Predictive maintenance: data-driven failure forecast of hard-disks.

Fabio Cecchi

May 2, 2018

Introduction

In every aspect of our everyday life we rely on machines to perform the most diverse tasks. From the car that we use for commuting to the electric lines that serve our home, from the pacemaker that keeps our heart beating to the laptop we work with. Although very different, all these machines inevitably deteriorate over time, and eventually are going to stop functioning. The consequences of a failure depend on the specific task and may largely vary in magnitude. A car malfunctioning may lead to an accident, an electrical line failing may overcharge the network and cause major blackouts, a pacemaker stopping may coerce the patient into an urgent recovery, and a laptop not working may damage hours of work.

In general the "cost" incurred after a failure is tremendously larger than what has to be paid to maintain or even replace the specific machine. This is the reason why regular inspections are usually scheduled to track their functioning, and replacements are suggested as they get too old.

An easily-implementable policy consists in *preventive replacement*: machines are checked and replaced regularly, thus minimizing the cost due to unplanned events. The efficiency of this policy largely depends on how much is known about the time-to-failure distribution. In real-life system the state of the machine evolves continuously but observations are sampled only in a discrete manner, at the moment a checking is scheduled, these models have been largely studied in literature [2, 4]. It is arguable that when the time-to-failure distribution is sufficiently known, an optimal preventive replacement policy is the best offline policy, i.e., the best policy that cannot be modified in between two inspections, we can aim for.

However, we are now experiencing a paradigm shift due to the explosion of Internet of Things [1] and the large availability of data [3]. It is now easy to place sensors in machines so as to capture some of their metrics almost real-time. In specific cases, the metrics sensed can then be used to establish whether a machine is working properly or not, i.e., a correlation with the time-to-failure can be proved and exploited. The continuous monitoring of the machine is thus used to raise warnings and eventually schedule additional checking and replacements which can be carried out in parallel to the regular (preventive replacement based) inspections. This policy is named *predictive replacement* in the sense that it attempts to predict from data when a machine is close to a failure and makes you act accordingly.

The major challenge in predictive replacement consists in establishing the relevant metrics to observe. In fact, we rarely have at our disposal a metric that directly captures the remaining life of a certain machine. The latter must be inferred through the observable metrics at our disposal, by means of an application-based extrapolation. Hence, the role of the data scientist should first of all understand which observable metrics are correlated to the time-to-failure of a machine and then establish under which circumstances it is appropriate to raise a warning and suggest a deeper checking.

Motivation and case study

In this project we aim to understand whether preventive replacement can be used in forecasting failures of hard drive disks (HDDs). It is expected that by

2020 70% of all data will be in HDDs and 90% of all data center data will be in hard disk drives (Forbes article), thus preventing the failure of HDDs will play a key role in designing efficient data centers.

Unfortunately, the time-to-failure of an HDD cannot be easily captured. However, by placing sensors on an HDD we may capture the Self-Monitoring, Analysis, and Reporting Technology (S.M.A.R.T.) metrics indicating whether the performance of the HDD are up to the required standards. We aim to understand whether and which of the S.M.A.R.T. metrics are correlated to the time-to-failure and eventually be able to raise warnings if the behavior of the HDD is suspicious. Obviously, we do not want to raise too many false alarms, but false alarms are definitely less costly with respect to a missed failure and a trade-off has to be determined.

In a recent Google study [5] it as been observed that "Further, 36% of failed drives did so without recording any S.M.A.R.T. error at all, except the temperature, meaning that S.M.A.R.T. data alone was of limited usefulness in anticipating failures." This means that predictive replacement alone is not enough to capture every failures, nevertheless the silver lining is that the remaining 64% of the failures were somehow (perhaps minimally) correlated to a certain level of disruption in at least one of the S.M.A.R.T. metric. Thus a constant monitoring of these metrics, together with a thorough statistical analysis may drastically improve the reliability of data centers.

Dataset description

In this project we rely on the open dataset made available by Backblaze, a data storage provider which allows the user to back up data continuously, manually, when the computer is idle, or on an hourly schedule. We focused on the data relative to the period 2015-2017. The dataset for these three-years period consists of 7.2 millions of entries in which the HDDs are monitored on a daily basis. Each entry is features the following features:

- Serial Number: The unique ID of the HDD observed.
- Model: The model of the HDD observed.
- Capacity: The capacity of the HDD observed (same models have same capacity).
- Date: The time-stamp of the observation (measures are sampled on a daily basis).
- Failure: A 0/1 value, if equal to 0 the observed HDD did not fail on such day, if equal to 1 it did.
- SMART$n$raw: Raw value of the SMART metric n of the HDD observed.
- SMART$n$normalized: Normalized value of the SMART metric n of the HDD observed.

The SMART metrics reported are 45 and are model dependent. Observe that the same SMART metric for two different models may have completely different meaning and be measured differently. A specific model may lack a specific SMART metric which is indicated by NaN in the dataset. The raw values for

Model	Failures	Entries	Ratio of failed HDDs
WDC WD30EFRX	122	1261	9.67%
ST3000DM001	106	1170	9.06%
ST4000DM000	2590	36700	7.06%
ST31500541AS	112	1693	6.61%
Hitachi HDS723030ALA640	44	1018	4.32%
Hitachi HDS722020ALA330	152	4683	3.24%
ST6000DX000	60	1938	3.10%
Hitachi HDS5C3030ALA630	96	4608	2.08%
Hitachi HDS5C4040ALE630	42	2660	1.58%
ST8000DM002	141	10029	1.40%
HGST HMS5C4040ALE640	97	8661	1.12%
HGST HMS5C4040BLE640	135	16306	0.83%
ST8000NM0055	88	14510	0.61%
ST10000NM0086	3	1225	0.24%
ST12000NM0007	17	7244	0.23%

Table 1: List of models with at least 1000 entries

the SMART metrics are real numbers whose meaning depends on the specific metric and model type. Their normalized version take an integer value which ranges from 1 to 253 (with 1 representing the worst case, 253 representing the best, and 100 representing the typical behavior).

Observe that with respect to the study in [5] the sensing is done way more occasionally, here we have daily measurements there they had measures every few minutes. Thus we do not expect to capture as many failures as they did, nevertheless we aim to provide a computationally, memory, and energy-efficient procedure to support the nevertheless essential preventive replacement approach.

Data wrangling

The SMART metrics are the foundations of our predictive maintenance policy and, since they are model dependent, a different analysis has to be carried out for every model. As a first step, we read the dataset (PLACEHOLDER *Link to git and cells number*) and identify the models which have enough entries to be suitable for the analysis.

In Table 1 we list the number of unique HDD per model and how many of them failed in the three-year period considered. We reported only those models with at least 1000 unique HDDs and ordered them by fraction of failed HDDs. Note that the ratio is always below 10% and the unbalancedness between the number healthy and failing HDDs is an issue we will have to deal with in the following analysis.

Model-based data cleaning

In the following we will focus on the model "Hitachi HDS722020ALA330" and report only the result associated to such model. The code in (PLACEHOLDER)

is fully automated and the same analysis has been performed for the other models as well.

Observe that since we consider a unique model, we can immediately drop the columns 'Model' and 'Capacity' from the dataset. Moreover, for every SMART metric we have both the raw and the normalized value, for the sake of simplicity we will work only with the normalized quantities. Thus, the original dataset is immediately reduced to one with 48 columns (date, serial number, failure, and the normalized SMART metrics). We further reduce the dataset by dropping the *non-informative* SMART metrics. In particular, given a specific model some of the SMART metrics are absent and others do not vary sufficiently to provide meaningful information. While the former are easy to identify, we need to provide a way to judge which metrics belong to the latter.

Non-informative SMART metrics

Given a SMART metric, we claim that it is non-informative if it does not vary sufficiently over the dataset. We came up with some methods to test the meaningfulness of a series taking values $\mathbf{x} = x_1, \dots, x_N$:

- Consider the variance $v(\mathbf{x}) = \mathrm{Var}\big(x_1, \dots, x_N\big)$;
- Consider the range $r(\mathbf{x}) = \max\big(x_1, \dots, x_N\big) - \min\big(x_1, \dots, x_N\big)$;
- Consider the minimum $m(\mathbf{x}) = \min\big(x_1, \dots, x_N\big)$.

We claim that a metric is non-meaningful if

$$\Big(m(\mathbf{x}) \geq 100\Big) \text{ or } \Big(\big(r(\mathbf{x}) < T_r\big) \text{ and } \big(v(\mathbf{x}) < T_v\big)\Big)$$

with T_r and T_v chosen thresholds. We selected $T_r = 20$ and $T_v = 1$.

Note that if $m(\mathbf{x}) \geq 100$ it means that the metric never takes values below the typical and thus do not provide information of eventual malfunctions. To capture the variability of the metric, we believe that the variance alone would not be sufficient, in fact in some cases the SMART metric may drop abruptly just before a failure. In such case the large majority of the values for the metric would be close to a typical value and the variance would be too low due to the small quantity of unusual values.

Model-based data wrangling

The model-based data cleaning reduces drastically the dimensions of the dataset making it way more manageable while retaining most the information that could lead to the preventive identification of a failure. For the model "Hitachi HDS722020ALA330" we drop most of the metrics and keep only the columns associated to the normalized SMART metrics number 1, 5, 8, 192, 193, and 196 out of the initial 45 metrics.

The initial dataset is further wrangled and we obtain the following datasets:

- `df_serialnumber`. This dataset is indexed by the unique ID of the HDD and has the following columns:
  - 'date_first': First appearance of the HDD

  - 'date_last': Last appearance of the HDD
  - 'state': Type of HDD. This categorical feature can be 'failing' if the HDD fails at a certain point, 'healthy' if it does not, or 'wrong' if the date of failure does not coincide with its last appearance. A unique HDD (JK1101B9JPJ4NF) is categorized as wrong for the model "Hitachi HDS722020ALA330" and we removed it from the dataset.

- `df_smartmetrics`. For each of the SMART metric which is filtered we keep track of its mean value, std, min, and maximum.

- `df`. This is a multi-indexed dataframe in which the first level is given by the HDD serial number and the second by the date and the columns are given by the different SMART metrics. Note that when we slice for a specific serial number we obtain a time series for each SMART metrics whose last entry is related to the day of failure of an HDD with a failing state.

Exploratory data analysis

So far we excluded most of the SMART metrics by saying that are not informative, we now focus on those that we kept and aim to understand whether their analysis could help in predicting failures of HDDs.

First of all, we group the dataframe `df` according to the state of the HDD. In particular, we obtain a dataframe `df_healthy` containing only the observations associated to HDD which do not fail and `df_failing` with the others. The latter dataframe is further wrangled and we obtain `df_failure` and `df_failure_K` containing only the observations of the HDDs on the day of the failure and K days before the failure. We aim to answer the following questions:

- Does the distribution of the SMART metrics differ in between `df_healthy` and `df_failure`?

- If it does, how fast the SMART metrics deteriorate? That is to say, for which values of K we can distinguish the distribution of `df_healthy` and `df_failure_K`? Note that a larger K would help, it would provide more time to identify a close-to-failure HDD.

In Figure 1 we present the different distributions for `df_healthy`, `df_failure`, and `df_failure_K` for $K = 7$. Note that the y-scale is logarithmic, nevertheless metrics 1, 5, 8, and 196 have an interesting trend. On the one hand not only the SMART metric on the day of failure seems to be stochastically smaller than for healthy HDDs, but also the SMART metrics seem to worsen already one week before the failure. On the other hand, the vast majority of HDDs do not show a worsening in any of the SMART metrics even in the day of their failure. The latter failures will not be detected by any kind of analysis based on this dataset, a finer sensing policy (hourly instead of daily perhaps) might help.

In order to answer the second question, we now group the entries of the `df_failing` dataframe by the days to failure. Note that with respect to the SMART metrics 1,5, and 196 we have that the performance start to worsen about two weeks ahead of the failure. This can be observed in Figure 2 where

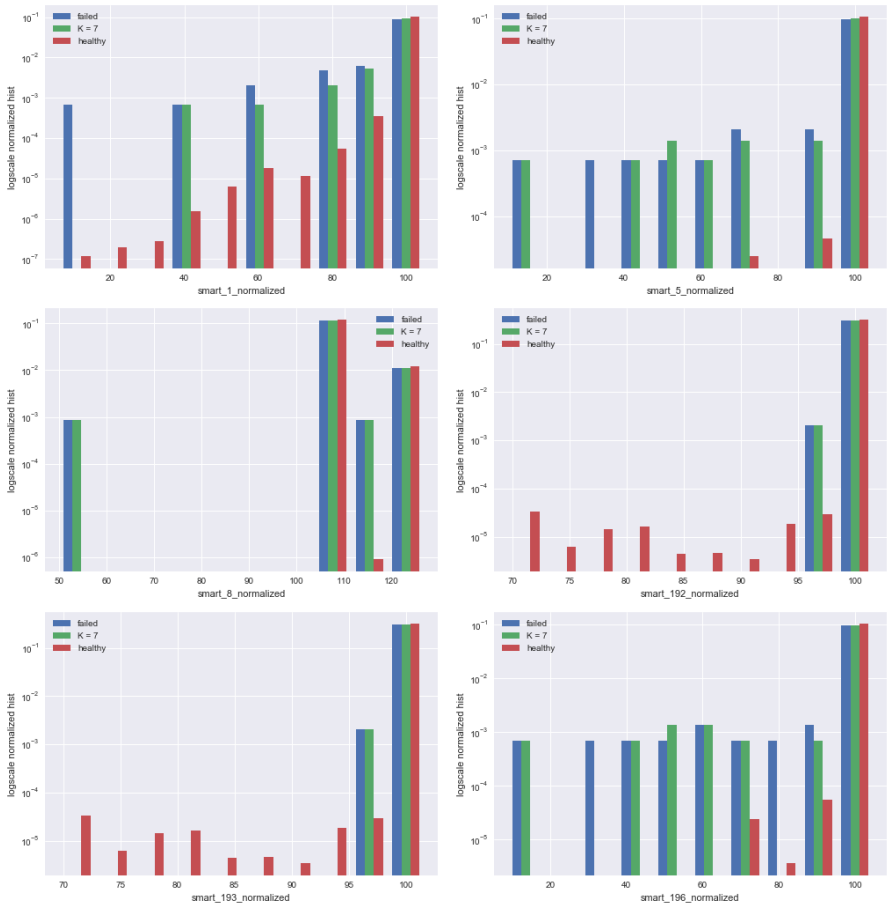

Figure 1: Normalized distributions for the SMART metrics for HDDs in different states.

the quantiles of the SMART metrics of the HDDs are plotted versus the days to failure.

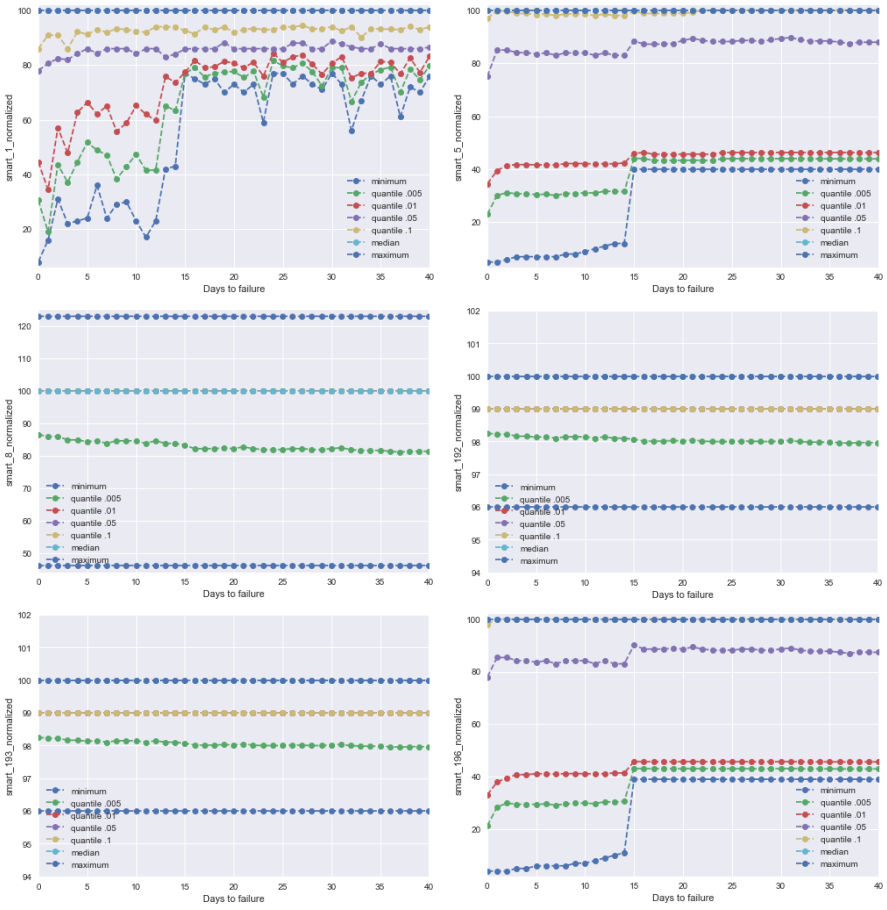

Figure 2: Normalized distributions for the SMART metrics for HDDs in different states.

Bibliography

[1] D. Evans (2011). The internet of things. *How the next evolution of the internet is changing everything*, Whitepaper, Cisco (IBSG).

[2] A.H. Jazwinski (2007). Stochastic processes and filtering theory.

[3] S. John Walker (2014). Big data: A revolution that will transform how we live, work, and think.

[4] V. Makis, X. Jiang (2003). Optimal replacement under partial observations. *Mathematics of Operations Research* **28 (2)**, 382–394.

[5] E. Pinheiro, W.D. Weber, L.A. Barroso (2007). Failure Trends in a Large Disk Drive Population. *FAST* **7 (1)**.