Save the Data! An Intelligent Approach to Avoid Data Loss

Marcos Iseki, Bruno Nogueira, Brivaldo Junior[1]

[1]Faculty of Computing (Facom) – Federal University of Mato Grosso do Sul (UFMS)
79.070-900 – Campo Grande – MS – Brazil

{iseki, bruno, brivaldo}@facom.ufms.br

Abstract. *Data loss can harm customers, business strategies and companies reputation. While enterprise environments commonly employ data replication technologies as RAID, small business and customers rely on the lifetime of their storage devices, mostly hard drives. Thus, as these hard drives fail, massive data losses may happen. When important data is at stake, being aware of possible disk fails is crucial. In this sense, hard disks use SMART technology to try to detect failures. These analysis, however, are carried out only when operational system requires or during boot process. Moreover, these predictions are not very accurate, presenting small accuracies and high false positive rates. To avoid such problems, we propose a machine learning approach to detect hard drive failures. We use a huge and recent dataset from Blackblaze. Decision trees achieved the best performance with 80% in accuracy rate and less than 12% in false positive rate in failure predictions.*

1. Introduction

Failures in computer hard disk drives (HDDs) may cause significant data loss. Either permanent or temporary, data losses are normally associated with costs of availability and working time, not to mention the importance of data itself. In 2010, industrial level sectors like energy, telecommunications, or manufacturing could have a lost revenue that ranged between \$1.6 and \$2.8 million per hour of computer downtime, or data loss [Kroll Ontrack Inc. 2010]. For small businesses, that could mean their business survival, and for customers, the loss of important personal data.

Despite backup being a typical countermeasure against that, in 2016, around 24% of American computer users never replicated all data [Klein 2016]. That percentage reaches 66% if we add yearly and less often distributions. On the opposite side, only 8% of users backed up personal information once a day or more often, meaning not only that a large amount of data is in danger of loss, but that the associated risks and costs are not widespread yet.

At the same time, cloud storage fees are still high, mainly for small businesses and customers. Considering average prices of services in the first quarter of 2017, in less than two years, a client paying for a 1 TB storage service could purchase a brand new external HDD of same capacity.

In the high scale scenario, data center storage installed capacity doubles every year since 2015. Cisco Systems predicts that, by 2020, the amount of data stored in data centers and on client devices will reach 915 EB and 5.3 ZB, respectively [Cisco Inc. 2016]. Also, every year, HDDs' vendors sell hundreds of millions of devices, being HDDs the most common computer storage appliance for both enterprise and customer segments.

Another survey conducted by Kroll Ontrack Inc. revealed some causes of data loss and the impact to businesses and home users [Kroll Ontrack Inc. 2014]. The survey was carried out in ten countries across North America, Europe and Asia Pacific, in 2014. Hardware crash or failure was accountable for 66% of data loss, and within that segment, HDD crashes for 72% of imputed causes. Individual response segments also confirmed HDD failure as the main reason of data loss, among 71% of businesses and 72% of home users.

Hard disk Annualized Failure Rate (AFR) is a disk failure probability within one year of full use. It is calculated by manufacturers, and specified in device datasheets, being an important parameter to calculate drive reliability. Manufacturers get this metric during stress tests, reporting a scaled score. Data centers and other companies that work with large HDD population may calculate their annual failure rate, calling it either AFR or Annual Replacement Rate (ARR). Hence, ARR is not an estimation, but the yearly ratio between failed drives and total number of a drive population.

Schroeder and Gibson confirmed a disagreement between datasheet AFRs, and ARRs acquired from field-gathered disk replacement data [Schroeder and Gibson 2007]. The authors analyzed a set of $100,000$ HDDs from different vendors, running different systems, and interfaces (SCSI, FC, and SATA). Nominal AFRs ranged between 0.58% and 0.88%. However, ARRs started from 0.5%, reaching 13% in some systems, more than 14.7 times higher than stated in datasheets.

Besides, failure rates are expected to follow a pattern known as "bathtub curve", which means both high early-failure (also referred to as infant mortality rate) and wear-out rates, and in-betweens a low constant failure rate, as shown in Figure 1. However, Schroeder and Gibson also confirmed that ARRs increase within a period of five years, contradicting the low constant failure rate. Pinheiro [Pinheiro et al. 2007] also verified similar analysis over a different dataset, as we will discuss on Section 2.

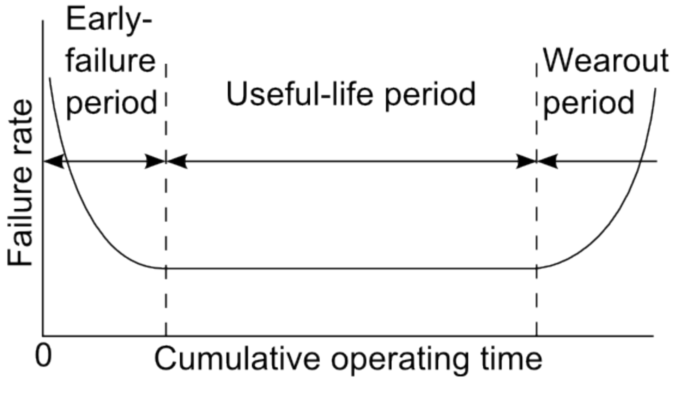

Figure 1. HDD failure pattern, or the "bathtub curve" [Yang and Sun 1999].

After a HDD crash, there are some cases when data retrieval is possible through drive recovery, but cost of services are considerably high. It changes mainly according to severity of the damage, with an average price of $\$1,325$ [Dell inc. 2016] per drive, at least 4-5 times more expensive than a brand new HDD.

When even backups may fail, **S**elf-**M**onitoring, **A**nalysis, and **R**eporting

Technology (SMART) is a feature that allows failure prediction of SMART-capable drives. SMART is an Advanced Technology Attachment (ATA) standard since 2003. In the current version, the system works by taking a snapshot of 45 device health parameters, comparing them with vendor specific thresholds. At operational system boot screen, SMART is able to alert the user when a failure is imminent, providing an estimated 3-10% detection rate with 0.1% false alarms [Murray et al. 2005].

This detection rate could be significantly improved using machine learning algorithms [Mitchell 1997]. Previous works exploit the construction of predictive models in order to anticipate disk failures using SMART information presenting significantly better results in terms of precision and false alarms [Murray et al. 2005, Pinheiro et al. 2007, Pitakrat et al. 2013]. Some of these works, however, are focused on predicting failures in specific HDD models. Besides, disk failures evolve as the technology evolves. Thus, some predictive models may be now outdated.

From these considerations, in this work we evaluate whether different machine learning algorithms increase accuracy classification of impending failures of diverse SMART-capable devices. We compare the models by their accuracy and execution time. We also intend to test whether our findings either validate or contradict previous work found in the literature.

This paper is organized as follows. In Section 2 we discuss some previous related works. In Section 3 we describe the dataset. In Section 4 we describe the HDD failure prediction model, and analyze the gathered results. Finally, in Section 5 we highlight conclusions, and future works.

2. Related work

Murray et al. analyzed a set of $68,411$ instances of 64 HDD attributes from a specific vendor, including the class value, which can be either *good* or *failed*, represented as boolean values [Murray et al. 2005]. Each instance has SMART logs collected every two hours from 369 single model HDDs, with 191 failed units, and 178 good ones. The authors compared the performance of two machine learning models: SVM and Multiple Instance Naïve Bayes (mi-NB). The Support Vector Machines achieved 50.6% in detection rate (accuracy) and 0% in false alarm. On the other hand, mi-NB achieved 34.5% in detection rate and 1.0% in false alarms. They mentioned that SVM was computationally expensive, although it had an excellent false positive rate. Also, mi-NB was suitable for on-line tests due to lower memory and computation requirements.

We were able to identify at least three weaknesses in Murray's study, two regarded to dataset, and one to result. It is important to mention that these weak points are opportunities to expand research in this area, and are not necessarily mistake notations. First, the dataset covered a small collection of HDDs all from the *same model*, possibly reducing capacity of the trained algorithms to interpret test samples from different models. Second, SMART logs of failed drives were taken *after* drives were declared failed by customers. This means that the algorithms learned to classify drives after they failed. The dataset we used covers $41,980,868$ entries, 55 different HDD models from five major manufacturers. All data related to failed drives were taken on their last day in operation.

Finally, when we analyze results, it is possible to say that 50.6% or 34.5% detection rate for binary classification is statistically close to randomly categorize the object in focus, like flipping a coin. In this study, we have reached both higher accuracy and less computational time.

Pinheiro et al. studied failure patterns of HDDs in a large disk drive population from different manufacturers, all running some of Google's systems [Pinheiro et al. 2007]. Google developed a System Health infrastructure which basically collected, stored, and mined data from servers and other data bases. The time-series information includes environment temperature, utilization rate, repair events, and some SMART parameters.

The data analysis revealed four important findings: first, either temperature or activity level have little correlation with failure rates; second, a total of four SMART parameters had a large impact on failure probability, so that drives are significantly more likely to fail after an occurrence of those SMART variables; third, AFRs does not follow the "bathtub curve", as manufacturers expected; and fourth, 36% of all failed drives had zero counts on all SMART variables. Based on this third finding, the authors concluded that SMART data alone is not effective to build HDD failure prediction models. The results we have found, though, add a new comprehension over SMART parameters, and their contribution to HDD failure prediction.

Pitakrat et al. preprocessed the dataset used in Murray et al. [Murray et al. 2005], classifying instances into two classes, 7 days before failing (as DAY_7 class) and not failing (as INF class) [Pitakrat et al. 2013]. They also selected 26 out of 64 original parameters, eliminating drive serial numbers, hours of operation, and other constant attributes. The authors compared prediction quality and time required to build training and prediction phases of 21 machine learning algorithms. The results suggested that algorithms with high prediction quality are nearest neighbor classifier, random forest, C4.5, REPTree, RIPPER, PART, and K-Star (accuracy above 97%). Those suitable for online predictions are Bayesian network and OneR (less than 0.2s demanded in prediction phase and 89% accuracy). And those with very low false alarm rates are SMO, and SVM.

In this case, we were able to use the dataset to reproduce some aspects of the original training-prediction experiment, achieving good results in terms of accuracy and processing time, as shown in Table 1. However, as all HDDs were from the same manufacturer, SMART parameters were identified by specific names that did not match with SMART logs we can acquire from different HDDs now. Hence, we could not verify whether the algorithms would disclose similar results when facing updated datasets.

Table 1. Performances of Machine Learning Algorithms considering 68,411 inputs from Murray's dataset. Results represent the average of ten runs, with ten folds.

Machine Learning Algorithm	**Accuracy**	**Time (sec)**
Naive Bayes (NB)	0.86±0.03	0.08
Multinomial Naive Bayes (MNB)	0.84±0.06	0.12
Multilayer Perceptron (MLP)	0.87±0.00	0.51
Decision Tree (DT)	0.80±0.19	0.73
KNN Euclidean (KNN)	0.77±0.16	0.91
Logistic Regression (LR)	0.84±0.08	1.75
SVM with kernels (SVM)	0.87±0.00	772.71

Another obstacle we found was evaluating the effectiveness of data preprocessing. The authors stated that some instances were reclassified as DAY_7, meaning that the algorithm would be able to detect or anticipate a HDD impairment 7 days before it actually happens, allowing the user to plan the device replacement with some success. Nevertheless, the main problem is that SMART stats that could identify failures were taken from HDDs *after* they were declared failed, as we have mentioned before.

Given this scenario, in this work we carried out a comparison of some of the state-of-the-art classifiers in predicting disk failures. These algorithms were applied in a recent dataset containing data of multiple HDD vendors which is described in the next section.

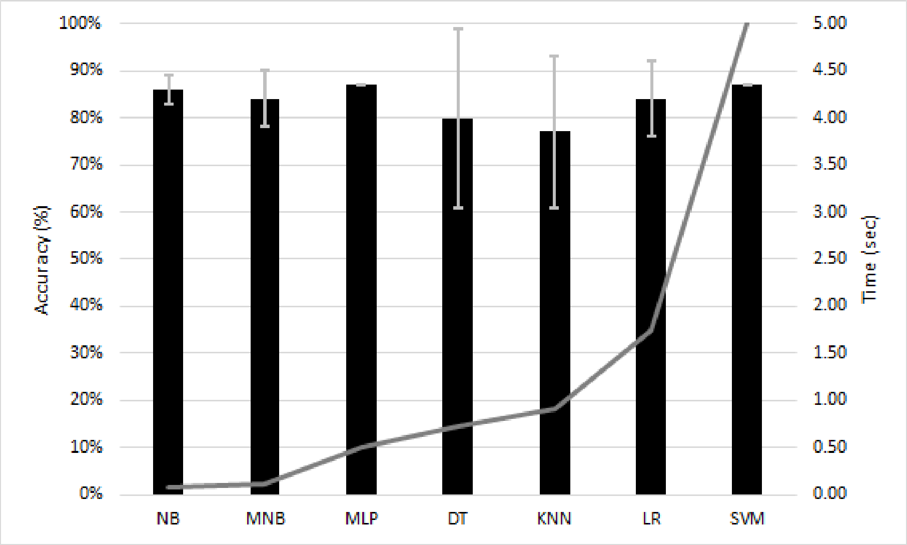

Figure 2. Representation of Table 1. The execution time of SVM was cutoff.

3. Dataset Description

Backblaze Inc. updates the dataset used in this study since April, 10 2013, available with free access on their website[1]. It consists of daily files in comma-separated values (CSV) format.

Each file has a set of SMART logs of every HDD in operation in that specific day. The number of HDDs in use shifts occasionally as the company discards defective devices, adding and/or replacing drives. By the end of 2013, Backblaze had $27,223$ HDDs in operation, increasing this number to $73,653$ in 2016, a 271% change. During the same period of four years, they kept an average AFR of 3%, as shown in Figure 3, which is 2.2 to 6 times higher than datasheet AFR.

Files ranging from 2013 to 2014 have 80 columns related to 40 different SMART stats, and those ranging from 2015 have 90 columns related to 45 different SMART stats, with both normalized and raw values. Because of that difference in the number of SMART stats, and due to general SMART data quality, we have decided to use only datasets of 2015 and 2016 in the training and prediction of algorithms.

[1] The dataset can be downloaded on https://www.backblaze.com/b2/hard-drive-test-data.html

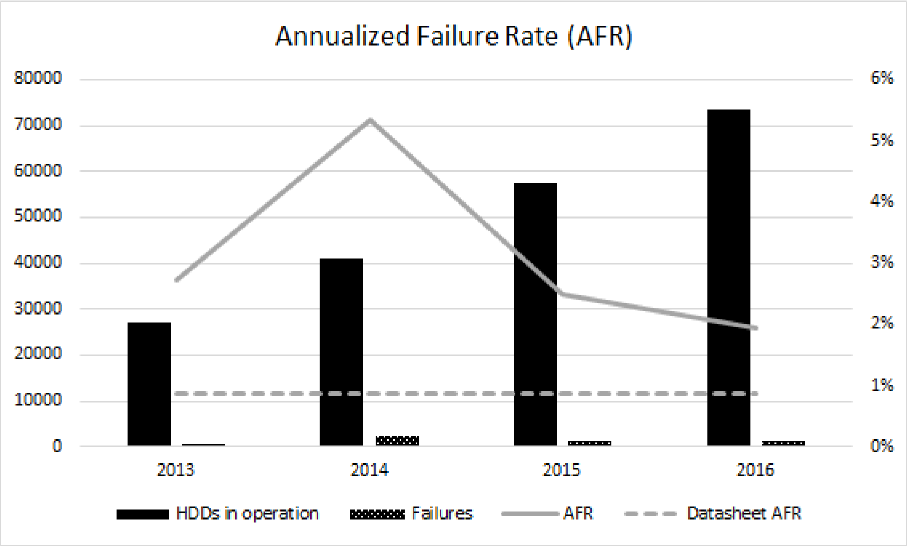

Figure 3. Dataset AFR from 2013 to 2016. In 2014, field-gathered AFR was six times than the highest datasheet AFR informed by manufacturers, which is 0.88% (dashed line).

Each row is related to a single HDD data, and the columns are date, serial number, model, capacity in bytes, failed (boolean value), followed by all SMART parameters available. Figure 4 illustrates rows and columns of the dataset. The serial number uniquely identifies a HDD, and by its model, it was possible to describe the dataset by manufacturer, as shown in Figure 5, in terms of failure events, HDD population, and failure rate. HDD set had models from five major manufacturers.

```
date,serial_number,model,capacity,failure,smart_1_normalized,smart_1_raw,...
2016-12-31,ZA10Q2F7,ST8000DM002,8.00E+12,1,64,134452400,...
2016-12-31,MJ0351YNGA62NA,Hitachi HDS5C3030ALA630,3.00E+12,1,76,22676646,...
```

Figure 4. Illustration of rows and columns of the dataset (excerpt).

Western Digital's HDDs displayed the highest failure rate, 20.23% in 2015-2016, while HGST's, a subsidiary of Western Digital, the lowest failure rate, of 0.5% in the same period. HGST had the second largest HDD population, with $24,545$ drives by the end of 2016, and Seagate the largest, with $45,531$ drives. The total of failed drives was of $2,857$ units. Although Seagate had $2,086$ failed HDDs, which corresponds to 73.01% of the total of failures, the failure rate was of 4.58% proportionally to its large drive population.

Once a HDD is declared failed, the serial number is removed from dataset on the next day. For Backblaze, a drive is considered to have failed if any of the three events happens: 1) The drive will not spin up or connect to the OS; 2) The drive will not sync, or stay synced, in a RAID Array; 3) The SMART stats they use show values above their thresholds.

We must emphasize three important notes. First, the dataset does not inform the root cause of failure. It implies that a drive declared failed may have "health" SMART stats if a threshold was not exceeded. Also, it is not possible to establish a correlation between SMART logs and other causes of failure.

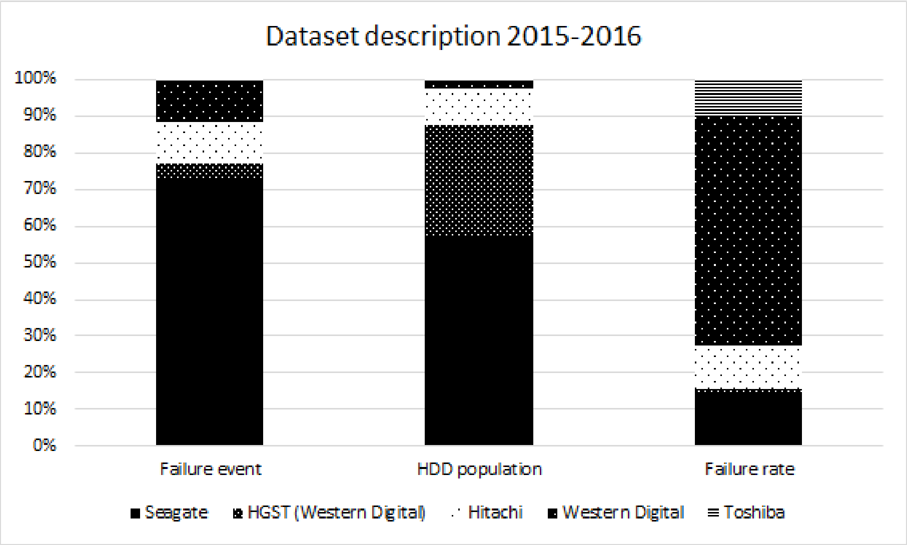

Figure 5. Dataset description from 2015 to 2016 in terms of failure event, HDD population, and failure rate. We highlight the failure rate of Western Digital HDDs, of 20.23%.

Second, Backblaze Inc. keeps track of five SMART attributes: SMART 5 - Reallocated Sectors Count; SMART 187 - Reported Uncorrectable Errors; SMART 188 - Command Timeout; SMART 197 - Current Pending Sector Count; SMART 198 - Uncorrectable Sector Count.

Hence, if the mentioned conditions *one* and *two* do not happen, only when any of the five SMART attributes overreaches the thresholds established by Backblaze, the drive is classified as failed. Within these conditions, 4.2% of operational drives and 76.7% of failed drives show one or more of those SMART stats with values greater than zero.

And third, SMART-capable devices show both raw and normalized attributes. Raw values are occurrence counters, with the exception of temperature. And normalized values are calculated by manufacturers according to ATA standard. However, not all manufacturers follow this convention.

Our last dataset analysis refers to the number of non-zero SMART stats per failed HDD, as shown in Figure 6. We identified 58 models with four to 21 non-zero parameters. Three out of $2,860$ (0.001%) failed drives showed all SMART stats equal to “Null”, or “Not a Number” (NaN). By convention, and in order to process training and prediction algorithms, “Null” and “NaN” values are filled with zeros.

“Null” and “NaN” SMART parameters may happen because of three reasons. First, if a device was not SMART-capable. Second, if any problem blocked log extraction. Or third, if a manufacturer hid data following internal policies.

The very low (0.001%) occurrence of all-zero SMART stats allowed us to review Pinheiro’s conclusion about SMART data effectiveness on HDD failure prediction [Pinheiro et al. 2007]. The difference between the dataset of 2013 and 2016 in terms of data completeness is notable. Extending that to 2005, when Pinheiro collected data for their study, we concluded that the quality of data increased considerably, allowing us to run tests on SMART data to predict HDD failure. Figure 7 shows the number of drives with four to 21 non-zero SMART stats. There is a consistent amount of useful SMART parameters to be used in this experiment.

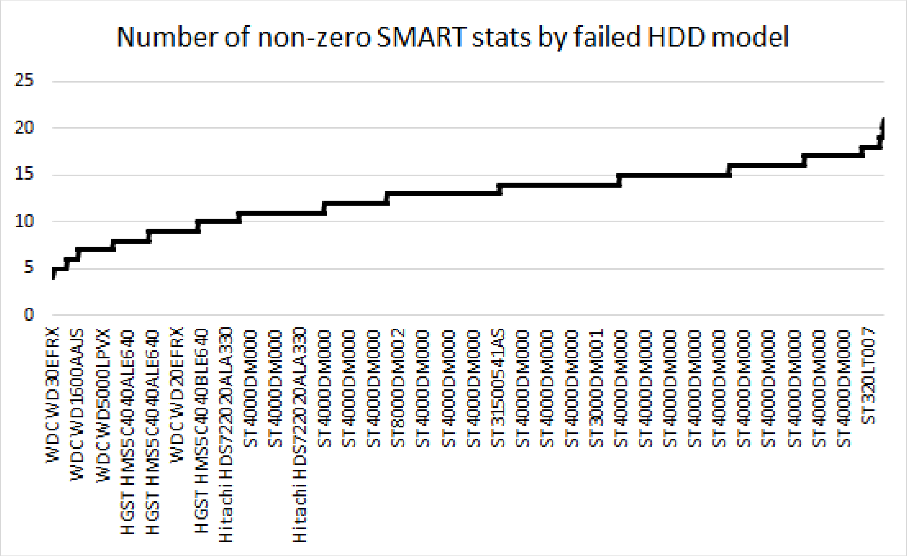

Figure 6. Models of failed HDDs and number of non-zero SMART stats. Not all models are listed due to figure size limits.

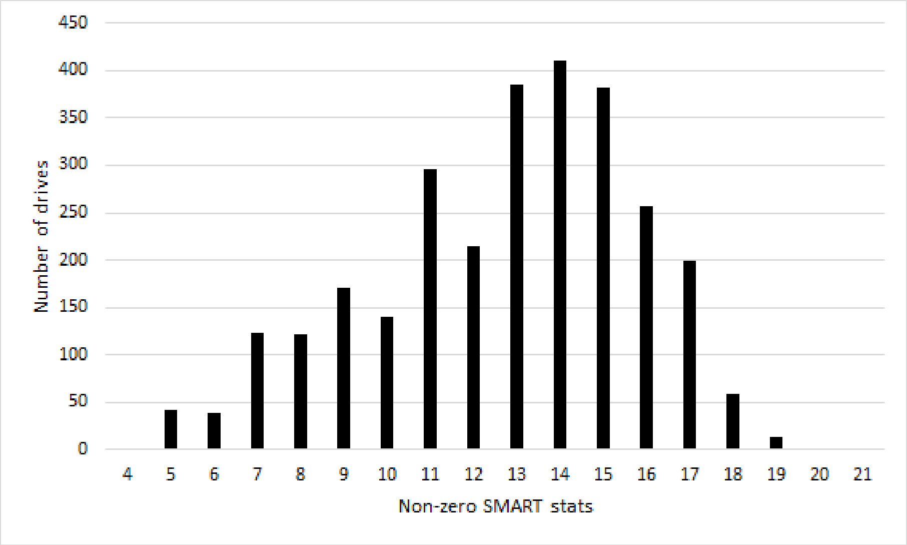

Figure 7. Number of drives per number of non-zero SMART parameters.

4. Experimental Evaluation

Now, we will describe the modeling of data preprocessing and machine learning algorithms. We developed all applications using Python 2.7, and Python scikit-learn 0.18.1. The computer used to run all applications was equipped with a Intel Core i7 3.4GHz processor, 32GB RAM, Ubuntu 16.04 LTS, Linux Kernel 4.8.0. The source code used in our experiments is public available [2].

[2] http://lnk.ufms.br/smart

Firstly, we preprocessed the original dataset, extracting all failure instances, and excluding a total of three occurrences because they had all SMART stats equal to “Null”, or “Not a Number” (NaN), as explained before.

Next, as we had an imbalanced dataset, we have chosen a random undersampling approach [Chawla 2005], we concatenated to the failure instances the same amount of non-failing examples from 10 different files, selected in a pseudo-random process, so executing the next steps 10 times to calculate the mean values of accuracy rate and time spent in training and prediction phases.

Then, we split the data in ten train/test sets, using Stratified K-Folds cross-validation. We compared the performance of nine different classification algorithms: Decision Tree (DT), SVM (with linear and RBF kernels), KNN, Multi Layer Perceptron (MLP), Naïve Bayes (NB), Multinomial Naïve Bayes (MNB), Bernoulli Naïve Bayes (BNB) and Logistic Regression (LR).

We carried out a parameter tuning for the first four algorithms through a cross-validated randomized search. The compared values can be observed in Table 2. In the end, considering the best parameters achieved, we gathered the average of ten executions of ten folds each one, that is, the average of 100 different runs for each algorithm, and its respective standard deviation.

Table 2. Parameters tested on tuning procedure.

Algorithm	Parameters	Tested values
SVM	Kernel	Radial Basis Function (RBF) and Linear
	C	Randomly in an exponential function with scale 100.
	gamma	Randomly in an exponential function with scale 0.1.
MLP	solver	SGD (stochastic gradient descent)
	learning_rate	Constant
	momentum	Randomly in an exponential function with scale 0.1.
	alpha	Randomly in an exponential function with scale 0.001.
	activation	Logistic
	learning_rate_init	Randomly in an exponential function with scale 0.01.
	hidden_layer_sizes	(200,50,20), (100,20), (100,50), (200,20), (200,50), (50,10), (200,), (100,), (50,)
	max_iter	500
Decision Tree	criterion	Gini Index
	max_depth	[3,15]
KNN	metric	Euclidean distance
	n_neighbors	[1,15]

Finally, we prepared a model persistence of the best estimator according to its performances and purposes, for future use without having to retrain the model. The fastest algorithms are suitable for short or online tests. The most accurate ones for long or offline tests.

We also developed an application that extracts SMART parameters from HDDs with exactly same format as Backblaze's does, to run our tests with some diverse data. It should work for both Linux and Windows operating systems with Python 2.7, `smartctl` 6.6, pySMART 0.3, and other dependencies installed. pySMART is a Python wrapper for the `smartctl` component of smartmontools, requesting administrative privileges to access all SMART parameters correctly.

The results are shown in Table 3 and Figure 8. In particular, we have special interest in results in terms of accuracy, which is equivalent to detection rate, and precision, which is complementary to false alarm rate.

It is possible to observe that the decision tree algorithm achieved better results in most of the measures. However, there is no significant difference in terms of performance when comparing decision tree to SVM with RBF kernel. Both algorithms performed very well especially in terms of accuracy and precision. However, decision trees have presented a very competitive performance in terms of execution time. The training and testing procedure in SVM takes much longer time than in decision trees. These results allow us to use decision trees in both online and offline applications to predict disk failures.

As the dataset we used was up-to-date, and more complex than Murray's in terms of disk population, heterogeneity, and completeness, our results show advantages on HDD failure prediction. Also, the persistent models achieved close-to-zero execution time, ensuring performance for both online and offline tests.

Table 3. Performance comparison of machine learning algorithms on training and testing procedures.

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1**	**AUC**	**Time (sec)**
DT	0.80±0.06	0.88±0.03	0.70±0.13	0.78±0.09	0.80±0.06	0.02±0.01
MLP	0.71±0.04	0.72±0.06	0.70±0.10	0.70±0.05	0.71±0.04	6.74±4.84
KNN	0.76±0.07	0.82±0.04	0.66±0.14	0.73±0.10	0.76±0.07	0.05±0.01
NB	0.55±0.03	0.77±0.13	0.14±0.07	0.22±0.11	0.55±0.03	0.01±0.01
MNB	0.61±0.04	0.58±0.03	0.79±0.09	0.67±0.04	0.61±0.04	0.01±0.01
BNB	0.59±0.05	0.57±0.04	0.72±0.09	0.64±0.05	0.59±0.05	0.01±0.00
LR	0.75±0.04	0.81±0.03	0.66±0.07	0.73±0.05	0.75±0.04	0.37±0.10
SVM RBF	0.79±0.06	0.87±0.03	0.69±0.12	0.76±0.09	0.79±0.06	1.84±0.18
SVM Linear	0.78±0.07	0.80±0.05	0.75±0.14	0.76±0.10	0.78±0.07	0.85±0.11

5. Conclusion

This paper has three main contributions. First, SMART data is suitable to solve the problem of predicting HDD failures. A previous work was opposed to this statement because the quality of SMART data in terms of non-zero stats did not allow acceptable results. However, we have identified some remarkable data quality improvements, from 36% all-zero stats down to only 0.001%.

Second, despite of these improvements, not all manufacturers follow ATA standards, probably because of their internal policies. The consequence is that models are

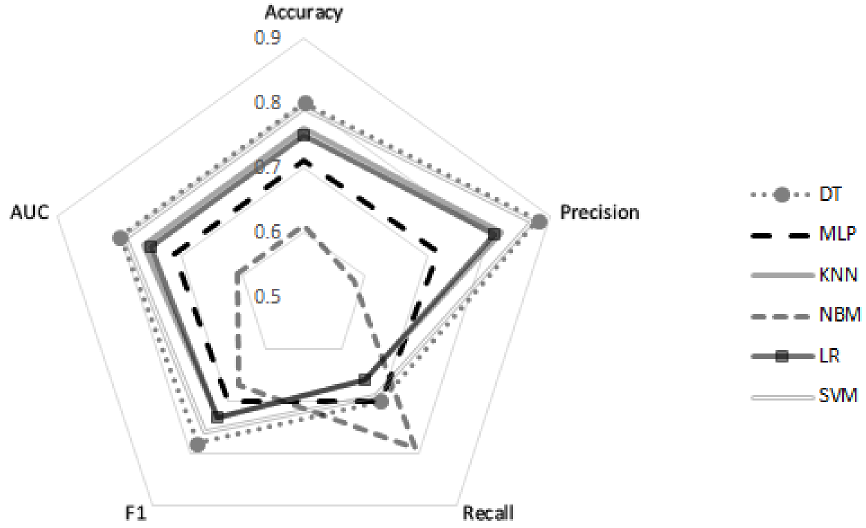

Figure 8. Performance of selected algorithms. We highlight results of decision tree (DT) and Support Vector Machine (SVM).

not totally comparable to each other. Even though, we were capable of summarizing the dataset, and apply the machine learning techniques to detect failures.

Finally, we were able to achieve similar results with decision tree and SVM with RBF kernel algorithms, with around 80% accuracy, 88% precision rates, and 12% in false positive rate in failure predictions. Nevertheless, decision tree was 92 times faster than SVM, fitting both online and offline implementations. Although these rates are lower than those reached in other studies, we presented a more complex problem, with an up-to-date dataset composed of millions of entries from five major HDD manufacturers and 55 different drive models.

Some future works that may enhance our results include improvements on parameter tuning of algorithms, on feature selection of the best SMART stats, and on data clustering. Another approach is to observe new technologies like SSDs (Solid-State Drive), and how to use similar strategies to detect failure on these devices. Finally, we would also consider to separate the dataset by disk manufactures in order to build specialized classifiers to improve failure detection.

References

Chawla, N. V. (2005). *Data Mining for Imbalanced Datasets: An Overview*. Springer US, Boston, MA.

Cisco Inc. (2016). Cisco Global Cloud Index: Forecast and Methodology 2015–2020, White Paper. Online; accessed 2017-02-15.

Dell inc. (2016). Data Loss: Understanding the Causes and Costs, white paper. Online; accessed 2017-02-10.

Klein, A. (2016). Data Backup: Are You a Hero or a Zero? Online; accessed 2017-01-30.

Kroll Ontrack Inc. (2010). Lost Data? Reasons and Costs of Data Loss. Online; accessed 2017-03-25.

Kroll Ontrack Inc. (2014). Hard Disk Drive (HDD) Crashes Remain the Leading Cause of Data Loss, Survey Says. Online; accessed 2017-03-25.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill Higher Education.

Murray, J. F., Hughes, G. F., and Kreutz-Delgado, K. (2005). Machine learning methods for predicting failures in hard drives: A multiple-instance application. *J. Mach. Learn. Res.*, 6:783–816.

Pinheiro, E., Weber, W.-D., and Barroso, L. A. (2007). Failure trends in a large disk drive population. In *5th USENIX Conference on File and Storage Technologies (FAST 2007)*, pages 17–29.

Pitakrat, T., van Hoorn, A., and Grunske, L. (2013). A Comparison of Machine Learning Algorithms for Proactive Hard Disk Drive Failure Detection. In *Proceedings of the 4th International ACM Sigsoft Symposium on Architecting Critical Systems*, ISARCS '13, pages 1–10, New York, NY, USA. ACM.

Schroeder, B. and Gibson, G. A. (2007). Disk Failures in the Real World: What Does an MTTF of 1,000,000 Hours Mean to You? In *Proceedings of the 5th USENIX Conference on File and Storage Technologies*, FAST '07, Berkeley, CA, USA. USENIX Association.

Yang, J. and Sun, F.-B. (1999). A comprehensive review of hard-disk drive reliability. In *Annual Reliability and Maintainability. Symposium. 1999 Proceedings (Cat. No.99CH36283)*, pages 403–409.