A Bayesian Metric for Evaluating Machine Learning Algorithms

Lucas R. Hope and Kevin B. Korb

School of Computer Science,
and Software Engineering,
Monash University,
Clayton, VIC 3168, Australia
{lhope, korb}@csse.monash.edu.au

Abstract. How to assess the performance of machine learning algorithms is a problem of increasing interest and urgency as the data mining application of myriad algorithms grows. Rather than predictive accuracy, we propose the use of information-theoretic reward functions. The first such proposal was made by Kononenko and Bratko. Here we improve upon our alternative Bayesian metric, which provides a fair betting assessment of any machine learner. We include an empirical analysis of various Bayesian classification learners.

Keywords: Predictive accuracy, Bayesian evaluation, information reward.

1 Introduction

As interest in machine learning and data mining grows, the problem of how to assess machine learning algorithms becomes more urgent. The standard practice for supervised classification learners has been to measure predictive accuracy (or its dual, error rate) using a fixed sample divided repeatedly into training and test sets, accepting a machine learner as superior to another if its predictive accuracy passes a statistical significance test. This represents an improvement over historical practices, particularly when the statistical dependencies introduced by resampling are taken into account (cf. [1, 2]).

Nevertheless, there are a number of objections to the use of predictive accuracy, the most telling being that it fails to take into account the uncertainty of predictions. For example, a prediction of a mushroom's edibility with a probability of 0.51 counts exactly the same as a prediction of edibility with a probability of 1.0. We might rationally prefer to consume the mushroom in the second case. Predictive accuracy shows no such discernment. According to common evaluation practice, every correct prediction is as good as every other. Hence, we advocate that classification learners should be designed, or redesigned, so as to yieled probabilistic predictions rather than catagorical predictions.

G.I. Webb and Xinghuo Yu (Eds.): AI 2004, LNAI 3339, pp. 991–997, 2004.

We believe a cost-sensitive assessment, favouring the machine learner which maximizes expected reward is, in principle, the best way of evaluating learning algorithms. Unfortunately, finding appropriate cost functions is often difficult or impossible. Provost and Fawcett [3] use receiver operating characteristic (ROC) convex hulls for evaluation independent of cost functions. This has the useful meta-learning feature of selecting the best predictor for a given performance constraint, in the form of a selected false negative classification rate. Unfortunately, the ROC curves underlying this method again ignore the probabilistic aspect of prediction, as does predictive accuracy simpliciter.

Here we examine metrics which attend to the probability of a classification, namely information-theoretic measures and in particular, *information reward* (*IR*). We illustrate its application by comparing Naive Bayes with other classification learners, contrasting *IR* with predictive accuracy assessments.

2 Information-Theoretic Metrics

2.1 Good's Information Reward

The original information reward (*IR*) was introduced by I.J. Good [4] as *fair betting fees* — the cost of buying a bet which makes the expected value of the purchase zero. Good's *IR* positively rewarded binary classifications which were informative relative to a uniform prior. *IR* is split into two cases: that where the classification is correct, indicated by a superscripted '+', and where the classification is incorrect, indicated by a superscripted '−'.

Definition 1. *The IR of a binary classification with probability p' is*

$$I^+ = 1 + \log_2 p' \qquad \textit{(for correct classification)} \tag{1a}$$

$$I^- = 1 + \log_2(1 - p') \quad \textit{(for misclassification)} \tag{1b}$$

IR has the range $(-\infty, 1)$. For successful classification, it increases monotonically with p', and thus is maximized as p' approaches 1; for misclassification, *IR* decreases monotonically. While the constant 1 in (1a) and (1b) is unnecessary for simply ranking machine learners, it makes sense in terms of fair fees. When the learner reports a probability of 0.5, it is not communicating any *information* (given a uniform prior), and thus receives a zero reward.

Our work generalizes Good's to multinomial classification tasks, while also relativizing the reward function to non-uniform prior probabilities.

2.2 Kononenko and Bratko's Metric

The measure introduced by Kononenko and Bratko [5] also relativizes reward to prior probabilities. Furthermore, it too is nominally based upon information theory. This foundation is seriously undermined, however, by their insistance that when a reward is applied to a correct prediction with probability 1 and an incorrect prediction also with probability 1, the correct and incorrect predictions ought precisely to counterbalance, resulting in a total reward of 0. This conflicts

with the supposed information-theoretic basis: on any account in accord with Shannon, a reward for a certain prediction coming true can only be finite, while a penalty for such a *certain* prediction coming false must always be infinite. Putting these into balance guarantees there will be no proper information-theoretic interpretation of their reward function.

Kononenko and Bratko introduce the following reward function, where p' is the estimated probability and p is the prior:

$$I^+_{KB} = \log p' - \log p \qquad \text{(for } p' \geq p\text{)} \tag{2a}$$

$$I^-_{KB} = -\log(1-p') + \log(1-p) \qquad \text{(for } p' < p\text{)} \tag{2b}$$

This measure is assessed against the *true* class only. Since the probabilities of other classes are not considered, in multinomial classification a miscalibrated assessment of the alternative classes will go unpunished. For all these reasons we do not consider the Kononenko and Bratko function to be adequate.[1]

2.3 Bayesian Information Reward

The idea behind fair fees, that you should only be paid for an *informative* prediction, is simply not adequately addressed by Good's *IR*. Suppose an expert's job is to diagnose patients with a disease that is carried by 10% of some population. This particular expert is lazy and simply reports that each patient does not have the disease, with 0.9 confidence. Good's expected reward per patient for this strategy is $0.9(1+\log_2 0.9)+0.1(1+\log_2 0.1) = 0.531$, so the uninformed strategy is rewarded substantially! The expected reward per patient we should like is 0, which our generalization below provides. Good's *IR* breaks down in its application to multinomial classification: any *successful* prediction with confidence less than 0.5 is penalized, even when the confidence is greater than the prior. Good's fair fees are actually fair only when both the prior is uniform and the task binary.

We presented a Bayesian metric similar to below in Hope and Korb [6]. Unfortunately, it failed to reward perfect calibration maximally,[2] and thus we abandoned it in favour of the following. For classification into classes $\{C_1, \ldots, C_k\}$ with estimated probabilities p'_i and priors p_i, where $i \in \{1, \ldots, k\}$:

$$IR_B = \frac{\sum_i I_i}{k} \tag{3}$$

where $I_i = I^+_i$ for the true class and $I_i = I^-_i$ otherwise, and

$$I^+_i = \log \frac{p'_i}{p_i}$$

$$I^-_i = \log \frac{1-p'_i}{1-p_i}$$

[1] We did, however, include it in the empirical evaluation of [6].

[2] David Dowe pointed this out.

Clearly, when $p' = p$, the reward is 0. IR_B also retains an information-theoretic interpretation: the measure is finitely bounded in the positive direction, since prior probabilities are never zero, and misplaced certainty (i.e., when the probability for the true value is 0) warrants an infinite negative reward. Finally, correct probabilities are now rewarded maximally in the long run. The proof of this is available in [7] and [8–§10.8].

A non-uniform prior p can be obtained any number of ways, including being set subjectively (or arbitrarily). In our empirical studies we simply use the frequency in the test set given to the machine learner to compute the prior.[3] This is because we have no informed prior to work with, and because it is simple and unbiased relative to the learning algorithms under study.

Bayesian IR_B reflects the gambling metaphor more adequately than does Good's IR. Book makers are required to take bets for and against whatever events are in their books, with their earnings depending on the spread. They are, in effect, being rated on the quality of the odds they generate for all outcomes simultaneously. Bayesian IR does the same for machine learning algorithms: the odds (probabilities) they offer on all the possible classes are simultaneously assessed, extracting maximum information from each probabilistic classification.

3 Empirical Study: Bayesian Models

Our empirical evaluation focuses on machine learners that form Bayesian models, partially in response to recent work showing the power of Naive Bayes learners (e.g., [9, 10, 11]). The machine learners are Naive Bayes (NB) [12], Tree Augmented Naive Bayes (TAN) [10, 13], Averaged One Dependence Estimators (AODE) [9] and Causal MML (CaMML) [14].

For the experiment, we artificially generated data from a series of Bayesian model types. Three model types are chosen, each designed to favour a particular machine learner: Naive Bayes, TAN or AODE. Thus, we compare how the learners perform when their assumptions are broken and when they are exactly matched. To test the threshold at which a simpler model outperforms the more complex, we also systematically vary the amount of training data.

Of our machine learners, CaMML finds models of the greatest generality. Given standard convergence results, CaMML must better or equal every other machine learner in the limit. Again, AODE's and TAN's models generalize the Naive Bayes models, and given sufficient data they should perform at least on par with Naive Bayes. This suggests a converse phenomenon. At low levels of data, and *if the learner's representations include the true model*, the simpler learner should outperform the more complex, because complex machine learners converge to their optimum models slower, due to a larger search space.

Experimental Method. For statistical analysis, we regard each model type as a separate experiment. For each experiment we sample the space of appropriate

[3] We start the frequency counts at 0.5 to prevent overconfident probabilities.

Table 1. Table of results for the three Bayesian model experiments. The results are average information reward for each machine learner, given 50, 500 or 5000 training instances. Confidence intervals at 95% are shown

(a) Naive Bayes Models

Training samples	NBayes	AODE	TAN	CaMML
50	0.250 ± 0.053	0.238 ± 0.049	0.234 ± 0.053	0.092 ± 0.051
500	0.466 ± 0.051	0.443 ± 0.051	0.456 ± 0.053	0.459 ± 0.051
5000	0.496 ± 0.051	0.494 ± 0.051	0.495 ± 0.051	0.496 ± 0.051
(b) Tree Augmented Naive Bayes Models				
50	-0.003 ± 0.039	0.124 ± 0.031	0.003 ± 0.039	-0.019 ± 0.033
500	0.214 ± 0.031	0.351 ± 0.029	0.361 ± 0.033	0.400 ± 0.033
5000	0.247 ± 0.031	0.411 ± 0.029	0.498 ± 0.035	0.504 ± 0.035
(c) Averaged One Dependence Models				
50	-0.228 ± 0.041	-0.134 ± 0.025	-0.277 ± 0.041	-0.013 ± 0.014
500	0.020 ± 0.010	0.050 ± 0.012	0.000 ± 0.014	0.003 ± 0.004
5000	0.057 ± 0.008	0.113 ± 0.010	0.089 ± 0.010	0.098 ± 0.012

models. Each model has 4–8 attributes (including the target attribute), with each attribute having 2–5 values. The probabilities in each attribute are determined randomly. We sample 40 models and perform a two-factor repeated measures ANOVA, in order to provide a statistical test independent of our Bayesian assumptions. The two factors are (1) the machine learner and (2) the amount of training data (50, 500 or 5000). It is advantageous to use a repeated measure ANOVA because this design controls for the individual differences between samples (where each model is considered a sample).

We use information reward on a single test set of 1000 instances for each model to measure the 'treatment' of each machine learner at different 'doses' (amounts of training data). We don't report accuracy nor Kononenko and Bratko's measure, for the reasons given in Sections 1 and 2.2. Where we report confidence intervals, these have been adjusted by the Bonferroni method for limiting the underestimation of variance [15].

Naive Bayes Models follow the Naive Bayes assumptions: attributes are pairwise independent, given the class. This is the simplest model type we use in this evaluation, so we expect that all learners will perform reasonably.

Table 1a shows the performance of the machine learners for each amount of training data. Naive Bayes, TAN and AODE perform similarly for each level — unsurprising, since they share the correct assumption that the target class is a parent of all other attributes. For small amounts of data, CaMML performs significantly worse than the others: it cannot reliably find the correct model. As more data become available, it achieves a score similar to the others.

Tree Augmented Naive Models are formed by creating a tree-like dependency structure amongst the (non-target) attributes, with all of them directly dependent upon the target. This is more complicated than the Naive Bayes

model above. Each model we generate has a random tree structure amongst the non-target attributes.

Surprisingly, Table 1b shows that TAN is not the best learner with low amounts of training data: AODE stands superior. This is likely because AODE has a richer representation than Naive Bayes (i.e., with averaged predictions), yet doesn't need to search for the tree structure. Once there are enough data both TAN and CaMML seem to find the right structure and both outperform AODE. This illustrates the additional difficulty of model selection. Although TAN assumes the correct model type, it still has to find a particular tree structure, thus TAN's performance is dependent on its search capabilities. Naive Bayes, with its inaccurate assumptions, is clearly inferior to the other learners once an adequate amount of training data is given.

Averaged One-Dependence Models are each a series of n models; in the ith model, attribute i is the parent of each other (non-target) attribute. As in Naive Bayes, each attribute is also directly dependent on the target. Thus, each AODE model is a hybrid of one-dependence models, with each model having equal chance to be selected from when sampling the model for data.

This hybrid model seems to be very difficult for the machine learners to learn, with Table 1c showing the information reward ranging from only -0.3 to 0.1. Recall that a reward of zero corresponds to a machine learner which finds no associations, returning the observed frequency of the target class as its estimate. It takes more than 50 training instances to achieve an average score higher than zero! CaMML performs slightly better with sparse data, near the level of total ignorance. The explanation of the poor performance with little data perhaps lies in each learner's assumptions: Naive Bayes, TAN and AODE assume a model where all attributes depend on the target, regardless of whether this model decreases performance. CaMML is not beholden to any particular model, and thus is free to choose no association at all. This conservatism wins out, even against Naive Bayes with small datasets. After enough training data, AODE (the only learner that can model the data properly) obtains an advantage over the other learners.

We also evaluated the learners on a set of well known datasets, including many from the UCI data repository. For this we used Dietterich's 5×2cv evaluation method [2], modified to incorporate stratified sampling. These are reported in [7]. Briefly, we found that AODE seemed to outperform the other learners, consistent with its performance above, and also reconfirmed that accuracy and IR_B often return conflicting results.

4 Conclusion

We have reviewed a variety of metrics for the evaluation of machine learners. Accuracy is too crude, optimizing only domain knowledge while ignoring calibration. We have developed a new metric which is maximized under the combination of domain knowledge and perfect calibration. This information reward evaluates

learners on their estimate of the whole class distribution rather than on a single classification. In rewarding calibration, it provides a valuable alternative to cost-sensitive metrics when costs are unavailable.

References

1. Kohavi, R.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: IJCAI. (1995) 1137–1145
2. Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation **7** (1998) 1895–1924
3. Provost, F., Fawcett, T.: Robust classification for imprecise environments. Machine Learning **42** (2001) 203–231
4. Good, I.J.: Rational decisions. Journal of the Royal Statistical Society. Series B **14** (1952) 107–114
5. Kononenko, I., Bratko, I.: Information-based evaluation criterion for classifier's performance. Machine Learning **6** (1991) 67–80
6. Hope, L.R., Korb, K.B.: Bayesian information reward. In Mckay, B., Slaney, J., eds.: Lecture Notes in Artificial Intelligence. Springer (2002) 272–283
7. Hope, L.R., Korb, K.B.: A Bayesian metric for evaluating machine learners. Technical report, Monash University (2004)
8. Korb, K.B., Nicholson, A.E.: Bayesian Artificial Intelligence. Chapman & Hall/CRC (2004)
9. Webb, G.I., Boughton, J., Wang, Z.: Averaged One-Dependence Estimators: Preliminary results. In: Australasian Data Mining Workshop, ANU (2002) 65–73
10. Friedman, N., Goldszmidt, M.: Building classifiers using Bayesian networks. In: AAAI-96. (1996) 1277–1284
11. Zheng, Z., Webb, G.I.: Lazy learning of Bayesian rules. Machine Learning **41** (2000) 53–84
12. John, G.H., Langley, P.: Estimating continuous distributions in Bayesian classifiers. In: UAI 11, Morgan Kaufmann, San Mateo (1995) 338–345
13. Keogh, E., Pazzani., M.: Learning augmented Bayesian classifiers. In: AI and Statistics. (1999) 225–230
14. Wallace, C., Boulton, D.: An information measure for classification. The Computer Journal **11** (1968) 185–194
15. Keppel, G.: Design and Analysis: A Researcher's Handbook. Prentice-Hall (1991)