Proc. ICANN/ICONIP 2003, Istanbul, Turkey

Hard drive failure prediction using non-parametric statistical methods

Joseph F. Murray, Gordon F. Hughes and Kenneth Kreutz-Delgado

Abstract— **We present a case study of a difficult real-world pattern recognition problem: predicting hard drive failure using attributes monitored internally by individual drives. We compare the performance of support vector machines (SVMs), unsupervised clustering, and non-parametric statistical tests (rank-sum and reverse arrangements). Somewhat surprisingly, the rank-sum method outperformed the other methods, including SVMs. We also show the utility of using non-parametric tests for feature set selection.**

Keywords— **failure prediction, hard drive reliability, ranksum, reverse arrangements, support vector machines, Autoclass.**

I. Introduction

Since 1994, hard drive manufacturers have been developing self-monitoring technology in their products, in an effort to predict failures early enough to allow users to backup their data [1]. The Self-Monitoring and Reporting Technology (SMART) system uses attributes collected during normal operation (and during off-line tests) to set a failure prediction flag. The SMART flag is a one-bit signal that can be read by operating systems and third-party software, designed to warn users of impending drive failure. Some of the attributes used to make the failure prediction include counts of track seek retries, write faults, reallocated sectors, head fly heights, and high temperature. Most attributes are error count data, implying positive integer data values, and a pattern of increasing attribute values over time is indicative of impending failure. (Drives internally detect and correct many of these errors to access user data properly). Each manufacturer develops and uses their own set of attributes and their own algorithm for failure prediction. Because every time a failure warning is triggered, the drive could be returned to the factory for warranty replacement, manufacturers are very concerned with reducing the false alarm rate of the algorithm. Currently, all manufacturers use a threshold algorithm which triggers a SMART flag when any attribute exceeds a predefined value. The thresholds are set conservatively to avoid false alarms at the expense of predictive accuracy, with an acceptable false alarm rate on the order of 0.1%. For the SMART algorithm currently implemented in drives, manufacturers estimate the detection rate to be $3-10\%$. Our previous work has shown that by using non-parametric statistical tests, the accuracy of correctly detected failures can be improved over the manufacturer's threshold rules while maintaining low false alarm rates [1], [2].

E-mail: J. F. Murray jfmurray@ucsd.edu, G. F. Hughes gfhughes@ucsd.edu, K. Kreutz-Delgado kreutz@ece.ucsd.edu

II. Data Description

The dataset consists of time series of SMART attributes from one drive model from a single manufacturer (a different manufacturer and drive model than in [1]). Data from 369 drives was collected, and each drive was labelled *good* or *failed*. Drives labelled as good were from a reliability demonstration test, run in a controlled environment by the manufacturer. Drives labelled as failed were returned to the manufacturer from users after a failure. It should be noted that since the good drive data was collected in a controlled uniform environment and the failed data comes from drives that were operated by users, it is reasonable to expect that there will be differences between the two populations due to the different manner of operation. Algorithms that attempt to learn the difference between the good and failed populations may in fact be learning this difference and not the desired difference between good and nearly-failing drive samples. We highlight this point to emphasize the importance of understanding the populations in the data and considering alternative reasons for differences between classes.

Each SMART sample was taken at two hour intervals in the operating drives, and the most recent 300 samples are saved on the disk. Each sample contains the drive's serial number, the total power-on-hours, and 60 other performance-monitoring attributes.

III. Feature Selection

As will be demonstrated below, some attributes are not strongly correlated with future drive failure and including these attributes can have a negative impact on the classifier performance. Because it is computationally expensive to try all combinations of attribute values, we use a fast non-parametric test to identify potentially useful attributes.

A. Reverse arrangements test

The *reverse arrangements test* is a non-parametric test for trend, which is applied to each variable in the dataset [3], [4]. Suppose we have a ordered sequence of observations of a random variable, $x_i, i = 1...N$. The test statistic A is the sum of all reverse arrangements, where a reverse arrangement is defined as an occurrence of $x_i > x_j$ when $i < j$. To find A we use the intermediate sums A_i and the indicator h_{ij},

$$A = \sum_{i=1}^{N-1} A_i \quad , \tag{1}$$

where,

$$A_i = \sum_{j=i+1}^{N} h_{ij} \tag{2}$$

$$h_{ij} = \begin{cases} 1 & \text{if } x_i > x_j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

An example of calculating A is given in Bendat [4], pg 107. For sample size N and significance level α, Appendix Table A.6 of [4] gives the acceptance regions,

$$A_{N;1-\alpha/2} < A \leq A_{N;\alpha/2}\,, \tag{4}$$

for the null hypothesis of no trend in the sequence x_i (i.e. that x_i are independent observations of the same underlying random variable).

The test is formulated assuming that the values are drawn from a continuous distribution. SMART error count data values are discrete and allow the possibility of ties. It is conventional in rank-based methods to add random noise to break the ties, or to use the *midrank* method described in Section IV-C.

B. Application to SMART data

To apply the reverse arrangements test to the SMART data for the purpose of feature extraction, the test is performed on a set of 100 samples taken at the end of the time series available. To break ties, uniform random noise within the range $[-0.1, 0.1]$ is added to each value (which are initially non-negative integers). The percentage of drives for which the null hypothesis of no trend is rejected is calculated for good and failed drives. Table I lists attributes and the percent of drives that have significant trends for the good and failed populations. The null hypothesis (no trend) was accepted for $1968 \leq A \leq 2981$, for a significance level higher than 99%. We are interested in attributes that have both a high percentage of failed drives with significant trends and a low percentage of good drives with trends, in the belief that an attribute that increases over time in failed drives while remaining constant in good drives is likely to be informative in predicting impending failure. From Table I we can see that attributes such as BadSamp, MaxVisibleReadErr and MaxSeekErr could be useful predictors.

TABLE I

Percent of drives with significant trends by the reverse arrangements test for selected attributes.

Attribute	% Good	% Failed
MinimumTemp	34.8%	42.9%
MaximumTemp	8.4%	58.9%
FHSigmaOD	15.7%	21.4%
FHSigmaID	0.6%	10.7%
GListEntries	0.6%	10.7%
SListEntries	0.6%	3.6%
BadID	0.0%	0.0%
BadSamp	0.6%	30.4%
NoTMD	0.6%	0.0%
Spinups	97.2%	92.9%
ThirdSeek	53.9%	46.4%
OneTrackSeek	1.7%	25.0%
SectorsRead	1.1%	37.5%
SectorsWritten	0.6%	32.1%
L1ReadErr	0.0%	0.0%
L2ReadErr	0.6%	5.4%
MaxWriteFaults	1.1%	0.0%
MaxVisibleReadErr	0.0%	41.1%
MaxHiddenReadErr	0.0%	0.0%
MaxTAErrors	0.0%	0.0%
MaxGListEntries	0.0%	0.0%
MaxSListEntries	0.0%	8.9%
MaxSeeks	0.6%	5.4%
MaxSeekErr	1.7%	39.3%

IV. Failure detection algorithms

We describe how the pattern recognition algorithms and statistical tests are applied to the SMART dataset for failure prediction.

A. Support vector machines

Support vector machines (SVM) are popular modern pattern recognition and regression algorithms. First developed by Vapnik [5], the principle of the SVM classifier is to project the data into a higher dimensional space where the classes are separated by a linear hyperplane which is defined by a small set of support vectors. We use the widely-available MySVM package developed by Ruping [6].

To apply the SVM to the SMART dataset, drives are randomly assigned into training and test sets. The training set consists of 25% of good drives and 10% of failed drives. For validation, means and standard deviations of detection and false alarm rates are found over 10 trials with different training and test sets.

Two types of preprocessing must be done before presenting the data to the SVM classifier. A vector x of five consecutive samples of each attribute is used to make the classification, and every five consecutive samples in the history of the drive is used. (Sample lengths of between 2 and 15 were tried, with 5 providing the best performance). The length of x is $(5 \times \#\ of\ attributes)$. If any x is classified as failed, then the drive is predicted to fail. Since the classifier is applied repeatedly to different vectors of samples from the same drive, each test must be very resistant to false alarms. The first type of preprocessing is *magnitude sorting*; the vector x is sorted in descending value for each attribute, with the largest samples placed in lower numbered indices of x. Because the range of values the different attributes may take can differ widely (for drive technology reasons), the attribute values are binned into quartiles, with a special category for zero values. Thus, the range that each attribute can take is restricted to $[0, 1]$.

Parameters for the MySVM program are set as follows (see [6] for details): $epsilon = 10^{-2}$, $max\_iterations = 10000$, $convergence\_epsilon = 10^{-3}$, no_scale. The parameters C, $L+$ and $L-$ were varied to adjust the tradeoff between detection and false alarms.

B. Clustering (Autoclass)

Unsupervised clustering algorithms can be used for anomaly detection. Here, we use the Autoclass package [7] to learn a probabilistic model of the training data from only good drives. Any sample that is an anomaly (outlier) from the learned statistical model of good drives is used as a failure prediction. The *expectation maximization (EM)* algorithm is used to find the highest-likelihood mixture model that fits the data. A number of forms of the probability density function (pdf) are available, including Gaussian, Poisson (for integer count data) and nomial (unordered discrete, either independent or covariant). For the hard drive problem, they are all set to independent nomial to avoid assuming a parametric form for any attribute's distribution. This choice results in an algorithm very closely related to the *naive Bayes EM* algorithm [2], which was found to perform well on earlier SMART data.

Before being presented to Autoclass the attribute values are discretized into equal-sized bins, where the bin range is determined by the maximum range of the attribute in the training set (of only good drives). Four bins were used, with an extra bin for zero-valued attributes. The training procedure attempts to find the most likely mixture model to account for the good drive data. The number of clusters can also be determined by Autoclass, but here we have restricted it to a small fixed number. During testing, the estimated probability that a sample belongs to each cluster is calculated, and the sample is assigned to the most likely one. A failure prediction warning is triggered for a drive if the probability of any of its samples is below a threshold. To increase robustness, the input vector consists of two consecutive samples of each attribute (as described above for the SVM). The Autoclass threshold parameter was varied to adjust tradeoff between detection and false alarm.

C. Rank-sum test

The Wilcoxon-Mann-Whitney rank-sum test is used to determine if the two random data sets arise from the same probability distribution [8] (pg. 5). One set T comes from the drive under test and the other R is a *reference set* composed of samples from good drives. The use of this test requires some assumptions to be made about the distributions underlying the attribute values and the process of failure. Each attribute has a *good distribution* G and an *about-to-fail distribution* F. For most of the life of the drive, each attribute value is chosen from the G, and then at some time before failure, the values begin to be chosen from F. This model posits an abrupt change from G to F, however, the test should still work if the distribution changes gradually over time, and only give a warning when it has changed significantly from the reference set.

The test statistic W_S is calculated by ranking the elements of R (of size n) and T (of size m) such that each element of R and T has a rank $S \in [1, n+m]$ with the smallest element assigned $S = 1$. The rank-sum W_S is the sum of the ranks S of the test set.

The rank-sum test is often presented assuming continuous data. The attributes in the SMART data are discrete which creates the possibility of ties. Tied values are ranked by assigning identical values to their *midrank* [8] (pg. 18), which is the average rank that the values would have if they were not tied. For example, if there were three elements tied at the smallest value, they would each be assigned the midrank $\frac{1+2+3}{3} = 2$.

If the sample sizes are large enough (usually, if the smaller sample $m > 10$ or $n+m > 20$), the rank-sum statistic W_S is normally distributed under the null hypothesis (T and R are from the same population) due to the central limit theorem, with mean and variance:

$$E(W_S) = \frac{1}{2}n(n+m+1) \quad (5)$$

$$Var(W_S) = \frac{nm(n+m+1)}{12} - C_T \quad (6)$$

where C_T is the ties correction, defined as,

$$C_T = \frac{mn \sum_{i=1}^{e} (d_i^3 - d_i)}{12(m+n)(m+n-1)} \quad (7)$$

where e is the number of distinct values in R and T, and d_i is the number of tied elements at each value. The probability of a particular W_S can be found using the standard normal distribution, and a critical value α can be set at which to reject the null hypothesis. In cases of smaller samples where the central limit theorem does not apply, an exact method of calculating the probability of the test statistic can be used (see [1] and [9] for details).

For application to the SMART data, the reference set R for each attribute (size $n = 50$) is chosen at random from the samples of good drives. The test set T (size $m = 5$) is chosen from consecutive samples of the drive under test. If the test set for any attribute over the history of the drive is found to be significantly different from the reference set R the drive is predicted to fail. The significance level α is adjusted in the range $[10^{-7}, 10^{-1}]$ to vary the tradeoff between false alarms and detections. We use the one-sided test of T coming from a larger distribution than R, against the hypothesis of identical distributions.

D. Reverse arrangements tests

The reverse arrangements test described above for feature selection can also be used for failure prediction. No training set is required, as the test is used to determine if there is a significant trend in the time series of an attribute. For use with the SMART data, 100 samples are used in each test, and every consecutive sequence of samples is used. For each drive, if any test of any attribute shows a significant trend, then the drive is predicted to fail. As with the rank-sum, the significance level α controls the detection/false alarm tradeoff.

V. Results

Figure 1 shows the failure prediction results using the SVM and Autoclass classifiers with 25 attributes that were selected because of promising reverse arrangements test or

z-score values. Although both classifiers appear to have learned some aspects of the problem, the SVM was clearly superior, yet even with the modest 17.5% detection, the 2.3% false alarm is much higher than desirable (compared to the low actual failure rates of hard drives).

Using the reverse arrangements test (Table I) we postulate that using an individual attribute could improve the performance over that shown in Figure 1. The MaxVisibleReadErr attribute appears promising (with 41.1% of failed drives showing significant trends). Figure 2 shows the failure prediction results using the only MaxVisibleReadErr attribute. The rank-sum test provided the best performance, with 24.3% detection with false alarms too low to measure, and 33.2% detection and 0.5% false alarms. While Autoclass performed better than the reverse arrangements test, the false alarm rate with Autoclass could not be adjusted lower than 1.0% (as could be done with the other two tests). The SVM was unable to detect failures using this attribute.

Using combinations of attributes in the rank-sum test can lead to improved results over single-attribute classifiers. With MaxVisibleReadErr and MaxHiddenReadErr attributes, 43.1% of failures are detected with 0.6% false alarms. If even lower false alarm rates are needed, this combination of attributes can detect 25.0% of failures with no measured false alarms.

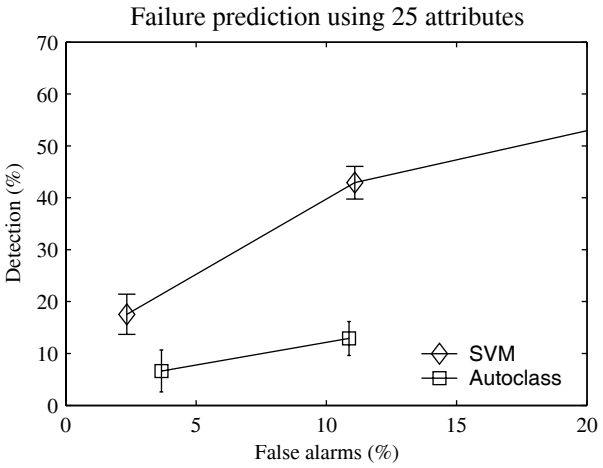

Fig. 1. Failure prediction performance of SVM and Autoclass using 25 attributes. Error bars are ±1 standard error.

VI. Conclusions

We have shown that the non-parametric rank-sum test can be used for pattern recognition, and that it can have higher performance than SVMs or unsupervised clustering on the hard drive failure prediction problem. The best performance occurred when using a small set of attributes (or a single attribute). Adding additional features increased the rate of false alarms. Attributes useful for failure prediction were selected by using the reverse arrangements test for increasing trend.

Improving the performance of hard drive failure prediction will have many practical benefits. Increased accuracy of detection will benefit users by giving them an opportunity to backup their data. Very low false alarms (in the range of 0.1%) will reduce the number of returned good drives, thus lowering costs to manufacturers of implementing improved SMART algorithms. While we believe the algorithms presented here are of high enough quality to be implemented in drives, it is still important to test them on larger number of drives (on the order of thousands) to measure accuracy to the desired precision of 0.1%.

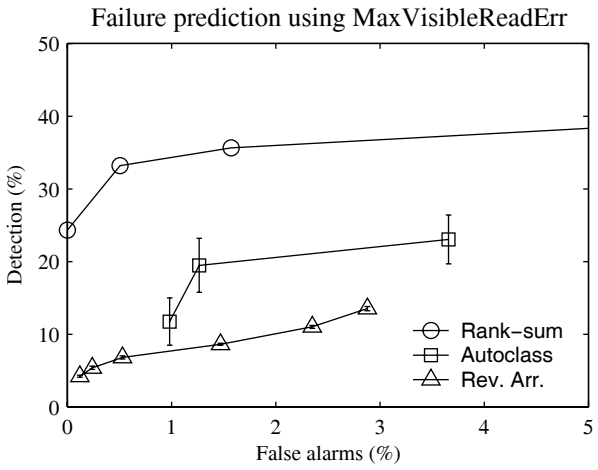

Fig. 2. Failure prediction performance of classifiers using MaxVisibleReadErr attribute. Error bars are ±1 standard error. For rank-sum and reverse arrangements, error bars are smaller than line markers.

Acknowledgments

This work is part of the UCSD Intelligent Disk Drive Project funded by the Information Storage Industry Center (a Sloan Foundation Center), and by the UCSD Center for Magnetic Recording Research. J. F. Murray gratefully acknowledges support by the Arcs Foundation.

References

[1] Gordon F. Hughes, Joseph F. Murray, Kenneth Kruetz-Delgado, and Charles Elkan, "Improved disk-drive failure warnings," *IEEE Transactions on Reliability*, vol. 51, no. 3, pp. 350–357, September 2002.

[2] Greg Hamerly and Charles Elkan, "Bayesian approaches to failure prediction for disk drives," in *Eighteenth International Conference on Machine Learning*, 2001, pp. 1–9.

[3] Henry B. Mann, "Nonparametric tests against trend," *Econometrica*, vol. 13, no. 3, pp. 245–259, 1945.

[4] Julius S. Bendat and Allan G. Piersol, *Random Data*, Wiley, New York, 3rd edition, 2000.

[5] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer Verlag, New York, 1995.

[6] Stefan Ruping, "mySVM manual," Tech. Rep., University of Dortmund, CS Department, AI Unit, October 2000.

[7] P. Cheeseman and J. Stutz, *Advances in Knowledge Discovery and Data Mining*, chapter Bayesian Classification (AutoClass), pp. 158–180, AAAI Press, Menlo Park, CA, 1995.

[8] E. L. Lehmann and H. J. M. D'Abrera, *Nonparametrics: Statistical Methods Based on Ranks*, Prentice Hall, Upper Saddle River, NJ, 1998.

[9] Joseph F. Murray, "Hard drive failure prediction: Description of data sets and methods," Center for Magnetic Recording Research (CMRR) Technical Report, UCSD, 2000.