# The Use of Misclassification Costs To Learn Rule-based Decision Support Models for Cost-Effective Hospital Admission Strategies

Richard Ambrosino[1], Bruce G. Buchanan[2], Gregory F. Cooper[1], and Michael J. Fine[3]
[1]Section of Medical Informatics, [2]Intelligent Systems Laboratory, [3]Department of Internal Medicine, University of Pittsburgh

## ABSTRACT

*Cost-effective health care is at the forefront of today's important health-related issues. A research team at the University of Pittsburgh has been interested in lowering the cost of medical care by attempting to define a subset of patients with community-acquired pneumonia for whom outpatient therapy is appropriate and safe. Sensitivity and specificity requirements for this domain make it difficult to use rule-based learning algorithms with standard measures of performance based on accuracy. This paper describes the use of misclassification costs to assist a rule-based machine-learning program in deriving a decision-support aid for choosing outpatient therapy for patients with community-acquired pneumonia.*

## INTRODUCTION

The Cost-Effective Health-Care (CEHC) project at the University of Pittsburgh Medical Center is interested in lowering the cost of medical care by attempting to define a subset of patients with community-acquired pneumonia for whom outpatient therapy is appropriate and safe. The CEHC group is comprised of members of the Pneumonia Patient Outcomes Research Team (Pneumonia PORT) project, and machine-learning researchers from the University of Pittsburgh and Carnegie-Mellon University. The CEHC group is investigating numerous machine-learning and statistical classification models, to predict whether a patient with a diagnosis of community-acquired pneumonia may be safely treated on an outpatient basis. Models under investigation include logistic regression, neural networks, Bayesian and non-Bayesian belief networks [1,2], and rule-based systems. A physician who is uncertain about outpatient treatment for a patient with community-acquired pneumonia could consult the model for assistance. The ideal model would use only a few key patient findings to make a prediction, and would be simple enough for a physician to perform the calculations on a piece of paper.

To be clinically useful, the group decided that such a model must have a very high predictive value when outpatient therapy is recommended. In other words, the model should make very few predictions suggesting outpatient therapy in patients who should ideally be treated as inpatients.

Learning a model with a very high positive predictive value is a problem not typically seen in the machine-learning literature. Most machine-learning algorithms are evaluated and compared using classification accuracy or classification error as the measure of performance [3]. (Accuracy is defined as the ratio of number of correct classifications divided by the total number of cases to be classified. Error rate is one minus the accuracy.) Learning a model with maximal classification accuracy will not necessarily produce a model with the desired sensitivity and specificity requirements (or predictive value requirements) for a given domain.

For learning methods whose output is continuous (e.g., a probability of risk of mortality between zero and one inclusive), an ROC curve[1] [4] can be computed by varying the cutoff threshold for the two predicted classes. A cutoff threshold can be selected to produce a model with an appropriate sensitivity and specificity for the given domain. (If it is preferred to use predictive values to select the cutoff threshold, as suggested by the Pneumonia PORT team for this domain, then a table of predictive values can be calculated for each cutoff value.) The ability to generate an ROC curve from a single model allows two or more models to be compared using the area under their ROC curves (or area under the clinically relevant portion of the ROC curve). Learning methods that produce continuous value outputs include logistic regression, neural networks, and Bayesian networks.

Learning methods with categorical outputs (e.g., 'admit' versus 'outpatient therapy') include traditional decision tree programs [5], classification trees/CART [6], and most rule induction programs [7,8,9]. These methods can *not* produce an ROC curve from a single model. To obtain a model with a different sensitivity and specificity, a completely new model must be induced (learned). In this paper we explain how misclassification error costs can be used with a rule

[1] A receiver operator characteristic (ROC) curve is a plot of true positive rate (sensitivity) versus false positive rate (one minus specificity).

0195-4210/95/$5.00 

induction system to produce an *ROC-like* curve and to learn a clinically relevant model (i.e., one with a high predictive value). Although we chose a rule induction system, this method is general and can be used with any learning method that produces categorical outputs.

## BACKGROUND

For a two-class learning problem, classification accuracy is calculated as the sum of true positive and true negative cases divided the total number of cases. The classification error rate (which is one minus classification accuracy) can be calculated as the sum of false positive and false negative cases divided the total number of cases. Misclassification cost is similar to classification error, except that different weights can be assigned to the false positive and false negative cases. Misclassification cost is calculated as the number of false positive cases multiplied by the false positive cost plus the false negative cases multiplied by the false negative cost divided by the total number of cases. Table 1 shows a sample misclassification matrix. The matrix shown gives a large false positive cost (10) when a patient who should be admitted (e.g., according to expert opinion) is predicted by a computer model to best receive 'outpatient therapy (Rx).' There is a smaller false negative cost (1) when a patient which should be treated as an outpatient is predicted as 'admit.'

Table 1: A misclassification matrix with FP:FN cost ratio of 10.

| | True Class | |
|---|---|---|
| Predicted Class | Outpatient Rx | Admit |
| Outpatient Rx | 0 (fp) | 10 (fp) |
| Admit | 1 (fn) | 0 (tn) |

In a misclassification matrix, the forward diagonals typically have zero values (no cost for correct classifications) [3,10]. For a two-class problem, a misclassification matrix (with zero diagonals) can be completely defined by the ratio of the false positive cost to the false negative cost (FP:FN cost ratio).

## RELATED WORK

Pazzani et al. [10] has applied several misclassification cost matrices to standard machine-learning databases and has shown that cost can be used as a measure of performance. Catlett [11] has applied misclassification costs to a blackjack domain by augmenting a standard decision tree program. He has shown how a variation in the FP:FN cost ratio can produce a ROC-like curve. The work described in this paper was done independently of Catlett's research.

## METHODS

The data used for this study were obtained from the 1989 MedisGroups Comparative Hospital Database [12]. Data were collected on 772,000 patients admitted to 78 hospitals in 23 states during the period from 7/87 to 12/88. The database consists of patient demographics and over 250 *key clinical findings* which are obtained from hospital information systems and standardized chart review.

The Pneumonia PORT study team identified 14199 adult patients with community-acquired pneumonia from the 1989 MedisGroups data using the following criteria: 1) ICD-9-CM principal diagnosis of pneumonia, 2) age 18 or greater, and 3) admission from home or a nursing home. Excluded were: 1) patients with AIDS or HIV positive titers (131 patients), and 2) patients with a previous hospitalization within the prior week (346 patients). Patients with AIDS or HIV positive titers were excluded due to the distinct differences in pneumonia etiologies and prognoses. For patients with more than one hospitalization for pneumonia in the database during the study period, (1008 patients), only the initial episode of pneumonia was evaluated.

From the MedisGroups key clinical findings, the Pneumonia PORT study team selected 47 of the most likely patient attributes (findings) considered to be useful for the prediction of risk of mortality in patients with community-acquired pneumonia. These attributes included patient demographics (age, sex), presence of comorbid conditions (diabetes mellitus, asthma, cancer, congestive heart failure, chronic renal insufficiency or failure, etc.), physical findings (vital signs, presence of wheezing, heart murmur, altered mental status, etc.), laboratory findings (pH, $pO_2$, $pCO_2$, electrolytes, BUN, creatinine, liver function tests, WBC, etc.), and radiologic findings (infiltrate, effusion, pneumothorax, mass, collapse, etc.). All of the 47 attributes typically are available to an emergency department physician who is deciding the disposition of a patient with community-acquired pneumonia.

The data were randomly divided into two mutually exclusive sets. One set, consisting of 9847 patients, was used for training (model learning) and the other set, consisting of 4352 patients, was used for testing. The dependent variable used for learning was the outcome of vital status (mortality) in the MedisGroups database. 'Alive' is defined as surviving to discharge or surviving more than 60 days in the hospital. The mortality rate in the training and test sets were 11.1% (1091 out of the 9847) and 10.4% (451 out of the 4352), respectively.

Learning a model to suggest a fraction of community-acquired patients which can be treated safely at home, should, ideally, involve data on *both* inpatients and

outpatients. Since the MedisGroups database contains only inpatients, any model derived from this data has, in reality, learned to predict 'low risk for inpatient mortality' versus 'higher risk for inpatient mortality.' Therefore, we will subsequently refer to the two predicted classes as 'low risk' and 'high risk.' We define 'low risk' as the positive class and 'high risk' as the negative class. Thus, a false positive prediction corresponds to the classification model predicting 'low risk' in a patient who was admitted and died in the hospital. Similarly, a false negative prediction corresponds to the model predicting 'high risk' in a patient who was admitted and survived in the hospital.

Model learning for this study was done in two stages. First, a rule induction machine-learning program, Rule Learner (RL), was used to generate a large set of plausible rules. Second, this large set of rules was given to a post-processing program, Optimizer (OP), whose task was to select a subset of these rules, based on misclassification error costs. Both the rule generation stage and rule selection stage used the training set only. Model testing was performed with the test set only.

RL is a knowledge-based, rule induction program under development in the Intelligent Systems Laboratory at the University of Pittsburgh [7,8]. The rules generated by this program are of the form 'if <feature 1> and <feature 2> and ... then the case is a member of <class X>.' We define a *feature* as an *attribute:value pair*; thus 'age' is an attribute, while 'age greater than 80' is a feature. If a test case satisfies all of the features on the *left-hand-side* of the rule (the *if* part), then the rule would cause that case to be predicted as the class specified by the *right-hand-side* of the rule (the *then* part). An example of a rule that predicts 'low risk' is "if (age < 23) and ($pO_2$ > 47.5) then predict the case as 'low risk.'" An example which predicts 'high risk' is "if (systolic blood pressure < 60) then predict the case as "high risk."

OP is a ruleset post-processor under development in the Section of Medical Informatics at the University of Pittsburgh [13]. It is designed to select a subset of rules from a ruleset (from any source) with the goal to improve classification on unseen test cases. OP can use either cost or accuracy as a measure of performance. The use of a misclassification cost matrix allows it to assign different costs to errors and to select a ruleset whose predicted total error cost on the test set is minimized.

In our study, the rules generated by RL predicted some cases to be 'low risk' and others to be 'high risk.' When two or more conflicting rules predicted the *same* case to be both 'low risk' and 'high risk,' it was necessary for OP to use a *evidence gathering procedure* to decide the class of the case. The conflict-resolution strategy used in this study was based on a weighted sum of rule strength, which is best illustrated with an example. Assume that two rules predicted a case to be 'high risk,' and three rules predicted the case to be 'low risk.' The case will be predicted to be 'high risk' if the sum of the ratings for the rules predicting 'high risk' exceeds the sum of the ratings for the rules predicting 'low risk.' If no rule predicts the class of a case (or if the weighted voting results in a tie), a default class is chosen. The default class used in this study was 'high risk' because this class has the *least expected cost* (i.e., it is the class with the least cost if all cases were predicted to be in that class).

OP selects a subset of rules from a larger ruleset using a two stage process. The first stage, which we shall call *rule ordering*, involves assigning an order to the rules. The second stage, which we shall call *ruleset evaluation and selection*, involves selecting a ruleset using the ordering from the first stage.

OP currently implements three different post-processing algorithms for the rule ordering stage. The algorithm used in this study begins by rating the strength of each of the rules. The rating function used was based on the FP:FN cost ratio and the performance of the given rule on the training data. The rule with the highest strength (rating) was selected as the first in an *ordered list*. The algorithm then recalculated the rule strength for each remaining rule. This time, however, the rating function also included additional information provided by the performance on the training data of the ruleset consisting of the ordered list. That is, given the ordered list of selected rules, the rating function was able to assign more weight to training cases not correctly classified by these rules (on the ordered list). After the ratings were recalculated, the best rule was added to the ordered list, and the process was repeated.

During the ruleset evaluation and selection stage, OP selected the single ruleset that performed best (i.e., has the lowest cost) on the training data. The first ruleset evaluated was the empty ruleset. (This corresponds to the rule 'predict all cases as high risk,' since the default class is 'high risk.' The second ruleset contains only the first rule from the ordered list, while the third ruleset contains the first two rules from the ordered list, and the fourth ruleset contains the first three rules from the ordered list, etc.

Table 2: A misclassification matrix used with OP.

| | **True Class** | |
|---|---|---|
| **Predicted Class** | Outpatient Rx | Admit |
| Outpatient Rx | 0 | FP:FN cost ratio |
| Admit | 1 | 0 |

OP was run a single time for each value of the misclassification matrix FP:FN cost ratio. Table 2 above shows the misclassification matrix used. The

Table 3: Table of partial results. The headings of the table are FP:FN (FP:FN cost ratio), TP (true positive predictions), FP (false positive predictions), FN (false negative predictions), TN (true negative predictions), SENS (sensitivity), SPEC (specificity), PPV (predictive value of an 'low risk' prediction), NPV (predictive value of an 'high risk' prediction), ACC (classification accuracy), and % Low Risk (percent of total test cases predicted as 'low risk.'). A positive prediction is a prediction of 'low risk.' A negative prediction is a prediction of 'high risk.'

| Cost Ratio | TP | FP | FN | TN | SENS | SPEC | PPV | NPV | ACC | % Low Risk |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 3883 | 422 | 18 | 29 | 0.995 | 0.064 | 0.902 | 0.617 | 0.899 | 98.9 |
| 2 | 3792 | 348 | 109 | 103 | 0.972 | 0.228 | 0.916 | 0.486 | 0.895 | 95.1 |
| 3 | 3675 | 294 | 226 | 157 | 0.942 | 0.348 | 0.926 | 0.410 | 0.881 | 91.2 |
| 4 | 3516 | 243 | 385 | 208 | 0.901 | 0.461 | 0.935 | 0.351 | 0.856 | 86.4 |
| 6 | 3324 | 182 | 577 | 269 | 0.852 | 0.596 | 0.948 | 0.318 | 0.826 | 80.6 |
| 8 | 3055 | 121 | 846 | 330 | 0.783 | 0.732 | 0.962 | 0.281 | 0.778 | 73.0 |
| 10 | 2838 | 88 | 1063 | 363 | 0.728 | 0.805 | 0.970 | 0.255 | 0.736 | 67.2 |
| 12 | 2514 | 60 | 1387 | 391 | 0.644 | 0.867 | 0.977 | 0.220 | 0.668 | 59.1 |
| 16 | 2347 | 50 | 1554 | 401 | 0.602 | 0.889 | 0.979 | 0.205 | 0.631 | 55.1 |
| 20 | 1943 | 35 | 1958 | 416 | 0.498 | 0.922 | 0.982 | 0.175 | 0.542 | 45.5 |
| 24 | 1776 | 25 | 2125 | 426 | 0.455 | 0.945 | 0.986 | 0.167 | 0.506 | 41.4 |
| 30 | 1506 | 15 | 2395 | 436 | 0.386 | 0.967 | 0.990 | 0.154 | 0.446 | 34.9 |
| 36 | 1405 | 14 | 2496 | 437 | 0.360 | 0.969 | 0.990 | 0.149 | 0.423 | 32.6 |
| 44 | 1277 | 11 | 2624 | 440 | 0.327 | 0.976 | 0.991 | 0.144 | 0.395 | 29.6 |
| 56 | 1029 | 9 | 2872 | 442 | 0.264 | 0.980 | 0.991 | 0.133 | 0.338 | 23.9 |
| 70 | 610 | 4 | 3291 | 447 | 0.156 | 0.991 | 0.993 | 0.120 | 0.243 | 14.1 |
| 80 | 286 | 1 | 3615 | 450 | 0.073 | 0.998 | 0.997 | 0.111 | 0.169 | 6.6 |

FP:FN cost ratio was initially set to 1, and was increased until no further increase in positive predictive value was obtained.

## RESULTS

Table 3 is a partial listing of the results. Each line in the table represents the results of the ruleset model selected by OP for the given FP:FN cost ratio. The results shown are on the *test* data only.

As the FP:FN cost ratio increases, Table 3 shows that the number of false positives (FP) decreases faster than the number of true positives (TP), so that the positive predictive value of a 'low risk' prediction (PPV) increases. Classification accuracy (ACC) and the percent predicted as 'low risk' also decrease as the positive predictive value and FP:FN cost ratio increase.

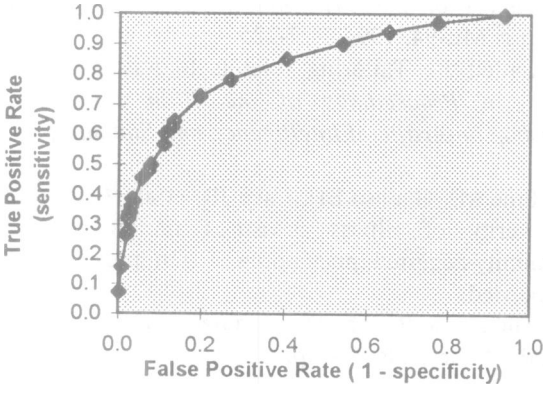


**Figure 1**

Figure 1 shows a plot of true positive rate (sensitivity) versus false positive rate (one minus specificity). Each line in Table 3 represents one point in Figure 1. The point in the upper right corner of the figure corresponds to the FP:FN cost ratio of 1, while the point in the lower left corner corresponds to the FP:FN cost ratio of 80.

## DISCUSSION

We have chosen to call the curve in Figure 1 an *ROC-like* curve because, each point represents a different learned model. Thus, in contrast to a traditional ROC curve, an ROC-like curve may not be strictly monotonically increasing. Nevertheless, this curve still provides the clinician a graphical representation of the data and can aid in determination of the most suitable model.

By varying the FP:FN cost ratio, a rule-based learning algorithm has been able to learn models with varying values of sensitivity and specificity. Clinician researchers can use the data in Table 3 or the curve plotted in Figure 1 to select an appropriate model for clinical application or to compare these results to another learning method. The clinicians involved in CEHC and PORT projects believe that an appropriate model will require a positive predictive value in the range of 0.990 (1% mortality rate in hospitalized patients when 'low risk' is predicted). Given this clinically useful range for positive predictive value, Table 3 shows that the 'best' model would *not* have the highest classification accuracy.

Since the purpose of this study is to describe a general methodology for developing a rule-based model in domains with specific sensitivity and specificity requirements (or predictive value requirements), we have purposely not emphasized the actual results of this rule-based learning system. For this domain, our clinician researchers have determined that the important parameters are the fraction of patients able to be assigned to the 'low risk' group (% Low Risk) and the prediction accuracy when 'low risk' is predicted (positive predictive value). (Table 3 shows the tradeoff between these two important parameters.) Models with similar positive predictive value can be compared using percent predicted as 'low risk.' Alternatively, models with similar percent predicted as 'low risk' can be compared using positive predictive value.

This study has some limitations. First, as previously stated, a model whose goal is to select community-acquired pneumonia patients for outpatient therapy, should be learned from a database which contains outcome data for both inpatients and outpatients. Since the MedisGroups data consist only of inpatients, *any model learned from this data makes the assumption that hospitalized patients with a very low mortality rate (i.e., those predicted as 'low risk'), will not have a higher mortality rate when treated as outpatients.* Second, this study also assumes that low risk patients demonstrate favorable clinically relevant outcomes other than mortality, such as morbidity, symptom resolution, and return to usual activity. To validate these assumptions, the chosen classification model will be evaluated using data from the Pneumonia PORT study, which includes both inpatient and outpatient data, and an assessment of other health outcome measures.

## CONCLUSIONS

We have shown how to use misclassification costs with statistical or machine-learning algorithms which produce categorical outputs to adjust for the sensitivity and specificity (or predictive value) requirements of a particular domain. We have illustrated this general technique by applying it to the construction of rule-based systems that predict inpatient pneumonia mortality.

## ACKNOWLEDGMENTS

We would like to thank Constantin Aliferis for his comments on earlier drafts of this paper. We are grateful to Wishwa Kapoor, the principal investigator of the Pneumonia PORT study, and to the other co-investigators, for the use of the data. Foster Provost provided guidance in the development of OP. This work was supported as part of the Cost-Effective Health-Care (CEHC) Project, which is supported by grant BES-9315428 from the National Science Foundation. This work was also supported in part by the National Library of Medicine under grant T15-LM07059, and in part by the Agency for Health Care Policy and Research as part of the Pneumonia Patient Outcomes Research Team (Pneumonia PORT) Project (R01 HS 06468). Dr. Fine is supported as a Robert Wood Johnson Foundation Generalist Physician Faculty Scholar.

## References

1. Cooper GF, Herskovits E. A Bayesian method for the induction of probabilistic networks from data. Machine Learning 9:309-347, 1992.

2. Spirtes P, Glymour C, Scheines R. Causation, Prediction, and Search. Springer-Verlag, New York, 1993.

3. Weiss S, Kulikowski C. *Computer Systems that Learn.* San Mateo: Morgan Kaufmann 1990.

4. McNeil BJ, Keeler E, Adelstein SJ. Primer on certain elements of medical decision making. N Engl J Med 293(5):211-15, 1975.

5. Quinlan JR. *C4.5: Programs for Machine Learning.* San Mateo: Morgan Kaufmann 1993.

6. Breiman L, Freidman JH, Olshen RA, Stone CJ. *Classification and Regression Trees.* Belmont: Wadsworth 1984.

7. Clearwater SH, Provost F. RL4: A tool for knowledge-based induction. In *Proceedings of the 2nd International IEEE Conference on Tools for Artificial Intelligence.* IEEE CS Press 1990, 24-30.

8. Buchanan BG. The Role of Experimentation in Artificial Intelligence. Phil Trans R Soc Lond A 349:153-166, 1994.

9. Clark P, and Niblett T. The CN2 algorithm. Machine Learning 3:261-283, 1989.

10. Pazzani M, Merz C, Murphy P, Ali K, Hume T, Brunk C. Reducing misclassification costs. In *Proceedings of the 11th Intern. Conf. on Machine Learning,* ML-94, 1994, 217-225.

11. Catlett J. Tailoring rulesets to misclassification cost. In *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics, 1995, 88-94.*

12. Fine MJ, Hanusa BH, Lave JR, et al. Comparison of severity of illness measures in patients with community-acquired pneumonia. (In press) J. Gen. Intern. Med.

13. Ambrosino R. The Development and Evaluation of a Method for Rule Post-processing. Masters Project Report, Intelligent Systems Program, University of Pittsburgh, December 1994.