Lifespan and Failures of SSDs and HDDs: Similarities, Differences, and Prediction Models

Riccardo Pinciroli, Lishan Yang, Jacob Alter, and Evgenia Smirni

Abstract—Data center downtime typically centers around IT equipment failure. Storage devices are the most frequently failing components in data centers. We present a comparative study of hard disk drives (HDDs) and solid state drives (SSDs) that constitute the typical storage in data centers. Using six-year field data of 100,000 HDDs of different models from the same manufacturer from the Backblaze dataset and six-year field data of 30,000 SSDs of three models from a Google data center, we characterize the workload conditions that lead to failures. We illustrate that their root failure causes differ from common expectations and that they remain difficult to discern. For the case of HDDs we observe that young and old drives do not present many differences in their failures. Instead, failures may be distinguished by discriminating drives based on the time spent for head positioning. For SSDs, we observe high levels of infant mortality and characterize the differences between infant and non-infant failures. We develop several machine learning failure prediction models that are shown to be surprisingly accurate, achieving high recall and low false positive rates. These models are used beyond simple prediction as they aid us to untangle the complex interaction of workload characteristics that lead to failures and identify failure root causes from monitored symptoms.

Index Terms—**Supervised learning, Classification, Data centers, Storage devices, SSD, HDD**

1 Introduction

Storage devices, such as hard disk drives (HDDs) and solid state drives (SSDs), are among the components that affect data center dependability the most [40, 43, 23, 6], contributing to the 28% of data center failure events [50]. Accurate prediction of storage component failures enables on-time mitigation actions that avoid data loss and increases data center dependability. Failure events observed in HDDs and SSDs are different due to their distinct physical mechanics, it follows that observations from HDD analysis cannot be generalized to SSDs and vice-versa. Existing studies of HDD dependability in the literature generally focus on a few disk models that are analyzed using disk traces of collected disk data during a period of up to two years [7, 48]. Most of the SSD studies in the literature are done on a simulated or controlled environment [16, 54, 8, 38, 27] or focus on specific error types [31, 29, 42].

In this paper, we focus on failures of HDDs and SSDs by analyzing disk logs collected from real-world data centers over a period of six years. We analyze *Self-Monitoring, Analysis and Reporting Technology* (SMART) traces from five different HDD models from the Backblaze data center [5] and the logs of three multi-level cell (MLC) models of SSDs collected at a Google data center. Though we are unaware of the data centers' exact workflows for drive repairs, replacements, and retirements (e.g., whether they are done manually or automatically, or what are the replacement policies in place), we are able to discover key correlations and patterns of failure, as well as generate useful forecasts of future failures. Being able to predict an upcoming drive retirement could allow early action: for example, early replacement before failure happens, migration of data and VMs to other resources, or even allocation of VMs to disks that are not prone to failure [51].

In this paper, we study the various error types accounted by the logs to determine their roles in triggering, or otherwise portending, future drive failures. We note that although both the Backblaze and the Google logs have ample data, simple statistical methods cannot be used for predictions: we find no evidence that failures are triggered by any deterministic decision rule, e.g., a failure cannot be predicted using simple rules such as a feature exceeding a threshold. To address this, we resort to machine learning predictors to detect which quantitative measures provide strong indications of upcoming failures. We show that machine learning models that are trained from monitoring logs achieve failure prediction that is both remarkably accurate and timely, both in the HDD and SSD domains. The models confirm that drive replacements are not triggered by naive feature thresholds. Beyond prediction, the models are interpreted to provide valuable insights on which errors and workload characteristics are most indicative of future catastrophic failures.

The models are able to anticipate failure events with good accuracy up to several days in advance, despite some inherent methodological challenges. Although the frequency of failures is significant, the datasets in hand are highly imbalanced (the healthy-faulty ratio is 10,000:1). This is a common problem in classification, and makes achieving simultaneously high true positive rates and low false positive ones very difficult. Here, we illustrate that there exist different ways to partition the HDD and SSD datasets to increase model accuracy. This partitioning is based on workload analysis that was first developed in [2] for SSDs and focuses on the discovery of certain drive attributes. We saw that similar partitioning can be also successfully applied for the case of HDDs.

Although the dataset partitioning is based on already available drive attributes (e.g., drive age for SSDs, head flying hours for

• *R. Pinciroli is with the Computer Science Department, Gran Sasso Science Institute, Italy. E-mail: riccardo.pinciroli@gssi.it*
L. Yang, E. Smirni are with the Computer Science Department, William and Mary, Williamsburg, VA 23185. E-mail: {lyang11,esmirni}@cs.wm.edu
J. Alter is with Ipsos North America. E-mail: jralter@email.wm.edu

978-1-5386-5541-2/18/$31.00 ©2022 IEEE

HDDs), to the best of our knowledge this is the first time that such an approach is used to improve the accuracy of failure prediction models of storage devices. We also focus on the interpretability of the machine learning models and derive insights that can be used to drive proactive disk management policies. Our findings are summarized as follows:

- Although a consistent number of drives fail during the observed six years (i.e., 14% of SSDs and 7% of HDDs), only a few of them are repaired and put back into production (i.e., 50% of failed SSDs and 7% of failed HDDs). The percentage of failed SSDs that are put back into production within a month after the failure event is 8%, while almost all the repaired HDDs are back in the data center within a day from the failure.
- Drive failures are triggered by a set of attributes and different drive features must be monitored to accurately predict a failure event. There is no single metric that triggers a drive failure after it reaches a certain threshold.
- Several machine learning predictors are quite successful for failure prediction. Random forests are found to be the most successful for SSDs and HDDs.
- Datasets may be partitioned to improve the performance of the classifier. Partitioning SSDs on *drive age* attribute and HDDs on *head flying hours* (i.e., SMART 240) increases model accuracy. This is enabled by training a distinct classifier for each partition of the dataset.
- The attributes that are most useful for failure prediction differ depending on the dataset split (i.e., split based on age for SSDs and on head flying hours for HDDs).
- No relationship between SSD program-erase (P/E) cycles and failures are observed, suggesting that write operations do not affect the state of SSDs as previously thought.

The remainder of the paper is organized as follows. Section 2 describes the datasets analyzed in this paper and Section 3 characterizes the data and summarizes observations. Section 4 investigates main reasons causing SSD and HDD failures. Section 5 studies different classification algorithms to predict failure events. Section 6 presents related work and Section 7 concludes the paper.

2 SSD AND HDD DATASET

SSD dataset: The data consists of daily performance logs for three MLC SSD models collected at a Google data center over a period of six years. All three models are manufactured by the same vendor and have a 480GB capacity and a lithography on the order of 50nm. They utilize custom firmware and drivers, meaning that error reporting is done in a proprietary format rather than through standard SMART features [3]. This dataset has been previously examined in [41, 42, 29]. We therefore refer to the three SSD models as MLC-A, MLC-B, and MLC-D in accordance with the naming in [41, 29]. The dataset provides over 10,000 unique drives for each drive model, totaling over 40,000,000 daily drive reports overall.

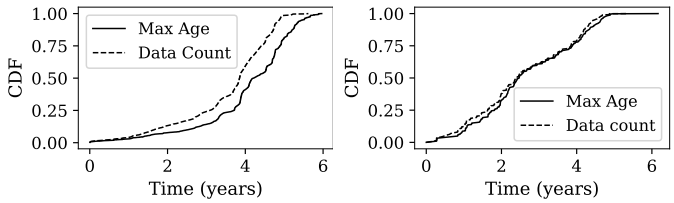
The logs report daily summaries of drive activity. Drives are uniquely identified by their ID, which is a hashed value of their serial number. For each day, the following metrics are reported:

- The timestamp since the beginning of the drive's lifetime, given in microseconds.
- The number of read, write, and erase operations performed by the drive over the course of the day.
- The cumulative number of program/erase (P/E) cycles over the drive lifetime. P/E is the process by which a memory cell is written to and subsequently erased. The cumulative amount of P/E cycles is a measure of device wear.
- Two status flags indicating whether the drive has died and whether the drive is operating in read-only mode.
- The number of *bad blocks* in the drive. A block is marked bad either when it is non-operational upon purchase (denoted as *factory bad block*) or when a non-transparent error occurs in the block and it is subsequently removed from use. Cumulative counts of these two measures are provided in the log.
- The counts of different errors that have occurred over the course of the day. Table 1 lists the logged error types.

HDD dataset: The HDD dataset contains daily logs for HDDs collected at a Backblaze data center over six years. Basic information such as serial number, model, and date is reported, as well as standard SMART features. Some SMART features are not reported for Hitachi drives and also for Seagate drives before 2014, for this reason we only consider Seagate drives starting from January 17, 2014, until December 31, 2019. The dataset contains more than 17 different HDD models, over 100,000 unique HDDs each identified by its serial number, with more than 100,000,000 daily reports per HDD. Of the 17 different models, we consider the 5 most popular ones. Every day, a snapshot operation is performed for every operational hard drive, reporting the following metrics:

- The timestamp of the snapshot, which is the date of the snapshot.
- The serial number and model of the hard drive.
- Flag of failure: 0 shows that the drive is functioning, while 1 means this is the last operational day before failure.
- SMART features: All SMART features are included in the dataset, however, some of them are empty. We are interested in the non-null features, shown in Table 2.

In addition to the raw features, we also consider the cumulative effect of SMART 187 (Reported Uncorrectable Errors). For cumulative features (i.e., SMART 4, 5, 7, 9, 10, 12, 192, 193, 197, 198, 199, 240, 241, and 242), we also calculate their non-cumulative version (the difference with the previous observation).



(a) SSD (b) HDD

Fig. 1: CDFs of maximum observed drive age (solid) and number of observed drive days within the error logs (dashed). The HDD maximum age is shorter than the SSD one.

Drive age and data count: For a given drive, the error log may have observations spanning over a period of several days up to several years. This is depicted in the "Max Age" CDF in Fig. 1, which shows the distribution over the "oldest" observations for each drive, i.e., the length of the observational horizons in the logs. We show the CDF of SSDs and HDDs separately in Figs. 1(a) and

TABLE 1: Name and description of different error types logged in the SSD dataset.

Name	Description	Name	Description
correctable error	Number of bits that were found corrupted and corrected using drive-internal error correction codes (ECC) during read operations during that day	read error	Number of read operations that experienced an error but succeeded on retry (initiated drive-internally)
erase error	Number of erase operations that failed	response error	Number of bad responses from the drive
final read error	Number of read operations that fail, even after (drive-initiated) retries	timeout error	Number of operations that timed out after some wait period
final write error	Number of write operations that fail, even after (drive-initiated) retries	uncorrectable error	Number of uncorrectable ECC errors encountered during read operations during that day
meta error	Number of errors encountered while reading drive-internal metadata	write error	Number of write operations that experienced an error but succeeded on retry (initiated drive-internally)

TABLE 2: ID, name, and description of considered SMART features for HDDs.

ID	Name	Description	ID	Name	Description
1	Read Error Rate	Frequency of errors when performing a read operation	190	Temperature Difference	Difference between a manufacturer-defined threshold (usually 100) and the measured temperature
3	Spin-Up Time	Average time of spindle spin-up in milliseconds	192	Power-off Retract Count	Number of power-off or retract cycles
4	Start-Stop Count	Number of spindle start-stop cycles	193	Load/Unload Cycle Count	Number of load/unload cycles
5	Reallocated Sectors Count	Number of reallocated sectors	194	Temperature	Temperature of the hard drive
7	Seek Error Rate	Frequency of errors while the drive head is positioning	197	Current Pending Sector Count	Number of sector pending for reallocation
9	Power-On Hours	Number of hours of the drive in power-on state (drive age)	198	Uncorrectable Sector Count	Number of uncorrectable sector
10	Spin Retry Count	Number of spin start retries	199	UltraDMA CRC Error Count	Number of errors identified by the Interface Cyclic Redundancy Check (ICRC) during data transfer
12	Power Cycle Count	Number of cycles when the drive is in power-on state	240	Head Flying Hours	Number of hours when the drive head is positioning
187	Reported Uncorrectable Errors	Number of uncorrectable ECC errors	241	Total LBAs Written	Number of Logical Block Addressing (LBAs) written
188	Command Timeout	Number of aborted operations due to HDD timeout	242	Total LBAs Read	Number of Logical Block Addressing (LBAs) read

TABLE 3: Proportion of drive days that exhibit each error type. 1.0 represents a proportion of 100%.

	Error Type	correctable error	final read error	final write error	meta error	read error	response error	timeout error	uncorrectable error	write error
SSD	MLC-A	0.828895	0.001077	0.000026	0.000014	0.000090	0.000001	0.000009	0.002176	0.000117
	MLC-B	0.776308	0.001805	0.000027	0.000016	0.000103	0.000004	0.000010	0.002349	0.001309
	MLC-D	0.767593	0.001552	0.000034	0.000028	0.000133	0.000002	0.000014	0.002583	0.000162

	Error Type	uncorrectable error	ultraDMC CRC error	reallocated sector count	power-off retract count	uncorrectable sector count
HDD	ST12000NM0007	0.004301	0.007232	0.014590	0.330956	0.003755
	ST3000DM001	0.216522	0.303442	0.092016	0.001818	0.074695
	ST4000DM000	0.014837	0.022238	0.004435	0.000549	0.006698
	ST8000DM002	0.005044	0.007391	0.022137	0.010971	0.00377
	ST8000NM0055	0.004534	0.007392	0.020378	0.020668	0.003742

(b), respectively. We observe that, for over 50% of drives, we have data extending over a period of 4 to 6 years for SSDs, and 2 to 5 years for HDDs, which shows that generally HDDs have a shorter lifespan. Note that days of activity are not logged for all drives. Accordingly, we may ask what magnitude of data we have access to for a given drive. The accompanying "Data Count" CDF shows exactly this: the number of drive days that are recorded in the error log for each drive. "Data Count" is a measure of the volume of the log entries. Measuring them as a function of time is reasonable as there is one log entry per day per drive. Fig. 1 clearly shows that there are ample data available for both SSDs and HDDs, and therefore amenable to the analysis presented in this paper.

Collected errors are separated into two types: *transparent* and *non-transparent* errors. Transparent errors (i.e., correctable, read, write, and erase errors) may be hidden from the user while non-transparent errors (i.e., final read, final write, meta, response, timeout, and uncorrectable errors) may not. Error type statistics are listed in Table 3 where 1.0 represents a proportion of 100%. Note that meta, response, and timeout errors in the SSD dataset are very rare. The number of uncorrectable and final read errors is at least one order of magnitude larger than other errors. Correctable errors are the most common type of errors, as also reported in [41]. For a detailed analysis of the SSD dataset that focuses on raw bit error rates, uncorrectable bit error rates, and their relationships with workload, age, and drive wear out, we direct the interested reader to [41]. For HDDs, only two types of errors are reported (i.e., uncorrectable error and ultraDMC CRC error). In addition to these two errors, we also consider reallocated and uncorrectable sector counts, which are highly related to errors [36], and power-off retract counts. Comparing different HDD models, ST3000DM001 has the highest rate of different errors and unhealthy behaviors as also observed in [11].

TABLE 4: Matrix of Spearman correlations among different features. We calculate the features in SSDs and HDDs separately. Bolded text indicates a large correlation value.

		erase	final read	final write	meta	read	response	timeout	uncorrect.	write	P/E cycle	bad block
SSD	erase	1.00										
	final read	0.21	1.00									
	final write	0.24	0.12	1.00								
	meta	0.17	0.19	**0.35**	1.00							
	read	0.22	0.20	0.30	**0.40**	1.00						
	response	0.02	0.06	0.24	0.02	0.03	1.00					
	timeout	0.01	0.12	**0.44**	0.02	0.03	**0.53**	1.00				
	uncorrectable	0.20	**0.97**	0.06	0.16	0.15	0.03	0.03	1.00			
	write	**0.32**	0.28	0.13	0.14	0.25	0.02	0.02	0.28	1.00		
	P/E cycle	**0.32**	0.18	−0.05	−0.02	0.03	0.03	0.00	0.19	0.23		
	bad block count	**0.38**	**0.37**	0.19	0.19	0.18	0.01	0.01	**0.37**	**0.34**	0.16	1.00
	drive age	0.20	**0.36**	0.06	0.05	0.06	0.04	0.05	**0.36**	0.14	**0.73**	0.18
		failure	reallocated sector	current pending sector	ultraDM CRC	uncorrect.	power-off retract	timeout	power-on hour	power cycle	temp.	
HDD	failure	1.00										
	reallocated sector	0.03	1.00									
	current pending sector	0.05	0.14	1.00								
	ultraDMC CRC	0.01	0.02	0.03	1.00							
	uncorrectable	0.04	0.28	**0.39**	0.07	1.00						
	power-off retract	0.00	0.05	0.00	0.02	-0.02	1.00					
	timeout	0.01	0.04	0.05	0.30	0.11	0.03	1.00				
	power-on hours	0.00	0.04	0.04	0.06	0.08	-0.21	0.07	1.00			
	power cycle	0.00	0.02	0.04	0.13	0.08	-0.07	0.21	**0.54**	1.00		
	temperature	0.00	0.03	-0.01	-0.05	-0.03	**0.33**	-0.06	**-0.32**	-0.30	1.00	
	head flying hours	0.00	0.03	0.01	0.03	0.01	-0.09	0.02	-0.10	0.03	-0.11	

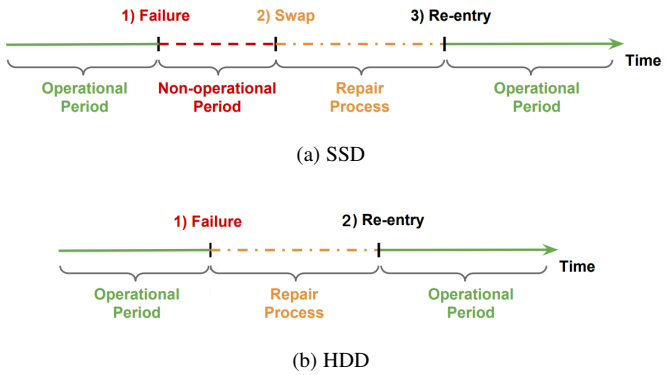
Correlation among errors and different features: Table 4 illustrates the Spearman correlation matrix across pairs of measures for the SSD and HDD sets, aiming to determine whether strong co-incidence relationships between different error types exist. Spearman correlations are used as a non-parametric measure of correlation, with values ranging between -1 and $+1$. The Spearman correlation differs from the more common Pearson correlation in that it is able to detect all sorts of monotonic relationships, not just linear ones [10]. Bolded values are those with magnitude greater than or equal to 0.30, indicating a non-negligible relationship between the pair of measures.

We first discuss the correlation of P/E cycle count with other errors in the SSD dataset. Since SSDs can only experience a finite number of write operations before their cells begin to lose their ability to hold a charge, manufacturers set a limit to the number of P/E cycles that a given drive model can handle. For our drive models, this limit is 3000 cycles [41]. Due to these limits, it is believed that errors are caused in part by wear of write operations on the drive, which one can measure using either a cumulative P/E cycle count or a cumulative count of write operations. Using either measure is equivalent since they are very highly correlated. It is interesting to observe in Table 4 that there is little-to-no correlation of P/E cycle count with any of the other errors, except for some moderate correlation with erase errors, which contradicts common expectations. One reason for this is the aforementioned argument regarding device wear. Another is that drives which experience more activity simply have more opportunities for other errors to occur. Note that the correlation value of P/E cycle count and uncorrectable error count (which reflects bad sectors and eventual drive swap) is mostly insignificant. The age of a drive gives a similar metric for drive wear, which correlates highly with the P/E cycle count. The drive age also has very small correlation with cumulative error counts, with the exception of uncorrectable/final read errors.

Bad blocks, another likely culprit for drive swaps, shows some mild correlation with erase errors, final read errors, and uncorrectable errors. The high value of the correlation coefficient of 0.97 between uncorrectable errors and final read errors is not useful as the two errors represent essentially the same event: if a read fails finally, then it is uncorrectable. Yet, we see some moderate correlation values between certain pairs of measures that eventually show to be of use for swap prediction within the context of machine learning-based predictors, see Section 5.

For the HDD dataset, we present the correlation of power-on hours (which reflects drive age), power cycle, and temperature with different errors and abnormal behaviors. All of them have little-to-no correlation with other drive errors and drive failures. Only temperature shows mild correlation with power-off retract count. Uncorrectable errors and current pending sector count have some correlation because the pending sector count reflects the number of unstable sectors in an HDD, and as more and more pending sectors appear in the drive, the more uncorrectable errors are observed. Power cycle and power-on hour are correlated, because they are both power-related metrics.

Observation #1: For SSDs, there is no clear relationship between non-transparent error types and uncorrectable error counts that presumably result in bad sectors and eventual drive failures. Program–erase (P/E) cycle counts, an indicator of drive wear, show low correlations with most uncorrectable errors and mild correlation with erase errors (transparent errors). Drive age shows a similar pattern of correlation.



(a) SSD

(b) HDD

Fig. 2: Overview of failure timeline. Different from SSDs, we cannot identify non-operational periods and swaps in HDDs.

TABLE 5: High-level failure statistics. This includes, for each model, the number of failures observed and the proportion of drives that are observed to fail at least once.

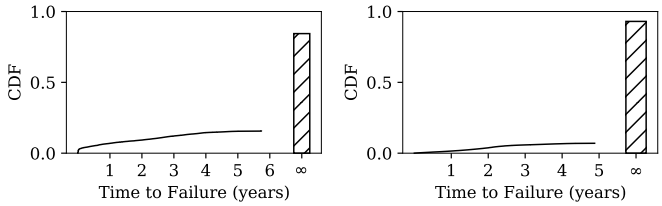
	Model	#Failures	%Failed
SSD	MLC-A	734	6.95
	MLC-B	1565	14.3
	MLC-D	1580	12.5
	All	3879	11.29
HDD	ST12000NM0007	1448	3.76
	ST3000DM001	1357	31.89
	ST4000DM000	3710	10.21
	ST8000DM002	354	3.47
	ST8000NM0055	435	2.91
	All	7743	7.01

Observation #2: For both datasets, correlations among all pairs of transparent and non-transparent errors show that some pairs may be mildly correlated and can be useful in prediction. Yet, there is no strong indication as to which measures are most useful for prediction.

3 Drive Failure and Repair

In this section, we focus on drive failures and actions following a failure. Table 5 shows the percentage of failures for each drive model in SSD and HDD datasets. On average, the failure rate of SSDs is higher than the one of HDDs. The largest failure rate of 31.89% is observed for ST3000DM001, an HDD model. This high frequency of failures poses a large pressure in terms of maintenance costs, since each failure requires manual intervention. Table 6 provides more insights by reporting statistics on the frequency of failures for the *same* drive. Unexpectedly, we find that some SSDs have failed as many as four times over the course of their lifetime. Nonetheless 89.6% of drives with failures, fail only once. HDDs have no more than 2 failures per drive, and only 0.2% of the failed drives has 2 failures, which is very rare for HDDs comparing to 9.2% for SSDs. Differences in the number of failures of HDDs and SSDs may be due to different replacement strategies adopted by the owners of the two datasets.

To better characterize the conditions of failure that lead to these repairs, we pinpoint the failures in the timeline. For HDDs, the failure event is directly logged using the *failure* feature. Fig. 2(b) shows the timeline of an HDD: 1) a failure can happen during the operational period, then the failed HDD is sent for repair. 2) The failed drive may or may not be repaired successfully. If it is repaired, it re-enters its operational period.



(a) SSD (b) HDD

Fig. 3: CDF of the length of the drive's operational period. The bar indicates what proportion of operational periods are not observed to end. Failure rate of SSDs is larger than HDDs.

For SSDs, in addition to daily performance metrics, special "swap" events are also reported in the data. These events indicate the time at which failed drives are extracted to be repaired. Swaps denote repairs – and not simply a swapping out for storing spare parts, or moving a healthy SSD to a storage cabinet. All swaps follow drive failures, and accordingly, each swap documented in the log corresponds to a single, catastrophic failure.

We now discuss what we consider to be "operational activity." It is often the case (roughly 80% of the times) that swaps are preceded by at least one day for which *no performance summaries* are documented in the SSD log. This indicates that the drive was non-operational during this period, having suffered a complete failure. A natural way to proceed is to define failure events with respect to swap events: a failure occurs on a drive's last day of *operational activity* prior to a swap. *Inactivity* refers to an absence of read or write operations provisioned to the drive, this is experienced prior to 36% of swaps. and amounts to a "soft" removal from production before the drive is physically swapped. Accordingly, we define a failure as happening directly prior to this period of inactivity, if such a period exists.

To summarize, SSD repairs undergo the following sequence of events, see Fig. 2(a): 1) At some point, the drive undergoes a failure, directly after which the drive may cease read/write activity, cease to report performance metrics, or both, in succession. 2) Data center maintenance takes notice of the failure and swap the faulty drive with an alternate. Such swaps are notated as special events in the data. 3) After trying to repair a failed drive, it may or may not be returned to the field to resume normal operation.

We will now characterize each of the stages. We start by examining the operational periods observed in the log. Fig. 3 presents the CDF of the length of operational periods (alternately denoted "time to failure"). The CDF includes both operational periods starting from the beginning of the drive lifetime and operational periods following a post-failure re-entry. It is interesting to note that more than 80% and 90% of operational periods of SSDs and HDDs, respectively, are not observed to end with failure during the 6 year sampling period; this probability mass is indicated by the bar centered at infinity. The figure shows that there is substantial variability in the drive operational time, with the majority of operating times being long. Yet, there is a non-negligible portion of operating times that result in failure. Comparing SSDs with HDDs in Figs. 3(a) and (b), the maximum time to failure of HDDs is less than 5 years, while the one of SSDs is close to 6 years. Only 7% of HDDs fail after an operational period, while for SSDs it is near to 20%. This is due to 1) the lower failure rate of HDDs compared to the one of SSDs and 2) the different maximum number of failures on a single drive (i.e., 2 for HDDs

TABLE 6: Statistics of failure counts. Statistics are given with respect to the entire population of drives and with respect to those drives which fail at least once ("failed drives").

	Number of Failures	% of drives	% of failed drives
SSD	0	88.72	—
	1	10.10	89.6
	2	1.04	9.2
	3	0.13	1.2
	4	0.01	0.1
HDD	0	93.35	—
	1	6.64	99.8
	2	0.01	0.2

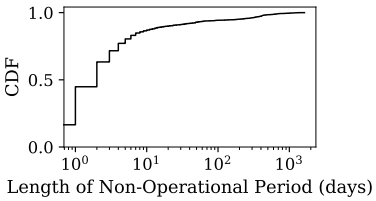


Fig. 4: CDF of length of non-operational period preceding a swap. This is the number of days between the swap-inducing failure and the physical swap itself. There is no non-operational period reported in the Backblaze dataset for HDDs.

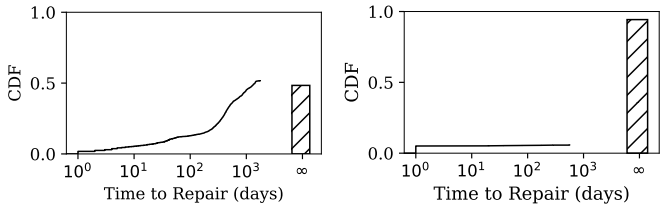
and 4 for SSDs, see Table 6).

Fig. 4 shows the CDF of the length of the pre-swap non-operational period (only for SSDs), i.e., the elapsed time between the drive failure and when it is swapped out of production. Roughly 20% of failed drives are removed within a day and 80% of failed drives are swapped out of the system within 7 days. However, this distribution has a very long tail (note the logarithmic scale on the x-axis). A non-negligible proportion of failed drives (roughly 8%) remain in a failed state up to 100 days before they are removed from production. Since these faulty drives can remain in limbo for upwards of a year, the data suggest that these drives may simply have been forgotten in the system. Note that the length of non-operational periods might differ among SSDs due to different policies adopted by Google at each data center. The data set provides no information about the physical location of its drives or the used policy.

Fig. 5 depicts the CDF of the *time to repair*. There are significant differences in the time to repair for SSDs and HDDs. Half of SSDs do not re-enter the field (i.e., their time to repair is infinite – their share of probability mass is again indicated by the bar), while more than 90% of HDDs are never successfully repaired. Among drives that *are* returned to the field, the majority of SSDs remains in for repair for upwards of a year, while most of the HDDs are repaired in less than one day.

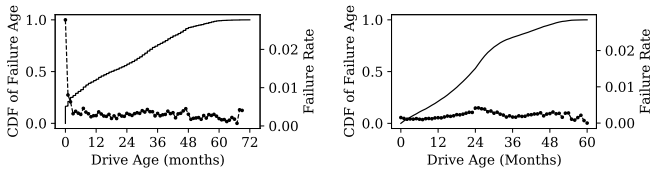
Restating the results in Fig. 5, Table 7 illustrates the percentage of failed drives that are repaired and re-enter the system after a period of n days in repair (the percentage of the successfully repaired drives as a function of all drives is also shown within the parentheses). These metrics show that HDDs are rarely repaired, probably due to their low cost (i.e., it is cheaper to substitute an HDD than repair it). Although roughly half of SSDs return to the production environment, their repair time is lengthy.

Observation #3: Failed SSDs are often swapped out of production within a week, though a small portion may remain in the system even longer than a year.



(a) SSD (b) HDD

Fig. 5: CDF of time to repair, ranging from 1 day to 4.85 years for SSDs, and ranging from 1 day to 1.54 years for HDDs. The proportion of repairs that are not observed to terminate is depicted with the bar graph. Half of the failed SSDs get repaired, while only 6% of failed HDDs are repaired.



(a) SSD (b) HDD

Fig. 6: The CDF of the age of failed drives (solid line) and the proportion of functioning drives that fail at a given age level, in months (dashed/dotted line).

Observation #4: While a significant percentage of SSDs (up to 14.3% for MLC-B, slightly smaller percentages for other MLC types) are swapped during their lifetime, only half of failed drives are seen to be successfully repaired and re-enter the field.

Observation #5: Of those repairs that do complete, only a small percentage of them finish within 10 days. About half of SSDs that are swapped out are not successfully repaired.

Observation #6: The failure rate of HDDs is lower than SSDs, and only 7% of failed HDDs are successfully repaired.

4 Symptoms and Causes of Failures

In this section we make an effort to connect the statistics from the two logs, aiming to identify causes of drive failures. Since the features/information provided by the two datasets are different, we analyze SSDs and HDDs separately.

4.1 Age and Device Wear

Recall that Table 5 shows that, among the drives represented in the datasets, 11.29% of SSDs and 7.01% of HDDs fail at least once. A natural question is *when* do these failures occur in the drive's lifetime: i.e., what is the role of age in drive failure? Fig. 6(a) reports the CDF of the failure age (solid line) as a function of the drive age for SSDs. The figure shows that there are many more drive failures in the first 90 days of drive operation than at any other point in the drive lifetime. In fact, 15% of observed failures occur on drives less than 30 days old and 25% occur on drives less than 90 days old. This implies that these drives have an infancy period during which drive mortality rate is particularly high, such patterns have been noticed in past studies of SSDs in the wild [31].

TABLE 7: Percentage of swapped drives for SSD and percentage of repaired drives for HDD that re-enter the workflow within *n* days. The percentage of repaired drives as a function of all drives is also reported within the parentheses.

	Model	1 day	10 days	30 days	100 days	1 year	2 years	3 years	∞
SSD	MLC-A	1.2 (0.09)	3.4 (0.23)	5.0 (0.34)	6.1 (0.43)	17.4 (2.61)	37.6 (2.61)	43.6 (3.03)	53.4 (3.71)
	MLC-B	2.5 (0.36)	6.8 (0.98)	9.4 (1.34)	12.7 (1.81)	25.3 (3.62)	36.1 (5.16)	42.7 (6.11)	43.9 (6.28)
	MLC-D	1.1 (0.14)	4.9 (0.61)	8.1 (1.01)	15.8 (1.97)	28.1 (3.51)	43.5 (5.44)	50.2 (6.28)	57.6 (7.20)
	All	1.7 (0.19)	5.4 (0.61)	8.0 (0.91)	12.7 (1.43)	25.0 (2.82)	39.4 (4.45)	45.9 (5.19)	51.3 (5.79)
HDD	ST3000DM001	24.98 (7.97)	25.28 (8.06)	25.42 (8.11)	25.42 (8.11)	25.42 (8.11)	25.42 (8.11)	25.42 (8.11)	74.58 (23.78)
	ST4000DM000	0.62 (0.06)	0.62 (0.06)	0.7 (0.07)	0.81 (0.08)	0.97 (0.1)	0.97 (0.1)	0.97 (0.1)	99.03 (10.11)
	ST8000NM0055	0.0 (0.0)	0.0 (0.0)	0.23 (0.01)	0.69 (0.02)	1.15 (0.03)	1.15 (0.03)	1.15 (0.03)	98.85 (2.87)
	ST8000DM002	0.28 (0.01)	0.28 (0.01)	0.85 (0.03)	1.69 (0.06)	2.54 (0.09)	2.82 (0.1)	2.82 (0.1)	97.18 (3.37)
	ST12000NM0007	0.48 (0.02)	0.55 (0.02)	0.76 (0.03)	1.04 (0.04)	1.45 (0.05)	1.45 (0.05)	1.45 (0.05)	98.55 (3.7)
	All	4.83 (0.32)	4.9 (0.33)	5.05 (0.34)	5.29 (0.35)	6.17 (0.41)	6.21 (0.42)	6.21 (0.42)	93.79 (6.28)

The slope of the CDF in Fig. 6(a) gives us an estimate of the rate at which swaps occur at a given drive age. However, this estimate is skewed since not all drive ages are equally represented in the data. For example, the rate of failures seems to slow down following the four year mark, but this is due to the fact that drives of this age level that *fail* are not as common in the data (i.e., the total number of drives that fail and are older than 48 months is only 389). We hence normalize the number of swaps within a month by the amount of drives represented in the data at that month to produce an unbiased failure *rate* for each month (dashed/dotted line in Fig. 6). We see that this rate evens out after the third month, indicating that the length of this observed high-failure infancy period is approximately 90 days. Accordingly, for the remainder of this paper, we distinguish drive swaps as *young* versus *old*, i.e., those swaps occurring before vs. after the 90-day mark. Beyond the 90-day mark, we observe that the failure rate is roughly constant, suggesting that, even if drives become very old, they are not more prone to failure.

A potential explanation for the spike in failures for infant drives is that they undergo a "burn-in" period. This is common practice in data centers, wherein new drives are subjected to a series of high-intensity workloads in order to test their resilience and check for manufacturing faults. Such increased workloads stress the drive, leading to a heightened rate of failure. To test this hypothesis, we looked at the intensity of workloads over time. For each month of drive age, we examined drives of that age and how many write operations they processed per day. The distributions of these write intensities are presented in Fig. 7. It is clear that younger drives do *not* tend to experience more write activity than usual (in fact, they tend to experience markedly *fewer* writes!). A similar trend is apparent for read activity (not pictured). We conclude that there is no burn-in period for these drives and that the spike in failure rates is caused by manufacturing malfunctions not caught by drive testing. It is worth noting that the daily write intensity is higher for SSDs that are older than 48 months. Given that 17,805 *healthy* drives in the dataset (i.e., drives that do not have any failure) are older than 48 months, observations for these drives are statistically significant. This phenomenon may be the result of company policy to route write requests to old drives or of a workload variation experienced by the data center in the last 48 months.

Beyond drive age, we are also interested in the relationship between failure and device wear, which is given by P/E cycles. In the same style as Fig. 6, Fig. 8 illustrates the relationship of cumulative P/E cycles and probability of failure in the form of a CDF (solid line) and an accompanying failure rate (dashed line). The CDF illustrates that almost 98% of failures occur before the drive sees 1500 P/E cycles. This is surprising, considering that the manufacturer guarantees satisfactory drive performance up until 3000 P/E cycles. Conversely, the failure rate beyond the P/E cycle limit is very small and roughly constant. The spikes at 4250 and 5250 P/E cycles are artifactual noise attributed to the fact that the number of drives that fail at these P/E levels are so few in number.

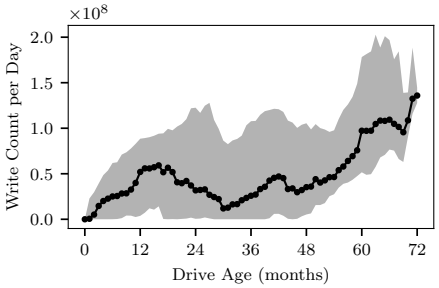


Fig. 7: Quartiles of the daily write intensity per month of drive age. The line shows the median write intensity for each month. The 1st and 3rd quartiles are shown as the boundaries of the shaded area.

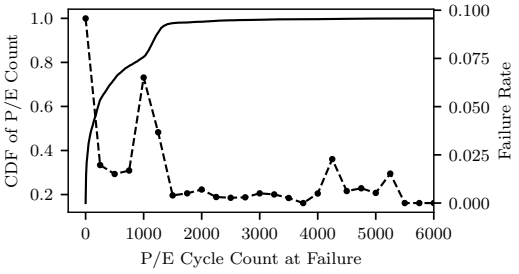


Fig. 8: The distribution P/E cycle counts of failed drives (solid) and the proportion of drives that fail (dashed) at a given P/E level, binned in increments of 250 cycles.

In the figures discussed, we observe high failures rates for both SSDs younger than three months and SSDs with fewer than 250 P/E cycles. Due to the correlation between age and P/E cycles, it might appear that these two characterizations are roughly equivalent and describe the same phenomenon. To illustrate that this is not the case, we plot two CDFs in Fig. 9: one for young failures and one for old ones. It is clear that the young failures inhabit a distinct, small range of the P/E cycle distribution. Since this range is so small, the individual P/E cycle counts are not informative to young failures.

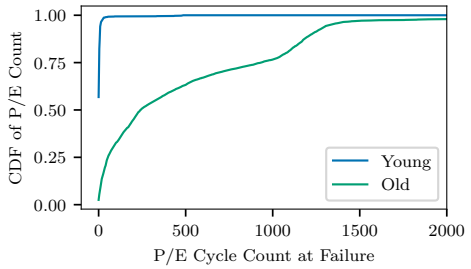


Fig. 9: The CDF in Fig. 8 split across infant failures (occurring at age ≤ 90 days) and mature failures (occurring at age > 90 days).

Observation #7: Age plays a crucial role in the SSD failure rate. Drives younger than 90 days have markedly higher failure rates. This phenomenon is characteristic to young drives and cannot be explained with P/E cycle counts.

Observation #8: Beyond the infancy period, age does not seem to play an important role. The oldest drives seem to fail with roughly the same frequency as young, non-infant drives.

Observation #9: The vast majority of drive failures happen well before their P/E cycle limit is reached. Furthermore, drives operating beyond their P/E cycle limit have low rates of failure.

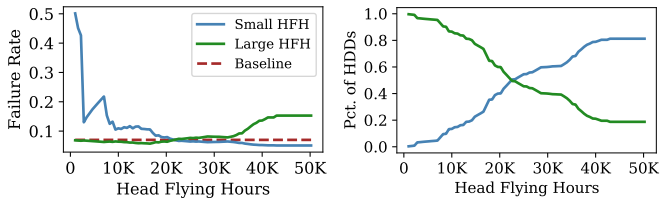
4.2 Head Flying Hours in HDDs

Similar to the drive age of SSDs shown in Fig. 6(a), we also present a similar plot for HDDs in Fig. 6(b). Differently from the high failure rate of young SSDs, the failure rate of young HDDs is relatively small (less than 1%). Therefore, we need to find other features which may be related to failure rate.

We examine *all SMART features* for HDDs, and find out that *head flying hours (HFH, SMART 240)* is highly related to failures even if it is not correlated with other HDD features (see Table 4). Here we define two kinds of disks regarding head flying hours with a certain *threshold*: 1) *Large HFH* disks are observed at least once with head flying hours larger than the *threshold*; 2) *Small HFH* disks always have head flying hours smaller than or equal to the *threshold*.

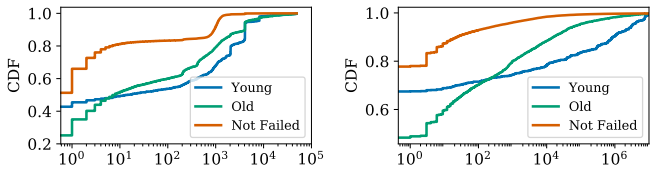
Fig. 10(a) shows the failure rate of small and large HFH drives as a function of the threshold. The *average* failure rate of all HDDs is also reported (baseline, see dashed line). We observe two situations with high failure rate in Fig. 10: 1) small HFH when the threshold is less than 3000 (the beginning of small HFH line), and 2) large HFH when threshold is larger than 40,000 (the end of large HFH line). The percentage of HDDs, if we partition the dataset according to HFH, is also shown in Fig. 10(b). When the threshold is less than 3000, the percentage of small HFH drives is less than 3%, therefore it is not representative. When the threshold is larger than 40,000, the percentage of large HFH is about 20%, which is worth to be considered. The failure rate of these 20% large HFH disks is 17%, which is much higher than the 7% failure average (baseline). We select as HFH $threshold = 40,000$ to split the dataset because this threshold balances well the failure rate and percentage of large HFH disks. This observation (small and large have different resilience behavior) guides us to split the dataset for better prediction, see Section 5.3 for more details.

Observation #10: HDDs with large head flying hours (SMART 240) show a greater failure rate than disks with small head flying hours.



(a) Failure Rate (b) HDD Percentage

Fig. 10: Head flying hours (SMART 240). Many HDDs fail when they have large head flying hours. The dashed-line in subfigure (a) marks the mean failure rate.



(a) Cumulative Bad Block Count (b) Cumulative Uncorrectable Error Count

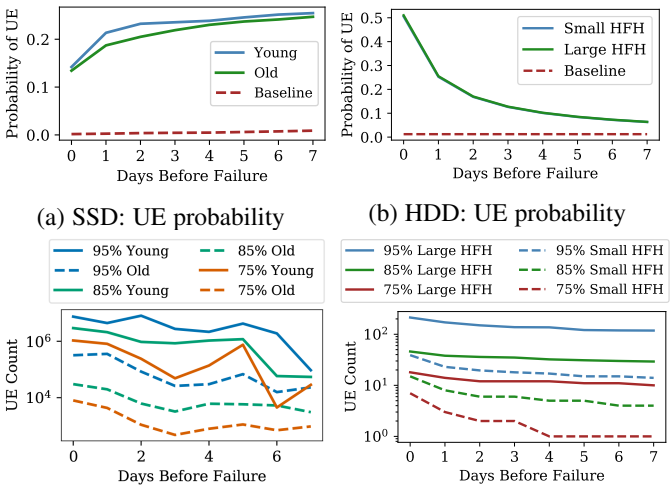
Fig. 11: CDF of cumulative bad block counts and uncorrectable error counts of SSDs based on the age at which the swap occurred. The "Not Failed" CDF corresponds to the distribution over drives that are not observed to fail.

4.3 Errors for SSDs

Intuitively, we would expect that catastrophic drive failure is preceded by previous lapses in drive function, i.e., non-transparent errors. We focus on uncorrectable errors and bad blocks since they are by far the most common of these errors. Other errors occur far too rarely to give much insight. We test the validity of our intuition by comparing the cumulative counts of errors seen by failed drives to a baseline of cumulative error counts taken across drives that are not observed to experience failure. We are also particularly interested to see if there is any difference in error rate between young failures (≤ 90 days) and old failures (> 90 days). This is illustrated with CDFs in Fig. 11.

We find that drives that fail tend to have experienced orders of magnitude more uncorrectable errors and bad blocks than other drives. This is exemplified by the fact that in roughly 80% of cases, non-failed drives are not observed to have experienced any uncorrectable errors. On the other hand, for failed drives, this proportion is substantially lower: 68% for young failures and 45% for old drives. In fact, broadening our scope, we find that 26% of failures happen to drives which have experienced no non-transparent errors *and* which have developed no bad blocks. Furthermore, we find that, if errors *are* observed, then young failures tend to see more of them than old failures. This is most easily seen in the tail behavior of the aforementioned CDFs; for example, the 90th percentile of the uncorrectable error count distribution is two orders of magnitude larger for young failures than for old failures, in spite of the fact that the young drives have been in operation for much less time.

Overall, the presence of errors is not a good indicator of drive failure since most failures occur without having seen *any* uncorrectable errors. However, drives that experience failure do have a higher error rate, which means that we expect error statistics to be of some utility in failure prediction. Furthermore, we find that



(a) SSD: UE probability (b) HDD: UE probability

(c) SSD: UE count (d) HDD: UE count

Fig. 12: (Top) Probability of uncorrectable error (UE) happening within the last n days before a swap. The baseline curve on the top graph is the probability of seeing an uncorrectable error within an arbitrary n-day period. (Bottom) Provided a UE happens, how many occur? Upper percentiles of the distribution of uncorrectable error counts preceding to failure, excluding zero counts.

the patterns of errors are markedly different among young and old failures. In particular, young failures have a predilection toward extremely high error counts.

Moving into a finer temporal granularity, we are interested in error rates directly preceding the failure. This behavior is of particular importance for failure forecasting and prediction. We ask: do drives tend to be more error-prone right before a failure occurs? How long before the drive failure is this behavior noticeable? Fig. 12 shows two relevant uncorrectable error statistics in the period before a drive swap.

Fig. 12(a) shows the probability that a faulty drive has an error within the last N days before its failure. The baseline is the probability of seeing an uncorrectable error within an arbitrary N-day period. We see that failed drives see uncorrectable errors with a much higher than average probability and that this behavior is most noticeable in the last two days preceding the failure. However, the proportions of failed drive that do not see any errors in its last 7 days is very high (about 75%).

Fig. 12(c) shows the distribution of those uncorrectable error counts that are nonzero on each day preceding the swap. We find that error counts tend to increase as the failure approaches. We also find that young failed drives, if they suffer an error, tend to experience orders of magnitude more errors than older ones, note the log-scale on the y-axis of the graph.

To summarize, we zoom in specifically on the period directly preceding a failure. We show that error rates depend on two factors: (1) the age of the drive (young vs. mature) and (2) the amount of time until the swap occurs. The resulting increase in error rate is most noticeable in the two days preceding the swap, suggesting that the lookahead window within which we can accurately forecast failure may be small.

Observation #11: Non-transparent errors are not strongly predictive of catastrophic drive failures. Moreover, a substantial proportion of drives experience failure without having seen non-transparent errors.

Observation #12: Failures of young drives are more likely to have seen higher error rates than failures of mature drives.

4.4 Errors for HDDs

We also show the percentage and count of uncorrectable errors (SMART 187) for HDDs, in Figs. 12(b) and 12(d). For HDDs, the probability of uncorrectable errors increases when a failure approaches. When a failure occurs, the probability of uncorrectable errors is around 50%. The probability of uncorrectable errors of small HFH and large HFH is essentially the same. For both cases, the probability is much higher than the baseline.

When looking into the number of uncorrectable errors, the uncorrectable error count of HDDs is several orders of magnitude lower than SSDs. For HDDs, we also observe higher uncorrectable error count for large HFH disks before failure events.

Observation #14: Error rates become extremely high in the two days before an HDD fails.

5 Failure Prediction

In this section, we use machine learning models to predict SSD and HDD failures that will occur within $N \geq 0$ days. The failure characterization of SSDs and HDDs presented in Section 4 is used to improve model accuracy and specifically by splitting the SSD and HDD datasets by drive age and head flying hours, respectively. The machine learning models are compared with results observed with state-of-the-art approaches [7, 29].

5.1 Model Description

Input. Both SSD and HDD datasets present daily statistics and are extremely imbalanced, i.e., the number of healthy disks (majority class) is larger than the number of faulty ones (minority class). In the SSD case, the ratio of healthy and defective drives is 10,000:1, for HDDs it is 13,000:1. To deal with such imbalanced datasets, we under-sample the majority class, use cross-validation for training and testing the model, and evaluate its performance with measures that are not affected by imbalanced datasets.

Under-sampling. For both datasets, the training set is under-sampled to result in a 1:1 healthy-faulty drives ratio to avoid the classifier being biased toward *healthy* drives. For this purpose, we use a random strategy, i.e., observations to be removed are randomly chosen from the majority class. We observe that the model performance is not profoundly affected by considering different under-sampling strategies and healthy-faulty ratios.

Cross-validation. Classifiers are cross-validated by splitting each dataset into five different folds. The dataset is partitioned by drive ID (i.e., all observations of a drive belong to the same fold and are not used concurrently for training and testing). If folds are created by randomly partitioning the dataset, i.e., the strategy adopted in [7, 29], it is possible that future observations of a drive are used to predict the past failure state of the same drive. This is undesirable since no future information is available in real scenarios. This is also avoided by using online prediction [51], i.e., the model is trained on observations collected before a specific date and tested on the remaining data. Unfortunately, this cannot be applied to the SSD dataset: its observations do not have a global timestamp attribute that would allow synchronizing the various traces across time. Four out of five folds are used for training, while the remaining one is used for testing. Five different classifiers are trained (each one tested on a different testing set).

TABLE 8: AUROC for different predictors and lookahead windows, N. The cross-validated average AUROC is provided with the standard deviation across folds.

	N (lookahead days)	0	1	2	7
SSD	Logistic Reg.	0.796 ± 0.010	0.765 ± 0.009	0.745 ± 0.007	0.713 ± 0.010
	k-NN	0.816 ± 0.013	0.791 ± 0.009	0.772 ± 0.008	0.716 ± 0.008
	SVM	0.821 ± 0.014	0.795 ± 0.011	0.778 ± 0.011	0.728 ± 0.011
	Neural Network	0.857 ± 0.007	0.828 ± 0.004	0.803 ± 0.009	0.770 ± 0.008
	Decision Tree	0.872 ± 0.007	0.840 ± 0.007	0.819 ± 0.005	0.780 ± 0.006
	XGBoost	0.904 ± 0.001	0.873 ± 0.002	0.851 ± 0.001	0.809 ± 0.001
	Random Forest	0.905 ± 0.008	0.859 ± 0.007	0.839 ± 0.006	0.803 ± 0.008
HDD	Logistic Reg.	0.668 ± 0.001	0.669 ± 0.001	0.668 ± 0.001	0.668 ± 0.001
	k-NN	0.699 ± 0.022	0.699 ± 0.026	0.701 ± 0.029	0.691 ± 0.029
	SVM	0.679 ± 0.012	0.689 ± 0.009	0.685 ± 0.011	0.684 ± 0.009
	Neural Network	0.684 ± 0.065	0.683 ± 0.076	0.685 ± 0.077	0.682 ± 0.076
	Decision Tree	0.886 ± 0.051	0.870 ± 0.053	0.862 ± 0.053	0.837 ± 0.052
	XGBoost	0.904 ± 0.001	0.888 ± 0.001	0.878 ± 0.001	0.854 ± 0.001
	Random Forest	0.903 ± 0.013	0.888 ± 0.010	0.878 ± 0.008	0.854 ± 0.006

Global performance is obtained by averaging the performance of each classifier.

Output. The model returns a continuous value in the interval $(0,1)$, i.e., the probability that the drive fails. In real-world scenarios, a binary output (i.e., failure vs. non-failure) is preferred. For this purpose, we set a discrimination threshold, α, that discretizes the returned probability: if the output is larger than α, then the model predicts a failure; otherwise, the model predicts a non-failure. Due to its insensitiveness to imbalanced datasets, receiver operating characteristic (ROC) is generally used to evaluate the accuracy of binary classifiers [12, 29] and is adopted also in this paper. ROC curve plots the true positive rate (TPR or recall) against the false positive rate (FPR) of the analyzed classifier by considering different values of α. TPR and FPR are defined as:

$$TPR = \frac{TP}{TP+FN} \quad \text{and} \quad FPR = \frac{FP}{FP+TN},$$

where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives. We also consider the area under the ROC curve (i.e., AUROC) as a further measure to determine the goodness of the proposed classifier. The AUROC is always in the interval $(0.5,1)$: it is 0.5 if the prediction is not better than the one of a random classifier; its value is 1 for a perfect predictor.

5.2 Prediction Accuracy

To compare classifiers, we report in Table 8 the AUROC of different predictors for different lookahead windows. The table illustrates that Random Forest and XGBoot result in superior performance comparing to other predictors, this is consistent with the literature ([2] and [29] for SSDs and HDDs, respectively). Although the performance of XGBoost and Random Forest models are similar (in fact, XGBoost models may be slightly more accurate than Random Forest ones), the training time of Random Forest is only 5% of the time required for training XGBoost. Table 8 also shows that, independently of the classifier used, the smaller the lookahead window, the larger the AUROC.

Fig. 13 plots the AUROC of the Random Forest prediction on HDD (solid line) and SSD (dashed line) datasets against different lookahead windows. Each value, obtained by averaging the AUROC of different cross-validation folds, is plotted with its standard deviation. In both cases, the Random Forest performance decreases for longer lookahead windows and better AUROC values are observed for the HDD dataset. Fig. 13 suggests that

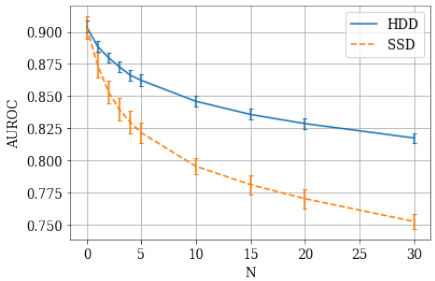
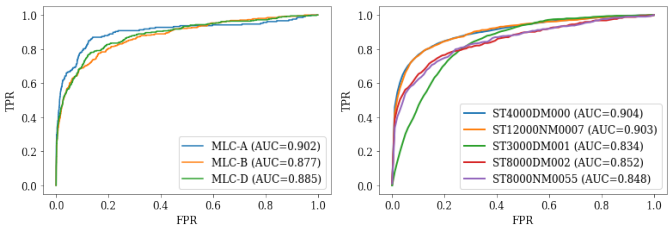


Fig. 13: Random Forest AUROC as a function of lookahead window size, N. Error bars indicate the standard deviation of the cross-validated error across folds.



(a) SSD (b) HDD

Fig. 14: Performance of Random Forest models when they are trained on all drive models and tested on a single one. The lookahead window is set to $N = 0$ days.

Random Forests can efficiently predict SSD and HDD failures for $N \leq 2$ and $N \leq 8$ days lookahead, respectively.

Besides testing the predictor on all drive models, we also evaluate the performance of the Random Forest with test set that consists of a *specific* drive model. Results, for lookahead window set to $N = 0$ days, are shown in Fig. 14. When applied to the SSD dataset, Random Forests can efficiently predict the status of the SSD independently of the drive model. Instead, in the HDD case, the predictor provides good AUROC for *ST4000DM000* and *ST12000NM0007*, acceptable predictions for *ST8000DM002* and *ST8000NM0055*, and mild performance for *ST3000DM001*.

Tables 9 and 10 show the robustness of the Random Forest across HDD and SSD models, respectively. The tables report the AUROC of the predictor when it is trained with a given drive model and it is tested on a different one. While the Random

TABLE 9: Random forest on the HDD dataset for $N = 0$.

Test	Training					
	ST4000DM000	ST12000NM0007	ST3000DM001	ST8000DM002	ST8000NM0055	All
ST4000DM000	0.902	0.839	0.828	0.827	0.788	0.904
ST12000NM0007	0.872	0.901	0.828	0.878	0.848	0.903
ST3000DM001	0.745	0.694	0.863	0.664	0.654	0.834
ST8000DM002	0.809	0.822	0.774	0.844	0.832	0.852
ST8000NM0055	0.793	0.818	0.743	0.831	0.850	0.848

TABLE 10: Random forest on the SSD dataset for $N = 0$.

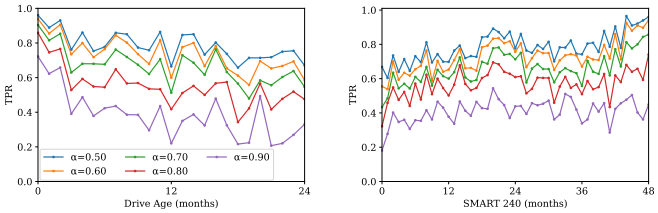
Test	Training			
	MLC-A	MLC-B	MLC-D	All
MLC-A	0.891	0.871	0.887	0.901
MLC-B	0.832	0.892	0.849	0.893
MLC-D	0.868	0.857	0.863	0.901

Forest is very robust when applied to the SSD dataset, it is less accurate on the HDD workload. The worst AUROC is observed for *ST3000DM001*, independently of the HDD model used for training. Good prediction for this HDD model is observed only when the random forest is trained using the same drive model. This is consistent with what is observed in Tables 3 and 5 that show different error and failure rates for *ST3000DM001* comparing to other HDD models. In most cases, training the Random Forest with all available models improves its prediction. Exceptions are observed for ST8000NM0055 and ST3000DM001 HDD models, for which the AUROC decreases if all disk models are used for training the random forest. While for the former disk the difference is negligible (i.e., 0.002), a higher AUROC reduction is observed in the latter case (i.e., almost 0.03). It is worth noting that ST3000DM001 presents more uncorrectable errors and re-enters the workflow more likely than other disk models (see Tables 3 and 7, respectively). These differences between ST3000DM001 and other HDD models may affect the prediction accuracy of the classifier.

5.3 Model Improvement

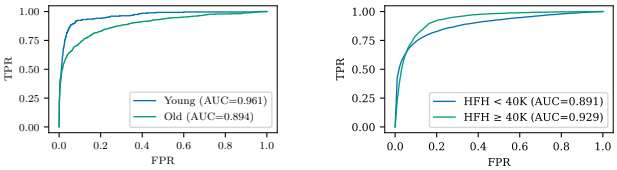
Section 4 shows that many SSD failures are related to the drive age, while HDD ones are affected by the head flying hours (i.e., SMART 240). Here, we use those attributes to improve the performance of the model. Each dataset is split based on the value of the considered feature (i.e., drive age or head flying hours). Then, the model is trained and validated on each sub-dataset and the performance of each new model is compared to the performance obtained without splitting the dataset.

First, we evaluate the TPR of the model against different splitting on drive age (for SSDs) and head flying hours (for HDDs). Since Random Forest models return a prediction probability, a threshold α is set to obtain a binary prediction. A conservative threshold (i.e., $\alpha \simeq 1$) is practical for problems that require a low false positive rate. We test different thresholds between 0.5 (the default one) and 1.0. Results are shown in Fig. 15. Fig. 15(a) shows that, independently of the chosen threshold, the TPR for young SSDs (i.e., younger than 3 months) is substantially greater than the TPR for older drives. Fig. 15(b) depicts the effect of the time (in months) spent for positioning the disk heads (i.e., head flying hours, SMART 240) on the performance of the model. Especially for small values of α, the TPR increases with the time spent by the drive for positioning its heads.



(a) SSD, Drive Age (b) HDD, Head Flying Hours

Fig. 15: True positive rate as a function of drive attributes for SSD and HDD for different prediction thresholds (α). The lookahead window is set to N=0 days.



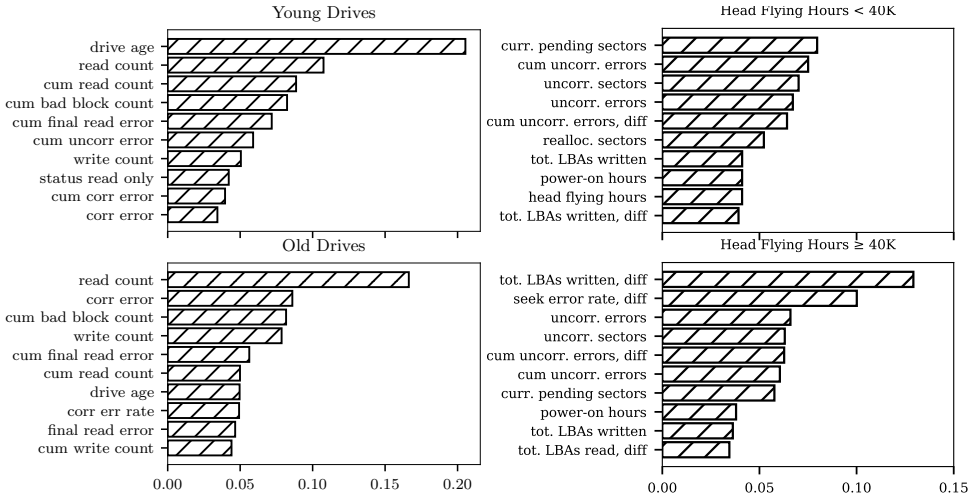
(a) SSD (b) HDD

Fig. 16: ROC curves after splitting the dataset on drive features. Prediction model is random forest with lookahead window $N = 0$ days. In (b), *HFH* is for *Head Flying Hours* (i.e., SMART 240).

Fig. 16 shows the ROC curve of the model when it is trained on different sub-dataset and predicts the state of each drive in the next 24 hours. As depicted in Fig. 16(a), the prediction model works better with young SSDs (i.e., drive age smaller than 3 months) since its AUROC is significantly larger (0.961) than the one shown in Fig. 13 (0.906). This comes at the expense of slightly reduced performance for older drives (0.894). Similarly, Fig. 16(b) shows the performance improvement observed by splitting the HDD dataset on the *head flying hours* feature (i.e., 40,000 hours). The model can better predict the state of HDDs that spend a longer time in positioning their heads (0.929). Also in this case, improvements are observed comparing to the default strategy (i.e., no split) when the measured AUROC is 0.902, while the performance for drives with small head flying hours slightly decreases (0.890). It is worth noting that the 20% of swap-inducing failures in the SSD dataset are young failures, while the 25% of HDDs has large head flying hours.

5.4 Model Interpretability

Besides providing the best performance, Random Forest models also assign a score to each attribute based on its relevance for solving the classification problem. This greatly increases the model interpretability since it is possible to identify the features that are more related to drive failures.



(a) SSD, split on drive age (b) HDD, split on head flying hours

Fig. 17: Feature importance for Random Forest models after splitting the dataset on drive features.

Fig. 17 shows the TOP-10 features for each sub-dataset considered in Section 5.3 (i.e., young and old SSDs, HDDs with short and large head flying hours). Fig. 17(a) shows the feature ranking for the SSD dataset. When considering young drives, the drive age is the most important feature, followed by the read count, its cumulative value, and the cumulative number of bad blocks. For old SSDs, features counting correctable errors and read/write operations, and the cumulative number of bad blocks are in the TOP-4. It is expected that read and write counts are more relevant for the state prediction of old drives since young drives may have only a few activities at the failure time. Fig. 17(b) depicts the feature importance for the HDD dataset. The number of *current pending sectors*, *uncorrectable errors*, *uncorrectable sectors*, and *reallocated sectors* are among the most important features for detecting failing drives. This is similar to what is observed in [29]. The attribute ranking of HDDs with large head flying hours provides new insights. In this case, the two most relevant features are the incremental step of *written logical block addressing* (LBA) and *seek error rate*, followed by the number of *uncorrectable errors* and *uncorrectable sectors*. The *seek error rate* and the *uncorrectable sector* count are observed to be important features also in [26]. The *reallocated sector* count is not in the TOP-10 important features for HDDs with large head flying hours.

5.5 Comparison with Other Models

In this section we illustrate that splitting the SSD and HDD datasets based on drive attributes is superior to state-of-the-art approaches. Specifically, we use the HDD dataset (i.e., Backblaze) to evaluate the failure prediction accuracy of the proposed approach against the regularized greedy forest in [7]. To compare the effect of splitting the SSD based on drive age, we compare prediction of error incidence with the model presented in [29]. Note that [7] and [29] use respectively, the HDD and SSD datasets as in this paper, so a comparison is approprate.

Backblaze dataset: In [7], a regularized greedy forest is used for predicting HDD failures. To deal with the imbalance problem, the whole dataset is undersampled and the remaining observations are split into training and test sets. This biases the classifier because if separate drive IDs are not used to generate the training and testing sets, then we cannot preclude using future observations to predict the past. In addition, [7] creates the testing set by sampling on the actual data which is another source of bias since the testing set does not resemble anymore the real data. This problem for the results presented in [7] has been also reported by [4]. For our comparison, we first use the biased way to define the training and testing as in [7] and use the same hyper-parameters (i.e., leaf nodes, L2 regularization parameters, and informative under-sampling using K-means clustering) to train the regularized greedy forest. For testing, we use the same HDD model (ST4000DM000) from Backblaze and observation period (February 2014 to June 2015). Table 11 presents the AUROC obtained with this biased approach and is in good agreement[1] with the results presented in [7]. The table also presents results with its unbiased version (i.e., by defining the training and testing sets based on drive IDs), and by partitioning the dataset based on head flying hours (i.e., the methodology proposed in this paper). As expected, if under-sampling and cross-validation are not performed correctly (i.e., the first row of Table 11), the regularized greedy forest returns extremely accurate predictions. When training and test sets are generated considering drive IDs and only the training set is under-sampled (i.e., the second row of Table 11), results are much closer to those obtained with our approach. The AUROC slightly decreases when a random forest is evaluated using HDDs with short head flying hours (i.e., less than 40K), but it is similar to the one obtained with the regularized greedy forest for small lookahead windows (i.e., less than or equal to 2 days). This is shown in the third row of Table 11. The AUROC improves when the classifier is trained and evaluated only on disks whose head flying hours is longer than 40K (i.e., the fourth row of Table 11). In this case, the random forest provides accurate failure prediction up to 7 days.

Google dataset: Although in this paper we predict SDD failures, our methodology is extensible to estimate error incidence as done in [29]. For example, when predicting uncorrectable errors, we achieve AUROC scores of 0.960 ± 0.009 and 0.931 ± 0.007

1. For comparison purposes, we use the AUROC metric instead of the precision and recall used in [7] but it essentially presents the same data in a different format.

TABLE 11: AUROC obtained using the approach presented in this paper and the one discussed in [7]. Both classifiers are trained and tested on logs from ST4000DM000 disks in the Backblaze dataset collected from February 2014 to June 2015.

N (lookahead days)	0	1	2	7
Regularized Greedy Forest with Informative Under-sampling [7] (biased)	0.980 ± 0.004	0.980 ± 0.005	0.982 ± 0.007	0.984 ± 0.004
Regularized Greedy Forest with Informative Under-sampling	0.911 ± 0.011	0.878 ± 0.019	0.866 ± 0.015	0.824 ± 0.025
Random Forest with Random Under-sampling (HFH $<$ 40K)	0.891 ± 0.065	0.873 ± 0.040	0.866 ± 0.031	0.787 ± 0.025
Random Forest with Random Under-sampling (HFH $\geq$ 40K)	0.947 ± 0.017	0.936 ± 0.006	0.926 ± 0.016	0.908 ± 0.012

if the dataset used to evaluate the random forest is partitioned into young (i.e., less than 3 months) and old SSDs, respectively. This leads to improved performance for young drives and stable accuracy for old SSDs when AUROC is compared to the one obtained with unpartitioned data (i.e., 0.933 ± 0.006). A similar pattern is observed also for other non-transparent errors as shown in Table 12.

TABLE 12: AUROC obtained with the random forest when predicting various error types of SSDs and $N = 1$. Response errors are too rare to predict for different age granularities.

Error	Combined [29]	Young	Old
Bad block	0.877	0.878	0.873
Erase	0.889	0.934	0.882
Final read	0.906	0.959	0.852
Final write	0.841	0.937	0.780
Meta	0.854	0.890	0.842
Read	0.971	0.917	0.973
Response	0.806	—	—
Timeout	0.755	0.812	0.735
Uncorrectable	0.933	0.960	0.931
Write	0.916	0.911	0.914

6 Related Work

Large-scale data centers serve millions of requests every day [46, 14, 45] and their dependability is critical for services hosted on their infrastructure. Much prior work investigates the main components that affect the data center dependability [17, 28, 42, 47, 34, 33] and storage systems are pointed out as extremely important components to determine the data center reliability [40, 43, 23, 50].

Storage drives, such as HDDs and SSDs, include a monitor system (originally designed by IBM and called SMART) that allows logging data about the drive reliability [3]. Disk failures are investigated by Schroeder and Gibson in [40], while Pinheiro et al. [36] explore their relationship with SMART features. Meza et al. [31] conduct a large-scale reliability study of SSDs mainly focusing on hardware events. Maneas et al. [30] explore reasons for breakdown replacement and correlations among drives in the same RAID group. Jaffer et al. [24] evaluate the ability of different file systems to get through SSD faults. Hao et al. [21] study HDDs and SSDs to highlight the importance of masking storage tail latencies to increase performance stability. Han et al. [20] analyze SSD traces from Alibaba and investigate correlated failures from spatial and temporal perspectives. None of these works attempts to predict the failure state of drives.

Different approaches are proposed to predict drives failures and errors. Main ones are reported in Table 13 with their primary characteristics and discussed in more details in the following. Hamerly et al. [18] and Hughes et al. [22] use statistical inference (i.e., Bayesian approach and Rank sum hypothesis test, respectively), while Ma et al. [28] rely on thresholds to predict component failures. Goldszmidt [15] combines statistical models and thresholds to predict disks that are soon to fail. However, machine learning prediction models are shown to be more efficient than other strategies. Murray et al. [32] develop a multi-instance naive Bayes classifier to reduce the number of false alarms when predicting disk failures, while Tan and Gu [44] investigates the performance of a tree augmented naive Bayesian method to predict the future drive status. Agarwal et al. [1] investigate the performance of a rule-based classifier for discovering disk failures, Li et al. [25] address the same problem by using decision trees, and Zhu et al. [55] adopt neural networks and SVM. Lu et al. [26] implement a convolutional neural network with long short-term memory that predicts HDD failures with a 10-day lookahead window by considering SMART features, disk performance metrics, and disk physical location. Queiroz et al. [37] propose a new method (i.e., Gaussian Mixture based Fault Detection) to build a statistical model of SMART attributes and predict imminent HDD failures.

To fairly compare the performance of our method with other approaches, the same dataset must be used. As discussed by Yu [53], the performance may be extremely different when considering different data sources. The Backblaze dataset is used in [7, 4] to train and validate their approaches for predicting disk failures while Mahdisoltani et al. [29] use the Backblaze and the Google datasets for investigating faulty HDDs and SSDs. The regularized greedy forest adopted by Botezatu et al. [7] shines when compared to other approaches used to predict failures in the Backblaze dataset with 98% precision and accuracy but as discussed in Section 5.5 this is due to the incorrect formation of the training and test sets. Moreover, they also under-sample both training and test sets as also noted by [4]. Similarly, also Wang et al. [48] test a deep architecture for predicting HDD failures on an under-sampled dataset. Typically, under- or over-sampling the whole dataset (training and test sets) generates overoptimistic results [39]. Aussel et al. [4] train and evaluate a random forest on a small subset of the Backblaze dataset (i.e., only data from 2014). Although they show high precision and a considerable recall (i.e., 95% and 65%, respectively), they filter out observations with similar features and different failure states. We believe this is hardly doable in real scenarios since it requires a priori knowledge of the drive state. Mahdisoltani et al. [29] investigates different prediction models to predict uncorrectable errors and bad blocks in HDDs and SSDs, and they show that random forests provide good

TABLE 13: Related work that uses ML to predict storage device failures. For each paper, the adopted approach, its goal, datasets, used model, the resampling strategy, and the cross-validation technique are reported.

Paper	Goal	Dataset(s), Drive Type, Availability	ML Model(s)	Random Sampling	Cross-Validation
[1, 32, 37, 44]	Failure Pred.	Quantum, HDD, N/A	Various	–	Random
[26, 55]	Failure Pred.	Baidu, HDD, Public	Various	–	Random
[25]	Failure Pred.	N/A, HDD, Private	Decision Tree	Change Class Weight of Training Set	Random
[7]	Failure Pred.	Backblaze, HDD, Public	Regularized Greedy Forest	Under-sample the whole dataset	Random
[4]	Failure Pred.	Backblaze, HDD, Public	Random Forest	SMOTE on Training Set, No "Hard-to-Classify" Samples	Drive-by-drive
[29]	Error Pred.	Backblaze and Google, HDD and SSD, Public and Private	CART, Neural Network, Linear Regression, SVM, Random Forest	Under-sample the Training Set	Random
[49]	Online Failure Pred.	Backblaze, HDD, Private	Online Random Forest	–	Time-based
[48]	Failure Pred.	Backblaze, HDD, Public	Attetion-augmented Deep Architecture	Under-sample the whole dataset	Drive-by-drive
[9]	Failure Pred.	Google, SSD, Private	One-class ML	–	Unknown
[19]	Online Failure Pred.	Backblaze and Alibaba, HDD, Public and Private	Stream Mining	Under-sample the whole dataset	Time-based

performance for this task. Although our approach is similar to the one presented in [29], we aim to predict drive failures. A recent work [9] aims to predict SSD failures by using 1-class ML models that are trained only on the majority class (i.e., healthy drives). This approach allows obtaining high prediction accuracy even if the occurring error has never been observed. Although the authors claim that their approach provide better AUROC results than [2], close inspection illustrates that the look-ahead prediction in [2] is superior to that reported in [9]: compare Figure 8 in [9] with Figure 12 in [2] and Figure 13 in this paper. Approaches for online prediction of HDD errors and failures are proposed in [49, 19, 52] by using online random forests, concept drift adaptation [13], and tensor decomposition [35]. Unfortunately, the SSD dataset does not provide a global timestamp, hence online prediction cannot be implemented here.

In this paper, we explore the capability of random forests for predicting drive failures and investigate possible enhancements by statistically analyzing drive features and by using different models based on observed attributes. We consider a conceivably long lookahead window and use two large and real datasets for training and validating the proposed machine learning approach. To the best of our knowledge, storage device failures have never been studied by splitting the dataset based on attribute values.

7 Conclusion

In this paper, we investigate SSD and HDD failures using two traces, each one six-year long, from production environments. Daily logs for 30,000 SSDs are collected at a Google data center, while 100,000 HDDs are observed at a Backblaze data center. Analyzing the available traces, we draw remarkable conclusions. Unexpectedly, we observe that features that are commonly thought to cause SSD failures (i.e., write operations and error incidence) are not highly related to faulty SSDs. We train and test different classifiers to predict faulty SSDs and HDDs, and note that Random Forest models provide accurate predictions with a short training time. Their high interpretability makes them the best predictor for the analyzed problem. We observe that splitting each dataset based on attribute values of its observations allows increasing the performance of random forests. The drive age is a critical attribute for predicting SSD failures; drives failing before being three months old can be detected easier than other drives. Similarly, when predicting faulty HDDs, a higer detection rate is observed for drives with *head flying hours* (i.e., SMART 240) longer than 40,000 hours.

We are aware that new technologies for SSDs (e.g., flash design and interface) and HDDs (e.g., internal track design) have been developed in recent years. To the best of our knowledge, dataset containing logs of new generation storage devices are not publicly available yet. In the future, we aim to apply the methodologies presented in this paper to more recent devices and evaluate their performance.

References

[1] Vipul Agarwal et al. "Discovering rules from disk events for predicting hard drive failures". In: *2009 International Conference on Machine Learning and Applications (ICMLA)*. IEEE. 2009, pp. 782–786.

[2] Jacob Alter et al. "SSD failures in the field: symptoms, causes, and prediction models". In: *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. 2019, pp. 1–14.

[3] *American National Standards Institute. AT attachment 8 - ATA/ATAPI command set (ATA8-ACS), 2008.* http://www.t13.org/documents/uploadeddocuments/docs2008/d1699r6a-ata8-acs.pdf. [Online; accessed on 2020-04-19].

[4] Nicolas Aussel et al. "Predictive models of hard drive failures based on operational data". In: *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE. 2017, pp. 619–625.

[5] Backblaze. *Hard Drive Data and Stats*. https://www.backblaze.com/b2/hard-drive-test-data.html. [Online; accessed on 2020-04-28].

[6] Robert Birke et al. "(Big)data in a virtualized world: volume, velocity, and variety in cloud datacenters". In: *Proceedings of the 12th USENIX Conference on File and Storage Technologies, (FAST 14)*. 2014, pp. 177–189.

[7] Mirela Madalina Botezatu et al. "Predicting disk replacement towards reliable data centers". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016, pp. 39–48.

[8] Yu Cai et al. "Data retention in MLC NAND flash memory: Characterization, optimization, and recovery". In: *21st IEEE International Symposium on High Performance Computer Architecture, HPCA 2015*. 2015, pp. 551–563.

[9] Chandranil Chakraborttii and Heiner Litz. "Improving the accuracy, adaptability, and interpretability of SSD failure prediction models". In: *Proceedings of the 11th ACM Symposium on Cloud Computing*. 2020, pp. 120–133.

[10] G.W. Corder and D.I. Foreman. *Nonparametric Statistics: A Step-by-Step Approach*. Wiley, 2014.

[11] *CSI: Backblaze—Dissecting 3TB Drive Failure, 2015*. https://www.backblaze.com/blog/3tb-hard-drive-failure/. [Online; accessed on 2021-05-20].

[12] Tom Fawcett. "An introduction to ROC analysis". In: *Pattern recognition letters* 27.8 (2006), pp. 861–874.

[13] João Gama et al. "A survey on concept drift adaptation". In: *ACM computing surveys (CSUR)* 46.4 (2014), pp. 1–37.

[14] Phillipa Gill, Navendu Jain, and Nachiappan Nagappan. "Understanding network failures in data centers: measurement, analysis, and implications". In: *ACM SIGCOMM Computer Communication Review*. Vol. 41. 4. ACM. 2011, pp. 350–361.

[15] Moisés Goldszmidt. "Finding Soon-to-Fail Disks in a Haystack". In: *4th USENIX Workshop on Hot Topics in Storage and File Systems, HotStorage'12*. 2012.

[16] Laura M. Grupp et al. "Characterizing flash memory: anomalies, observations, and applications". In: *42st Annual IEEE/ACM International Symposium on Microarchitecture (MICRO-42 2009)*. 2009, pp. 24–33.

[17] Chuanxiong Guo et al. "Pingmesh: A large-scale system for data center network latency measurement and analysis". In: *ACM SIGCOMM Computer Communication Review*. Vol. 45. 4. ACM. 2015, pp. 139–152.

[18] Greg Hamerly, Charles Elkan, et al. "Bayesian approaches to failure prediction for disk drives". In: *ICML*. Vol. 1. 2001, pp. 202–209.

[19] Shujie Han et al. "Toward Adaptive Disk Failure Prediction via Stream Mining". In: *2019 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. IEEE. 2020.

[20] Shujie Han et al. "An In-Depth Study of Correlated Failures in Production SSD-Based Data Centers". In: *19th USENIX Conference on File and Storage Technologies (FAST 21)*. 2021, pp. 417–429.

[21] Mingzhe Hao et al. "The Tail at Store: A Revelation from Millions of Hours of Disk and SSD Deployments". In: *14th USENIX Conference on File and Storage Technologies (FAST 16)*. 2016, pp. 263–276.

[22] Gordon F Hughes et al. "Improved disk-drive failure warnings". In: *IEEE transactions on reliability* 51.3 (2002), pp. 350–357.

[23] Andy A Hwang, Ioan A Stefanovici, and Bianca Schroeder. "Cosmic rays don't strike twice: understanding the nature of DRAM errors and the implications for system design". In: *ACM SIGPLAN Notices*. Vol. 47. 4. ACM. 2012, pp. 111–122.

[24] Shehbaz Jaffer et al. "Evaluating file system reliability on solid state drives". In: *2019 USENIX Annual Technical Conference (USENIX ATC 19)*. 2019, pp. 783–798.

[25] Jing Li et al. "Hard drive failure prediction using classification and regression trees". In: *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*. IEEE. 2014, pp. 383–394.

[26] Sidi Lu et al. "Making Disk Failure Predictions SMARTer!" In: *18th USENIX Conference on File and Storage Technologies (FAST 20)*. 2020, pp. 151–167.

[27] Yixin Luo et al. "Improving 3D NAND Flash Memory Lifetime by Tolerating Early Retention Loss and Process Variation". In: *Abstracts of the 2018 ACM International Conference on Measurement and Modeling of Computer Systems, SIGMETRICS 2018*. 2018, p. 106.

[28] Ao Ma et al. "RAIDShield: characterizing, monitoring, and proactively protecting against disk failures". In: *ACM Transactions on Storage (TOS)* 11.4 (2015), pp. 1–28.

[29] Farzaneh Mahdisoltani, Ioan Stefanovici, and Bianca Schroeder. "Proactive error prediction to improve storage system reliability". In: *2017 USENIX Annual Technical Conference (USENIX ATC 17)*. 2017, pp. 391–402.

[30] Stathis Maneas et al. "A Study of SSD Reliability in Large Scale Enterprise Storage Deployments". In: *18th USENIX Conference on File and Storage Technologies (FAST 20)*. 2020, pp. 137–149.

[31] Justin Meza et al. "A Large-Scale Study of Flash Memory Failures in the Field". In: *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*. 2015, pp. 177–190.

[32] Joseph F Murray, Gordon F Hughes, and Kenneth Kreutz-Delgado. "Machine learning methods for predicting failures in hard drives: A multiple-instance application". In: *Journal of Machine Learning Research* 6 (2005), pp. 783–816.

[33] Bin Nie et al. "Characterizing Temperature, Power, and Soft-Error Behaviors in Data Center Systems: Insights, Challenges, and Opportunities". In: *25th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, MASCOTS 2017, Banff, AB, Canada, September 20-22, 2017*. IEEE Computer Society, 2017, pp. 22–31.

[34] Bin Nie et al. "Machine Learning Models for GPU Error Prediction in a Large Scale HPC System". In: *Dependable Systems and Networks (DSN), 2018 48th Annual IEEE/IFIP International Conference on*. IEEE. 2018, pp. 1–12.

[35] Evangelos E Papalexakis, Christos Faloutsos, and Nicholas D Sidiropoulos. "Tensors for data mining and data fusion: Models, applications, and scalable algorithms". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 8.2 (2016), pp. 1–44.

[36] Eduardo Pinheiro, Wolf-Dietrich Weber, and Luiz André Barroso. "Failure trends in a large disk drive population". In: *5th USENIX Conference on File and Storage Technologies, (FAST 07)*. 2007, pp. 17–28.

[37] Lucas P Queiroz et al. "A fault detection method for hard disk drives based on mixture of Gaussians and non-parametric statistics". In: *IEEE Transactions on Industrial Informatics* 13.2 (2017), pp. 542–550.

[38] Moinuddin K. Qureshi et al. "AVATAR: A Variable-Retention-Time (VRT) Aware Refresh for DRAM Systems". In: *45th Annual IEEE/IFIP International Conference*

on Dependable Systems and Networks, DSN 2015. 2015, pp. 427–437.

[39] Miriam Seoane Santos et al. "Cross-validation for imbalanced datasets: Avoiding overoptimistic and overfitting approaches [research frontier]". In: *IEEE Computational Intelligence Magazine* 13.4 (2018), pp. 59–76.

[40] Bianca Schroeder and Garth A Gibson. "Disk failures in the real world: What does an MTTF of 1,000,000 hours mean to you?" In: *5th USENIX Conference on File and Storage Technologies, (FAST 07)*. 2007, pp. 1–16.

[41] Bianca Schroeder, Raghav Lagisetty, and Arif Merchant. "Flash Reliability in Production: the Expected and the Unexpected". In: *14th USENIX Conference on File and Storage Technologies, (FAST 16)*. 2016, pp. 67–80.

[42] Bianca Schroeder, Arif Merchant, and Raghav Lagisetty. "Reliability of NAND-based SSDs: What field studies tell us". In: *Proceedings of the IEEE* 105.9 (2017), pp. 1751–1769.

[43] Bianca Schroeder, Eduardo Pinheiro, and Wolf-Dietrich Weber. "DRAM errors in the wild: a large-scale field study". In: *ACM SIGMETRICS Performance Evaluation Review*. Vol. 37. 1. ACM. 2009, pp. 193–204.

[44] Yongmin Tan and Xiaohui Gu. "On predictability of system anomalies in real world". In: *2010 IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*. IEEE. 2010, pp. 133–140.

[45] Abhishek Verma et al. "Large-scale cluster management at Google with Borg". In: *Proceedings of the Tenth European Conference on Computer Systems, EuroSys 2015*. ACM. 2015, 18:1–18:17.

[46] Guohui Wang and TS Eugene Ng. "The impact of virtualization on network performance of Amazon EC2 data center". In: *Proceedings of Infocom 2010*. IEEE. 2010, pp. 1–9.

[47] Guosai Wang, Lifei Zhang, and Wei Xu. "What Can We Learn from Four Years of Data Center Hardware Failures?" In: *Dependable Systems and Networks (DSN), 2017 47th Annual IEEE/IFIP International Conference on*. IEEE. 2017, pp. 25–36.

[48] Ji Wang et al. "An Attention-augmented Deep Architecture for Hard Drive Status Monitoring in Large-scale Storage Systems". In: *ACM Transactions on Storage (TOS)* 15.3 (2019), pp. 1–26.

[49] Jiang Xiao et al. "Disk failure prediction in data centers via online learning". In: *Proceedings of the 47th International Conference on Parallel Processing*. 2018, pp. 1–10.

[50] Erci Xu et al. "Lessons and actions: What we learned from 10k ssd-related storage system failures". In: *2019 USENIX Annual Technical Conference (USENIX ATC 19)*. 2019, pp. 961–976.

[51] Yong Xu et al. "Improving Service Availability of Cloud Systems by Predicting Disk Error". In: *2018 USENIX Annual Technical Conference, (USENIX ATC 18)*. 2018, pp. 481–494.

[52] Amirhessam Yazdi et al. "SEFEE: lightweight storage error forecasting in large-scale enterprise storage systems". In: *2020 SC20: International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*. IEEE Computer Society. 2020, pp. 894–907.

[53] Jie Yu. "Hard disk Drive Failure Prediction Challenges in Machine Learning for Multi-variate Time Series". In: *Proceedings of the 2019 3rd International Conference on Advances in Image Processing*. 2019, pp. 144–148.

[54] Mai Zheng et al. "Understanding the robustness of SSDs under power fault". In: *Proceedings of the 11th USENIX Conference on File and Storage Technologies, (FAST 13)*. 2013, pp. 271–284.

[55] Bingpeng Zhu et al. "Proactive drive failure prediction for large scale storage systems". In: *2013 IEEE 29th symposium on mass storage systems and technologies (MSST)*. IEEE. 2013, pp. 1–5.

Riccardo Pinciroli received M.S. and Ph.D. degrees in computer engineering from Politecnico di Milano, in 2014 and 2018, respectively. He is currently a Postdoc Fellow in Computer Science at the Gran Sasso Science Institute. His research interests include stochastic modeling, performance evaluation, energy efficiency, and uncertainty propagation applied to cloud computing, data-centers, and cyber-physical systems.

Lishan Yang is a Ph.D. candidate in Computer Science at William & Mary. She received her B.S. degree in Computer Science from University of Science and Technology of China (USTC) in 2016. Her research interests are in GPU architecture, reliability analysis, and performance analysis. She is a member of IEEE and ACM.

Jacob Alter received an M.S. in Computer Science from William & Mary in 2019. He is currently a Data Scientist at Ipsos North America in Washington, DC. His research interests include storage systems, reliability analysis, and the application of data mining and machine learning to production computing systems.

Evgenia Smirni is the Sidney P. Chockley professor of computer science at William and Mary, Williamsburg, VA. Her research interests include queuing networks, stochastic modeling, resource allocation, storage systems, cloud computing, workload characterization, and modeling of distributed systems and applications. She is an ACM Distinguished scientist and an IEEE Fellow.