

# CASE STUDY - DISK FAILURE PREDICTION

WEI REN

**ABSTRACT.** In this report, the detailed illustration of data preprocessing and feature engineering, model choosing and parameter tuning, results evaluation and insights from this task are included.

## 1. BRIEF INTRODUCTION

**1.1. Introduction.** Various disk failures are not rare in large-scale IDCs and cloud computing environments, fortunately, we have S.M.A.R.T. (Self-Monitoring, Analysis, and Reporting Technology; often written as SMART) logs collected from computer hard disk drives (HDDs), solid-state drives (SSDs) and eMMC drives that detects and reports on various indicators of drive reliability, with the intent of enabling the anticipation of hardware failures. Hence, HDD vendors are highly motivated to reduce the rate of failures as a cost saving measure.

SMART attributes represent HDD health statistics such as the number of scan errors, reallocation counts and probational counts of a HDD, and a detailed list can be seen in [4]. If a certain attribute considered critical to HDD health goes above its threshold value, the HDD is marked as likely to fail[1].

This report focuses on applying machine learning to improve prediction accuracy over baseline heuristics in hard disk drives. The goal of this case study is twofold: 1) to achieve a higher recall, precision and accuracy than our baseline implementation modeled off of current enterprise failure detection models. 2) to analyze which of our subset of machine learning models is best suited towards predicting failure of HDDs. Three different algorithms are applied: Logistic Regression, Naive Bayes and Random Forest, to see which has the best performance when predicting HDD failures.

**1.2. Literature Review.** Pinheiro et al[1] studied failure trends in a large scale of enterprise HDDs at a Google data center. Their analysis identified that specific SMART parameters (scan errors, reallocation counts, offline reallocation counts, and probational counts) correlated highly with failures, while models based on SMART parameters alone were inadequate for accurate prediction because of the lack of occurrence of predictive SMART signals on a large fraction of failed drives. Besides, neither elevated temperature nor activity levels were found to be very little correlated with failure rates. It was noticed that vendors and end-users often saw different statistics (below 2% and 6% respectively) due to the difference in definition, and they proposed a concise definition: *a drive is considered to have failed if it was replaced as part of a repairs procedure*. In their study, filtering was applied to cope with data integrity issues and clearly impossible data.

Similarly, BackBlaze analyzed the correlation rates between its HDD failures and SMART attributes and found that SMART 5, 187, 188, 197, and 198 had the highest rates of correlation to HDD failure. Pitakrat et al[2] evaluated 21 machine learning (ML) algorithms for predicting HDD failure. It was found by testing 21 machine learning models that different algorithms are suitable for different applications based on the desired prediction quality and the tolerated

---

*Date:* March 24, 2018.

*Key words and phrases.* Machine Learning, SMART, Disk Failure.

The motivation of this report is to show a case study of disk failure prediction as a coding test/supporting material for the application of an AIOps position of Alibaba Group.

training and prediction time. Specifically, a Random Forest (RF) algorithm produced the largest area under a ROC Curve (AUC), while a Nearest Neighbor classifier (NNC) had the highest F1-score.

It can be summarized that the five mentioned SMART attributes are highly correlated with failures and various models should be tested to obtain the desired prediction.

## 2. DATA PREPROCESSING AND FEATURE ENGINEERING

Data sources from <https://www.backblaze.com/b2/hard-drive-test-data.html>

**2.1. First Glance at Datasets.** BackBlaze has published statistics and their insights based on the hard drives in the data center. Each day in their data center, they take a snapshot of each operational hard drive and output a file for recording. The first row of the each file contains the column names, shown in the followings:

- **Date** - The date of the file in *yyyy-mm-dd* format.
- **Serial Number** - The manufacturer-assigned serial number of the drive.
- **Model** - The manufacturer-assigned model number of the drive.
- **Capacity** - The drive capacity in bytes.
- **Failure** - Contains a "0" if the drive is operational. Contains a "1" if this is the last day the drive was OK before failing and replaced.
- **SMART Stats** - 90 columns of data, that are the Raw and Normalized vlaues for 45 different SMART stats are reported by the given drive. Each value is the number reported by the drive.

In Figure 1, a snapshot of Backblaze datasets can be seen. Typically, there are numerous blank fields in each record file because most drives do not report values for all SMART stats. Also, different drives may report different stats based on their model and/or manufacturer. In the meantime, reported stats for the same SMART stat can vary in meaning based on the drive manufacturer and the drive model. When processing the data, as suggested by Backblaze, one should conduct bounds checks and notice the changing number of drives.

date	serial_number	model	capacity_bytes	failure	smart_1_normalized	smart_1_raw	smart_2_normalized	smart_2_raw	smart_3_normalized	smart_3_raw	smart_4_normalized	smart_4_raw	smart_5_normalized	smart_5_raw
2017/1/8	MJ0351YNG9Z0XA	Hitachi HD55C3030ALA630	3 00059E+12	0	100	0	135	108	127	554	100	15	100	0
2017/1/8	MJ0351YNG9WJSA	Hitachi HD55C3030ALA630	3 00059E+12	0	100	0	136	104	126	555	100	21	100	2
2017/1/8	PL1321LAG34XWH	Hitachi HD55C4040ALA630	4 00079E+12	0	100	0	134	101	130	560	100	31	100	0
2017/1/8	MJ0351YNG8YAA	Hitachi HD55C3030ALA630	3 00059E+12	0	100	0	136	104	137	507	100	17	100	0
2017/1/8	Z30582QN	ST4000DM000	4 00079E+12	0	118	177436144			91	0	100	8	100	0
2017/1/8	PL2331LAGN2YTJ	HGST HM55C4040BLE640	4 00079E+12	0	100	0	133	104	100	448	100	5	100	0
2017/1/8	WD-WMCA4N2899475	WDC WD30EFRX	3 00059E+12	0	200	0			175	6250	100	12	200	0
2017/1/8	Z302A0YH	ST4000DM000	4 00079E+12	0	117	152016136			92	0	100	19	100	0
2017/1/8	Z3058TQW	ST4000DM000	4 00079E+12	0	117	135533416			93	0	100	7	100	0

FIGURE 1. A snapshot of Backblaze data sets.

### 2.2. Data Preprocessing.

- (1) Choose raw over normalized SMART Data points
- (2) Failure status smoothing/backtracking
- (3) Filter out all HDD models besides Seagate models
- (4) Balance out the data set

### 2.3. Feature Engineering.

*Feature Selection.* From experience, BackBlaze have found the following five SMART metrics indicate impending disk drive failure (see Figure 2).

Ninety variables and millions of data points not only take an extensive amount of time to train and test on, but can also lead to overfitting. To reduce the computational workload and improve the performance of our models, we chose to select only the most relevant features and avoid features with a large amount of unfilled data points.

Drive Status	SMART 5	SMART 187	SMART 188	SMART 197	SMART 198
Operational	1.1%	0.5%	4.8%	0.7%	0.3%
Failed	42.2%	43.5%	44.8%	43.1%	33.0%

FIGURE 2. Percentage of both failed and operational drives with a RAW value of the five SMART attributes.

As of the end of 2017, there are about 88 million entries totaling 23 GB of data. Each entry consists of the date, manufacturer, model, serial number, status (operational or failed), and all of the SMART attributes reported by that drive. 5, 187, 188, 197, 198

There are some critical questions to be answered if one needs to complete this case study:

- Define the amount of data, and which columns should be counted; Data inputs. Data of the year of 2017.
- Define the features; Some feature selection methods.
- Define the training data sets and test data sets; Cross-validation for model selection.
- Define model, cost/loss function, parameters, training methods. Machine Learning. Scikit-Learn
- Train the models and tune parameters.
- Conduct prediction for other data sets and evaluate results. Confusion Matrix.

Algorithm 1 summaries the above process in the pseudo code format.

### 3. MODEL SELECTION AND PARAMETERS TUNING

How to choose machine learning models and tune the parameters?

Reduce to a classification problem.

**Base Line.** Our baseline analysis mimics what BackBlaze currently implements in its failure prediction system. We analyze five SMART attributes ( SMART 5, 187, 188, 197, 198) and predict a HDD will fail if any of these critical raw SMART attributes are greater than 0. Our goal is therefore to maintain as high of a TPR with a maximum TPR equal to the baseline analysis, and to focus on reducing the FPR to 0.

**Logistic Regression.** Logistic Regression is one of the basic tools for performing binary classification. One of the assumptions made in order for Logistic Regression to potentially perform well is that the data is linear. This means that the score we obtain from Logistic Regression is affected proportionally to changes in the feature values in a linear fashion. The tool mainly served as a second baseline in some sense as it was our first attempt at the classification problem beyond implementing the simple baseline. We also employed L2 regularization.

**Naive Bayes.** The Naive Bayes classifier model makes the assumption that the value of a feature is conditionally independent of the value of another feature given some class label. Among the different techniques used for building Naive Bayes models, we chose Multinomial Naive Bayes, which assumes that the probability of a feature value given some class label is sampled from a multinomial distribution. For regularization, we use Laplace smoothing.

**Random Forest.** Random forest is an ensemble tool which takes a subset of observations and a subset of variables to build a group of decision trees. It builds multiple such decision trees and amalgamate them together to get a more accurate and stable prediction.

#### 4. RESULTS EVALUATION

How to evaluate the results?

To evaluate the classifier's performance, we measure precision, recall and F-score as defined below.

- Precision: to measure the ability of the classifier to correctly identify disks at risk;
- Recall: to measure the classifier's sensitivity. A higher recall is equivalent to minimizing the number of false negatives;
- $F_1$  score: the combined score between precision and recall, or the weighted harmonic mean;
- ROC: Receiver Operating Characteristic, is the plot of TPR vs FPR. In usual, we can also draw the Precision-Recall Curve.

$$(1) \quad P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \quad F_1 \text{ score} = \frac{2PR}{P + R}$$

where  $TP$  refers to true positives,  $FP$  is false positives and  $FN$  denotes false negatives.

		True condition			
	Total population	Condition positive	Condition negative	Prevalence = $\frac{\sum \text{Condition positive}}{\sum \text{Total population}}$	Accuracy (ACC) = $\frac{\sum \text{True positive} + \sum \text{True negative}}{\sum \text{Total population}}$
Predicted condition	Predicted condition positive	True positive, Power	False positive, Type I error	Positive predictive value (PPV), Precision = $\frac{\sum \text{True positive}}{\sum \text{Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\sum \text{False positive}}{\sum \text{Predicted condition positive}}$
	Predicted condition negative	False negative, Type II error	True negative	False omission rate (FOR) = $\frac{\sum \text{False negative}}{\sum \text{Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\sum \text{True negative}}{\sum \text{Predicted condition negative}}$
		True positive rate (TPR), Recall, Sensitivity, probability of detection = $\frac{\sum \text{True positive}}{\sum \text{Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm = $\frac{\sum \text{False positive}}{\sum \text{Condition negative}}$	Positive likelihood ratio (LR+) = $\frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR-}}$
		False negative rate (FNR), Miss rate = $\frac{\sum \text{False negative}}{\sum \text{Condition positive}}$	True negative rate (TNR), Specificity (SPC) = $\frac{\sum \text{True negative}}{\sum \text{Condition negative}}$	Negative likelihood ratio (LR-) = $\frac{\text{FNR}}{\text{TNR}}$	
				$F_1 \text{ score} = \frac{2}{\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}}$	

FIGURE 3. Confusion Matrix[3]

#### 5. CONCLUSIONS

What insights or lessons learned from this task?

#### REFERENCES

1. Eduardo Pinheiro, Wolf-Dietrich Weber, and Luiz André Barroso, *Failure trends in a large disk drive population*, 2007.
2. Teerat Pitakrat, Andre van Hoorn, and Lars Grunske, *A comparison of machine learning algorithms for proactive hard disk drive failure detection*, ISARCS, 2013.
3. Wikipedia, *Receiver operating characterisic*.
4. ———, *S.m.a.r.t.*

## APPENDIX

**Appendix A: SMART Attributes.**

- **SMART 5: Reallocated Sector Count.**

When the drives logic believe that a sector is damaged, it can remap the faulty sector number to a new physical sector drawn from a pool of spares.

- **SMART 187: Reported Uncorrectable Errors.**

The count of errors that could not be recovered using hardware ECC. Large scan error counts can be indicative of surface defects and therefore are believed to be indicative of lower reliability.

- **SMART 188: Command Timeout.**

The count of aborted operations due to HDD timeout.

- **SMART 197: Current Pending Sector Count.**

Disk drives put suspect bad sectors on probation until they either fail permanently and are reallocated or continue to work without problems.

- **SMART 198: Offline Uncorrectable.**

The total count of uncorrectable errors when reading/writing to a sector. A rise in the value of this attribute indicated defects of the disk surface and/or problems in the mechanical subsystem.

**Appendix B: Algorithms Bank.** In this appendix, several algorithms in pseudo code format are listed.

---

**Algorithm 1:** Data importing and preprocessing

---

**Input:** The original test data of the year of 2017 plus Q4 of 2016 from Backblaze

**Output:** Raw and normalized values of the five SMART attributes for the year of 2017, along with failure/operational status, date, serial number

- 1  $(X, Y) = \text{Data selection}(\text{SMART } 5, \text{SMART } 187, \text{SMART } 188, \text{SMART } 197, \text{SMART } 198, \text{date, serial number, Failure status});$
  - 2 Initialize a new column for each SMART attribute with 0, named SMART binary, to denote if this attribute is non-zero ;
  - 3 **if** *SMART attributes*  $\neq \text{null}$  **then**
  - 4   | **if** *SMART attributes*  $\neq 0$  **then** let the value be 1 ;
  - 5 **else**
  - 6   | let the value be 0 ;
  - 7 **if** *failure status is true* **then**
  - 8   | Mark last 60 days (if applicable) of the HDD's *failure status as true* ;
  - 9 **return**  $X$  and  $Y$ , with three versions (raw, normalized, binary values of SMART attributes);
- 

---

**Algorithm 2:** Model selection, training, and parameters tuning

---

**Input:**  $X$  and  $Y$  of the year of 2017

**Output:** Trained model

- 1 Split the datasets into two pieces, training datasets  $(X_{train}, Y_{train})$  and validation datasets  $(X_{test}, Y_{test})$ , by cross validation;
  - 2 Train different models with training datasets;
  - 3 **repeat**
  - 4   | Evaluate the models and tune parameters;
  - 5 **until** *parameters are optimized or maximum iterations exhausted*;
  - 6 **return** trained model;
- 

---

**Algorithm 3:** Results evaluation

---

**Input:**  $X_{test}$  and  $Y_{test}$  of Q4 of 2016, trained models

**Output:** Evaluation of models

- 1 Predict the failure status of  $X_{test}$  with trained models;
  - 2 Contrast to  $Y_{test}$  and compute  $TP$ ,  $TN$ ,  $FP$  and  $FN$ ;
  - 3 Calculate precision, recall, and  $F_1$  score;
  - 4 Draw the ROC and AUC plots;
  - 5 **return** true;
-