A Quantification Of Distance-Bias Between Evaluation Metrics In Classification

Ricardo Vilalta VILALTA@US.IBM.COM
IBM T.J. Watson Research Center, 30 Saw Mill River Rd., Hawthorne, N.Y., 10532 USA

Daniel Oblinger OBLINGER@US.IBM.COM
IBM T.J. Watson Research Center, P.O. Box 218, Yorktown Heights, N.Y., 10598 USA

Proceedings of the 17th International Conference on Machine Learning
(Stanford University), pp. 1087-1094. Morgan Kaufmann.

Abstract

This paper provides a characterization of bias for evaluation metrics in classification (e.g., Information Gain, Gini, χ^2, etc.). Our characterization provides a uniform representation for all traditional evaluation metrics. Such representation leads naturally to a measure for the distance between the bias of two evaluation metrics. We give a practical value to our measure by observing if the distance between the bias of two evaluation metrics correlates with differences in predictive accuracy when we compare two versions of the same learning algorithm that differ in the evaluation metric only. Experiments on real-world domains show how the expectations on accuracy differences generated by the distance-bias measure correlate with actual differences when the learning algorithm is simple (e.g., search for the best single-feature or the best single-rule). The correlation, however, weakens with more complex algorithms (e.g., learning decision trees). Our results show how interaction among learning components is a key factor to understand learning performance.

1. Introduction

In symbolic classification (e.g., decision trees, rule induction, decision lists), an evaluation metric serves to assess the value of potentially useful expressions (i.e., features or logical combination of features). The best-evaluated expression is then used as a building-block to construct the final hypothesis (e.g., select the best feature at each node of a decision tree). Most common metrics, *traditional* or *purity based*, quantify the degree of intra-class uniformity or purity of the example subsets induced by an expression. Instances of this kind include Information Gain (Quinlan, 1994), Gain Ratio (Quinlan, 1994), G statistic (Mingers, 1989), χ^2 (Mingers, 1989; White & Liu, 1994), Laplace (Quinlan, 1995), and Gini Index (Breiman et al., 1984).

This paper is a study of the bias adopted by traditional evaluation metrics. We limit ourselves to the case where expressions are binary-valued. Previous work has defined bias as the tendency of a metric to favor multi-valued expressions (White & Liu, 1994; Kononenko, 1995). Instead we follow the definition of bias as the preference for a partial ordering over the space of candidate expressions. The question we address is not how much metric M favors multi-valued expressions, but rather how metrics M_1 and M_2 differ in their ranking of all available binary-valued expressions. Under this framework, our characterization shows the bias adopted by each metric explicitly, in a uniform representation, and enables us to compare differences in bias.

We propose a measure for the distance between the bias of two evaluation metrics. A pair-wise comparison of several traditional metrics using this distance-bias measure quantifies the degree of similarity (conversely dissimilarity) between metrics (Vilalta, 1999). We explore the connection between the distance-bias measure and differences in predictive accuracy when the same learning algorithm is built using different evaluation metrics. Multiple experiments on real-world domains show how the expectations on accuracy differences generated by the distance-bias measure correlate with actual differences when the learning algorithm is simple: look for the best single feature or the best single rule (Pearson's correlation coefficient

$r \geq 0.97$). The usefulness of our proposed measure, however, is questionable under more complex algorithms, e.g., learning decision trees (Pearson's correlation coefficient $r \geq 0.52$). We conclude interactions among components in a learning algorithm must be taken into account to fully explain accuracy performance.

The organization of this paper follows. Section 2 reviews definitions of some traditional evaluation metrics. Section 3 gives a characterization of bias when the number of induced example subsets is binary, and defines a measure for the distance between the bias of two evaluation metrics. Section 4 describes our experiments on real-world domains with discussion. Finally, Section 5 extends a summary and our conclusions.

2. Preliminaries

Classification presupposes the availability of a training set T_{train} made up of examples, $T_{\text{train}} : \{(X_i, c_i)\}_{i=1}^{N}$, where each vector X_i is characterized as a point in a k-dimensional feature space. X_i is labeled with class c_i according to an unknown target function C, $C(X_i) = c_i$. We shall assume T_{train} consists of independently and identically distributed (i.i.d.) examples obtained according to a fixed but unknown joint probability distribution in the space of possible examples and classes $\mathcal{X} \times \mathcal{C}$. For simplicity, a splitting function or expression E will be restricted to a logical form: E can represent a feature value or the logical combination of feature values; target concept C will also be assumed of logical form.

An evaluation metric M is used to quantify the quality of the partitions induced by an expression E over a training subset $T \subseteq T_{\text{train}}$, where $|T| = N$ and $N \leq |T_{\text{train}}|$. Expression E divides T in two sets: $\{X \mid E(X) = 1\}$ and $\{X \mid E(X) = 0\}$; we say the former set is covered by E, whereas the latter set is covered by the complement E'. Similarly, set T can be divided according to the coverage of concept C, and its complement C'. Figure 1 shows the cross-classification of classes and values of E. Let n^1 and n^0 be the number of examples in T of class 1 and 0 respectively, where $n^1 + n^0 = N$. Let n_1^1 and n_1^0 be the number of examples covered by E of class 1 and 0 respectively, such that $n_1^1 + n_1^0 = n_1$, and let n_0^1 and n_0^0 represent the corresponding numbers in E', such that $n_0^1 + n_0^0 = n_0$. In addition, Figure 1 defines probabilities as estimated from the data.

The following are definitions of some traditional evaluation metrics (goal is to maximize the output value).

	E	E'	
C	n_1^1	n_0^1	n_1
C'	n_1^0	n_0^0	n_0
	n^1	n^0	N

Probabilities:

For C and C': $P^1 = \frac{n^1}{N}$ $P^0 = \frac{n^0}{N}$

For E: $P_1^1 = \frac{n_1^1}{n_1}$ $P_1^0 = \frac{n_1^0}{n_1}$ $P_1 = \frac{n_1}{N}$

For E': $P_0^1 = \frac{n_0^1}{n_0}$ $P_0^0 = \frac{n_0^0}{n_0}$ $P_0 = \frac{n_0}{N}$

Figure 1. Cross-classification of expression values and classes with probabilities estimated from the data.

INFORMATION GAIN

Let entropy $\mathrm{H}(x, y) = -x log_2(x) - y log_2(y)$

$$\mathrm{IG}(E) = \mathrm{H}(P^1, P^0) - \sum_{i=0}^{1} (P_i \ \mathrm{H}(P_i^1, P_i^0)) \qquad (1)$$

GAIN RATIO

$$\mathrm{GR}(E) = \frac{\mathrm{IG}(E)}{\mathrm{H}(P_0, P_1)} \qquad (2)$$

G STATISTIC

$$\mathrm{G}(E) = 2N \ \mathrm{IG}(E) \ \log_e 2 \qquad (3)$$

GINI

Let $\mathrm{GI}(x, y) = 1 - (x^2 + y^2)$

$$\mathrm{gini}(E) = \mathrm{GI}(P^1, P^0) - \sum_{i=0}^{1} (P_i \ \mathrm{GI}(P_i^1, P_i^0)) \qquad (4)$$

χ^2

$$\chi^2(E) = \frac{N \ (n_1^1 n_0^0 - n_0^1 n_1^0)^2}{n^1 n^0 n_1 n_0} \qquad (5)$$

LAPLACE

$$\mathrm{L}(E) = \begin{cases} \frac{n_1^1 + 1}{n_1^1 + n_1^0 + 2} & \text{if } n_1^1 \geq n_1^0 \\ \frac{n_1^0 + 1}{n_1^1 + n_1^0 + 2} & \text{if } n_1^1 < n_1^0 \end{cases} \qquad (6)$$

3. A Characterization of Bias

The metrics defined in Section 2 share a common characteristic: the value of an expression E is based on

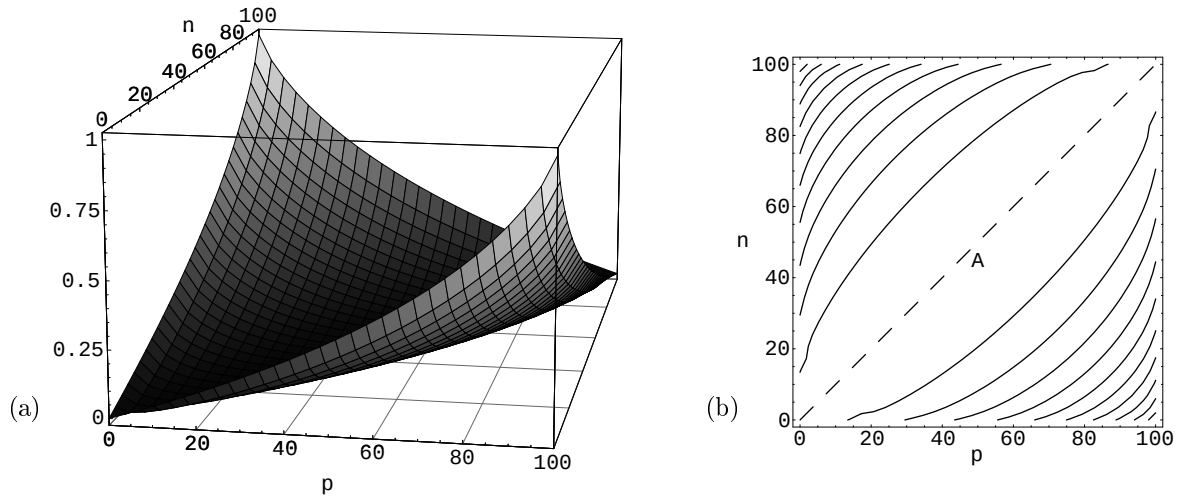

Figure 2. (a) Information Gain as a function of the possible coverage (number of positive and negative examples) of an expression. (b) A contour plot of (a) showing isometric lines over the coverage plane.

the intra-class purity of the induced example subsets. An evaluation metric M is a function of the number of positive and negative examples covered by expression E and of its complement E', $M : f(n_1^1, n_1^0, n_0^1, n_0^0)$ (Figure 1). Alternatively M could be defined as a function of the coverage of E and of the coverage of the whole set T, $f(n_1^1, n_1^0, n^1, n^0)$, since $n^1 = n_0^1 + n_1^1$ and $n^0 = n_0^0 + n_1^0$. For a given learning problem, n^1 and n^0 are fixed; by considering them as constants, we can simply express M as $f(n_1^1, n_1^0)$. For simplicity let's rename n_1^1 and n_1^0 as p and n (the positive and negative examples covered by E), such that $M : f(p, n)$. Metric M extends above the plane defined by these two variables. We define this plane as the *coverage plane*.

Each of the metrics defined in Section 2 can be plotted above a coverage plane bounded by the total positive and negative examples in T, n^1 and n^0. As an example, Figure 2(a) plots Information Gain when the number of positive and negative examples in T is the same ($n^1 = n^0 = 100$); each point (p, n) is evaluated according to equation 1. The fact that the value of $f(p, n)$ takes into account both the coverage of E and of its complement E' is reflected by the symmetry about the axis-line $((0, 0), (100, 100))$. The maximum values are attained at the extreme points $(100, 0)$ and $(0, 100)$, when the induced example subsets are class uniform.

3.1 Bias and Isometric Lines

We now answer the following question: how can we concisely represent the ordering imposed by an evaluation metric over all points in the coverage plane? (i.e., over all possible expressions?). To answer this question, consider first the result of projecting $M : f(p, n)$ over the coverage plane. Those points that have the same value for $f(p, n)$ project into the plane as isometric lines. An isometric line indicates a constant value for $M : f(p, n)$ throughout its extent on the coverage plane. As an example, Figure 2(b) shows isometric lines obtained by projecting Information Gain over the coverage plane. An isometric line comprises points having the same height on $f(p, n)$; the line actually joins such points, but has no meaning between non-integer values of p and n. The isometric line with lowest-value points, axis-line A, cuts the space in two, from $(0, 0)$ to $(100, 100)$. All other lines move parallel to A towards the extreme points $(100, 0)$ and $(0, 100)$.

Consider arbitrarily the right lower half of the coverage plane shown in Figure 2(b). As long as M monotonically increases from axis-line A to the extreme point $(100, 0)$, then any isometric line divides this half space into two regions; the right lower region encompasses points with a preference over those points in the left upper region.[1] Thus, *the bias of an evaluation metric is determined by the shape of the isometric lines (i.e., contour lines) obtained by projecting $M : f(p, n)$ over the coverage plane.* Two evaluation metrics projecting isometric lines having different shape over the coverage plane differ in their ordering or ranking of all possible expressions.

3.2 The Distance in Bias Between Evaluation Metrics

We now proceed to explain a method to quantify the distance between the bias of two evaluation metrics. To begin, let us first find a representation for a single

[1] Symmetrically, an isometric line on the left upper half of the coverage plane divides this half space into two regions; the left upper region having a preference over the right lower region.

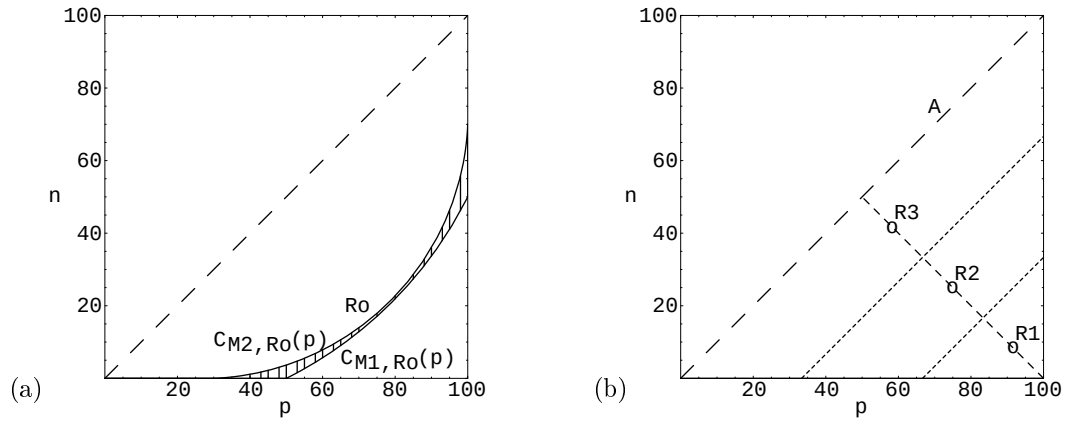

Figure 3. (a) The difference in bias between metrics M_1 and M_2 represented as the area between $\mathrm{C}_{M_1,R_0}(p)$ and $\mathrm{C}_{M_2,R_0}(p)$. (b) Three points positioned on different regions of the coverage plane.

isometric line. Let $M : f(p, n)$ be an evaluation metric and let $R_0 = (p_0, n_0)$ be a fixed point on the coverage plane. We can represent the isometric line passing through R_0 by a function that maps values of n (arbitrarily chosen as the dependent variable) onto values of p, the independent variable. We use the function, $\mathrm{C}_{M,R_0}(p)$, to denote the contour line of the metric M that passes through the point R_0. For example, for all points on the right lower half of the coverage plane (i.e., when $p > n$), the Laplace function (Section 2) is defined as follows:

$$\mathrm{L}(E) = \mathrm{L}(p, n) = \frac{p+1}{p+n+2} \tag{7}$$

For a fixed point $R_0 = \langle p_0, n_0 \rangle$, we solve this equation for n and generate the contour function for this metric:

$$\mathrm{C}_{L,R_0} = n = \frac{p+1}{\mathrm{L}(\mathrm{p_0}, \mathrm{n_0})} - (p+2) \tag{8}$$

In order to ensure that this definition is valid over the entire coverage plane we take n to be 0 if the contour function returns a negative number. More precisely, n must be represented as $n : (g(p))_+$, where $(\cdot)_+$ denotes the positive threshold function, i.e., $(g(p))_+ = 0$ if $g(p) < 0$, and $(g(p))_+ = g(p)$ if $g(p) \geq 0$. Equation 8 represents a family of isometric lines for the Laplace function according to the value of R_0.

We now consider a definition for the distance between the bias of two evaluation metrics. Given a fixed point R_0 and two metrics M_1 and M_2 we define the distance between these two metrics as the area between the isometric lines from each metric passing through the point R_0. Formally the area between the two isometric lines at R_0 is written as $\delta_{R_0}(M_1, M_2)$, and it is defined as:

$$\delta_{R_0}(M_1, M_2) = \int_0^{n^1} |\mathrm{C}_{M_1,R_0}(p) - \mathrm{C}_{M_2,R_0}(p)| \; dp \tag{9}$$

The integral goes from 0 to the total number of positive examples n^1, and captures the disagreement between M_1 and M_2. As an example, Figure 3(a) shows the projection of a single isometric line for two metrics. R_0 corresponds to the point of intersection between the lines. The shaded area between both lines quantifies the difference in bias between M_1 and M_2 at R_0. Equation 9 can be interpreted in the following way: Assume E is an expression whose coverage is characterized by the coordinate R_0. Now for another expression E', the metrics will disagree on the ordering of E and E' only if the coverage coordinates of E' fall within the area between the isometric curves for the two metrics. Thus Equation 9 is a measure of dissent between the two metrics at a single point R_0.

We can naturally extend equation 9 to a general measure of dissimilarity between two metrics. This is simply the average size of the dissent region over the space of all coverage coordinates. The overall measure of dissimilarity is defined as:

$$\delta(M_1, M_2) = \frac{1}{|\mathrm{R}|} \sum_{r \in \mathrm{R}} \delta_r(M_1, M_2) \tag{10}$$

where R denotes the set of all points r in the coverage plane. Because a numeric approximation of equation 10 is computationally expensive, in what follows we focus our attention on equation 9 exclusively.

3.3 A Quantitative Comparison

Let us now test the effect of the position of R_0 in the evaluation of $\delta_{R_0}(M_1, M_2)$ (equation 9). To that effect

Table 1. A pair-wise comparison of different evaluation metrics.

Pair of Metrics	R_1 $(92, 8)$	R_2 $(75, 25)$	R_3 $(58, 42)$
Gain-Ratio vs Gini/χ^2	20.23 0.4%	233.8 4.6%	152.62 3.0%
Gain-Ratio vs Info-Gain/G	6.76 0.13%	178.8 3.5%	145.42 2.9%
Gain-Ratio vs Laplace	327.5 6.5%	806.5 16%	573.6 11.4%
Gini/χ^2 vs Info-Gain/G	13.46 0.27%	55.02 1.1%	7.20 0.14%
Gini/χ^2 vs Laplace	327.5 6.5%	806.5 16%	567.3 11.3%
Info-Gain/G vs Laplace	327.5 6.5%	806.5 16%	567.3 11.3%
Info-Gain vs G Gini vs χ^2	0.0 0.0%	0.0 0.0%	0.0 0.0%

we conducted a series of experiments described as follows. Figure 3(b) shows a coverage plane with equal class proportions ($n^1 = n^0 = 100$); the right-lower half of the plane is divided into three equally-spaced regions. We took three different points along the line perpendicular to A ending on the extreme point $(n^1, 0)$ (i.e., $(100, 0)$). These points are labeled as R_1, R_2, and R_3. We carried out a pair-wise comparison of all evaluation metrics defined in Section 2 by applying equation 9 at each of these three points. The goal of this experiment is to determine which metrics are more similar to each other, and what is the effect of evaluating an expression on different regions of the coverage plane. On each comparison we used Mathematica (Wolfram, 1999) to obtain a numeric approximation of equation 9. The results are shown in Table 1. The coordinates for R_1, R_2, and R_3 are indicated on the first row. On each cell, the number on the top represents the result of evaluating equation 9; the number on the bottom shows the fraction of the area covered by the top number with respect to the area of the entire half space of the coverage plane. The area of the half space in Figure 3(b) equals $(100 \times 100)/2 = 5000$, and in general the area equals $(n^1 \times n^0)/2$.

An analysis of Table 1 shows no difference between Information Gain and the G statistic, and between Gini and χ^2 (Table 1, last row). We can prove the first two metrics share the same bias by observing that the G statistic is simply a constant factor times Information Gain, and thus both metrics project the same isometrics lines over the coverage plane. We leave as an open problem a proof (or disproof) that both Gini and χ^2 project the same isometric lines. Using the bottom value on each table entry, the smallest difference on average is between Gini (or χ^2) and Information Gain (or the G statistic), around 0.5%. Next in magnitude is the difference between Gain Ratio and Information Gain (or the G statistic), around 2.18%, followed by the difference between Gain Ratio and Gini (or χ^2), around 2.7%. Most distances are within 5%, except for the Laplace function showing differences up to 16%.

Notice in Table 1 that the difference between any pair of metrics increases from point R_1 to point R_2, and then decreases from R_2 to R_3. We conclude a ceiling effect (Cohen, 1995, Chapter 3, p. 79) occurs at points around R_1 because the quality of the expressions in this region is close to optimal. Conversely, a floor effect occurs at points around R_3 because the proximity to axis-line A forces most metrics to agree in assigning low credit to any expression in this region. The highest difference among biases is expected along the region comprising R_2, where no upper or lower bounds restrain the degree of agreement —or disagreement— in bias.

3.4 Skewed Class Distributions

We now answer the question: what is the effect of skewed class distributions over the coverage plane? A domain with unequal class proportions maps to a coverage plane that stretches into a wide or tall rectangle, depending on the dominating class. More formally, consider the line perpendicular to axis line A connecting A with any of the extreme points $(n^1, 0)$ and $(0, n^0)$ (Figure 3(b)). This line attains maximum size when $n^1 = n^0$, decreasing as the population of one class grows large relative to the other class. The shorter the size of this line the more difficult to distinguish differences in bias because most expressions would lie close to either line A or one of the uniform-class points. Therefore, the degree of skewness in the class distribution is an important factor when comparing the effect of different evaluation metrics during learning. A highly skewed distribution may lead to the conclusion that two metrics yield similar generalization effects, when in fact a significant difference could be detected under equal class distributions.

4. Experiments

In this section we attempt to give a practical value to the distance-bias measure defined in equation 9. If two metrics, M_1 and M_2, are used to generate hypotheses h_1 and h_2 respectively, we expect the disagreement between h_1 and h_2 to increase as the distance in bias $\delta_{R_0}(M_1, M_2)$ increases too. Of particular interest is then to compare differences in predictive accuracy resulting from using different evaluation metrics. We wish to determine if the distance-bias measure corre-

lates with such performance differences.

In the experiments below, we first use a single-rule learning algorithm. The algorithm conducts a search over the space of logical expressions, each expression represented as the conjunction of feature literals (feature values or their negations). The final hypothesis is a rule of the form: **if** (cond_1 & cond_2 & $\cdots$ & cond_p) **then** Class = c, where cond_i is a boolean feature or its negation, and c is the class assigned to any example satisfying the rule antecedent. The model is justified from the fact that we wish to obtain hypotheses not always consistent with the training data (expected to underfit the target concept), and thus not necessarily lying on an optimal, class-uniform point over the coverage plane (Section 3). This will allow us to test the distance-bias measure (equation 9) on different regions of the coverage plane. The algorithm performs a beam-search over the space of possible conjunctions of literals (Vilalta, 1999), and is outlined in Figure 4.

Algorithm 1: Single-Rule Learning Algorithm
Input: beam width α, Metric M
Output: A single hypothesis rule
BEST_RULE()
(1) Let L_{literals} be the list of all literals
(2) $L_{\text{beam}} \leftarrow$ best α literals in L_{literals}
(3) **while** (**true**)
(4) $L_{\text{new}} \leftarrow$ Form the conjunction of
(5) every $E_i \in L_{\text{beam}}$ with every $E_j \in L_{\text{literals}}$
(6) Keep track of best expresion E_{best}
(7) according to evaluation metric M
(8) Eliminate poor expressions from L_{new}
(9) **if** $L_{\text{new}} = \emptyset$
(10) **break**
(11) $L_{\text{beam}} \leftarrow$ best α combinations in L_{new}
(12) **end while**
(13) Let c be the majority class in E_{best}
(14) **return if** E_{best} **then** c

Figure 4. A learning algorithm that returns a single rule as the final hypohesis.

4.1 Experimental Methodology

Table 2 compares the performance of the learning algorithm described above when the search for the best single rule is conducted using different evaluation metrics. The table reports on four different versions of the algorithm (columns 4-7). The algorithms using Information Gain and the G statistic, and Gini and χ^2, perform each pair the same and thus collapse into the same column (Section 3.3).

The first column lists real-world domains extracted from the UCI database repository (Merz & Murphy, 1998), except for the star-cluster domain extracted from Table 22.1 in Andrews and Herzberg (1985).

The second column in Table 2 indicates, on average and for each domain, the location of the final expression (i.e, rule antecedent) over the coverage plane. Region 1 is the closest to the class-uniform points, region 2 is half-way between the class-uniform points and axis-line A, and finally region 3 is closest to axis-line A (Figure 3b). Each entry on column 2 is the result of first averaging for each algorithm and over all runs the region on the coverage plane where the final expression lies. We then take the average over all algorithms. Domains in Table 2 are ordered based on the values in column 2. We assign a domain to region 1 if the entry on column 2 is within $[1.0, 1.5)$, region 2 if it is within $[1.5, 2.5)$, and region 3 if it is within $[2.5, 3.0]$.

The third column in Table 2 shows the proportion of positive examples on the training set. Each entry is found by averaging the proportion of positive classes on the training set over all runs (all algorithms are presented the same training data).

The last four columns in Table 2 estimate, for each algorithm, the predictive accuracy achieved on each domain by using stratified 10-fold cross-validation, averaged over 10 repetitions. Numbers enclosed in parentheses represent standard deviations. Previous to each run, an initial discretization step makes all features boolean (numeric features are discretized following Catlett (1991); nominal features are decomposed into a boolean feature for each nominal value). On each run, the training set is divided in two: one half conducts a beam search for the best logical expression (as mentioned above), the other half serves to validate the best current expression to avoid statistical errors from multiple comparisons (Jensen & Cohen, 2000).

Runs were performed on a RISC/6000 IBM model 7043-140.

4.2 Testing the Utility of the Distance-Bias Measure

We now test the utility of the distance-bias measure defined in equation 9. Our first experiment uses the last row in Table 2 corresponding to the average predictive accuracy for each algorithm over all domains. We compute the absolute difference in predictive accuracy between each pair of algorithms. We also compute the average distance in bias between pairs of evaluation metrics in Table 1 (average over the three regions). The results can be paired-up by matching average distance in bias between metrics with the corresponding

Table 2. Predictive accuracy on real-world domains for a single-rule hypothesis. Numbers enclosed in parentheses represent standard deviations.

Concept	Region (1-3)	Proportion (+) class	Predictive Accuracy			
			Info. Gain G Statistic	Gini χ^2	Gain Ratio	Laplace
voting	1.0	0.38	95.0 (0.26)	95.13 (0.38)	94.95 (0.39)	92.53 (1.35)
cancer	1.01	0.65	94.16 (0.64)	94.64 (0.61)	94.51 (0.36)	90.34 (1.08)
new-thyroid-hyper	1.01	0.17	96.67 (1.33)	97.10 (1.05)	96.43 (0.75)	90.42 (1.79)
new-thyroid-hypo	1.0	0.15	96.19 (1.04)	96.10 (0.90)	95.62 (0.87)	94.19 (1.98)
star cluster	1.04	0.67	96.95 (0.71)	96.81 (0.80)	96.86 (0.83)	83.90 (2.68)
promoters	1.18	0.50	76.2 (4.07)	77.8 (2.71)	78.7 (3.82)	74.2 (4.02)
mushroom	1.25	0.45	99.73 (0.02)	99.53 (0.32)	99.73 (0.02)	78.39 (0.01)
ionosphere	1.25	0.66	89.63 (1.64)	89.74 (1.71)	90.31 (1.24)	66.06 (3.42)
crx	1.38	0.45	86.01 (0.34)	86.12 (0.26)	85.94 (0.32)	62.05 (2.39)
mean region 1			92.28	92.55	92.56	81.34
hepatitis	1.58	0.18	70.2 (2.93)	78.0 (1.90)	85.9 (2.34)	64.5 (3.11)
lymphography-2	1.6	0.52	81.71 (2.22)	81.93 (2.0)	80.21 (3.13)	72.29 (3.09)
lymphography-3	1.67	0.44	79.21 (3.03)	79.43 (2.85)	76.79 (1.87)	64.86 (4.54)
zoo	1.92	0.05	85.33 (2.27)	85.44 (3.20)	85.0 (3.55)	47.0 (10.27)
credit	1.95	0.67	67.36 (3.69)	69.81 (2.50)	73.09 (3.18)	54.90 (4.25)
chess-end game	2.01	0.50	79.41 (0.86)	80.73 (0.33)	75.44 (0.09)	72.80 (0.94)
heart	2.06	0.45	71.31 (1.03)	71.82 (1.35)	71.52 (1.27)	62.38 (4.77)
mean region 2			76.36	78.17	78.28	62.68
diabetes	2.51	0.32	67.62 (2.13)	70.86 (1.80)	69.89 (0.84)	43.32 (0.95)
bupa	2.61	0.45	59.18 (2.56)	60.63 (2.57)	57.72 (2.18)	47.12 (1.35)
tic-tac-toe	2.69	0.67	58.41 (2.37)	67.88 (4.15)	73.68 (0.88)	50.77 (1.97)
mean region 3			61.74	66.46	67.10	47.07
mean overall			77.16	78.94	79.30	65.02

average accuracy difference (e.g., the average distance bias between Gini and Laplace maps to the absolute accuracy difference of the two algorithms using Gini and Laplace). A linear regression model applied to this data yields a correlation coefficient (Pearson's coefficient) of $r = 0.97$, which points to a strong variable interdependence between the distance-bias measure and differences in accuracy performance. Results for other similar experiments are all summarized on Table 3. Each entry shows the correlation coefficient obtained from fitting a linear regression model to the data. Results are grouped in two columns. The first column uses all available domains. The second column eliminates the effect of skewed class distributions by filtering out domains with a positive-class proportion outside the range $[0.4, 0.6]$ (Section 3.4), leaving a total of eight (out of nineteen) domains. Without skewed distributions the experiment described above produces a correlation coefficient of $r = 0.99$. Such improvement indicates that the relation between the distance-bias measure and differences in predictive accuracy is more evident when the computation of equation 9 is done assuming a class distribution similar to that of the domain under analysis (Table 1 assumes equal class proportions). In addition, Table 3 shows results of experiments similar to the two above, except we group domains on three regions according to the coverage of the expression in the final rule hypothesis. The highest correlation is observed when domains are grouped into regions and skewed distributions are eliminated (bottom entries on column 3, Table 3).

4.3 Using Different Models

Our analysis is so far limited to single-rule hypotheses. One might ask if the results hold for other learning algorithms. We address the following question: does the relation between the distance-bias measure and differences in accuracy hold as the complexity of the algorithm increases? To answer this question we report on two additional experiments. The first experiment uses a simple algorithm that outputs a single-feature as the final hypothesis. After repeating the experiments reported above, correlation coefficients comparable to the first row of results on Table 3 (average over all domains with and without eliminating skewed distributions) take values of $r = 0.98$ and $r = 0.98$ respectively. The second experiment increases the complexity of the algorithm by using a decision tree as the hypothesis. The corresponding correlation coefficients are $r = 0.52$ and $r = 0.72$.

Our results show how increasing the number of learning components in the algorithm weakens the correlation between the distance-bias measure and differences

Table 3. Correlation coefficients on data comparing the distance-bias measure vs. differences in accuracy for a single-rule hypothesis.

Experiment	Correlation Coefficient	
	All Domains	Domains Without Skewed Dist.
Overall Average	0.97	0.99
Average Region 1	0.79	1.00
Average Region 2	0.97	0.98
Average Region 3	0.92	0.98

in predictive accuracy. Thus, even if the evaluation metric is the only component altered in the learning algorithm, failing to understand interactions among all other components may result in a poor understanding of performance. In decision-tree learning, for example, a robust analysis would additionally need to consider the effects of tree pruning, of a continuous partitioning of the feature space, of the tree-stopping criterion, etc.

5. Summary and Conclusions

This paper provides a characterization of bias for traditional or purity-based evaluation metrics. We show how the projection of an evaluation metric M over the coverage plane (i.e., the plane where axis i represents the number of examples of class i covered by an expression) yields isometric lines, or lines of constant value. It is the shape of these isometric lines that indicates the preference for one expression over another, i.e., indicates the bias of M (Section 3.1). The characterization above leads naturally to a measure for the distance in bias between two evaluation metrics (Section 3.2).

Our experimental results show a correlation between the distance-bias measure and differences in predictive accuracy when the same learning model is built using different evaluation metrics (e.g., look for the best single feature or the best single rule). Our results also show how the correlation tends to weaken as the degree of interaction between the evaluation-metric component and other components embedded in the learning algorithm increases (e.g., learning decision trees, Section 4.3). From this we conclude that a key element to understand differences in performance is to take into account interactions among learning components.

Future work will extend our results by trying to characterize the distance in bias between learning models apparently too far apart in their design. We learn from this study that an initial step to understand differences in performance is to produce an explicit representation of model bias, i.e., of the partial ordering imposed over the space of hypotheses. Such representation can then serve to quantify the amount of agreement (or disagreement) between the bias of two different models.

Acknowledgments

We are grateful to Se June Hong, Sholom Weiss, and Chid Apte for their valuable suggestions. This work was supported by IBM T.J. Watson Research Center.

References

Andrews, D., & Herzberg, A. (1985). *Data: A collection of problems from many fields for the student and research worker.* New York: Springer-Verlag.

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees.* Belmont, CA.: Wadsworth.

Catlett, J. (1991). On changing continuous attributes into ordered discrete attributes. *Proceedings of the Fifth European working Session on Learning* (pp. 164–178). New York: Springer-Verlag.

Cohen, P. (1995). *Empirical methods for artificial intelligence.* Cambridge, MA.: MIT Press.

Jensen, D., & Cohen, P. (2000). Multiple comparisons in induction algorithms. *Machine Learning*, *38*, 309–338.

Kononenko, I. (1995). On biases in estimating multi-valued attributes. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (pp. 1034–1040). San Mateo, CA.: Morgan Kaufmann.

Merz, C., & Murphy, P. (1998). *Uci repository of machine learning databases.* Available: www.ics.uci.edu/ mlearn/MLRepository.html.

Mingers, J. (1989). An empirical comparison of selection measures for decision-tree induction. *Machine Learning*, *3*, 319–342.

Quinlan, J. R. (1994). *C4.5: Programs for machine learning.* San Francisco: Morgan Kaufmann.

Quinlan, R. (1995). Oversearching and layered search in empirical learning. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (pp. 1019–1024). San Francisco: Morgan Kaufmann.

Vilalta, R. (1999). *Evaluation metrics in classification: A characterization of bias with an application to meta-learning.* Unpublished Manuscript, IBM T.J. Watson Research, Yorktown.

White, A., & Liu, W. (1994). Bias in information-based measures in decision tree induction. *Machine Learning*, *15*, 321–329.

Wolfram, S. (1999). *The mathematica book (4th ed.).* Champaign, IL: Wolfram Media.