision increment from 0.711 for CFT.FE.P, the poorest one, to RDTF is 17%. Among the single trees, C4.5P has the largest value.

5. Conclusions

We have carried out an empirical comparison of several PET algorithms with respect to a varying training size using nine metrics. By responding to the hypotheses proposed in section 1, we reach the following conclusions:

(1).Ensembles consistently outperform single trees on all of the nine metrics. Bagged trees are better than RDT on seven metrics, the exception being AUC and precision. RDT with full depth exhibits better overall performance than half depth tree, which is in accord with [3-5]. (2).Among single trees, C4.4 performs best on MSE, cross entropy, calibration, refinement and resolution. Pruned C4.5 is superior for error rate and precision. Unpruned CFT with laplace correction has the advantage on AUC and AULC. (3).As training size increases, the learning curves of these trees improve for all metrics but calibration. (4).MSE and cross entropy are quite consistent with each other; MSE can be better understood through decomposition; AUC and AULC are nearly isomorphic rank metrics. In summary, this paper can be viewed as a supplement of what has been done in [2-8]. Future work will focus on correcting some PETs' deficiencies through algorithm modification.

6. References

- [1].J.R.Quinlan. C4.5: programs for empirical learning. Morgan Kaufmann,1993
- [2]F.Provost and P.Domingos."Tree induction for probability based rankings".Machine Learning, 52(3), 2003, pp. 199-215
- [3] W.Fan, H.X.Wang, P.S.Yu and S.Ma ,"Is random model better? On its accuracy and efficiency",ICDM2003, pp.51-58
- [4] W.Fan, "On the optimality of probability estimation by random decision trees", AAAI 2004, pp.336-341
- [5] F. Liu, K. M. Ting, W. Fan, "Maximizing tree diversity by building complete-random decision trees. PAKDD2005
- [6] G.Weiss and F.Provost, "Learning when training data are costly: the effect of class distribution on tree induction", JAIR, 19, 2003, pp.315-354
- [7] C.Ling and R.Yan, "Decision Tree with better ranking", ICML,2003
- [8]J.Bradford,C.Kunz,R.Kohavi,C.Brunk,C.Brodley, Pruning decision trees with misclassification costs", ECML98
- [9]C.Blake and C.Merz, "UCI repository of machine learning database", UCI, 1998
- [10].G. P.Shapiro and B. Masand. "Estimating campaign benefits and modeling lift". KDD1999.

[11].T.Fawcett. "ROC graphs: notes and practical considerations for data mining researchers". HPL-2003-4.
 [12]M.DeGroot and S.Fienberg, "Assessing probability assessors: calibration and refinement", Statistica Decision Theory and Related Topics III, Vol 1, 1982, pp. 291-314

[13]A.H.Murphy, "A new vector partition of the probability score", J. Appl. Met.,12, 1973, pp.534-537

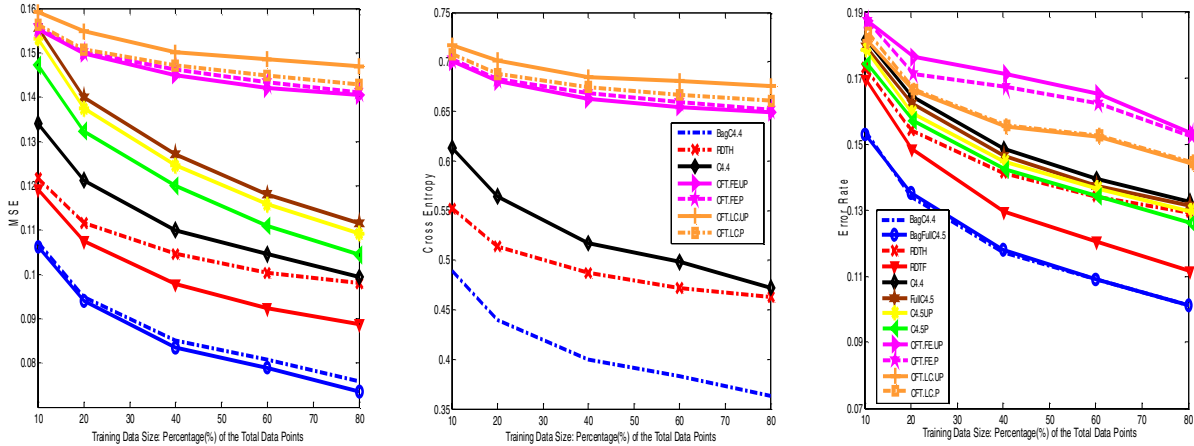


Figure 1: From left to right, learning curves for MSE, cross entropy and error rate as training set size increases

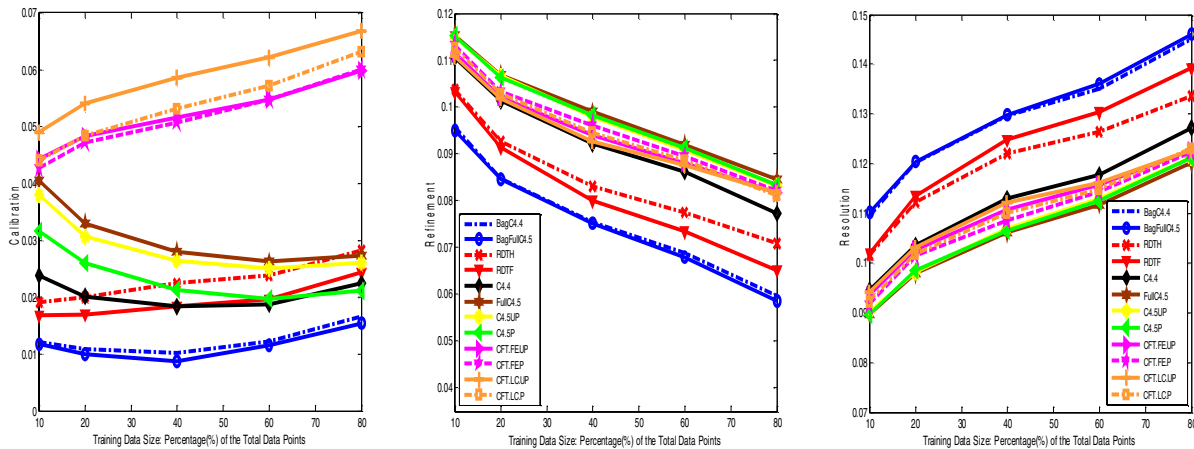


Figure 2: From left to right, learning curves for calibration, refinement and resolution as training set size increases

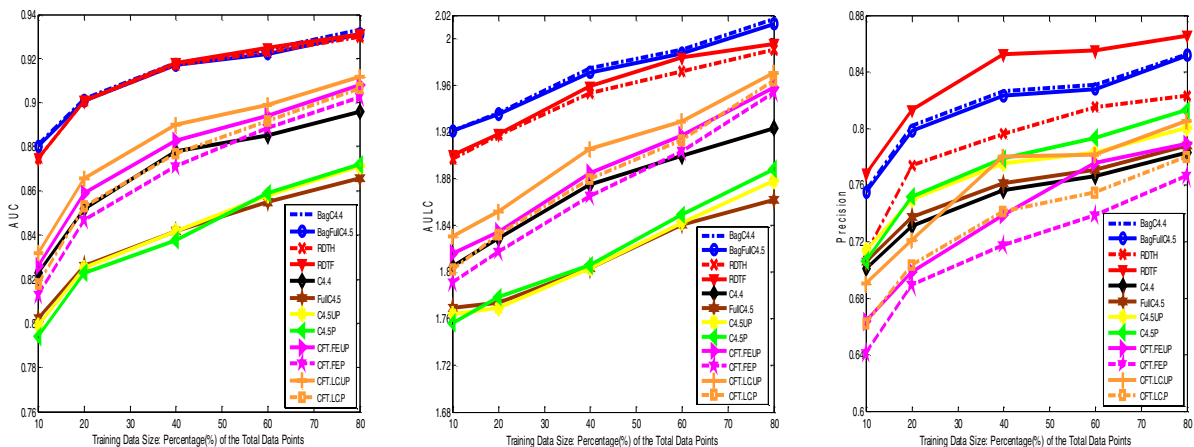


Figure 3: From left to right, learning curves for AUC, AULC and precision as training set size increases