EXTENDED ABSTRACT

Input Accuracy in Classification Tasks: Which Errors are More Painful?

I. Askira Gelman
DQIQ Research and Solutions

Abstract

Consider the following scenario: A bank develops a decision tree classifier for supporting loan approval decisions from a sample of error-free customer loan applications. The classification accuracy of this tree over test data is shown to be high, and the bank decides to use the classifier to support loan approval decisions in several branches of the bank. In general, the performance of such a classifier is not guaranteed to stay constant. An important source of variability in a classifier's performance is input data quality. Changes in input data quality may arise if some branches of the bank are less strict than others about data quality, such that respective loan application data are of lower quality, or if the bank's data quality policies or procedures change over time, etc. However, recent data quality studies hint to the possibility that errors in different subsets of the input data may not all have the same significance for the classifier's performance. Consequently, given the potential for variations in the quality of the input of the classifier over its useful lifetime, an ability to identify data subsets whose quality is critical to the performance of the classifier can translate, under the proper actions, to a significant financial gain for the organization.

We propose a solution that provides this ability. In particular, we have developed a quick, intuitive method for partitioning the input data of a decision tree classifier into subsets such that errors in each subset have a distinct effect on the performance of the tree, and a related algorithm for estimating the potential magnitude of that effect. The proposed solution was evaluated using Monte Carlo simulations. In this short document we illustrate our approach through the example of a decision tree classifier that was constructed from publicly available loan application data. Both the classifier and the loan application data have been described in the Information Systems literature. Despite the fact that our discussion assumes that the performance of a classifier is expressed by its accuracy, the proposed solution is easily applicable in the general case where the classifier's performance is measured in economic terms using a misclassification cost matrix.

Keywords: Classification, Decision tree classifier, Data quality management, Information quality.

1. Overview

Decision tree learning algorithms generate a tree-like structure, which is used for predicting the value of a target categorical variable. Each interior node corresponds to one of the input variables, and each leaf represents a predicted value of the target variable given the values of the input variables on the path from the root to the leaf. Decision tree classifiers are utilized in numerous organizational classification problems. A few examples include retail target marketing, customer retention, customer loan approval, fraud detection, and medical diagnosis.

Consider, for example, the following scenario: a bank develops a decision tree classifier for supporting loan approval decisions from a sample of error-free customer loan applications. The classification accuracy of this tree over test data is shown to be high, and the bank decides to use the classifier to support loan approval decisions on a regular basis in several branches of the bank. In general, the performance of such a classifier is not guaranteed to stay constant. An important source of variability in a classifier's performance is input data quality. Changes in input data quality may arise if some branches of the bank are less strict than others about data quality, such that respective loan application data are of lower quality, or if the bank's data quality policies or procedures change over time, etc. However, recent data quality studies hint to the possibility that errors in different subsets of the input data of the classification tree may not all have the same significance for the classifier's performance [1], [2], [4]. The bank is planning to study the potential effect of the data quality factor on the performance of the tree before this classifier is commissioned. Given the potential variability in the quality of the input of the classifier over its useful lifetime, an ability to identify data subsets whose quality is critical to the performance of the classifier can translate to a significant financial gain. The bank can use it to focus data quality management efforts on sensitive data subsets.

We propose a solution that provides this ability. In particular, we have developed a quick, intuitive method for partitioning the input data of a decision tree classifier into subsets, such that errors in each subset have a distinct effect on the performance of the tree, and a related method for estimating the potential magnitude of that effect [3]. The proposed solution was evaluated using Monte Carlo simulations. In this short document we illustrate our approach through the example of a decision tree classifier that was constructed from publicly available loan application data [5]. Both the decision tree classifier and the loan application data have been described in the Information Systems literature [6]. Despite the fact that our discussion assumes that the performance of a classifier is expressed by its accuracy, the proposed methods are easily applicable in the general case where the classifier's performance is measured in economic terms using a misclassification cost matrix.

Notably, [1], [2], and related studies by Askira Gelman report theoretical findings on a seemingly similar problem. However, a critical difference that makes this theory largely irrelevant to the current problem concerns the treatment of risk and uncertainty. While previous studies refer to conditions in which the decision is perceived to involve no uncertainty (the challenge is simply to study the alternatives and choose the best solution), the current research explores decisions under conditions of risk.

2. Illustration

This section presents several details of the proposed solution and illustrates the output of this approach through the example of a bank loan approval decision.

In general, the proposed methods are applied over a decision tree classifier *T* and a respective test data set *I*, such that the classification accuracy over *I* of every decision rule that makes part of this tree has been measured and is therefore known. We assume that this classification accuracy is maintained when the classifier is put into active service if input data quality stays as high as the data quality of *I*. However, our methods focus on situations in which input data quality is *not* expected to stay as high as the data quality of *I*.

Our algorithm for quantifying the effect of input errors uses estimates of classification accuracy and information about potential distributions of input data errors among the different input values. The latter includes information about typical errors and potential dependencies between errors in different inputs. This information may be obtained from people who work closely with the data. Nonetheless, the algorithm may actually be useful even when input error distributions are unknown since outcomes can be largely insensitive to input error characteristics. Analysis of the sensitivity of the outcomes to input error characteristics is an integral part of our approach.

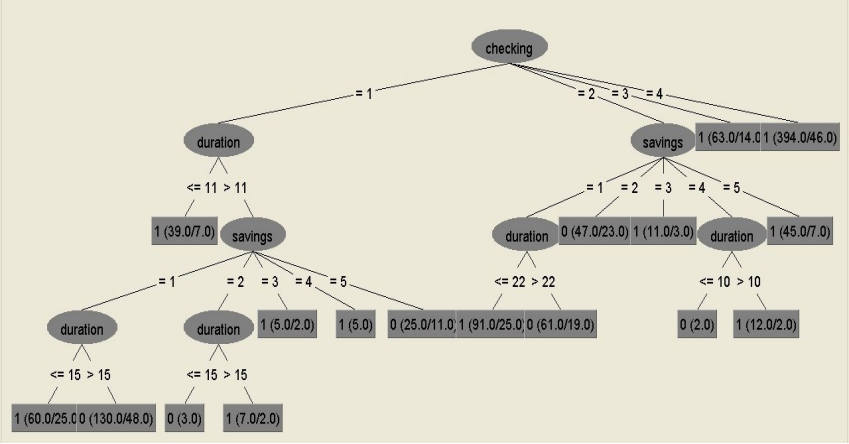

Figure 1: Loan decision tree classifier [6]

In the illustrative example that follows, *T* denotes a loan decision tree classifier that was recently presented by Zurada [6]. The loan data set that Zurada used in the development and testing of the classifier was originally contributed by a German financial institution and is now publicly available [5]. The data set contains 21 financial attributes for 1,000 customers, including an indicator of whether or not the loan was paid off. However, the inputs of *T* are limited to three of the 20 independent financial variables portrayed by that data set. In particular, the inputs of *T* consist of Checking

(checking account balance in Deutsche Marks (DM)—this is a categorical attribute with four possible values, 1, 2, 3, and 4; see the legend in Figure 2), Duration (duration of the loan in months—numeric), and Savings (savings account balance—five possible values, 1, 2, 3, 4, and 5; see Figure 2). These inputs are processed by *T* to predict whether or not the customer paid off the loan (see Figure 1 and Figure 2). Therefore, *I* denotes the publicly available collection of 1000 instances of the values of these three inputs. *I* is believed to be error-free and the classification accuracy over *I* of each of the 17 decision rules that *T* implies is specified in [6].

1. Checking=1	11<Duration≤15	Savings=1	payoff
2. Checking=1	Duration≤11		payoff
3. Checking=1	Duration>15	Savings=1	default
4. Checking=1	11<Duration≤15	Savings=2	default
5. Checking=1	Duration>15	Savings=2	payoff
6. Checking=1	Duration>11	Savings=3	payoff
7. Checking=1	Duration>11	Savings=4	payoff
8. Checking=1	Duration>11	Savings=5	default
9. Checking=2	Duration≤22	Savings=1	payoff
10. Checking=2	Duration>22	Savings=1	default
11. Checking=2		Savings=2	default
12. Checking=2		Savings=3	payoff
13. Checking=2	Duration≤10	Savings=4	default
14. Checking=2	Duration>10	Savings=4	payoff
15. Checking=2		Savings=5	payoff
16. Checking=3			payoff
17. Checking=4			payoff

Legend:
(a) Checking Account Balance
1 = less than 0 DM
2 = more than 0 but less than 200 DM
3 = at least 200 DM, and
4 = no checking account
(b) Duration of Loan [in months]
(c) Savings Account Balance
1 = less than 100 DM
2 = at least 100, but less than 500 DM
3 = at least 500, but less than 1000 DM
4 = at least 1000 DM
5 = unknown / no savings account

Figure 2: Loan classification rules

Partitioning *I*

Our partitioning method is based on two key observations:

1. The effect of an input error *e* on the accuracy of *T* depends on the rule that matches the instance in which *e* occurs (e.g., the effect of an error in an instance where Checking=1, Duration=18, and Savings=2 is related to rule 5 in Figure 2).

2. The effect of an input error *e* on the accuracy of *T* depends on the input in which *e* occurs (e.g., the effect of an error in the value of Checking is generally different from the effect of an error in the value of Duration).

In our example, accordingly, the outcome of partitioning *I* consists of 51 data subsets since *T* has three inputs and 17 rules.

Quantifying the effect of errors in individual data subsets

We quantify the effect of input errors on classification accuracy using a number that ranges from minus one (-1) and one (+1). Such a number, conveniently labeled *damage* [1],[2], designates the *mean change* of *T*'s classification accuracy when an error *e* is added to a specified subset of *I*. If *e* is guaranteed to add a classification error then the damage is equal to one; if *e* is guaranteed to eliminate a classification error, the damage is equal to minus one; if *e* has no effect on classification accuracy, the damage is equal to zero; if the mean number of classification errors added due to *e* is 0.2 then the damage is 0.2, and so on. Presumably, all other things being equal, one would prefer to prevent an error in a subset of *I* where the damage value is higher.

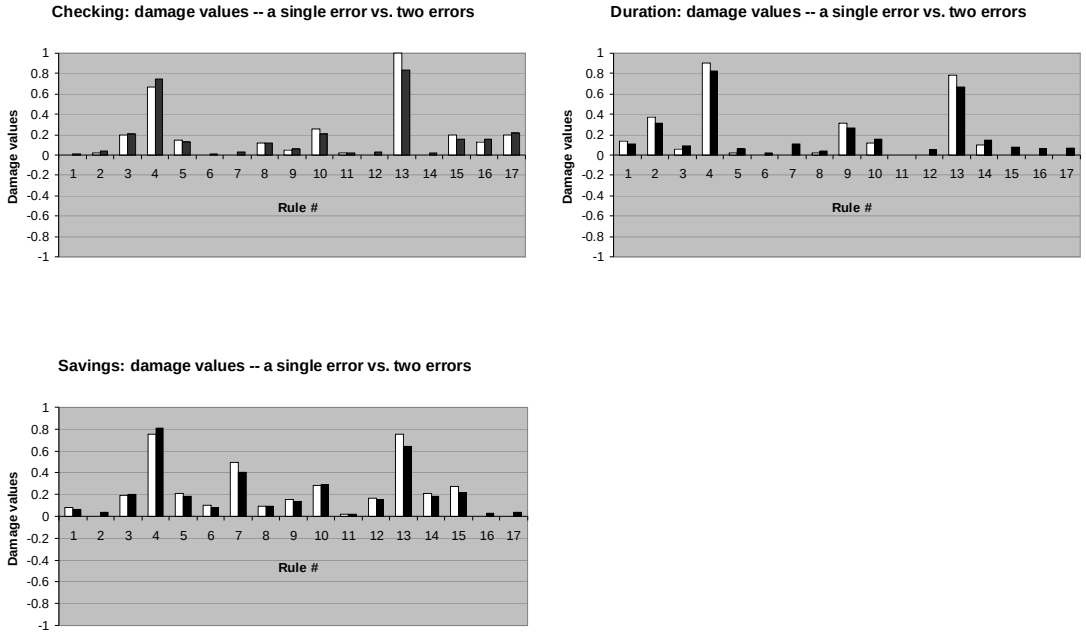

Figure 3: Damage values. Each of the three charts portrays the 17 data subsets that match the specified input; white columns correspond to the assumption that instances can have one error at the most; black columns correspond to the assumption that instances can have up to two errors.

As mentioned before, our algorithm for estimating damage values uses information about the potential distributions of input data errors among the different input values. In the loan decision example, such information may be obtained from bank employees who

work with customer loan applications in the branches that plan to use the classifier. In the absence of such data, our estimates of the damage in each of the 51 data subsets outlined previously assume that errors are randomly distributed and an error in an input produces a value that is uniformly distributed in the range of possible values of the input. As for dependencies between errors in different inputs, estimates of the damage are calculated under two scenarios: (1) a customer loan data instance (three values) contains one error at the most (2) a customer loan data instance contains up to two errors and the probability that an error in one input is accompanied by an error in another input is quite high, 0.25.

Figure 3 portrays the damage values that were derived for *T* and *I* as above. For the purpose of depicting the damage in the second scenario (the black columns in Figure 3), we calculated two damage values depending on the assumed location of the second error, and then depicted in Figure 3 the damage value that showed the highest gap from the value under the first scenario (the white columns in Figure 3).

As demonstrated by Figure 3, damage values vary dramatically. Some data subsets reveal absolute insensitivity of the classifier to error incidence, such that the damage is zero; while in other data subsets an error has a very high likelihood (even certainty) of triggering a classification error. This high variation in damage values signals that the new solution approach can indeed be useful.

Figure 3 also suggests that the difference between the damage values under the two scenarios considered is small. In other words, damage values are fairly insensitive to the difference between the two scenarios. Notably, if a practical application of our approach finds damage values to be insensitive to assumptions on error distributions, then data quality management should interpret this insensitivity as a positive sign of the validity of the damage estimates (as well as their potential usefulness in a diverse and/or dynamic environment).

References

1. Askira Gelman, I. “Simulations of Error Propagation for Prioritizing Data Accuracy Improvements in Multi-Criteria Satisficing Decision Making Scenarios.” *International Conference on Information Systems* (ICIS), 2009.
2. Askira Gelman, I. “GIGO or not GIGO: The Accuracy of Multi-Criteria Satisficing Decisions.” *ACM Journal of Data and Information Quality* (ACM JDIQ), 2011.
3. Askira Gelman, I. “Input accuracy in classification tasks: Which errors are more painful?” Submitted for publication.
4. Ballou, D. P., Pazer, H. L., Belardo, S., and B.D. Klein, Implications of Data Quality for Spreadsheet Analysis, *DATA BASE*, Vol. 18, No. 3, 1987.
5. Hofmann, H. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, 1994.
6. Zurada, J. “Could Decision Trees Improve the Classification Accuracy and Interpretability of Loan Granting Decisions?” *Hawaii International Conference on Systems Sciences* (HICSS-43), 2010.