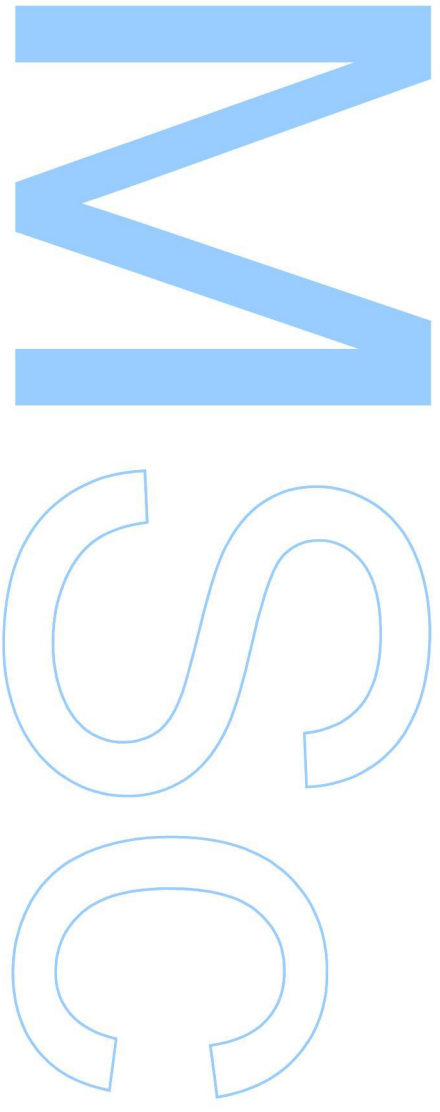
Predicting Hard Disk Drive Failures and Misbehavior

Henish Hemendra Balu
Mestrado Integrado em Engenharia de Redes e Sistemas Informáticos
Departamento de Ciência dos Computadores
2020

Orientador
Inês de Castro Dutra
Professora Auxiliar
Faculdade de Ciências da Universidade do Porto

Coorientador
Christopher David Harrison
Doutoramento em Ciência de Computadores
Faculdade de Ciências da Universidade do Porto



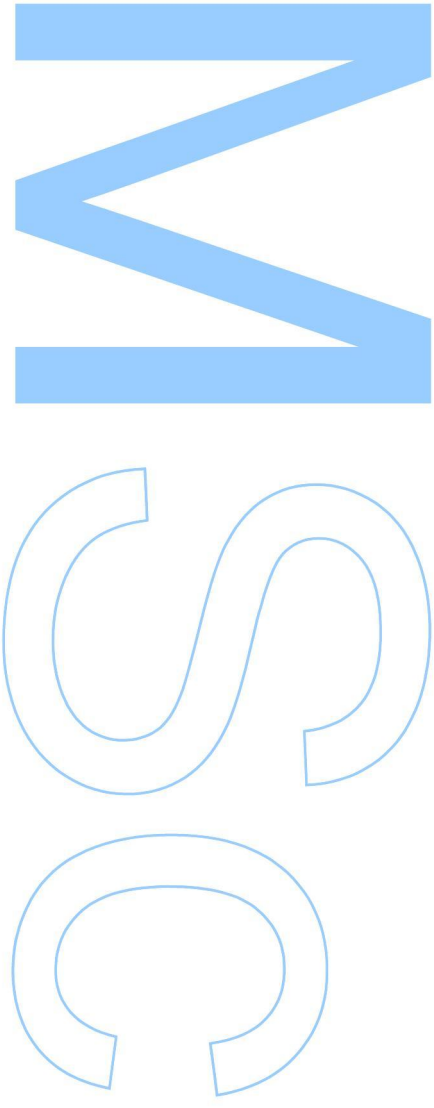
U.PORTO

FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

Todas as correções determinadas pelo júri, e só essas, foram efetuadas.

O Presidente do Júri,

Porto, _____/_____/_____



Abstract

Cloud Storage is a popular service used by large entities or by single users at a personal level. The responsibility of storage companies in maintaining external data stored on their servers has been growing in the last years. Any type of disk failure can result in large monetary costs for companies, even if the number of failed disks is small. In order to mitigate this problem, companies rely on monitoring variables provided by vendors. These are the S.M.A.R.T. (Self-Monitoring, Analysis and Report Technology) attributes, that provide information about various aspects of hard disks. However, even monitoring these variables, disks still fail unexpectedly.

This thesis' objective is based in the analysis and learning of hard disk behaviour in an attempt to uncover factors and relations among the S.M.A.R.T. attributes that may be worth investigating before a disk fails. By predicting anomalous behaviors and hard disk failures, the companies can be more proactive in taking preventive measures and improving the monitoring of their storage devices.

The data used in this work accounts for 90 days of hard disks observations made available by Backblaze. An thorough analysis of the disks is made, preparing the data for the machine learning methods used. Because the ratio of healthy disks over failed disks is very high, an undersampling method is applied to the majority class. As learning models, Random Forest and Support Vector Machine are used. Relations among the S.M.A.R.T. attributes, which expose relationship patterns of healthy and failed disks are extracted from the Random Forests. We also apply a Vector Autoregression (VAR) method in order to find multiple temporal correlations among the attributes and perform forecasting of disk attributes values.

Results indicate that S.M.A.R.T. variables 9, 193 and 240 are correlated over time. Also, variables 7, 9 and 240 are present quite often in the results for failed disks, which does not happen so often for the healthy disks.

The obtained results are promising and the methodology and statistical analysis carried out may prove to be useful for new projects in this area. For forecasting, our time series is not sufficiently large. Nevertheless, for some disks, predictions for a small period of time ahead have a low error. We expect that companies may investigate the methods used in this work in order to provide even better service to users.

Resumo

Cloud Storage é um serviço popular cada vez mais utilizado por grandes entidades ou até por utilizadores a nível pessoal. A responsabilidade das empresas de armazenamento em manter dados externos nos seus servidores, tem vindo a crescer nos últimos anos. Qualquer tipo de falha nos discos pode advir grandes custos monetários às empresas mesmo que o número de falhas seja pequeno. De forma a reduzir este problema, as empresas utilizam variáveis de monitorização disponibilizadas pelas diferentes marcas de discos. Estas variáveis são as S.M.A.R.T. (Self-Monitoring, Analysis and Report Technology) attributes, que fornecem informações sobre as diferentes características do disco. Contudo, mesmo com uma monitorização cuidada destas variáveis, alguns discos falham inesperadamente.

O objetivo desta tese baseia-se na análise e na aprendizagem do comportamento dos discos rígidos, com o intuito de descobrir fatores e relações das S.M.A.R.T. attributes que possam vir a ser investigados antes da falha do disco. Prevendo estes comportamentos anómalos e falhas em discos rígidos, as empresas podem tentar ser mais proativas a tomar medidas preventivas e a monitorizar os seus dispositivos de armazenamento.

Os dados usados neste projeto são de observações de discos rígidos, adquiridos durante 90 dias, disponibilizados pela empresa Backblaze. Uma análise por todos os discos é feita, para preparar os dados a serem processados pelos modelos de learning. Como o ratio de discos saudáveis sobre os discos que falham é muito elevado, um método de undersampling é aplicado à classe maioritária. Os modelos de learning utilizados são o Random Forest e o Support Vector Machine. Através da Random Forest são extraídas algumas relações, de acordo com as S.M.A.R.T attributes, que mostram padrões para dos discos saudáveis e para os discos que falham. É também aplicado um método de Vector Autoregression de maneira a encontrar múltiplas correlações temporais nas variáveis e executar uma previsão dos valores dos atributos.

Os resultados indicam que as variáveis S.M.A.R.T. 9, 193 e 240 se correlacionam ao longo do tempo. Além disso, as variáveis 7, 9 e 240 estão presentes com maior frequência nos resultados dos discos com falha, não se verificando o mesmo para discos considerados saudáveis.

Os resultados obtidos são promissores, sendo que a metodologia e a análise estatística feita podem vir a ser úteis para novos projetos nesta área. Para a previsão, a nossa série temporal não é suficientemente grande, contudo para alguns modelos de discos, a previsão tem uma percentagem de erro bastante baixa. Espera-se que as empresas possam investigar mais detalhadamente os

métodos utilizados neste trabalho de maneira a providenciar melhores serviços aos utilizadores.

Acknowledgement

I would like to thank Professor Inês Dutra for all the help and availability provided throughout the thesis, for all the hours of discussion and analysis we had, which made me learn a lot and increase my interest in the area. Thank you so much for all your patience and for cheering me up whenever you could.

A big thanks also to my co-advisor Christopher, for all the help and for the opinions more directed to the area, that helped me a lot in the decision making.

A huge thank you to my parents, for all the support they gave me not only in the thesis, but also throughout the years. Thank you for always being there for me and for giving so much of your life so I would never miss anything. What I am today, I owe to you. I hope I made you proud!

To my grandparents and my sister for always being there for me, and for giving me all the affection I needed to overcome the most difficult moments during these years!

To my friends, for all we lived during these years and for never leaving me alone at any moment.

Last but not least, a big thank to Faculdade de Ciências da Universidade do Porto for receiving me during these years, and CC&Redes for all the memories and moments we lived!

Dedicated to my Family and Friends

Contents

List of Tables

List of Figures

Acronyms

CRISP-DM Cross Industry Standard Process for Data Mining

CSV Comma Separated Values

DCC Departamento de Ciência de Computadores

ECC Error Correction Codes

FAR False Alarm Rate

FCUP Faculdade de Ciências da Universidade do Porto

FDR Failure Detection Rate

HDD Hard Disk Drive

IBM International Business Machines Corporation

KDD Knowledge Discovery in Databases

MB Mega Bytes

PATA Parallel Advanced Technology Attachment

RAID Redundant Array of Inexpensive Drives

SATA Serial Advanced Technology Attachment

SAS Serial Attached SCSI

S.M.A.R.T. Self-Monitoring, Analysis and Reporting Technology

SVM Support Vector Machine

USB Universal Serial Bus

VAR Vector Auto Regression

Chapter 1

Introduction

Hard drives are an essential component in today's data storage systems. Ten years ago a terabyte was considered a large amount of memory, but nowadays some applications manage to generate huge amounts of information per day. One example is Facebook, that can generate more than 500 terabytes daily [16].

The boom of cloud computing, online services and big data applications have resulted in a huge expansion of storage systems. More than 90% of the information produced annually in our world, is stored in magnetic devices [16]. Datacenters have the responsibility to ensure quality of service for their customers, who depend highly on their storage systems.

Although a hard drive failure event is relatively rare, when we consider cloud scale service providers, rare events occur frequently enough to be the norm and cause issues with large scale infrastructures. Hard drives are reported to be the components that most need to be replaced in storage systems. HDD failures cause service delays and sometimes data loss, costing big companies millions of euros per year. According to a research [17], the average cost of data center down time is 9000$ per minute.

S.M.A.R.T. (self monitoring, analysis and reporting technology) was for a long time, the mechanism used to evaluate the health status of hard drives. Once it was detected that the values were above the threshold, the system administrator would be informed. Unfortunately, the failure detection rate of only using S.M.A.R.T. variables is only between 3%-10% [30].

1.1 Objective

Given the poor performance of monitoring the S.M.A.R.T. attributes and the fact that many vendors choose specific attributes to monitor their HDD's, the main objectives of this work are: **(1)** to improve the performance in detecting possible failures and misbehaviour in hard disks, and **(2)** study the impact of other, less studied, S.M.A.R.T. attributes on healthy and failed hard disks.

1.2 Contribution

By examining daily observations of S.M.A.R.T. attributes from thousands of hard disk models, provided by Backblaze, patterns were successfully identified for specific attributes along time that could distinguish healthy from misbehaving disks. Were also found correlations for sets of S.M.A.R.T. attributes that are usually not associated with disk failure. With these correlations, companies can have a more precise monitoring of the disks, following the progress of several variables over time, instead of just a single critical variable. Moreover, we studied the applicability of a vector autoregressive model to perform a forecasting of the attributes values with the intention of emitting alerts to companies about the future behaviour of their disks.

1.3 Structure

This thesis is organized in 4 more chapters that are described below:

Background Provides some concepts on the most important topics related to this project, such as some basic concepts on hard disks functioning, S.M.A.R.T. attributes, Data Mining techniques, machine learning models, imbalanced domains, time series concepts and metrics to evaluate performance. The chapter ends with a literature review about learning models used to prevent disk failures.

Failure and Misbehavior Prediction In this Chapter the methodology used in this work is described to help finding ways to predict disk failures. This chapter also describes the dataset used, pre-processing methods and how the learning models were applied.

Results and Analysis Presents the results and main findings, as well a discussion about what was achieved.

Conclusion Presents a summary of the motivations and main contributions given by this work. And a summary about what could be done in future works.

Chapter 2

Background

In this chapter, the necessary concepts and methods to achieve our objectives are reviewed. It starts with a small overview on HDD's concepts and S.M.A.R.T. attributes. Next, some Data Mining techniques, followed by exploring some of the machine learning algorithms and time series concepts.

2.1 Hard Disk Drives

The hard disk industry has more than half a century of existence, and the first HDD was implemented in 1956 by the IBM (International Business Machines Corporation) company with only 5 MB (Mega Bytes) of capacity. Nowadays, anyone has access to a HDD on their computer, but a few decades ago, a HDD was a gigantic device that occupied an entire floor. The constant need to increase data storage digitally, made this area develop a lot in the recent years. The ability to meet this demand at a relatively low cost, makes the HDD the undisputed candidate for online storage [1].

A HDD is a magnetic data storage device that uses one or more rotating disks (platters) coated with magnetic material. The components of a HDD can be classified in 4 categories - magnetic, mechanical, electromechanical and electronic. The disks are paired with magnetic heads, that can read and write information on the surfaces of the disk. Data is written and read from HDD in chunks of data or data blocks and each block is mapped to a specific addressable place on the HDD. These blocks are the smallest unit of storage on any given HDD. HDD's can be connected to systems via PATA (Parallel Advanced Technology Attachment), SATA (Serial Advanced Technology Attachment), USB (Universal Serial Bus) or SAS (Serial Attached SCSI) cables [1]. In Figure 2.1 it is possible to verify some HDD essential components.

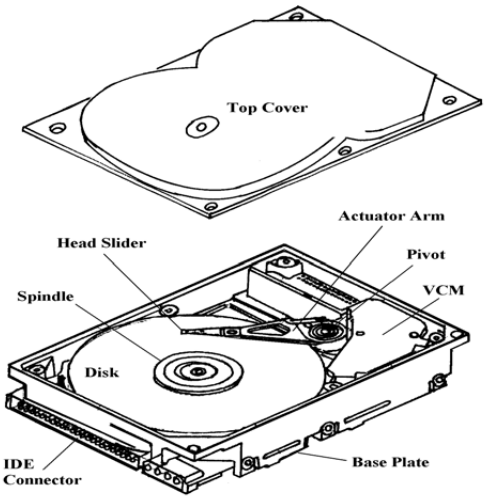


Figure 2.1: HDD components. [1]

2.1.1 Disk Failures

Schroeder *et al.* [28] mention a very important point, when saying that costumers and vendors might use different definitions of what is a faulty HDD. For a costumer, a disk misbehavior may consist in a reading operation that takes longer to execute than usual. For vendors, that value may not be an alarm to be considered because the threshold is not being passed. A disk manufacturer once published that 43% of all disks returned by costumers, because they supposedly found that the disks had some problem, were considered healthy by the vendors. While drive manufacturers often quote yearly failure rates below 2%, user studies have seen rates as high as 6% [27].

Failures can be categorized in two major groups, predictable and unpredictable [24]. Unpredictable failures, such as electronic and some mechanical problems, occur quickly without any chance of control from the user. For example, a power surge may cause chip or circuit damages on the hard disk. On the other hand, predictable failures are characterized by degradation of an attribute over time. Therefore, attributes can be monitored, making it possible for predictive-failure analysis. Many mechanical components suffer some kind of degradation over their life-time, this can indicate a potential problem in the future, so it is possible to the user to control the HDD components more carefully before they fail [24].

The study made by Schroeder *et al.* [28] also refers that even if the HDD is from the same model, they can differ on their behavior, because disks are manufactured using processes and parts that may change. A simple change in a drive's firmware or in a hardware component, or even in the assembly line on which a drive was manufactured, can change the failure behavior of a disk.

According to the Backblaze Company, a disk is considered failed when [5]:

"it is removed from a Storage Pod and replaced because it has 1) totally stopped working, or 2)

because it has shown evidence of failing soon. A drive is considered to have stopped working when the drive appears physically dead (e.g. won't power up), doesn't respond to console commands or the RAID system tells us that the drive can't be read or written."

2.1.2 S.M.A.R.T. Attributes

S.M.A.R.T. emerged from the need to protect critical information stored on disk drives. As system storage capacity requirements increased, the industry identified the importance of creating an early warning system that would allow enough lead time to back up data if failure was imminent, preventing catastrophic data loss [24].

S.M.A.R.T. includes a series of attributes, chosen specifically for each drive model. This individualism is important because HDD architectures vary from model to model. Attributes and thresholds that detect failure for one model may not be functional for another model. The same occurs between vendors.

In Table 2.1 some of the attributes and their meaning [9] are presented.

Table 2.1: S.M.A.R.T. Attributes

ID	Attribute Name	Description
smart_1	Read Error Rate	Rate of hardware read errors that occurred when reading data from a disk surface.
smart_2	Throughput Performance	Throughput performance of a hard disk drive.
smart_3	Spin-Up Time	Average time of spindle spin up (from zero RPM to fully operational [milliseconds]).
smart_4	Start/Stop Count	A tally of spindle start/stop cycles.
smart_5	Reallocated Sectors Count	Count of reallocated sectors.
smart_7	Seek Error Rate	Rate of seek errors of the magnetic heads.
smart_8	Seek Time Performance	Average performance of seek operations of the magnetic heads.
smart_9	Power-On Hours	Count of hours in power-on state.
smart_10	Spin Retry Count	A total count of the spin start attempts to reach the fully operational speed.
smart_11	Recalibration Retries	A count that recalibration was requested.
smart_12	Power Cycle Count	A count of full hard disk power on/off cycles.
smart_184	End-to-End error	A count of parity errors which occur in the data path to the media via the drive's cache RAM.
smart_187	Reported Uncorrectable Errors	The count of errors that could not be recovered using hardware ECC

smart_188	Command Timeout	The count of aborted operations due to HDD timeout.
smart_189	High Fly Writes	This attribute indicates the count of rewritten or reallocated information over the lifetime of the drive.
smart_190	Temperature Difference	Value is equal to (100-temp. C), allowing manufacturer to set a minimum threshold which corresponds to a maximum temperature.
smart_191	G-sense Error Rate	The count of errors resulting from externally induced shock and vibration.
smart_192	Power-off Retract Count	Number of power-off or emergency retract cycles.
smart_193	Load Cycle Count	Count of load/unload cycles into head landing zone position.
smart_194	Temperature	Indicates the device temperature.
smart_195	Hardware ECC Recovered	
smart_196	Reallocation Event Count	A count of attempts to transfer data from reallocated sectors to a spare area. Both successful and unsuccessful attempts are counted.
smart_197	Current Pending Sector Count	Count of "unstable" sectors (waiting to be remapped, because of unrecoverable read errors).
smart_198	Uncorrectable Sector Count	The total count of uncorrectable errors when reading/writing a sector.
smart_199	UltraDMA CRC Error Count	The count of errors in data transfer via the interface cable as determined by ICRC (Interface Cyclic Redundancy Check).
smart_200	Multi-Zone Error Rate	The count of errors found when writing a sector.
smart_201	Soft Read Error Rate	Count indicates the number of uncorrectable software read errors.
smart_223	Load/Unload Retry Count	Count of times head changes position.
smart_240	Head Flying Hours	Time spent during the positioning of the drive heads.

Some variables are considered to be critical, by the literature, in failure events. These variables are 5, 12, 187, 188, 189, 190, 198, 199 and 200 [3]. In this project, one of the objectives is also to pay attention to the variables that are not so observed, because the alert variables are generally already highly monitored by the storage companies.

Generally, in smart attributes that are already normalized by the vendors, higher values

are always better (except for temperature in some manufactures). The range is normally 0-100 and for some attributes 0-255. There is no standard on how manufacturers convert the raw values to the normalized ones: it can be a linear, exponential, logarithmic or any other range normalization. That said, it is really difficult to have a quick perception of the disks behavior on a cloud storage system, since they usually use dozens of different disk models.

2.2 Data Mining

As previously mentioned, the technological boom created a lot of information in the digital world. These data are a great source of knowledge extraction for all companies. The constant need to interpret data and discover relevant information has caused data analysis to develop rapidly in recent years.

Fayyad *et al.* [13] described the necessary steps to extract relevant information on databases. The KDD (Knowledge Discovery in Databases) process is a set of continuous activities that share the knowledge discovered from data. According to Fayyad *et al.* [13] this set is composed of five steps: data selection, pre-processing and data cleaning, processing of data, data mining, interpretation and evaluation of results (cf. Figure 2.2).

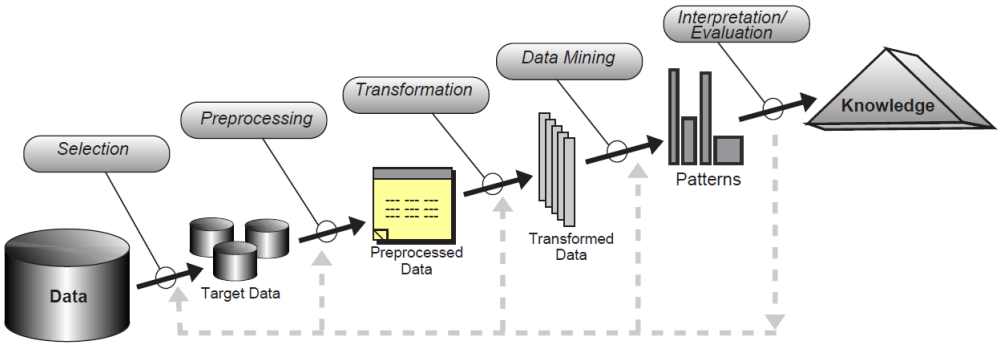


Figure 2.2: An overview of the KDD steps [13]

2.2.1 Machine Learning

Machine Learning principal objective is to understand the way data is related. It is based on algorithms that can learn models and make predictions. Machine learning tasks can be categorised as supervised, unsupervised or semi-supervised. In supervised learning the goal is to train the machine using data that is already labelled in order to give a learning basis for future processing. In this case, the labelled values correspond to what is called the target variable. In unsupervised learning, examples do not have a target variable associated, so the objective is to group similar observations without knowing what is represented by each group. Semi-supervised learning is a task using both labelled and unlabelled cases.

Supervised learning tasks can be split in two categories of algorithms: classification and regression. Classification is a form of data analysis that extracts models describing important data classes. Such models, called classifiers, predict categorical class labels. The classification model is built from the analysis of the training data set and is used to predict the class label for observations that are not categorized. Regression is used to predict a numeric or continuous values. Regression is a statistical methodology that is most often used for numeric prediction, so it is used to predict missing or unavailable numerical data values rather than (discrete) class labels [14].

2.2.2 Classification Algorithms

In the development of this work, two machine learning algorithms were used for the classification task such as Random Forest and SVM (Support Vector Machine). Below, the algorithms are described with more detail.

The goal of classification tasks is to obtain a good approximation of the unknown function that maps predictor variables toward the target value. The unknown function can be defined as $Y = f(X_1, X_2, ..., X_p)$, where Y is the target variable, $X_1, X_2, ..., X_p$ are features and $f()$ is the unknown function we want to approximate. This approximation is obtained using a training dataset $D = \{\langle x_i, y_i \rangle\}_{i=1}^{n}$

2.2.2.1 Random Forest

The Random Forest algorithm was first introduced by Breiman [8] and it is defined as an ensemble method. An ensemble is a set of multiple models, being in this case, a set of decision trees. As the name suggests, this algorithm creates a forest with a large number of decision trees, where each one considers a distinct random subset of features when forming the decision nodes, while accessing a subset of the training data. Each classifier tree is a predictor component.

In classification, Random Forest constructs is decision by counting the votes of the predictor components in each class and then selects the winning class by checking the number of votes accumulated. The process of this algorithm is represented in Figure 2.3: the first phase consists of training each decision tree with data subsets from the training set. Then, the test cases are classified by majority vote.

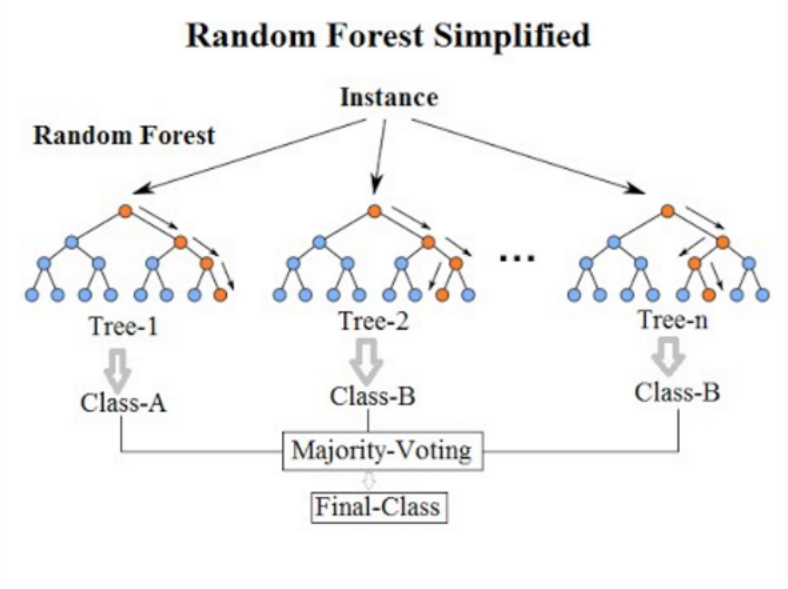


Figure 2.3: Random Forest example [21]

2.2.2.2 Support Vector Machine

Support Vector Machine, proposed by Boser *et al.* [7] in 1992, consists in a method that tries to find the largest margin to separate different classes of data. The objective of the SVM is to construct an optimal hyperplane that can separate different classes of data. In the Figure 2.4 it is possible to see how SVM works. There are several straight lines that can be drawn to separate the data, the support vectors are data points that are closer to the hyperplanes and they serve to choose the best one, represented by the filled points.

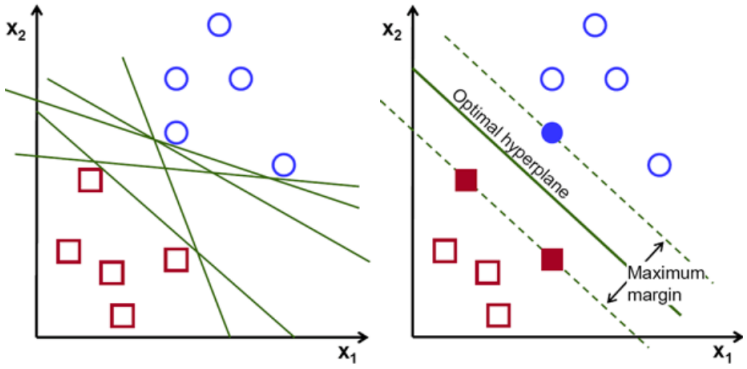


Figure 2.4: SVM example [18]

The data can not always be separable in a linear way. In these cases, the SVM maps the data to a space of higher dimension. At this point, the concepts of soft margin and kernel trick are introduced. The main idea of a soft margin is to allow some examples to be placed on the wrong side of the dividing hyperplane. The kernel transforms non-separable data to separable data by adding more dimension. Nonlinear kernel functions were proposed by Boser *et al.* [6] so SVM could be applied to data that couldn't be divided by linear hyperplanes. Table 2.2 provides some of the kernel functions.

Table 2.2: SVM Kernel Types

Kernel Type	**Formula**
Polynomial kernel	$(\mathrm{x}_i, x_j) = (x_i * x_j + r)^p, r \geq 0$
RBF kernel	$(\mathrm{x}_i, x_j) = exp(-\gamma \|\|x_i - x_j\|\|^2), \gamma > 0$
Sigmoid kernel	$(\mathrm{x}_i, x_j) = tanh(\eta x_i * x_j + v)$

2.2.3 Time Series

In almost every scientific field, some measurements are performed over time [12]. The purpose of time-series models is to extract the most meaningful knowledge from the shape (temporal) of the data. A time series is a collection of observations obtained chronologically. Time series can be regular if there is an equally spaced interval of time between the observations and irregular if the

opposite occurs. The values are typically measured at equal time intervals (e.g.,every minute, hour, or day). This type of data can be characterized in 4 different movements [14]:

- **Trend or long-term movements**: These indicate the general direction in which a time-series graph is moving over time.
- **Cyclic movements**: Are the long-term oscillations about a trend line or curve.
- **Seasonal variations**: Are nearly identical patterns that a time series appears to follow during seasons of successive time.
- **Random movements**: As the name refers, are random movements with no pattern associated.

In figure 2.5 it is possible to have a better perception of the 4 different types of movements.

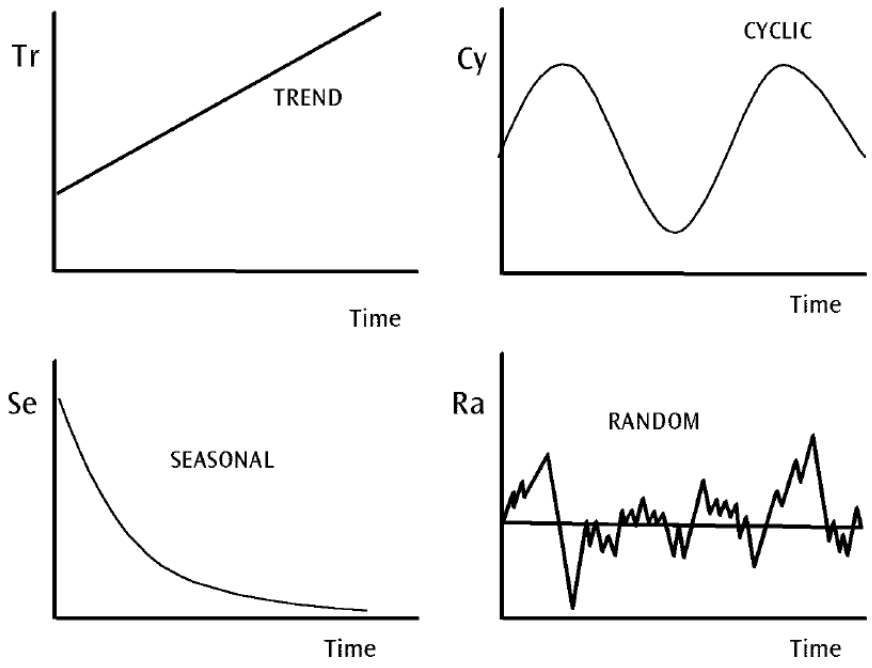


Figure 2.5: Time Series Types [2]

These movements referred above, can also be grouped in two kinds of data. Stationary and not stationary. In stationary data, the time series values do not depend on the time that the observations were collected, therefore it will not have predictable patterns in the long-term. Non stationary data typically have some kind of trend or seasonality over the time [23]. In figure 2.6 it is possible to see and example to have a better perception in the differences between these two types.

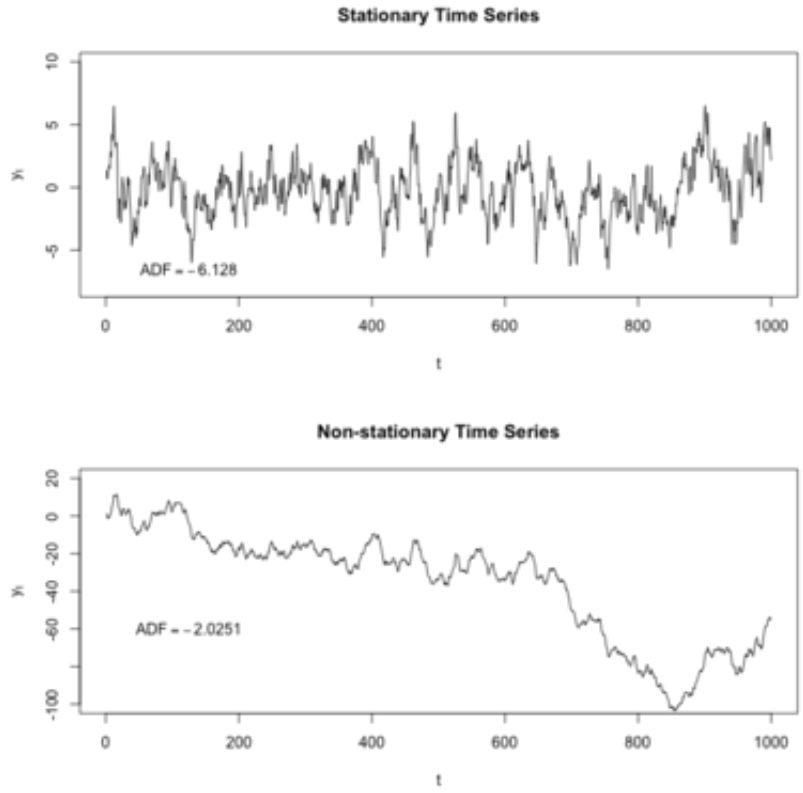


Figure 2.6: Stationary and not stationary data [26]

2.2.4 Vector Auto Regression

Lütkepohl *et al.* [23] say that if time series observations are available for a variable of interest and the past observations contain information about the future development of a feature, it is worth using as forecast some function of the data collected in the past.

The VAR model expresses each variable as a linear function of its **own past values**, the past values of all **other variables being considered**, and a serially **uncorrelated error term** [32]. Each variable has an equation explaining its progression over time. This equation includes the variable's lagged (past) values, the lagged values of the other variables in the model, and an error term.

Usually, equations of a bivariate autoregression typically take the form [15] :

$$y_t = \alpha_0 + \sum_{l=1}^{m} \alpha_l y_{t-l} + \sum_{l=1}^{m} \delta_l x_{t-l} + u_t \tag{2.1}$$

"where the α's and δ's are the coefficients of the linear projection of y_t onto a constant and past values of y_y and x_t, and the lag length m is sufficiently large to ensure that u_t is a white noise error term. While it is not essential that the lag lengths for y and x are equal, we follow typical practice by assuming that they are identical."

2.3 Imbalanced domain learning

This project faces an imbalance domain learning problem. This occurs whenever the user has an interest in cases that are rare in the training set. This can create several obstacles in the learning methods that are applied. The models created by standard learning algorithms tend to be biased towards the majority class and because of that, the evaluation metrics will not capture the competence of models in relevant cases.

Han *et al.* [14] say that *"given two-class data, the data are class-imbalanced if the main class of interest (the positive class) is represented by only a few tuples, while the majority of tuples represent the negative class."*

Therefore, it is really important to pay close attention to this type of data, and take the necessary steps to prevent getting wrong information from the data mining processes that are made.

Two of the methods utilized to handle imbalanced data are oversampling and undersampling. Oversampling works by resampling the positive tuples so that the training set contains an equal number of positive and negative tuples. Undersampling works by decreasing the number of negative tuples. It randomly eliminates tuples from the majority (negative) class until there are an equal number of positive and negative tuples [14].

In Figure 2.7 we can see a clear example of these two sampling methods:

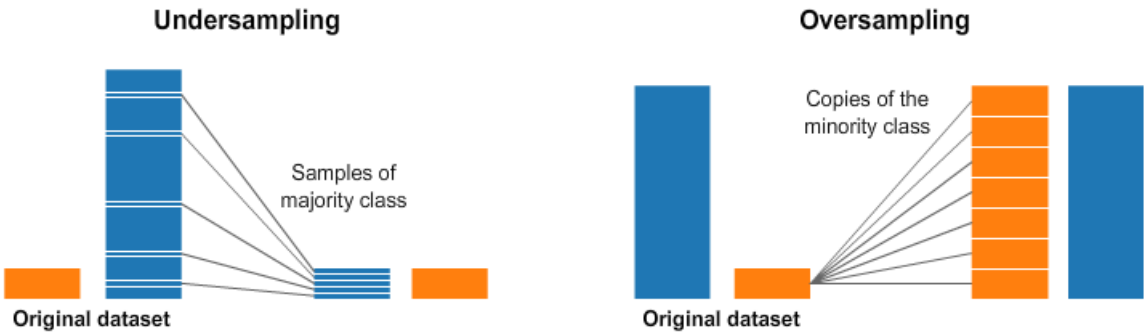


Figure 2.7: Undersampling and Oversampling examples. [20]

2.4 Evaluation Metrics

In machine learning, normaly are used the terms of positive tuples (tuples of the class of interest) and negative tuples (all the other tuples). Four more terms are used, and are the base-line of many evaluation metrics used. Below, each of the terms are explained [14]:

- **True positives (TP)**: Positive tuples that were correctly labeled by the classifier.

- **True negatives (TN)**: Negative tuples that were correctly labeled by the classifier.
- **False positives (FP)**: Negative tuples that were incorrectly labeled as positive.
- **False negatives (FN)**: Positive tuples that were incorrectly as negative.

With these four definitions, we can build a confusion matrix. It is a square matrix, with as many rows and columns as there are classes on the data. Each row represents the actual class of the observation while each column represents the predicted class. This matrix serves as a source of information for most of the metrics used. An example of a confusion matrix can be seen in Figure 2.8.

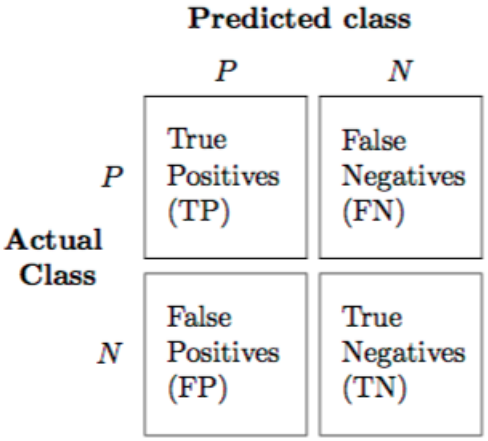


Figure 2.8: Confusion Matrix. [25]

In Figure 2.9 we can see the most used metrics to evaluate learning models.

Metric	Formula
True positive rate, recall	$\frac{TP}{TP+FN}$
False positive rate	$\frac{FP}{FP+TN}$
Precision	$\frac{TP}{TP+FP}$
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
F-measure	$\frac{2 \cdot precision \cdot recall}{precision + recall}$

Figure 2.9: Evaluation Metrics. [11]

2.5 Related Work

Several works on the subject of hard drive failure prediction have already been made. In recent years almost everyone works with the Backblaze dataset, that gathers more than 100 thousand hard disk drives and reports their respective S.M.A.R.T. variables daily [3], [33].

Aussel *et al.* [3] say that the existing predictive models no longer perform sufficiently well on the Backblaze dataset due to the extremely high unbalanced ratio of 5000:1 between healthy and failure disks, and the lack of control on the environment. For that reason they selected machine learning models for classification, like SVM, Random Forests and Gradient Boosting Trees. They achieved results of 95% precision and 67% recall with the Random Forests model and 94% precision and 67% recall with Gradient Boosting Trees. The SVM got a precision below 1%.

Wang *et al.* [33] defend that the reactive fault-tolerant measures, like RAID's (Redundant Array of Inexpensive Drives) and ECC (error correction codes) are not enough to mitigate or eliminate the negative effects of the HDD's failures. The proactive measures are more efficient because they will predict the failures in advance. However the built-in prediction models that the HDD's manufactures are using, have a quite weak prediction power, with only 4% of failure prediction rate. To overcome the issues and obtain results, they proposed a deep architecture called Amender (for Attention-augMENted Deep architEctuRe) composed of a feature integration layer, a temporal dependency extraction layer, an attention layer and a classification layer. After analyzing the results, they concluded that different S.M.A.R.T. attributes have different abilities to indicate failures. Compared with Recurrent Neural Networks (RNNs) the architecture proposed improves 8.3% on failure-prediction and 90.2% in the health status assessment. This will also help find the causes of HDD failures.

Shen *et al.* [31] propose a Random Forest predicting model capable of differentiating failure prediction for HDD's. They show that most of the statistical approaches, machine learning, and deep learning technologies are good at identifying failures that occur more frequently, but perform poorly when they face a less known behavior. A clustering-based under-sampling method is used, so the data imbalance problem is mitigated and the quality of training set is improved. The results show that the Random Forest model can achieve a FDR (Failure Detection Rate) of over 97.67% with a FAR (False Alarm Rate) of 0.017%.

Li *et al.* [22] propose two prediction models based on Decision Trees and Gradient Boosted Regression Trees in two different real-world datasets (one with 121,698 and other with 39,091 hard disks). In data preparation and pre-processing, they use quantile functions, to select the more important features on healthy and drives that fail.

- Bigger dataset: The Decision Trees model, helps in improving the hard drive failure prediction with a 93% FDR and a FAR under 0.01%. The Gradient Boosted Regression Trees also contributes in evaluating the health degree level and the results show a 90%

FDR and a 0% FAR.

- Smaller dataset:Both models show steady prediction performance, with failure detection rates of 80% to 96% and low false alarm rates of 0.006% to 0.31%.

They also mention a really interesting point by using a metric that calculates the expected number of data loss events per petabyte used by year in these companies.

Zhao *et al.* [34] believe that many works done in the area fail to consider the characteristics of the observed features, over time, and tend to make the predictions based on individual or a set of attributes. They also believe that it is reasonable that attribute values observed over time are not independent, and a sequence of observed values with certain patterns may be a good indicator on whether or not a drive may fail soon. Therefore, they consider the observations from the disks, as a time series and apply a Hidden Markov Model and a Hidden semi-markov model to build prediction models that could label disks as healthy or pre-failing. Although their FDR results are not high (up until 46% for single attributes and 52% for multiple attributes), they achieve a FAR of 0% in both cases.

Chapter 3

Failure and Misbehavior Prediction

In this chapter the methodology to achieve the proposed objectives is described. Starts by presenting the base method (CRISP-DM) used in this work. Then, a detailed description about the Backblaze data is presented, as well as the approach taken to pre-process, analyse and modeling these datasets.

3.1 Methodology

In this work we will use the popular CRISP-DM (Cross Industry Standard Process for Data Mining) methodology to analyse the data and build the learning models. This method, as shown in Figure 3.1, consists of six steps. Next, a brief describe of each one them is presented, highlighting our HDD domain.

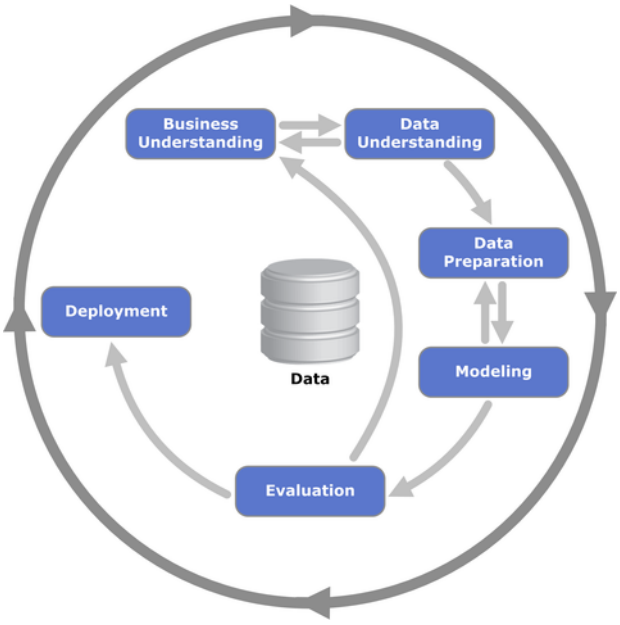


Figure 3.1: CRISP-DM process diagram.[**?**]

Business and data understanding

The proposed methodology begins by understanding the main concepts about the HDD market, and the cloud storage systems as a business, in order to get a better insight into the problem that is intended to solve. The data understanding makes a big part of this project, so a prior analysis to inspect the various attributes and observations is made, and a verification of the data quality to ensure that the pre-processing methods are done correctly.

Data Preparation

The dataset used is from Backblaze (explained further) so there is no need to create a program to collect the disks observations and merge them in a database. Therefore, it is just necessary to pre-process the data, based on the information extracted on the steps above. The usual tasks like removing missing values, duplicated observations and biased values are executed.

Modelling and Evaluation

In this phase, the learning models are selected according to the project objective. Since the dataset is a time series, a forecasting model is selected so the results can demonstrate some correlation through time between the variables, and a forecast can be calculated to predict some disk misbehavior. Lastly, two classification algorithms are applied, to failed and healthy disks, and the respective evaluation metrics are analysed.

3.2 Dataset Description

The Dataset utilized in this project is provided by Backblaze, a cloud storage and data backup company, founded in 2007. They provide B2 Cloud Storage and Computer Backup services, targeted at both business and personal markets. Since 2013, Backblaze has been publishing statistics and insights based on the hard drives in their data center. Along with that, they made their data available to the public [4].

Every day in the Backblaze data center, a snapshot of each operational hard drive is taken. The snapshots include basic drive information (explained below) along with the S.M.A.R.T. attributes reported by each drive. All the drives informations are collected into a file consisting of a row for each hard drive. The file format is a "csv" (Comma Separated Values) [4].

As mentioned, the company joins some drive information to the S.M.A.R.T. attributes. That information is described as follows:

- **Date** – The day when the snapshot was taken in yyyy-mm-dd format.
- **Serial Number** – The manufacturer-assigned serial number of the drive.
- **Model** – The manufacturer-assigned model number of the drive.
- **Capacity** – The drive capacity in bytes.

- **Failure** – Contains a "0" if the drive is OK. Contains a "1" if this is the last day the drive was operational before failing.

In figure 3.2 it is possible to have a better visualization of the type of data that is being used.

	date	serial_number	model	capacity_bytes	failure	smart_1_normalized	smart_1_raw	smart_2_normalized	smart_2_raw	smart_3_normalized	smart_3_raw	smart_4_normalized	smart_4_raw
0	2019-10-01	Z305B2QN	ST4000DM000	4000787030016	0	115.0	97236416.0	NaN	NaN	91.0	0.0	100.0	13.0
1	2019-10-01	ZJV0XJQ4	ST12000NM0007	12000138625024	0	67.0	4665536.0	NaN	NaN	96.0	0.0	100.0	3.0
2	2019-10-01	ZJV0XJQ3	ST12000NM0007	12000138625024	0	80.0	92892872.0	NaN	NaN	99.0	0.0	100.0	1.0
3	2019-10-01	ZJV0XJQ0	ST12000NM0007	12000138625024	0	84.0	231702544.0	NaN	NaN	93.0	0.0	100.0	6.0
4	2019-10-01	PL1331LAHG1S4H	HGST HMS5C4040ALE640	4000787030016	0	100.0	0.0	134.0	103.0	100.0	436.0	100.0	9.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
115254	2019-10-01	ZA10MCEQ	ST8000DM002	8001563222016	0	77.0	51786816.0	NaN	NaN	94.0	0.0	100.0	3.0
115255	2019-10-01	ZCH0CRTK	ST12000NM0007	12000138625024	0	73.0	20128440.0	NaN	NaN	97.0	0.0	100.0	3.0
115256	2019-10-01	AAGA7W2H	HGST HUH721212ALN604	12000138625024	0	100.0	0.0	132.0	96.0	100.0	0.0	100.0	1.0
115257	2019-10-01	PL1331LAHGD9NH	HGST HMS5C4040BLE640	4000787030016	0	100.0	0.0	134.0	100.0	100.0	459.0	100.0	5.0
115258	2019-10-01	ZJV5JLF1	ST12000NM0007	12000138625024	0	84.0	244009373.0	NaN	NaN	99.0	0.0	100.0	1.0

115259 rows × 131 columns

Figure 3.2: A dataset sample example from October 1st.

In this project, the data from the fourth trimester of 2019 will be used. 92 datasets from the period of 01-October to 31-December were downloaded from the Backblaze data center. After a brief analysis to all files, 125,731 different disks were identified during the three months.

From the 125,731 disks, only 678 failed showing a failure rate below 0.54%. This demonstrates a huge disparity between the two classes that categorize the disks. In Figure 3.3 it is possible to have a better visualization of the data distribution.

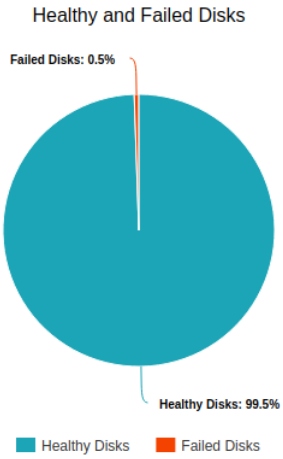


Figure 3.3: Diagram of Failed and Healthy Disks

The company has in its storage systems, dozens of different HDD's models and from different vendors too. In Table 3.1, it is possible to see all types of disks available on this dataset.

Table 3.1: All disk models and their numbers available on Backblaze Storage.

Disk model	**Number of Disks**	**Disk model**	**Number of Disks**
DELLBOSS VD	60	ST6000DM001	4
HGST HDS5C4040ALE630	26	ST6000DM004	1
HGST HMS5C4040ALE640	2833	ST6000DX000	887
HGST HMS5C4040BLE640	12758	ST8000DM002	9844
HGST HMS5C4040BLE641	1	ST8000DM004	3
HGST HUH721010ALE600	20	ST8000DM005	25
HGST HUH721212ALE600	1561	ST8000NM0055	14502
HGST HUH721212ALN604	10866	Seagate BarraCuda SSD ZA2000CM10002	4
HGST HUH728080ALE600	1002	Seagate BarraCuda SSD ZA250CM10002	157
HGST HUS726040ALE610	28	Seagate BarraCuda SSD ZA500CM10002	18
Hitachi HDS5C4040ALE630	2	Seagate SSD	107
ST10000NM0086	1205	TOSHIBA HDWE160	4
ST1000LM024 HN	1	TOSHIBA HDWF180	20
ST12000NM0007	37442	TOSHIBA MD04ABA400V	99
ST12000NM0008	7226	TOSHIBA MG07ACA14TA	3627
ST12000NM0117	15	TOSHIBA MQ01ABF050	475
ST16000NM001G	40	TOSHIBA MQ01ABF050M	425
ST4000DM000	19330	WDC WD5000BPKT	10
ST4000DM005	39	WDC WD5000LPCX	54
ST500LM012 HN	501	WDC WD5000LPVX	214
ST500LM021	33	WDC WD60EFRX	3
ST500LM030	259		

Not all vendors use the same variables, so it is important to note which ones will be kept in the dataset. In Table 3.2 the features that each vendor prefers to use, are summarised. It is important to note that not all variables are presented on the table, as vendors do not use many of them.

Table 3.2: S.M.A.R.T. Attributes used from each vendor.

Vendors / S.M.A.R.T.	Toshiba	Hitachi	Seagate	WDC
smart_1	✓	✓	✓	✓
smart_2	✓	✓	✓	
smart_3	✓	✓	✓	✓
smart_4	✓	✓	✓	✓
smart_5	✓	✓	✓	✓
smart_7	✓	✓	✓	✓
smart_8	✓	✓	✓	
smart_9	✓	✓	✓	✓
smart_10	✓	✓	✓	✓
smart_11			✓	✓
smart_12	✓	✓	✓	✓
smart_18			✓	
smart_22		✓		
smart_23	✓			
smart_24	✓			
smart_183			✓	
smart_184			✓	
smart_187			✓	
smart_188			✓	
smart_189			✓	
smart_190			✓	
smart_191	✓		✓	✓
smart_192	✓	✓	✓	✓
smart_193	✓	✓	✓	✓
smart_194	✓	✓	✓	✓
smart_195			✓	
smart_196	✓	✓	✓	✓
smart_197	✓	✓	✓	✓
smart_198	✓	✓	✓	✓
smart_199	✓	✓	✓	✓
smart_200			✓	✓
smart_220	✓			
smart_222	✓			
smart_223	✓	✓	✓	
smart_224	✓			
smart_225			✓	
smart_226	✓			
smart_240	✓		✓	✓
smart_241		✓	✓	
smart_242		✓	✓	
smart_254			✓	
Total	**26**	**21**	**34**	**19**

Backblaze gives some helpful considerations to be taken before working with the data available on the platform. Below some of them are presented [4].

Blank Fields

"The daily snapshots record the SMART stats information reported by the drive. Since most drives do not report values for all SMART stats, there are blank fields in every record. Also, different drives may report different stats based on their model and/or manufacturer."

Inconsistent Fields

"Reported stats for the same SMART stat can vary in meaning based on the drive manufacturer and the drive model. Make sure you are comparing apples-to-apples as drive manufacturers don't generally disclose what their specific numbers mean."

Out-of-Bounds Values

"The values in the files are the values reported by the drives. Sometimes, those values are out of whack. For example, in a few cases the RAW value of SMART 9 (Drive life in hours) reported a value that would make a drive 10+ years old, which was not possible. In other words, it's a good idea to have bounds checks when you process the data."

The Number of Drives Will Change

"When a drive fails, the "Failure" field is set to "1" on the day it fails. The next day, the drive is removed from the list and is no longer counted, reducing the overall number of drives. On the other hand, new drives are added on a regular basis increasing the overall number of drives. In other words, count the number of drives each day."

3.3 Data Preparation

For this project, data mining tasks were performed using the Python language, mainly using the pandas library, for data manipulation and analysis. Pandas offers structures and operations for manipulating numeric tables and time series. As a code development interface, Jupyter lab [19] and Google Collaboratory [10] platforms were used. The last one offers a virtual machine service, so the resources used were not from a personal machine. Access to a server at the DCC was also granted so that high-cost processing and memory executions could be performed.

The process of cleaning data is an important way to extract wrong values, missing values and information that could be biased. One of the first and natural approaches to take would be to remove disks that have less than 100 units per model, to make the dataset more homogeneous and work only with the most used models, as they would consequently have more information to extract. However, as this project works with imbalanced domains, the removed models could fail during the time period under analysis, and this would make the minority class even smaller.

As it is possible to see in Figure 3.2, each feature has a raw_ value and a normalized_value associated. As mentioned before, in Chapter 2, the normalized_value is a mapping from raw_values to discrete values chosen by vendors. As the objective would be to use classification algorithms, the ideal would be to work with normalized data, so that the values of the observations are not too dispersed. However, data normalization causes a lot of information to be lost during the process and raw values from different disks, which are very dispersed, end up belonging to the same value range when normalized.

Another natural approach would be to remove columns that have more than half of the observations as missing values. But as mentioned above, there are models that do not use some of the existing features in the dataset so the number of missing values will naturally increase, thus it is not possible to conclude that the variable is removable.

As discussed in the previous paragraphs, the pre-processing performed on this dataset requires a lot of attention and care. Since removing models with less than 100 disks would not be a good approach, it was decided that the models that never failed over the 3 months, would be removed. This measure was taken because the observations of these disks will never take our class of interest into account so it is not worth analyzing them.

Then it is necessary to deal with data imbalance. One of the methods presented in chapter 2 was undersampling, that consists of randomly choosing a smaller set of observations, capable of balancing the dataset. As the objective is to analyze the dataset in a temporal way, a function was executed to determine the day with most failures. After selecting it, the failed disks were gathered, and the respective observations were collected from day 1 to the selected day. With this, it is possible to obtain a dataset in the form of a time series, to better analyze the behavior of the disks over the days. From now, the undersampling method is used to select the disks that were healthy until the day, and collect the respective observations over the period. It is important to note, that the healthy disks must be from the same model as the disks that failed, so that the comparison would be correctly done, because as stated by Backblaze, different models make use of different variables. Therefore, 2 datasets are created: Healthy Disks and Failed Disks. In Figure 3.4 it is possible to have a better visualization of this process where blue corresponds to healthy disks and red to the failed disks.

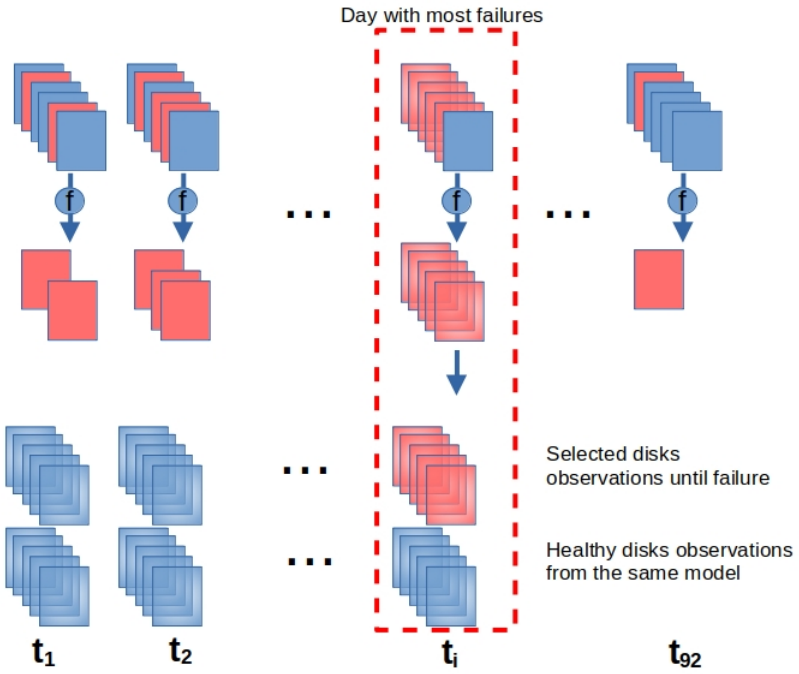


Figure 3.4: Pre-Processing Diagram Example.

3.4 Methodology for Data Visualization and Analysis

Before the models are applied, a temporal analysis is executed to both datasets created (Healthy Disks and Failed Disks) for a better visualization of the oscillations in the variables values, so as the differences between the healthy and failing disks. Also the Euclidean distances, between the failed disks and the healthy ones, are calculated for every feature. With this, it is possible to obtain a better numerical perception. Both processes will help extracting information from the data and turning the decision making, before applying the learning algorithms, more efficient and accurate.

The temporal analysis is performed by plotting the variables of interest for failing disks and healthy disks. The two graphs will be placed side by side for the comparison to be made.

The Euclidean distance is calculated for each variable. This distance helps understanding how dispersed are the values, of the same variables, between a healthy disk and a disk that ends up failing. To better clarify this process, the objective is to compare the smart_1, over time, of a healthy disk, with the smart_1 over time of a disk that will fail. To ensure that the execution of the algorithms is well made, the comparison between the disks is always done with the same models.

3.5 Learning Algorithms

As mentioned in Chapter 2, 3 learning models are applied in order to find more effective ways of preventing disk failures. An algorithm is used to analyze the dataset as a time series (VAR), and two are used as classification models (Random Forest and SVM).

3.5.1 Vector Auto Regression

To apply the VAR model, it was necessary to divide the two datasets created, into subdatasets that were grouped by serial_number. Thus, subdatasets would only contain observations over time of a given disk. By this way it is possible to apply the model to each disk and make a comparison between the healthy and the failing ones.

The observations from the last 5 days of each disk were removed so when the forecast was executed, the prediction could be compared with the real values. The drive information that was added by Backblaze is removed, except the Date values, so that the table is only constituted with S.M.A.R.T. variables, over time. This measure is taken, because the VAR model only performs operations on numeric variables and would not extract any information from the variables that were added only to describe the disk (Ex: Serial_number, Model).

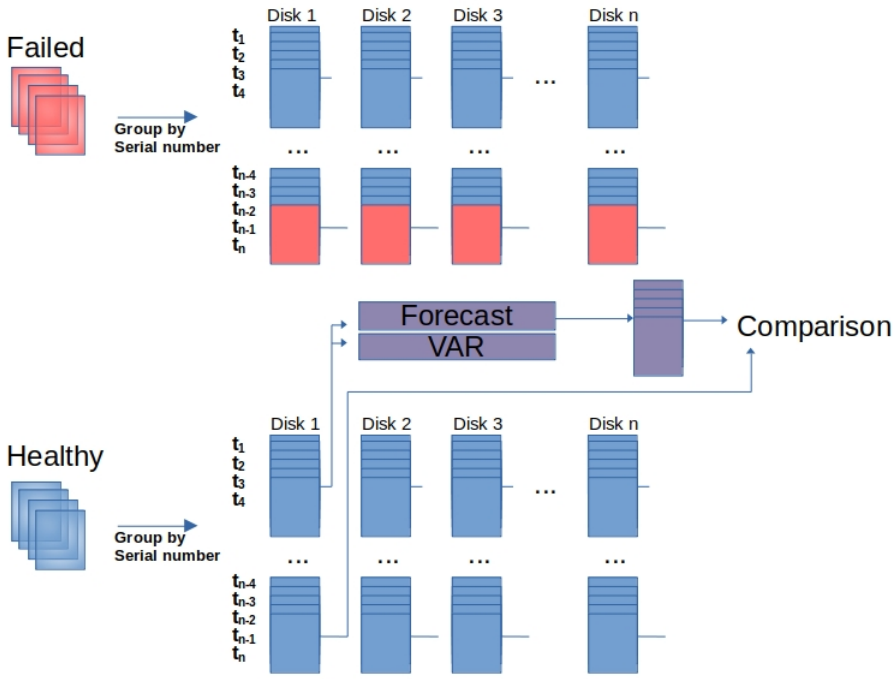


Figure 3.5: VAR Diagram Example.

Since one of the main algorithm goals is to show correlations between variables over time, features that remain constant over the days, were removed, because they don't fit in the model. Finally, the VAR model is applied to the subdatasets, the results are calculated using the different

lags, and the forecast is executed with the objective of predicting the behavior of each attribute.

3.5.2 Classification Algorithms

The first thing to do so the execution of the algorithms would be correctly done, is to divide the datasets (Healthy and Failed) in subdatasets once more, but this time, in subdatasets grouped by disk models. This can be done because there is no need to have a temporal view of the data, so more than 1 disk can be placed on the new subdataset. Then it is necessary to add the class variable to all observations. The disks that fail will have the class equal to 1 and those that remain healthy will have the class equal to 0. In these algorithms, only the S.M.A.R.T. attributes remain in the dataframe, the rest of the variables are eliminated for the same reasons referenced in the VAR model. After all these measures are taken, there may be observations that have S.M.A.R.T. variables with equal values, and since they no longer have serial_numbers to distinguish them, these observations will be removed so there are no duplicates and the classification models are not affected. All features were normalized to values between 0 and 1, since the learning algorithms had difficulties to perform the operations in the standard values. This normalization was made after the disks were divided by models, so the values range were not mixed up. The validation method utilized was the train-test-split with a 80 to 20 ratio. Both classification algorithms, SVM and Random Forest, were executed using default parameters.

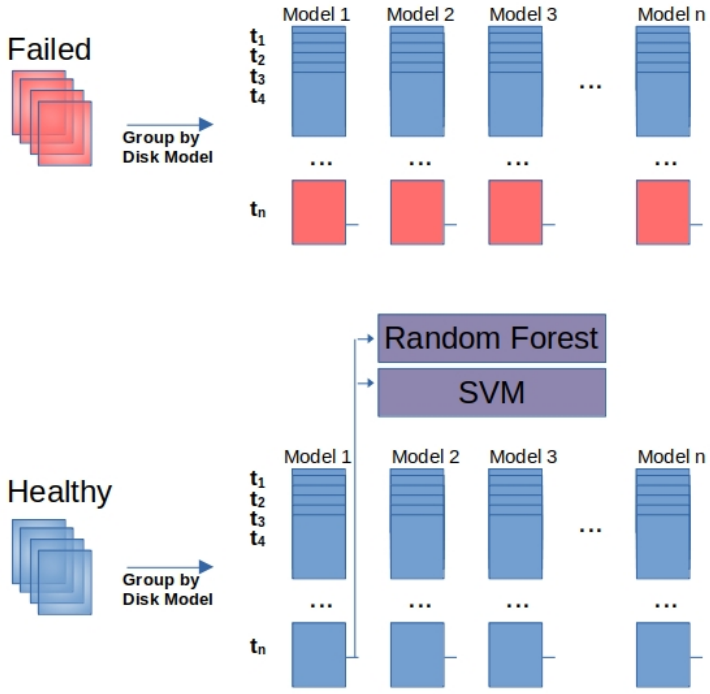


Figure 3.6: Classification models Diagram Example.

Chapter 4

Results and Analysis

In this chapter, the results of the methods applied to the pre-processed data are presented and analysed. The first step is to describe the statistical analysis made before the pre-processing and after it is done. Also the data used in the learning algorithms is shown for a better visualization of the features that were worked with. The second step is to present the results obtained through the temporal analysis and the Euclidean distances that were calculated between the two subdatasets (Healthy and Failed). Finally, the results of the learning models are observed and interpreted.

4.1 Pre-Processing

As described in the methodology, one of the first measures taken was counting the number of disks available throughout the trimester, as well as the respective failures. Table 4.1 shows a summary of the analysis made. All disks that never failed, can be removed, as they will not take our class of interest into account.

Table 4.1: Disks count and respective Failure Rate

Model	**Disk Count**	**Failure Count**	**Failure Rate**
DELLBOSS VD	60	0	0.0%
HGST HDS5C4040ALE630	26	0	0.0%
HGST HMS5C4040ALE640	2833	4	0.14%
HGST HMS5C4040BLE640	12758	12	0.09%
HGST HMS5C4040BLE641	1	0	0.0%
HGST HUH721010ALE600	20	0	0.0%
HGST HUH721212ALE600	1561	1	0.06%
HGST HUH721212ALN604	10866	7	0.06%
HGST HUH728080ALE600	1002	2	0.2%
HGST HUS726040ALE610	28	0	0.0%
Hitachi HDS5C4040ALE630	2	0	0.0%
ST10000NM0086	1205	5	0.41%
ST1000LM024 HN	1	0	0.0%
ST12000NM0007	37442	364	0.97%
ST12000NM0008	7226	10	0.14%
ST12000NM0117	15	0	0.0%
ST16000NM001G	40	0	0.0%
ST4000DM000	19330	119	0.62%
ST4000DM005	39	0	0.0%
ST500LM012 HN	501	13	2.59%
ST500LM021	33	0	0.0%
ST500LM030	259	6	2.32%
ST6000DM001	4	0	0.0%
ST6000DM004	1	0	0.0%
ST6000DX000	887	1	0.11%
ST8000DM002	9844	35	0.36%
ST8000DM004	3	0	0.0%
ST8000DM005	25	0	0.0%
ST8000NM0055	14502	53	0.37%
Seagate BarraCuda SSD ZA2000CM10002	4	0	0.0%
Seagate BarraCuda SSD ZA250CM10002	157	0	0.0%
Seagate BarraCuda SSD ZA500CM10002	18	0	0.0%
Seagate SSD	107	0	0.0%
TOSHIBA HDWE160	4	0	0.0%
TOSHIBA HDWF180	20	0	0.0%
TOSHIBA MD04ABA400V	99	0	0.0%
TOSHIBA MG07ACA14TA	3627	7	0.19%
TOSHIBA MQ01ABF050	475	23	4.84%
TOSHIBA MQ01ABF050M	425	10	2.35%
WDC WD5000BPKT	10	0	0.0%
WDC WD5000LPCX	54	0	0.0%
WDC WD5000LPVX	214	6	2.8%
WDC WD60EFRX	3	0	0.0%
Total	**125731**	**678**	**0.54%**

The second step, in the pre-processing methods, was based on analyzing the number of failures per day, so it would be possible to build a dataset that had the most number of failures and the chosen disks would have enough previous observations to perform a reasonable temporal

analysis.

In Figure 4.1 the results of this process are presented in a Bar Chart.

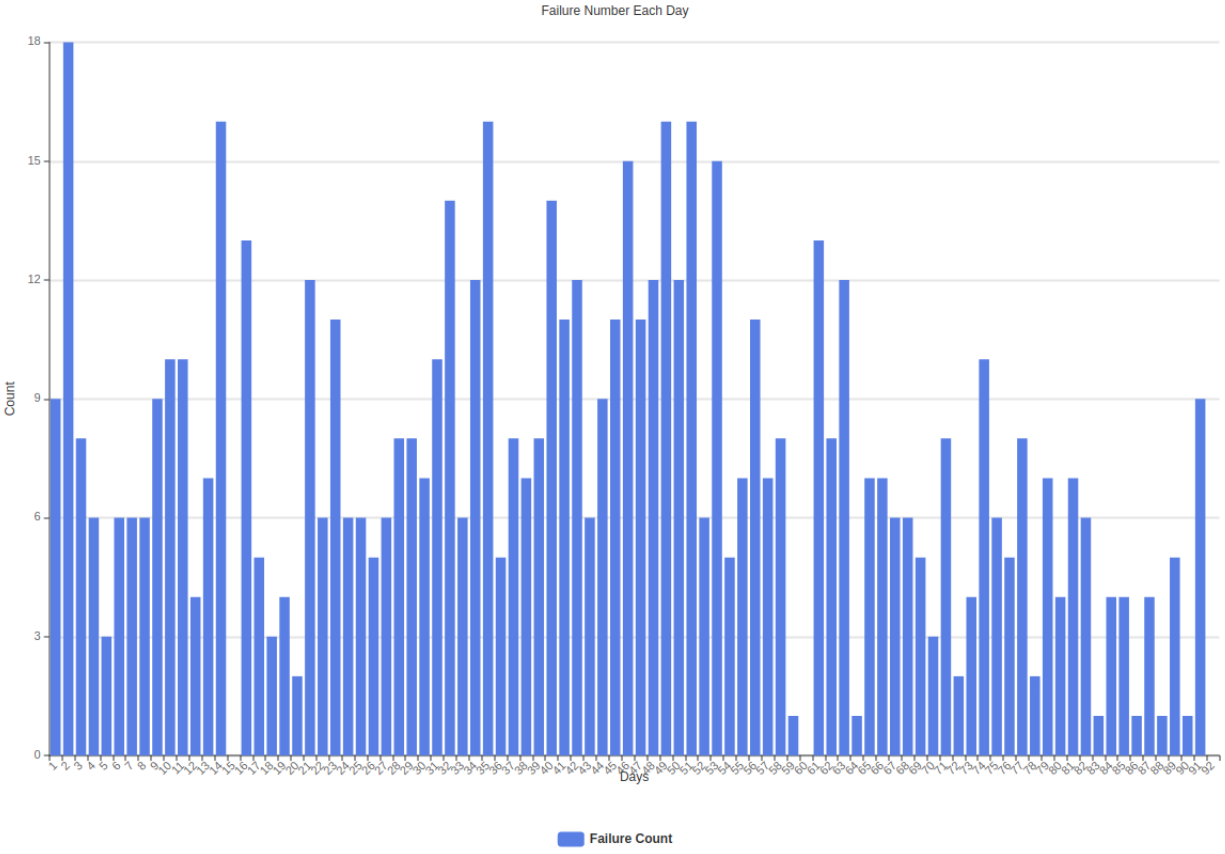


Figure 4.1: Bar Chart related to the number of failures per Day

As it is possible to observe, the second day was the day that had most failures (18 disks), however, since the objective is to have observations collected before the failing day, it is necessary to select another date, that is further from the beginning of the time scale. **14-10-2019**, **04-11-2019**, **18-11-2019** and **20-11-2019** were the next days with most failing disks (all with 16 disks). Therefore, **20-11-2019** was the selected date because there are more observations to be collected.

After the day was selected, all observations for the disks that failed, were collected. A 16 disks sampling, from disks that never failed, was executed to the rest of the dataset, with a condition that the collected observations needed to be from the same models as the failed disks. The results obtained can be observed in Table 4.2.

Table 4.2: Selected Healthy and Failed Disks

Healthy Disks		Failed Disks	
Serial_number	**Model**	**Serial_number**	**Model**
ZCH06YQ3	ST12000NM0007	ZCH097GA	ST12000NM0007
ZCH0D2V0	ST12000NM0007	ZJV00F20	ST12000NM0007
ZJV5GMSJ	ST12000NM0007	ZJV03JDV	ST12000NM0007
ZJV10J45	ST12000NM0007	ZJV00C88	ST12000NM0007
ZCH056VR	ST12000NM0007	ZJV03NQB	ST12000NM0007
ZJV501TY	ST12000NM0007	ZJV17SG6	ST12000NM0007
ZCH0BCML	ST12000NM0007	ZCH0AL23	ST12000NM0007
ZCH06HY1	ST12000NM0007	ZCH0A7G6	ST12000NM0007
ZCH0CDWV	ST12000NM0007	ZCH0C5JJ	ST12000NM0007
Z305D2CY	ST4000DM000	S301P6Y6	ST4000DM000
Z302DJZ6	ST4000DM000	Z302T88S	ST4000DM000
ZA18BTFV	ST8000NM0055	ZA1819DM	ST8000NM0055
ZA16YG7B	ST8000NM0055	ZA17ZNQ9	ST8000NM0055
ZHZ3PT1S	ST12000NM0008	ZHZ3MSH6	ST12000NM0008
X8B0A007F97G	TOSHIBA MG07ACA14TA	X8B0A00QF97G	TOSHIBA MG07ACA14TA
57RFWNHLT	TOSHIBA MQ01ABF050	17OYTGL3T	TOSHIBA MQ01ABF050

4.2 Temporal Analysis

Figure 4.2 and Figure 4.3 show the temporal behavior of the Smart_1_normalized variable for the failed and healthy disks respectively. It is possible to observe, for both cases, that four types of models have a really distinct behavior. These are ST4000DM000, ST12000NM0007 and the two Toshiba models (whose behavior overlap with each other). In both graphs, some models have peaks during time, indicating higher Read Error Rate than the regular behavior.

According to Backblaze documentation, it is common for different vendors and different models to have different reference values. That can explain why 3 groups with dispersed values are presented. According to Seagate, the use of third party software can also read different range values than the variable should report [29].

The remaining graphs for the variables that show significant variations are presented in the Appendix.

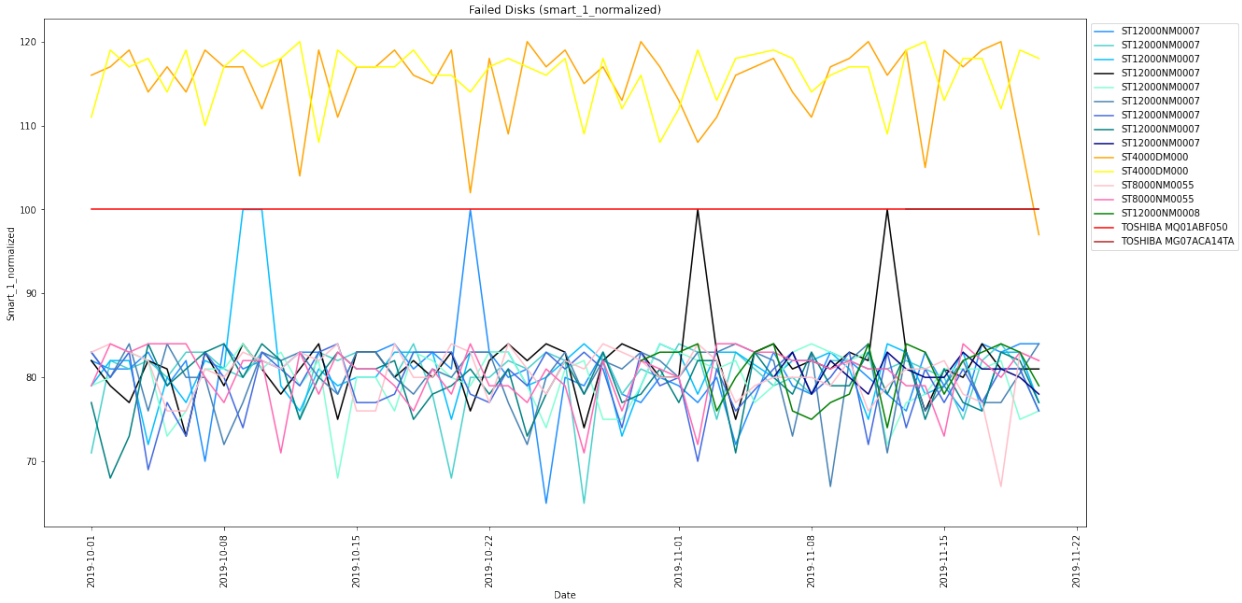


Figure 4.2: Smart_1 for the failed disks along time

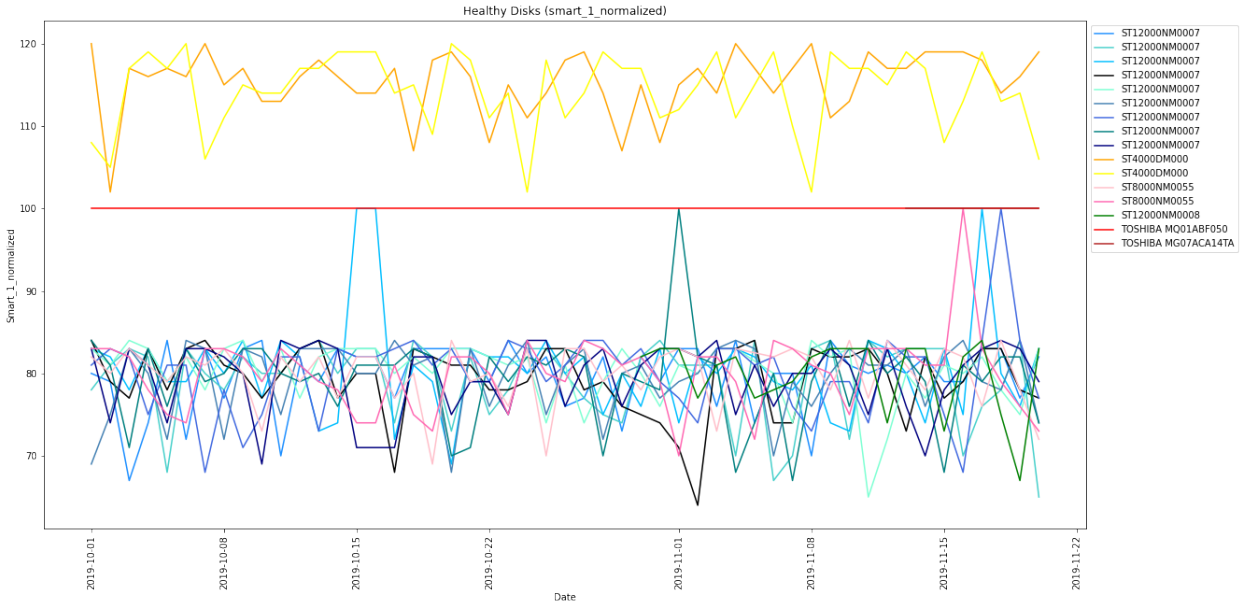


Figure 4.3: Smart_1 for the healthy disks along time

Euclidean distances were calculated over time since raw observational variables were difficult to obtain. In Tables 4.3, 4.4, 4.5 the results of the distances can be found.

Table 4.3: Euclidean distances between the healthy and failed disks

	EUCLIDEAN DISTANCES BETWEEN THE HEALTHY AND FAILED DISKS					
	smart_1_normalized	**smart_1_raw**	**smart_7_normalized**	**smart_7_raw**	**smart_9_normalized**	**smart_9_raw**
ST12000NM0007	2.896801	3.141699	N/A	0.186010	3.000000	0.056366
ST12000NM0007	2.054655	2.665402	1.092906	0.532815	**3.041381**	0.064692
ST12000NM0007	1.964036	3.185758	2.403701	0.704135	3.000000	0.040648
ST12000NM0007	3.832814	2.756828	**4.358899**	1.431158	3.041381	0.066226
ST12000NM0007	1.991425	2.650050	4.845187	**5.122032**	2.645751	0.053915
ST12000NM0007	2.565644	**3.179320**	1.471768	0.574069	2.449490	0.046023
ST12000NM0007	2.885591	2.882519	**3.122499**	0.871523	3.000000	0.056832
ST12000NM0007	3.051245	2.747076	3.752777	0.313711	3.000000	0.050737
ST12000NM0007	1.454620	1.372290	2.373880	0.160891	N/A	0.107380
ST4000DM000	2.293516	3.111510	2.719062	**3.947537**	2.236068	0.050780
ST4000DM000	2.394414	2.934578	N/A	0.299613	**3.000000**	0.031744
ST8000NM0055	2.323827	3.176403	**4.716991**	0.275336	3.041381	0.056363
ST8000NM0055	**3.533003**	3.427996	3.464102	0.530212	3.041381	0.057715
ST12000NM0008	1.980921	**2.321437**	0.139754	0.056393	N/A	0.005388
TOSHIBA MG07ACA14TA	N/A	N/A	N/A	N/A	0.881917	0.039677
TOSHIBA MQ01ABF050	N/A	N/A	N/A	N/A	N/A	0.135676

Table 4.4: Euclidean distances between the healthy and failed disks

	EUCLIDEAN DISTANCES BETWEEN THE HEALTHY AND FAILED DISKS					
	smart_190_normalized	**smart_190_raw**	**smart_192_raw**	**smart_193_raw**	**smart_194_normalized**	**smart_194_raw**
ST12000NM0007	2.483445	2.483445	1.619691	0.371148	2.483445	2.483445
ST12000NM0007	2.032753	2.032753	1.157226	0.628584	2.032753	2.032753
ST12000NM0007	**3.645545**	**3.645545**	1.084268	0.089325	**3.645545**	**3.645545**
ST12000NM0007	2.057304	2.057304	0.936340	0.229052	2.057304	2.057304
ST12000NM0007	2.390214	2.390214	0.391854	0.171993	2.390214	2.390214
ST12000NM0007	2.640707	2.640707	0.446825	0.232692	2.640707	2.640707
ST12000NM0007	2.315407	2.315407	1.379692	0.271378	2.315407	2.315407
ST12000NM0007	2.736989	2.736989	0.898376	0.262118	2.736989	2.736989
ST12000NM0007	**2.420973**	**2.420973**	1.262438	0.623084	**2.420973**	**2.420973**
ST4000DM000	2.190735	2.190735	N/A	2.023830	2.190735	2.190735
ST4000DM000	2.935652	2.935652	N/A	N/A	2.935652	2.935652
ST8000NM0055	3.040468	3.040468	N/A	0.475524	3.040468	3.040468
ST8000NM0055	2.808717	2.808717	N/A	0.488010	2.808717	2.808717
ST12000NM0008	0.634389	0.634389	0.971825	0.025017	0.634389	0.634389
TOSHIBA MG07ACA14TA	N/A	N/A	N/A	N/A	N/A	**3.133599**
TOSHIBA MQ01ABF050	N/A	N/A	N/A	0.124013	N/A	**0.490990**

Table 4.5: Euclidean distances between the healthy and failed disks

	EUCLIDEAN DISTANCES BETWEEN THE HEALTHY AND FAILED DISKS				
	smart_195_normalized	**smart_195_raw**	**smart_240_raw**	**smart_241_raw**	**smart_242_raw**
ST12000NM0007	**3.542220**	3.141699	0.054533	0.428738	0.492309
ST12000NM0007	2.331108	2.665402	0.066951	0.539049	0.929371
ST12000NM0007	1.964036	3.185758	0.038755	1.912447	1.286088
ST12000NM0007	3.832814	2.756828	0.065074	0.629421	2.144796
ST12000NM0007	1.830734	2.650050	0.052356	0.241918	0.438149
ST12000NM0007	2.565644	3.179320	0.055710	2.145395	1.042680
ST12000NM0007	2.967972	2.882519	0.060883	1.069801	1.628842
ST12000NM0007	**4.003278**	2.747076	0.055455	0.872949	0.444799
ST12000NM0007	1.684156	1.372290	0.113008	0.316462	0.831752
ST4000DM000	N/A	N/A	0.048300	0.306779	1.235419
ST4000DM000	N/A	N/A	0.032122	0.257148	0.909482
ST8000NM0055	2.323827	3.176403	0.057763	0.758562	0.214966
ST8000NM0055	**3.533003**	3.427996	0.061746	0.112897	0.645277
ST12000NM0008	1.000000	**2.321437**	0.013047	0.013032	0.014404
TOSHIBA MG07ACA14TA	N/A	N/A	N/A	N/A	N/A
TOSHIBA MQ01ABF050	N/A	N/A	N/A	N/A	N/A

The variables values that present bigger distances, for the comparisons made to the pairs of disks (healthy, failed), are selected in bold for a better visualization. The variables Smart_7 and Smart_194 are the ones with highest values and, therefore, their analysis may prove to be useful in discovering possible anomalies in the disks. It is also possible to notice that the

smart_7_normalized is the one that obtains the highest values in the Euclidean distances, with an approximate mean of 2.87.

As mentioned in Chapter 3, sometimes the values normalization can cause information loss in the data. Analyzing the values between the raw variables and the normalized variables, demonstrates that the difference between the distances can be quite notable.

For example, when comparing the raw variables of two disks of the same model (one healthy and one that fails), it is possible to find some differences in the behaviors, but if a comparison is made with the normalized values, this difference may no longer be shown. This can happen on both ways, the smart_raw shows no difference and the smart_normalized can show.

4.3 Classification Algorithms

After all steps, referred in Chapter 4, were executed, the classification models were applied to the subdatasets created (12 dataframes distinguished by disk model). From Table 4.6 to Table 4.17, the metrics results are presented along with the respective Confusion Matrix. The tables present metrics like precision, recall, f1-score and accuracy. It is also possible to observe the support of each class, that corresponds to how many observations are labeled for each class. In the confusion matrices the predicted cases for each classification algorithm are presented so it is possible to evaluate the respective performance. All the presented results are obtained from the test set.

Table 4.6: Metrics Results for model ST12000NM0007

	SVM				**RANDOM FOREST**			
	precision	**recall**	**f1-score**	**support**	**precision**	**recall**	**f1-score**	**support**
Healthy	1.00	0.99	0.99	92	0.99	0.99	0.99	92
Failure	0.99	1.00	0.99	83	0.99	0.99	0.99	83
accuracy			0.99	175			0.99	175
macro avg	0.99	0.99	0.99	175	0.99	0.99	0.99	175
weighted avg	0.99	0.99	0.99	175	0.99	0.99	0.99	175

Table 4.7: Confusion Matrix model ST12000NM0007

		SVM		**RANDOM FOREST**	
		Predicted Class		**Predicted Class**	
		Healthy	*Failure*	*Healthy*	*Failure*
Actual Class	*Healthy*	91	1	91	1
	Failure	0	83	1	82

Table 4.8: Metrics Results for model ST4000DM000

	SVM				**RANDOM FOREST**			
	precision	**recall**	**f1-score**	**support**	**precision**	**recall**	**f1-score**	**support**
Healthy	0.83	1.00	0.91	20	0.95	1.00	0.98	20
Failure	1.00	0.80	0.89	20	1.00	0.95	0.97	20
accuracy			0.90	40			0.97	40
macro avg	0.92	0.90	0.90	40	0.98	0.97	0.97	40
weighted avg	0.92	0.90	0.90	40	0.98	0.97	0.97	40

Table 4.9: Confusion Matrix model ST4000DM000

		SVM		**RANDOM FOREST**	
		Predicted Class		**Predicted Class**	
		Healthy	*Failure*	*Healthy*	*Failure*
Actual Class	*Healthy*	20	0	20	0
	Failure	4	16	1	19

Table 4.10: Metrics Results for model ST8000NM0055

	SVM				**RANDOM FOREST**			
	precision	**recall**	**f1-score**	**support**	**precision**	**recall**	**f1-score**	**support**
Healthy	0.95	1.00	0.98	20	1.00	1.00	1.00	20
Failure	1.00	0.95	0.98	21	1.00	1.00	1.00	21
accuracy			0.98	41			1.00	41
macro avg	0.98	0.98	0.98	41	1.00	1.00	1.00	41
weighted avg	0.98	0.98	0.98	41	1.00	1.00	1.00	41

Table 4.11: Confusion Matrix model ST8000NM0055

		SVM		**RANDOM FOREST**	
		Predicted Class		**Predicted Class**	
		Healthy	*Failure*	*Healthy*	*Failure*
Actual Class	*Healthy*	20	0	20	0
	Failure	1	20	0	21

Table 4.12: Metrics Results for model ST12000NM0008

	SVM				**RANDOM FOREST**			
	precision	**recall**	**f1-score**	**support**	**precision**	**recall**	**f1-score**	**support**
Healthy	0.40	0.40	0.40	5	0.50	0.40	0.44	5
Failure	0.25	0.25	0.25	4	0.40	0.50	0.44	4
accuracy			0.33	9			0.44	9
macro avg	0.33	0.33	0.33	9	0.45	0.45	0.44	9
weighted avg	0.33	0.33	0.33	9	0.46	0.44	0.44	9

Table 4.13: Confusion Matrix model ST12000NM0008

		SVM		**RANDOM FOREST**	
		Predicted Class		**Predicted Class**	
		Healthy	*Failure*	*Healthy*	*Failure*
Actual Class	*Healthy*	2	3	2	3
	Failure	3	1	2	2

Table 4.14: Metrics Results for model TOSHIBA MQ01ABF050

	SVM				**RANDOM FOREST**			
	precision	**recall**	**f1-score**	**support**	**precision**	**recall**	**f1-score**	**support**
Healthy	0.80	0.80	0.80	10	0.91	1.00	0.95	10
Failure	0.82	0.82	0.82	11	1.00	0.91	0.95	11
accuracy			0.81	21			0.95	21
macro avg	0.81	0.81	0.81	21	0.95	0.95	0.95	21
weighted avg	0.81	0.81	0.81	21	0.96	0.95	0.95	21

Table 4.15: Confusion Matrix model TOSHIBA MQ01ABF050

		SVM		**RANDOM FOREST**	
		Predicted Class		**Predicted Class**	
		Healthy	*Failure*	*Healthy*	*Failure*
Actual Class	*Healthy*	8	2	10	0
	Failure	2	9	1	10

Table 4.16: Metrics Results for model TOSHIBA MG07ACA14TA

	SVM				**RANDOM FOREST**			
	precision	**recall**	**f1-score**	**support**	**precision**	**recall**	**f1-score**	**support**
Healthy	1.00	0.50	0.67	2	1.00	0.50	0.67	2
Failure	0.67	1.00	0.80	2	0.67	1.00	0.80	2
accuracy			0.75	4			0.75	4
macro avg	0.83	0.75	0.73	4	0.83	0.75	0.73	4
weighted avg	0.83	0.75	0.73	4	0.83	0.75	0.73	4

Table 4.17: Confusion Matrix model TOSHIBA MG07ACA14TA

		SVM		**RANDOM FOREST**	
		Predicted Class		**Predicted Class**	
		Healthy	*Failure*	*Healthy*	*Failure*
Actual Class	*Healthy*	1	1	1	1
	Failure	0	2	0	2

It is possible to observe in Table 4.6 that the ST12000NM0007 model shows metrics values really close to 100%, this can demonstrate that the methodology used may prove to be quite accurate. It is important to note that the ST12000NM0008, TOSHIBA MQ01ABF050 and TOSHIBA MG07ACA14TA models, do not have a favorable support (very low number of observations and past behaviors information) for the algorithms execution, and therefore their results are not the most promising. In the future work section, some points that could improve these results are discussed.

It is essential to have a perception of the importance that the Random Forest model gives to variables in its decision making and in the trees creation. With this, it is easier to understand which features weight more in helping the algorithm to predict if the disk will fail or remain healthy. Table 4.18 shows the importance ranking given to the 6 different models.

Table 4.18: Random Forest Features Importance for each disk model

ST12000NM0007	
	importance
smart_7_normalized	0.228980
smart_193_raw	0.187561
smart_3_normalized	0.165387
smart_9_normalized	0.058750
smart_9_raw	0.057293
smart_241_raw	0.051673
smart_7_raw	0.041517
smart_240_raw	0.032224
smart_12_raw	0.026624
smart_242_raw	0.026087

ST4000DM000	
	importance
smart_193_raw	0.156771
smart_183_raw	0.127605
smart_3_normalized	0.099825
smart_190_normalized	0.083486
smart_183_normalized	0.080752
smart_194_raw	0.063729
smart_194_normalized	0.061899
smart_190_raw	0.056995
smart_7_normalized	0.046583
smart_240_raw	0.045435

ST8000NM0055	
	importance
smart_195_normalized	0.207715
smart_1_normalized	0.182538
smart_193_normalized	0.126374
smart_193_raw	0.075515
smart_191_raw	0.066237
smart_7_normalized	0.060627
smart_191_normalized	0.053196
smart_192_raw	0.043330
smart_190_raw	0.022044
smart_194_normalized	0.021385

ST12000NM0008	
	importance
smart_190_raw	0.165062
smart_194_normalized	0.146380
smart_190_normalized	0.139104
smart_194_raw	0.130639
smart_192_raw	0.061040
smart_1_raw	0.059038
smart_1_normalized	0.050796
smart_240_raw	0.038349
smart_7_raw	0.037081
smart_9_raw	0.036924

TOSHIBA MQ01ABF050	
	importance
smart_191_raw	0.400617
smart_194_raw	0.286671
smart_9_raw	0.134172
smart_222_raw	0.125146
smart_222_normalized	0.028102
smart_9_normalized	0.025292

TOSHIBA MG07ACA14TA	
	importance
smart_226_raw	0.278620
smart_222_raw	0.193245
smart_9_raw	0.181570
smart_220_raw	0.145886
smart_194_raw	0.104060
smart_193_raw	0.096619

Variables like smart_7, smart_9, and smart_193 have a high importance in almost all disk models, so trying to monitor them more often, is probably a good approach in the future. As mentioned in Chapter 2, the S.M.A.R.T. variables that are considered critical by the literature, are the ones that the storage systems follow the most. So giving more attention to these features could also help preventing some misbehavior.

Figure 4.4 shows one of the trees created after the Random Forest algorithm was executed in the ST12000NM0007 dataset. As is normal, the variables presented in the Decision Tree are shown in the importance table, showing that the algorithm uses them to classify the observations. In Table 4.19 the variables presented on the Decision Tree are described. It is important to note that this description was made after the variables were already normalized between 0 and 1.

Table 4.19: Decision Tree variables description

	smart__7__normalized	smart__7__raw	smart__9__normalized	smart__9__raw	smart__241__raw
count	699.000000	699.000000	699.000000	699.000000	699.000000
mean	0.705797	0.476673	0.270684	0.739938	0.769213
std	0.184539	0.306533	0.301359	0.301370	0.266498
min	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.600000	0.178866	0.100000	0.737158	0.804327
50%	0.685714	0.476555	0.150000	0.847509	0.853286
75%	0.900000	0.772911	0.263158	0.939120	0.925315
max	1.000000	1.000000	1.000000	1.000000	1.000000

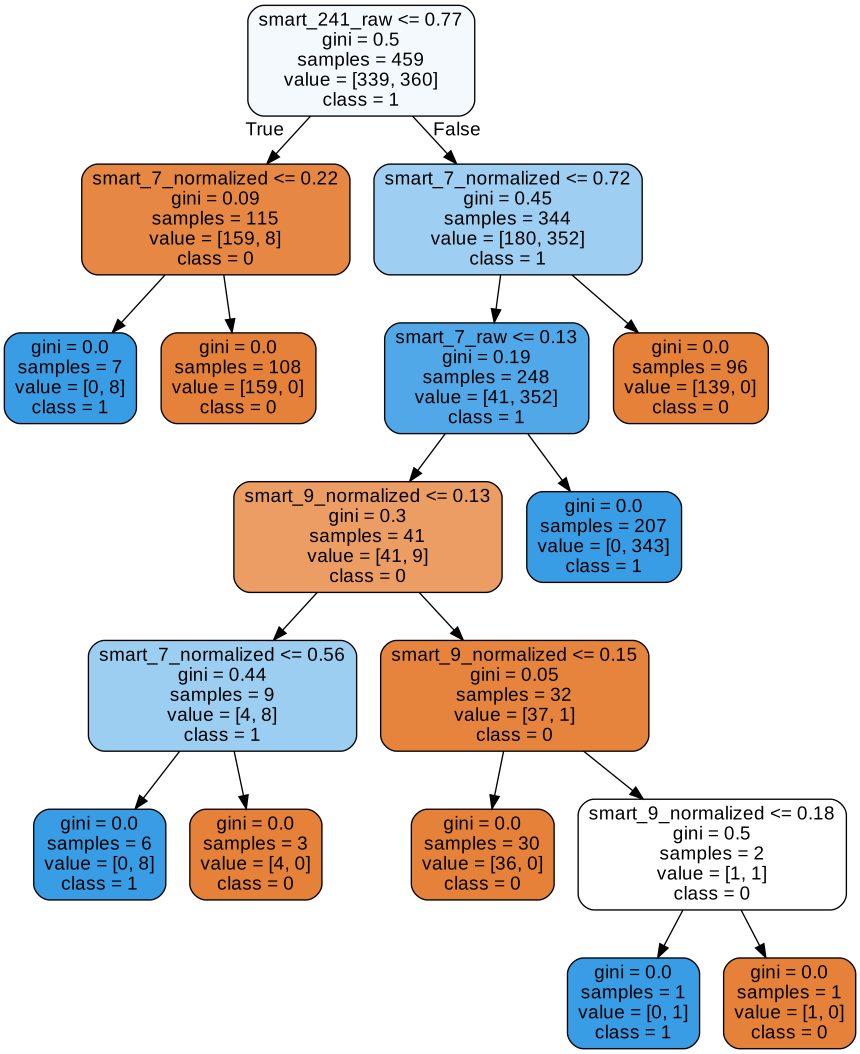


Figure 4.4: A Decision Tree from the Random Forest of the ST12000NM0007 model

4.4 VAR Model

As stated in the methodology, 32 subdatasets were created to apply this time series model, all of them grouped by serial_number, so the algorithm could be applied individually and could calculate the respective correlation matrices and a possible prediction of the variables behavior over time.

From Table 4.20 to Table 4.27 the correlation matrices from 4 disk models (one healthy and one failed) are presented. The selected disks from both Toshiba models (TOSHIBA MG07ACA14TA and TOSHIBA MQ01ABF050) could not provide enough information for the learning model to be executed, and therefore, their correlation matrices and the forecasting process were not built. This happened because all features that were filled with missing values were removed, as well as the ones that stay constant over the time, because the algorithm does not preform operations with them. Thus, there was not enough S.M.A.R.T. attributes to run the model.

Table 4.20: Correlation Matrix for failed disk from model ST12000NM0007

Failed Disk ST12000NM0007 (ZCH0C5JJ)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.253239	0.201611	-0.120629	-0.190959	0.187426
smart_7_raw	0.253239	1.000000	**0.852643**	**0.696433**	0.055252	**0.849855**
smart_9_raw	0.201611	**0.852643**	1.000000	**0.923364**	0.039857	**0.998544**
smart_193_raw	-0.120629	**0.696433**	**0.923364**	1.000000	-0.040797	**0.920185**
smart_194_raw	-0.190959	0.055252	0.039857	-0.040797	1.000000	0.068931
smart_240_raw	0.187426	**0.849855**	**0.998544**	**0.920185**	0.068931	1.000000

Table 4.21: Correlation Matrix for healthy disk from model ST12000NM0007

Healthy Disk ST12000NM0007 (ZCH06YQ3)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.166218	-0.085673	-0.141543	0.260900	-0.001269
smart_7_raw	0.166218	1.000000	0.351067	-0.135600	-0.120242	0.392243
smart_9_raw	-0.085673	0.351067	1.000000	0.254303	-0.457013	**0.992656**
smart_193_raw	-0.141543	-0.135600	0.254303	1.000000	-0.546709	0.235824
smart_194_raw	0.260900	-0.120242	-0.457013	-0.546709	1.000000	-0.406834
smart_240_raw	-0.001269	0.392243	**0.992656**	0.235824	-0.406834	1.000000

Table 4.22: Correlation Matrix for failed disk from model ST4000DM000

Failed Disk ST4000DM000 (Z302T88S)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.557678	0.641922	0.054301	0.027109	0.612161
smart_7_raw	0.557678	1.000000	**0.933525**	-0.456721	0.200886	**0.942728**
smart_9_raw	0.641922	**0.933525**	1.000000	-0.262591	0.029598	**0.998036**
smart_193_raw	0.054301	-0.456721	-0.262591	1.000000	-0.332683	-0.271897
smart_194_raw	0.027109	0.200886	0.029598	-0.332683	1.000000	0.036335
smart_240_raw	0.612161	**0.942728**	**0.998036**	-0.271897	0.036335	1.000000

Table 4.23: Correlation Matrix for healthy disk from model ST4000DM000

Healthy Disk ST4000DM000 (Z302DJZ6)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.105579	-0.208478	-0.153385	**0.724605**	-0.208246
smart_7_raw	-0.105579	1.000000	0.458922	-0.102037	-0.192726	0.471757
smart_9_raw	-0.208478	0.458922	1.000000	0.569640	-0.315389	**0.998916**
smart_193_raw	-0.153385	-0.102037	0.569640	1.000000	-0.432104	0.559510
smart_194_raw	**0.724605**	-0.192726	-0.315389	-0.432104	1.000000	-0.336655
smart_240_raw	-0.208246	0.471757	**0.998916**	0.559510	-0.336655	1.000000

Table 4.24: Correlation Matrix for failed disk from model ST8000NM0055

Failed Disk ST8000NM0055 (ZA1819DM)	***smart_1_raw***	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.257158	-0.061240	-0.286411	-0.268057	-0.022319
smart_7_raw	0.257158	1.000000	**0.830599**	-0.177065	-0.081025	**0.844100**
smart_9_raw	-0.061240	**0.830599**	1.000000	0.161528	0.070777	**0.995699**
smart_193_raw	-0.286411	-0.177065	0.161528	1.000000	0.370282	0.189769
smart_194_raw	-0.268057	-0.081025	0.070777	0.370282	1.000000	0.100202
smart_240_raw	-0.022319	**0.844100**	**0.995699**	0.189769	0.100202	1.000000

Table 4.25: Correlation Matrix for healthy disk from model ST8000NM0055

Healthy Disk ST8000NM0055 (ZA18BTFV)	***smart_1_raw***	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.455708	-0.264859	0.321201	0.271168	-0.249651
smart_7_raw	-0.455708	1.000000	**0.918986**	0.235787	-0.325022	**0.921027**
smart_9_raw	-0.264859	**0.918986**	1.000000	0.528043	-0.273634	**0.999511**
smart_193_raw	0.321201	0.235787	0.528043	1.000000	-0.302633	0.529167
smart_194_raw	0.271168	-0.325022	-0.273634	-0.302633	1.000000	-0.267527
smart_240_raw	-0.249651	**0.921027**	**0.999511**	0.529167	-0.267527	1.000000

Table 4.26: Correlation Matrix for failed disk from model ST12000NM0008

Failed Disk ST12000NM0008 (ZHZ3MSH6)	***smart_1_raw***	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.061050	0.395718	**0.741040**	**0.703804**	0.072431
smart_7_raw	0.061050	1.000000	**0.922226**	-0.374584	-0.446844	**0.961005**
smart_9_raw	0.395718	**0.922226**	1.000000	-0.097896	-0.156114	**0.927476**
smart_193_raw	**0.741040**	-0.374584	-0.097896	1.000000	**0.863018**	-0.457007
smart_194_raw	**0.703804**	-0.446844	-0.156114	**0.863018**	1.000000	-0.442773
smart_240_raw	0.072431	**0.961005**	**0.927476**	-0.457007	-0.442773	1.000000

Table 4.27: Correlation Matrix for healthy disk from model ST12000NM0008

Healthy Disk ST12000NM0008 (ZHZ3PT1S)	***smart_1_raw***	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.106104	-0.175308	-0.074616	0.041358	-0.206763
smart_7_raw	-0.106104	1.000000	**0.976768**	-0.217435	0.040426	**0.982719**
smart_9_raw	-0.175308	**0.976768**	1.000000	-0.054606	0.107858	**0.960382**
smart_193_raw	-0.074616	-0.217435	-0.054606	1.000000	0.578504	-0.306347
smart_194_raw	0.041358	0.040426	0.107858	0.578504	1.000000	-0.096563
smart_240_raw	-0.206763	**0.982719**	**0.960382**	-0.306347	-0.096563	1.000000

Undoubtedly, the choice of this model had as main objective the analysis of the correlation between variables. These correlations can be important, so the features are monitored together and not just those that present critical values because their thresholds were exceeded.

Variables 9 and 240 show in all disks (healthy and failed), correlations really close to 1. This is probably happening because both are hour counters, with 9 being the number of hours in power-state and 240 the time during the positioning of the drive heads.

It is possible to verify that higher correlations between variables (9-193) and (240-193) also happen more frequently in the disks that fail from ST12000NM0007 model. The high correlation between these variables, may alert that a more careful observation of the disks should be made. However, there are also healthy disks that show high correlations between these variables, but there is no guarantee that the disk will remain healthy in the future, so these disks may even already show some type of anomaly.

Even though there are not enough disks to draw a strong conclusion, the ST4000DM000 model shows correlations between the variables (9-7) and (240-7) for the disks that fail, and the ST12000NM0008 model with correlations in the variables (1-193), (1-194) and (193-194) also in the failing disks.

Finally, the forecasting was done to predict the disks behavior over the time. In Figures 4.5 and 4.7, inside the red dashed line, a 5-day forecast can be observed, for each variable, from a disk that fails and a disk considered healthy. In Figures 4.6 and 4.8, also inside the dashed line, it is possible to see how the disks actually performed in the last 5 days before being selected. The X axis represents a time scale, with daily periodicity and the Y axis represents the values for each variable. It is important to notice that these 5 days were removed from the datasets at the beginning of the learning model, so now that they could be compared with the respective forecasted values.

In short, good results were obtained, and it can be seen that the predicted results are not far from the real values.

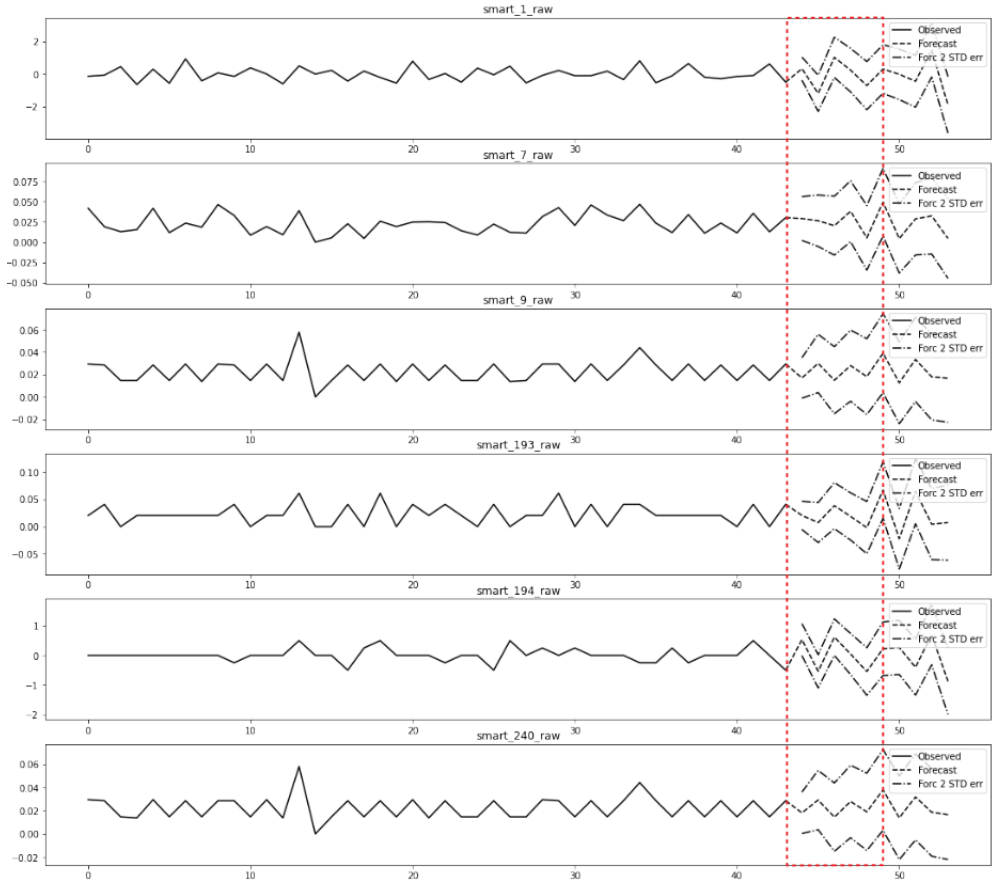


Figure 4.5: Forecast for the first failed disk with a FPE (failure prediction error) of $7x10^{-22}$

Figure 4.6: Real Values for the first failed disk

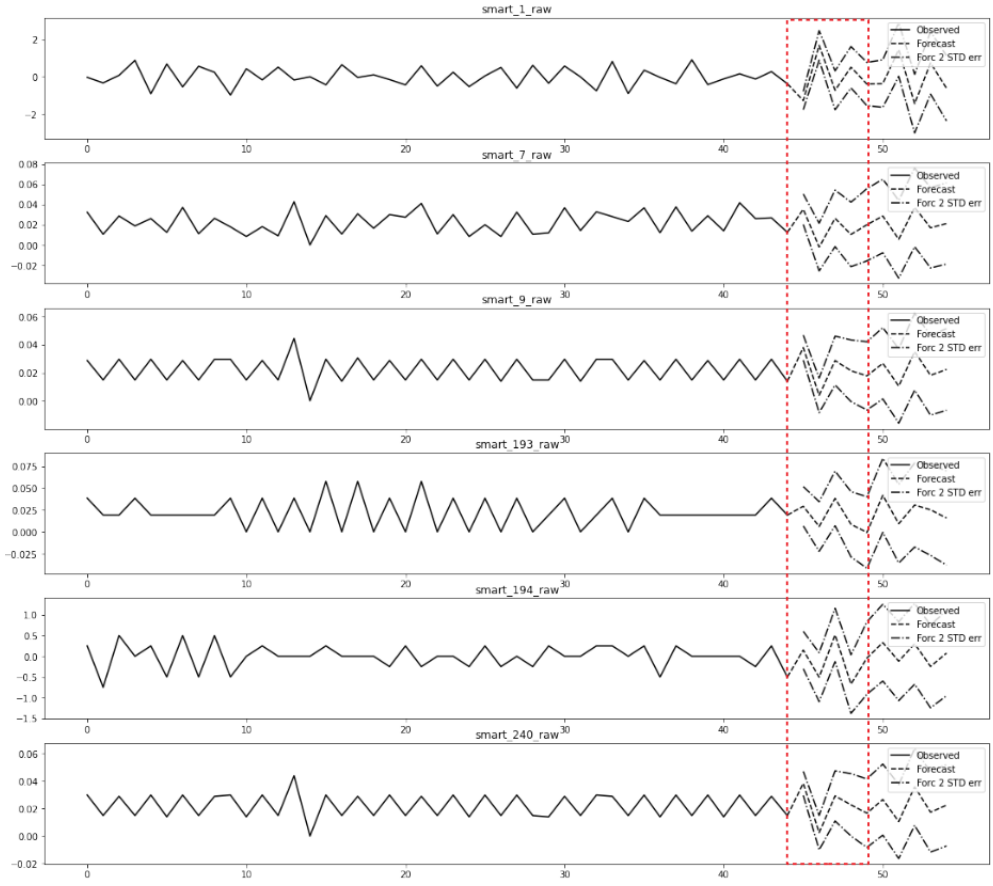


Figure 4.7: Forecast for the first healthy disk with a FPE (failure prediction error) of $4x10^{-22}$

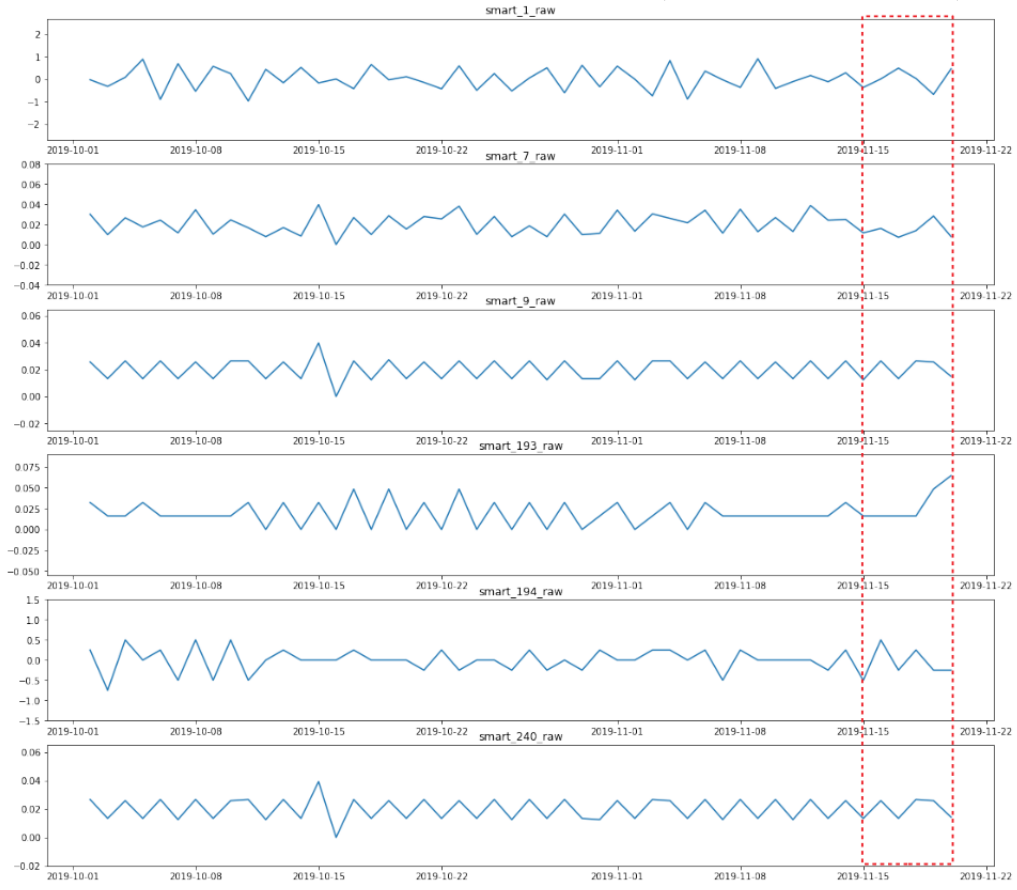


Figure 4.8: Real Values for the first healthy disk

Chapter 5

Conclusion and Future Work

The objective of the dissertation was to add, to the current state of the art, measures that would help to predict HDD's failures and anomalies, so the storage companies could reduce the problems resulting from this.

Working with imbalanced data reduces the effectiveness of prediction models. Because of this, it was necessary to take a very cautious approach to the data, so small information about the disks, that could be useful, was not lost during the process.

One of the biggest beliefs in this project was that the pre-processing and statistical analysis methods used to create the subdatasets were fundamental in the learning process of the data. Almost all studies made on the subject, analyze the disks together, without splitting them by models and vendors. This means that the variables standard values, the thresholds and even the features normalization process are not distinguished between them.

Although, most subdatasets created during the project, did not have the ideal number of observations, the metrics values for the classification models are quite promising, showing values really close to 100%, for the precision, recall and f-score of certain disk models.

The VAR model application, allowed to trace temporal correlation matrices between the features, showing that variables 9, 240 and 193 are related over time. The forecasting performed fulfilled the expectation, showing a relatively low forecasting error and may be an interesting method to predict the variables behavior in storage companies.

It should also be noticed that variables 7, 9 and 240 are present quite often in the results, and therefore, they must also be monitored carefully, together with the ones that are considered to be critical by the literature.

During the project, some obstacles appeared and had to be overcome. Working with such a large and imbalanced dataset was undoubtedly a great challenge, and helped to clarify the reality that in data science, the work is done with data that are not perfect to apply learning models. The lack of perception about the variables, the difference between vendors reference values and the amount of missing values presented in the dataset, made the decision making very

difficult, and many times, some methods had to be redone from scratch. Since the VAR is an algorithm that works with mathematical matrices, it proved to be a model with high complexity and that requires a lot of attention in the type of data that is used.

However, in overall, the objectives were achieved and the work carried out helped a lot in developing my knowledge in Data Science area and understanding how HDD's really work.

Although the objectives have been achieved, we believe that some measures can be taken to improve the results obtained and their reliability in future works:

- If a bigger time scale is used, the disk information will also be bigger and the learning models will probably prove to show a greater performance and better results for all disk models. This because, the processes executed by the algorithms, will have a greater support, and probably show that the methodology used in this work can also be used in datasets with a higher number of observations.
- A further study and analysis on the variables normalization should be done, in order to improve the state of the art and help future works to better understand the data.
- Use, in parallel with the methodology carried out in this project, a class variable that defines the disks lifetime. This classification can be done through variable 9, that counts the number of hours that the HDD was powered on.
- Apply the models built in this work to more recent Backblaze disks observations.

Appendix A

Subdatasets Description

Table 5.1: Failed Disks Dataset Description

		Smart_1	Smart_3	Smart_7	Smart_9	Smart_193	Smart_194
Disk1	count	46.0			46.0		46.0
	mean	79.6			80.6		37.1
	std	4.2			0.5		0.9
	min	67.0			80.0		35.0
	max	84.0			81.0		39.0
Disk2	count	46.0		46.0	46.0		46.0
	mean	79.6		83.1	90.3		23.9
	std	4.7		0.8	0.5		0.8
	min	67.0		82.0	90.0		22.0
	max	84.0		84.0	91.0		25.0
Disk3	count	46.0		46.0	46.0		46.0
	mean	80.4		87.3	87.5		24.4
	std	5.7		0.5	0.5		3.7
	min	69.0		87.0	87.0		20.0
	max	100.0		88.0	88.0		30.0
Disk4	count	46.0		46.0	46.0		46.0
	mean	79.2		89.5	80.6		35.3
	std	4.4		0.5	0.5		1.1
	min	64.0		89.0	80.0		33.0
	max	84.0		90.0	81.0		38.0
Disk5	count	45.0		45.0	45.0		45.0
	mean	80.3		80.6	83.5		21.9
	std	3.8		8.5	0.5		1.0
	min	65.0		60.0	83.0		20.0
	max	84.0		90.0	84.0		24.0
Disk6	count	46.0	46.0	46.0	46.0		46.0

Table 5.1: Failed Disks Dataset Description

		Smart_1	**Smart_3**	**Smart_7**	**Smart_9**	**Smart_193**	**Smart_194**
	mean	79.5	96.4	78.7	99.5		27.8
	std	4.4	1.5	4.6	0.5		0.8
	min	68.0	95.0	63.0	99.0		26.0
	max	84.0	98.0	83.0	100.0		29.0
	count	46.0		46.0	46.0		46.0
	mean	79.8		87.8	82.6		30.0
Disk7	std	3.9		0.4	0.5		0.9
	min	68.0		87.0	82.0		28.0
	max	84.0		88.0	83.0		32.0
	count	45.0		45.0	45.0		45.0
	mean	79.4		79.9	82.6		25.4
Disk8	std	5.6		1.7	0.5		0.7
	min	67.0		76.0	82.0		24.0
	max	100.0		82.0	83.0		27.0
	count	46.0		46.0	46.0		46.0
	mean	79.3		78.7	99.6		22.8
Disk9	std	4.4		4.8	0.5		1.1
	min	69.0		63.0	99.0		21.0
	max	84.0		83.0	100.0		25.0
	count	45.0		45.0	45.0		45.0
	mean	115.2		74.8	53.4		17.1
Disk10	std	4.0		4.9	0.5		0.4
	min	102.0		63.0	53.0		16.0
	max	120.0		90.0	54.0		18.0
	count	45.0			45.0		45.0
	mean	114.4			63.2		33.6
Disk11	std	4.8			0.4		1.0
	min	102.0			62.0		32.0
	max	120.0			64.0		35.0
	count	45.0		45.0	45.0	45.0	45.0
	mean	80.1		88.3	78.7	88.3	33.6
Disk12	std	3.5		0.7	0.4	0.4	0.6
	min	69.0		87.0	78.0	88.0	33.0
	max	84.0		89.0	79.0	89.0	35.0
	count	46.0		46.0	46.0	46.0	46.0
	mean	79.8		90.7	77.5	96.2	36.0
Disk13	std	3.7		0.5	0.5	0.4	0.9
	min	70.0		90.0	77.0	96.0	35.0

Table 5.1: Failed Disks Dataset Description

		Smart_1	**Smart_3**	**Smart_7**	**Smart_9**	**Smart_193**	**Smart_194**
	max	84.0		91.0	78.0	97.0	38.0
Disk14	count	17.0		17.0			17.0
	mean	80.4		71.6			30.6
	std	3.4		5.3			2.4
	min	73.0		63.0			25.0
	max	83.0		77.0			32.0
Disk15	count				17.0		
	mean				100.0		
	std				0.0		
	min				100.0		
	max				100.0		
Disk16	count						
	mean						
	std						
	min						
	max						

Table 5.2: Healthy Disks Dataset Description

		Smart_1	**Smart_3**	**Smart_7**	**Smart_9**	**Smart_193**	**Smart_194**
Disk1	count	45.0	45.0		45.0		45.0
	mean	80.4	88.8		85.9		26.9
	std	4.7	1.0		0.5		1.2
	min	65.0	88.0		85.0		25.0
	max	100.0	90.0		87.0		29.0
Disk2	count	45.0	45.0	45.0	45.0		45.0
	mean	80.2	91.3	80.8	82.9		26.5
	std	3.9	1.4	1.3	0.4		1.0
	min	65.0	90.0	78.0	82.0		22.0
	max	84.0	93.0	83.0	84.0		28.0
Disk3	count	45.0		45.0	45.0		45.0
	mean	81.3		84.4	82.9		37.7
	std	5.0		1.1	0.5		1.3
	min	72.0		83.0	82.0		35.0
	max	100.0		86.0	84.0		40.0
Disk4	count	46.0		46.0	46.0		46.0
	mean	81.7		89.1	85.2		36.5
	std	4.9		0.3	0.4		1.1
	min	73.0		89.0	85.0		35.0

Table 5.2: Healthy Disks Dataset Description

		Smart_1	**Smart_3**	**Smart_7**	**Smart_9**	**Smart_193**	**Smart_194**
	max	100.0		90.0	86.0		39.0
Disk5	count	46.0	46.0	46.0	46.0		46.0
	mean	79.9	89.6	88.3	82.6		30.6
	std	3.5	0.5	0.5	0.5		1.5
	min	68.0	89.0	88.0	82.0		28.0
	max	84.0	90.0	89.0	83.0		33.0
Disk6	count	45.0		45.0	45.0		45.0
	mean	80.0		86.2	82.6		26.4
	std	4.0		0.6	0.5		1.3
	min	67.0		85.0	82.0		24.0
	max	84.0		87.0	83.0		29.0
Disk7	count	45.0	45.0	45.0	45.0		45.0
	mean	79.4	89.8	85.8	82.3		30.7
	std	3.8	0.4	0.6	0.5		0.8
	min	69.0	89.0	85.0	82.0		29.0
	max	84.0	90.0	87.0	83.0		32.0
Disk8	count	46.0	46.0	46.0	46.0		46.0
	mean	79.4	89.6	88.2	85.2		32.8
	std	3.6	0.9	0.4	0.4		2.6
	min	68.0	89.0	88.0	85.0		27.0
	max	84.0	91.0	89.0	86.0		37.0
Disk9	count	10.0	10.0	10.0			10.0
	mean	80.5	97.8	71.8			23.3
	std	1.8	1.8	10.1			0.5
	min	78.0	93.0	65.0			23.0
	max	83.0	99.0	100.0			24.0
Disk10	count	45.0		45.0	45.0		45.0
	mean	115.3		74.1	59.3		28.5
	std	4.3		3.9	0.5		1.7
	min	102.0		62.0	59.0		26.0
	max	120.0		78.0	60.0		33.0
Disk11	count	45.0		45.0	45.0		45.0
	mean	115.9		88.7	60.6		21.6
	std	3.3		0.4	0.5		0.5
	min	108.0		88.0	60.0		20.0
	max	120.0		89.0	61.0		22.0
Disk12	count	46.0		46.0	46.0	46.0	46.0
	mean	80.8		92.1	78.3	95.7	46.3
	std	2.5		0.3	0.5	0.5	0.8

Table 5.2: Healthy Disks Dataset Description

		Smart_1	**Smart_3**	**Smart_7**	**Smart_9**	**Smart_193**	**Smart_194**
	min	76.0		92.0	78.0	95.0	45.0
	max	84.0		93.0	79.0	96.0	48.0
	count	46.0		46.0	46.0		46.0
	mean	80.2		92.4	78.3		39.4
Disk13	std	3.4		0.5	0.5		0.8
	min	71.0		92.0	78.0		38.0
	max	84.0		93.0	79.0		41.0
	count	17.0		17.0			17.0
	mean	80.2		71.8			24.6
Disk14	std	3.6		5.1			1.5
	min	74.0		63.0			21.0
	max	84.0		77.0			26.0
	count				17.0		
	mean				100.0		
Disk15	std				0.0		
	min				100.0		
	max				100.0		
	count						
	mean						
Disk16	std						
	min						
	max						

Temporal Analysis

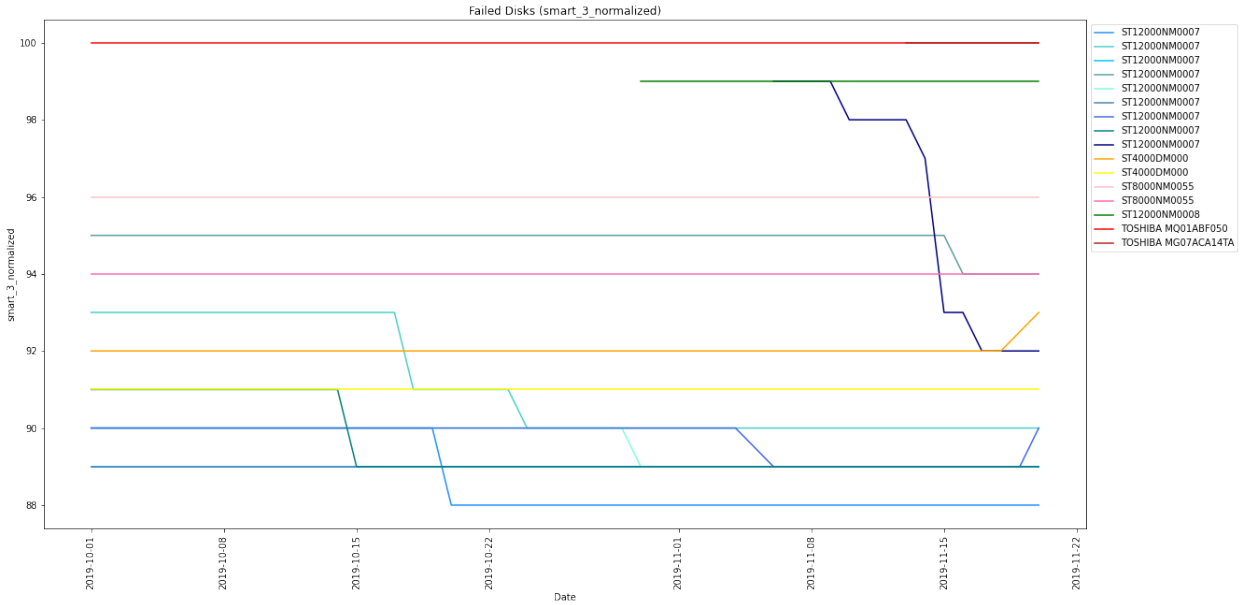


Figure 5.1: Smart_3 for the failed disks along time

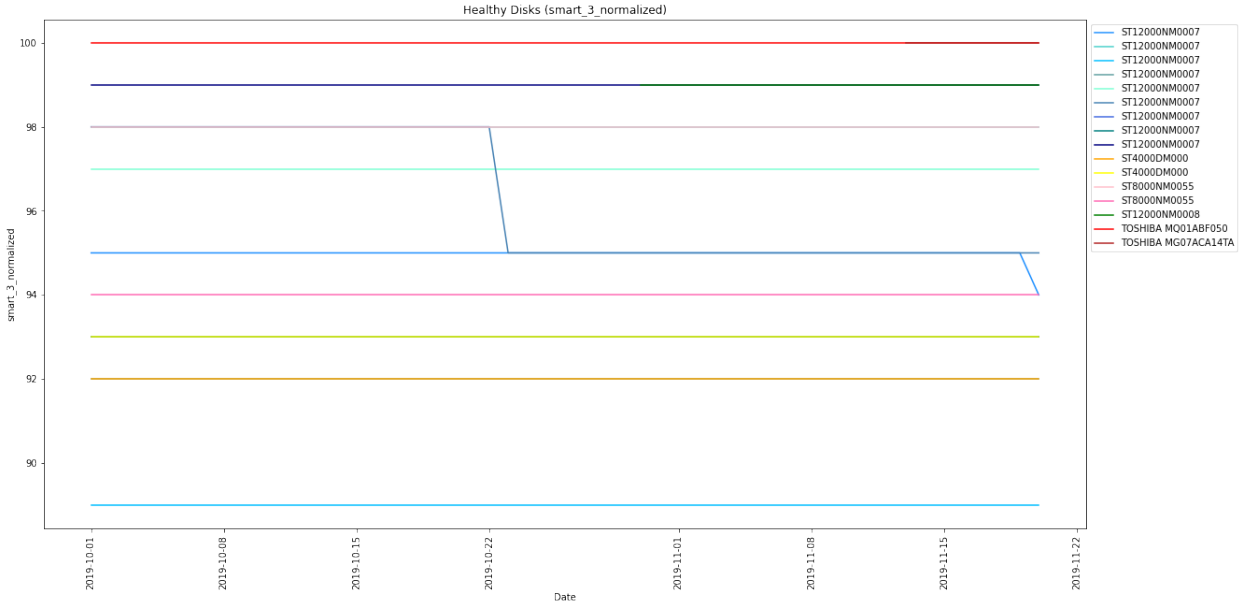


Figure 5.2: Smart_3 for the healthy disks along time

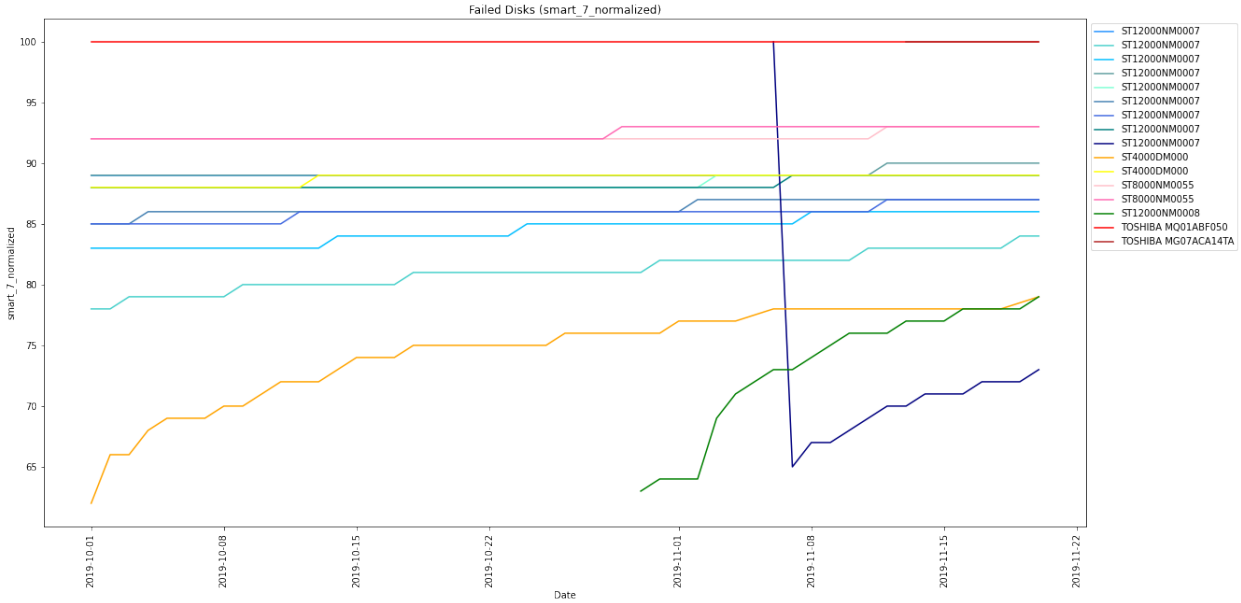


Figure 5.3: Smart_7 for the failed disks along time

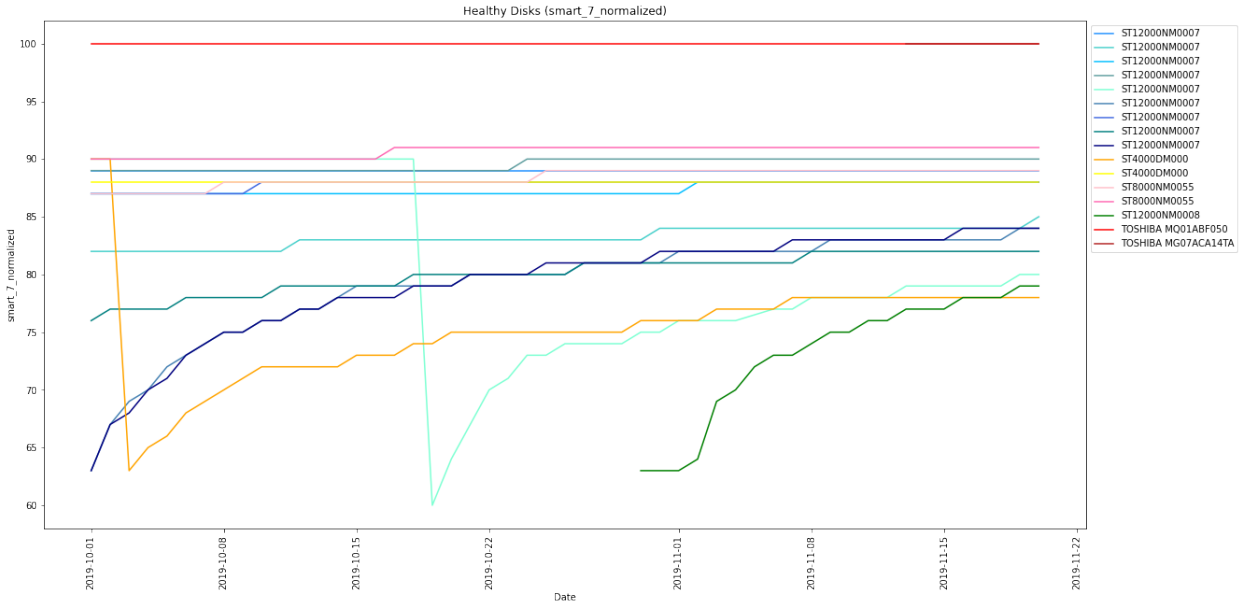


Figure 5.4: Smart_7 for the healthy disks along time

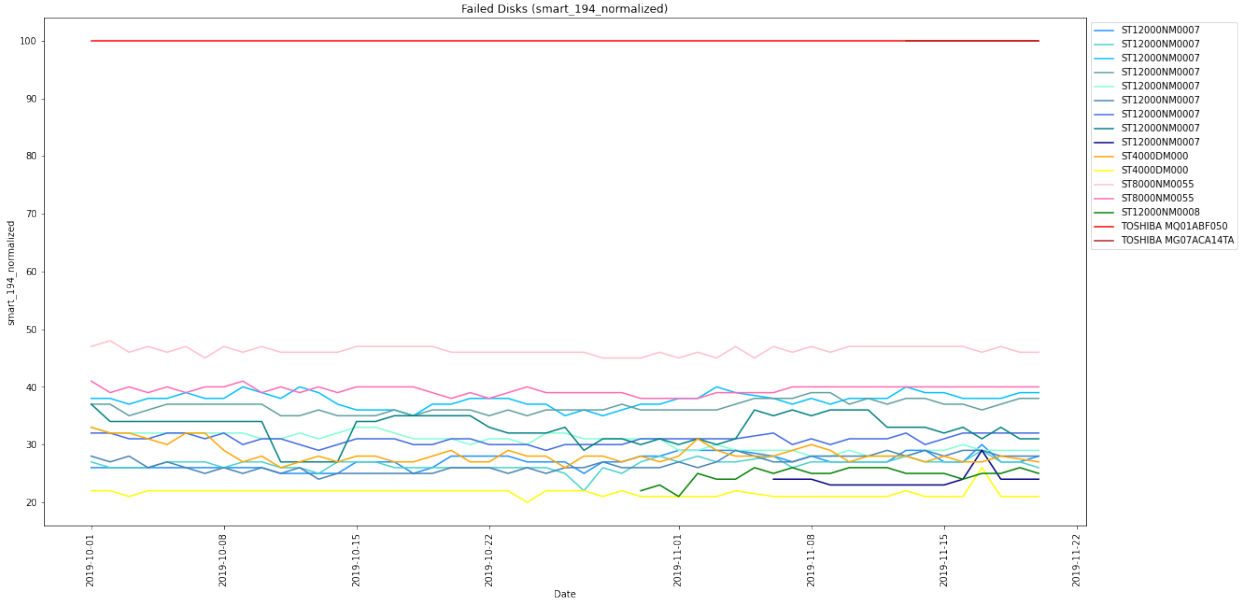


Figure 5.5: Smart_194 for the failed disks along time

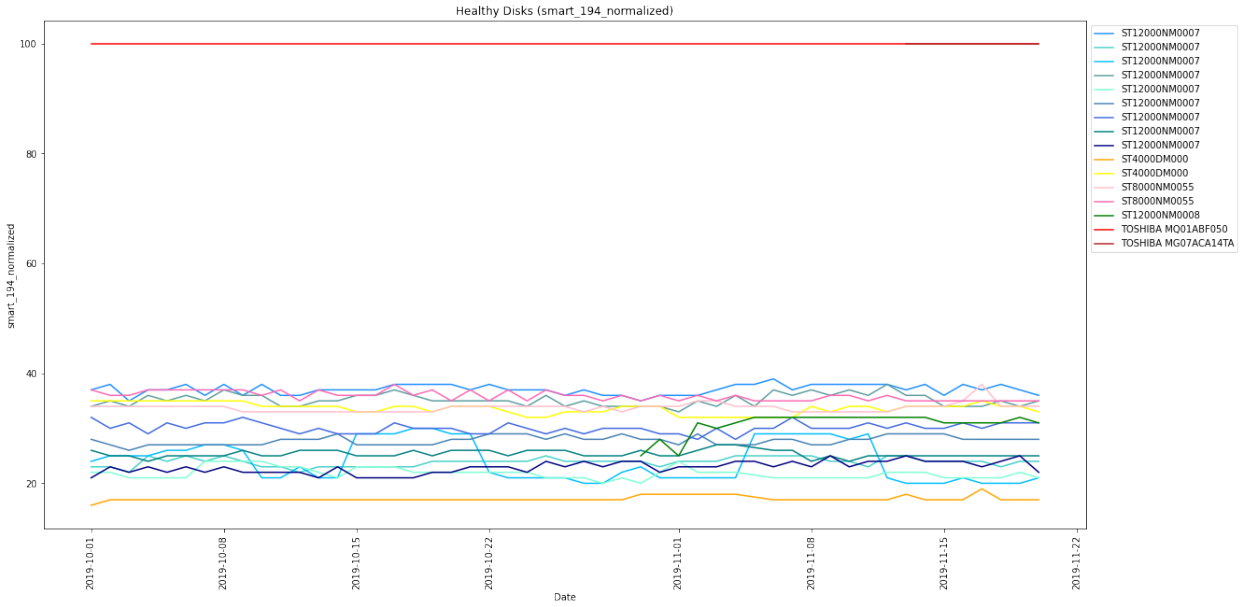


Figure 5.6: Smart_194 for the healthy disks along time

Correlation Matrices

Table 5.3: Correlation Matrix for failed disk from model ST12000NM0007

Failed Disk ST12000NM0007 (ZCH0A7G6)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.326879	0.470757	0.396524	0.429924	0.456820
smart_7_raw	0.326879	1.000000	0.539748	0.128281	0.585176	0.519329
smart_9_raw	0.470757	0.539748	1.000000	-0.020046	0.266706	**0.997736**
smart_193_raw	0.396524	0.128281	-0.020046	1.000000	0.335702	-0.079894
smart_194_raw	0.429924	0.585176	0.266706	0.335702	1.000000	0.228991
smart_240_raw	0.456820	0.519329	**0.997736**	-0.079894	0.228991	1.000000

Table 5.4: Correlation Matrix for healthy disk from model ST12000NM0007

Healthy Disk ST12000NM0007 (ZCH056VR)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.291492	0.040868	0.198374	-0.004008	0.046690
smart_7_raw	-0.291492	1.000000	**0.722485**	0.455088	-0.064878	**0.713665**
smart_9_raw	0.040868	**0.722485**	1.000000	0.642411	-0.326361	**0.999289**
smart_193_raw	0.198374	0.455088	0.642411	1.000000	-0.510889	0.642498
smart_194_raw	-0.004008	-0.064878	-0.326361	-0.510889	1.000000	-0.313906
smart_240_raw	0.046690	**0.713665**	**0.999289**	0.642498	-0.313906	1.000000

Table 5.5: Correlation Matrix for failed disk from model ST12000NM0007

Failed Disk ST12000NM0007 (ZCH0AL23)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_192_raw**	**smart_193_raw**	**smart_240_raw**
smart_1_raw	1.000000	**0.743022**	0.503125	0.451996	0.344509	0.513814
smart_7_raw	**0.743022**	1.000000	0.500417	0.405775	0.167038	0.486951
smart_9_raw	0.503125	0.500417	1.000000	0.601211	**0.839095**	**0.999076**
smart_192_raw	0.451996	0.405775	0.601211	1.000000	**0.759134**	0.614697
smart_193_raw	0.344509	0.167038	**0.839095**	**0.759134**	1.000000	**0.851500**
smart_240_raw	0.513814	0.486951	**0.999076**	0.614697	**0.851500**	1.000000

Table 5.6: Correlation Matrix for healthy disk from model ST12000NM0007

Healthy Disk ST12000NM0007 (ZJV10J45)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.155619	-0.166672	-0.016085	-0.659190	-0.170055
smart_7_raw	-0.155619	1.000000	0.173853	-0.020315	0.416513	0.191309
smart_9_raw	-0.166672	0.173853	1.000000	0.575493	-0.062259	**0.991522**
smart_193_raw	-0.016085	-0.020315	0.575493	1.000000	-0.311858	0.543125
smart_194_raw	-0.659190	0.416513	-0.062259	-0.311858	1.000000	-0.079599
smart_240_raw	-0.170055	0.191309	**0.991522**	0.543125	-0.079599	1.000000

Table 5.7: Correlation Matrix for failed disk from model ST12000NM0007

Failed Disk ST12000NM0007 (ZJV03NQB)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.544369	0.353490	-0.074066	0.373296	0.375813
smart_7_raw	0.544369	1.000000	-0.305087	0.011450	0.420312	-0.316823
smart_9_raw	0.353490	-0.305087	1.000000	0.018962	0.042215	**0.993542**
smart_193_raw	-0.074066	0.011450	0.018962	1.000000	0.047066	-0.027482
smart_194_raw	0.373296	0.420312	0.042215	0.047066	1.000000	0.067022
smart_240_raw	0.375813	-0.316823	**0.993542**	-0.027482	0.067022	1.000000

Table 5.8: Correlation Matrix for healthy disk from model ST12000NM0007

Healthy Disk ST12000NM0007 (ZCH06HY1)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	**-0.721362**	0.020591	**-0.832200**	0.202707	0.005923
smart_7_raw	**-0.721362**	1.000000	0.047877	**0.809806**	-0.043604	0.053896
smart_9_raw	0.020591	0.047877	1.000000	-0.087280	-0.063281	**0.991122**
smart_193_raw	**-0.832200**	**0.809806**	-0.087280	1.000000	-0.129605	-0.072844
smart_194_raw	0.202707	-0.043604	-0.063281	-0.129605	1.000000	0.016449
smart_240_raw	0.005923	0.053896	**0.991122**	-0.072844	0.016449	1.000000

Table 5.9: Correlation Matrix for failed disk from model ST12000NM0007

Failed Disk ST12000NM0007 (ZCH097GA)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.090013	-0.559926	-0.414314	-0.284392	-0.554120
smart_7_raw	-0.090013	1.000000	0.665603	0.419819	**0.818138**	0.673308
smart_9_raw	-0.559926	0.665603	1.000000	**0.842269**	0.640534	**0.999752**
smart_193_raw	-0.414314	0.419819	**0.842269**	1.000000	0.207130	**0.839416**
smart_194_raw	-0.284392	**0.818138**	0.640534	0.207130	1.000000	0.649345
smart_240_raw	-0.554120	0.673308	**0.999752**	**0.839416**	0.649345	1.000000

Table 5.10: Correlation Matrix for healthy disk from model ST12000NM0007

Healthy Disk ST12000NM0007 (ZCH0BCML)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.265662	0.288319	-0.222821	-0.149011	0.285033
smart_7_raw	0.265662	1.000000	-0.147119	-0.247221	**-0.699470**	-0.170810
smart_9_raw	0.288319	-0.147119	1.000000	**0.694975**	0.378101	**0.997797**
smart_193_raw	-0.222821	-0.247221	**0.694975**	1.000000	0.449099	0.677031
smart_194_raw	-0.149011	**-0.699470**	0.378101	0.449099	1.000000	0.412304
smart_240_raw	0.285033	-0.170810	**0.997797**	0.677031	0.412304	1.000000

Table 5.11: Correlation Matrix for failed disk from model ST12000NM0007

Failed Disk ST12000NM0007 (ZJV00F20)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.626566	-0.172211	-0.556303	0.240412	-0.109197
smart_7_raw	-0.626566	1.000000	0.494762	0.598639	0.003462	0.468865
smart_9_raw	-0.172211	0.494762	1.000000	0.531663	-0.138053	**0.996179**
smart_193_raw	-0.556303	0.598639	0.531663	1.000000	-0.299230	0.488389
smart_194_raw	0.240412	0.003462	-0.138053	-0.299230	1.000000	-0.085365
smart_240_raw	-0.109197	0.468865	**0.996179**	0.488389	-0.085365	1.000000

Table 5.12: Correlation Matrix for healthy disk from model ST12000NM0007

Healthy Disk ST12000NM0007 (ZJV501TY)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.221391	-0.503270	-0.396752	0.235408	-0.513900
smart_7_raw	-0.221391	1.000000	**0.885034**	0.088399	-0.281677	**0.896764**
smart_9_raw	-0.503270	**0.885034**	1.000000	0.032943	-0.346043	**0.998117**
smart_193_raw	-0.396752	0.088399	0.032943	1.000000	-0.007978	0.048129
smart_194_raw	0.235408	-0.281677	-0.346043	-0.007978	1.000000	-0.362542
smart_240_raw	-0.513900	**0.896764**	**0.998117**	0.048129	-0.362542	1.000000

Table 5.13: Correlation Matrix for failed disk from model ST12000NM0007

Failed Disk ST12000NM0007 (ZJV00C88)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.550420	**0.768246**	**0.703953**	-0.156656	**0.764068**
smart_7_raw	0.550420	1.000000	**0.923780**	**0.888632**	-0.435427	**0.921210**
smart_9_raw	**0.768246**	**0.923780**	1.000000	**0.933976**	-0.430801	**0.998644**
smart_193_raw	**0.703953**	**0.888632**	**0.933976**	1.000000	-0.407332	**0.916139**
smart_194_raw	-0.156656	-0.435427	-0.430801	-0.407332	1.000000	-0.425609
smart_240_raw	**0.764068**	**0.921210**	**0.998644**	**0.916139**	-0.425609	1.000000

Table 5.14: Correlation Matrix for healthy disk from model ST12000NM0007

Healthy Disk ST12000NM0007 (ZCH0CDWV)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.581867	-0.192267	0.126343	0.257818	-0.175353
smart_7_raw	-0.581867	1.000000	0.184252	0.041591	-0.142339	0.168532
smart_9_raw	-0.192267	0.184252	1.000000	**0.805019**	-0.560167	**0.998472**
smart_193_raw	0.126343	0.041591	**0.805019**	1.000000	-0.517995	**0.793278**
smart_194_raw	0.257818	-0.142339	-0.560167	-0.517995	1.000000	-0.547501
smart_240_raw	-0.175353	0.168532	**0.998472**	**0.793278**	-0.547501	1.000000

Table 5.15: Correlation Matrix for failed disk from model ST12000NM0007

Failed Disk ST12000NM0007 (ZJV03JDV)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.037589	-0.556252	-0.159486	-0.407908	-0.525013
smart_7_raw	-0.037589	1.000000	0.283011	0.049724	0.168751	0.261249
smart_9_raw	-0.556252	0.283011	1.000000	0.495007	0.422434	**0.998160**
smart_193_raw	-0.159486	0.049724	0.495007	1.000000	0.202369	0.487922
smart_194_raw	-0.407908	0.168751	0.422434	0.202369	1.000000	0.410003
smart_240_raw	-0.525013	0.261249	**0.998160**	0.487922	0.410003	1.000000

Table 5.16: Correlation Matrix for healthy disk from model ST12000NM0007

Healthy Disk ST12000NM0007 (ZCH0D2V0)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.041900	0.071323	0.507615	-0.437712	0.077877
smart_7_raw	-0.041900	1.000000	0.645153	0.122852	-0.205947	**0.696068**
smart_9_raw	0.071323	0.645153	1.000000	0.618001	-0.115715	**0.994133**
smart_193_raw	0.507615	0.122852	0.618001	1.000000	-0.289708	0.625184
smart_194_raw	-0.437712	-0.205947	-0.115715	-0.289708	1.000000	-0.140796
smart_240_raw	0.077877	**0.696068**	**0.994133**	0.625184	-0.140796	1.000000

Table 5.17: Correlation Matrix for failed disk from model ST4000DM000

Failed Disk ST4000DM000 (S301P6Y6)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.453119	0.366908	0.018715	0.365931
smart_7_raw	0.453119	1.000000	0.568157	-0.161762	0.558772
smart_9_raw	0.366908	0.568157	1.000000	-0.230670	**0.999374**
smart_194_raw	0.018715	-0.161762	-0.230670	1.000000	-0.234252
smart_240_raw	0.365931	0.558772	**0.999374**	-0.234252	1.000000

Table 5.18: Correlation Matrix for healthy disk from model ST4000DM000

Healthy Disk ST4000DM000 (Z305D2CY)	**smart_1_raw**	**smart_7_raw**	**smart_9_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.285876	0.242102	-0.167771	0.237615
smart_7_raw	-0.285876	1.000000	0.379360	-0.150866	0.374310
smart_9_raw	0.242102	0.379360	1.000000	-0.228496	**0.999596**
smart_194_raw	-0.167771	-0.150866	-0.228496	1.000000	-0.233356
smart_240_raw	0.237615	0.374310	**0.999596**	-0.233356	1.000000

Table 5.19: Correlation Matrix for failed disk from model ST8000NM0055

Failed Disk ST8000NM0055 (ZA17ZNQ9)	***smart_1_raw***	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	0.141209	0.133120	0.681647	0.148984	0.146289
smart_7_raw	0.141209	1.000000	**0.974040**	0.146295	0.473009	**0.972681**
smart_9_raw	0.133120	**0.974040**	1.000000	0.256401	0.485871	**0.999378**
smart_193_raw	0.681647	0.146295	0.256401	1.000000	0.421099	0.266986
smart_194_raw	0.148984	0.473009	0.485871	0.421099	1.000000	0.501445
smart_240_raw	0.146289	**0.972681**	**0.999378**	0.266986	0.501445	1.000000

Table 5.20: Correlation Matrix for healthy disk from model ST8000NM0055

Healthy Disk ST8000NM0055 (ZA16YG7B)	***smart_1_raw***	**smart_7_raw**	**smart_9_raw**	**smart_193_raw**	**smart_194_raw**	**smart_240_raw**
smart_1_raw	1.000000	-0.207226	-0.305566	-0.400377	-0.211327	-0.321603
smart_7_raw	-0.207226	1.000000	**0.907206**	-0.212340	-0.541860	**0.901361**
smart_9_raw	-0.305566	**0.907206**	1.000000	-0.216698	-0.650402	**0.996615**
smart_193_raw	-0.400377	-0.212340	-0.216698	1.000000	-0.038489	-0.259011
smart_194_raw	-0.211327	-0.541860	-0.650402	-0.038489	1.000000	-0.612920
smart_240_raw	-0.321603	**0.901361**	**0.996615**	-0.259011	-0.612920	1.000000

Bibliography

[1] Abdullah Al Mamun, GuoXiao Guo, and Chao Bi. *Hard disk drive: mechatronics and control*. CRC press, 2017.

[2] Australian Transport Assessment and Planning. 6. forecasting and evaluation, Oct 2019.

[3] Nicolas Aussel, Samuel Jaulin, Guillaume Gandon, Yohan Petetin, Eriza Fazli, and Sophie Chabridon. Predictive models of hard drive failures based on operational data. In *ICMLA 2017 : 16th IEEE International Conference On Machine Learning And Applications*, pages 619 – 625, Cancun, Mexico, 2017. IEEE Computer Society. doi:10.1109/ICMLA.2017.00-92.

[4] Backblaze. Hard drive data and stats, Sep 2020.

[5] Brian Beach. Hard drive smart stats, Sep 2020.

[6] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, page 144–152, New York, NY, USA, 1992. Association for Computing Machinery. ISBN: 089791497X. doi:10.1145/130385.130401.

[7] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, page 144–152, New York, NY, USA, 1992. Association for Computing Machinery. ISBN: 089791497X. doi:10.1145/130385.130401.

[8] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[9] Zbigniew Chlondowski. [link].

[10] Google Colaboratory. Google colaboratory.

[11] DeepAI. Evaluation metrics, May 2019.

[12] Philippe Esling and Carlos Agon. Time-series data mining. *ACM Comput. Surv.*, 45(1), December 2012. ISSN: 0360-0300. doi:10.1145/2379776.2379788.

[13] Usama M Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, et al. Knowledge discovery and data mining: Towards a unifying framework. In *KDD*, volume 96, pages 82–88, 1996.

[14] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.

[15] Douglas Holtz-Eakin, Whitney Newey, and Harvey S Rosen. Estimating vector autoregressions with panel data. *Econometrica: Journal of the econometric society*, pages 1371–1395, 1988.

[16] Song Huang, Shuwen Liang, Song Fu, Weisong Shi, Devesh Tiwari, and Hsing-bung(HB) Chen. Characterizing disk health degradation and proactively protecting against disk failures for reliable storage systems. pages 157–166, 06 2019. doi:10.1109/ICAC.2019.00027.

[17] Ponemon Institute. Cost of data center outages. 2016.

[18] Pier Paolo Ippolito. Support vector machines, Jun 2019.

[19] Jupyter. Jupyter, Sep 2020.

[20] KDnuggets. The 5 most useful techniques to handle imbalanced datasets.

[21] Will Koehrsen. Random forest simple explanation, Aug 2020.

[22] Jing Li, Rebecca J Stones, Gang Wang, Xiaoguang Liu, Zhongwei Li, and Ming Xu. Hard drive failure prediction using decision trees. *Reliability Engineering & System Safety*, 164: 55–65, 2017.

[23] Helmut Lütkepohl. *New introduction to multiple time series analysis*. Springer Science & Business Media, 2005.

[24] Seagate Product Marketing. Get smart for reliability. Technical report, Technical report, Seagate Technology Paper, 1999.

[25] MC.AI. Confusion matrix no more confusing, Sep 2018.

[26] Shay Palachy. Stationarity in time series analysis, Sep 2019.

[27] Eduardo Pinheiro, Wolf-Dietrich Weber, and Luiz André Barroso. Failure trends in a large disk drive population. 2007.

[28] Bianca Schroeder and Garth A. Gibson. Understanding disk failure rates: What does an mttf of 1,000,000 hours mean to you? *ACM Trans. Storage*, 3(3):8–es, October 2007. ISSN: 1553-3077. doi:10.1145/1288783.1288785.

[29] Seagate. How do i interpret smart diagnostic utilities results?: Seagate support us.

[30] Jing Shen, Jian Wan, Se-Jung Lim, and Lifeng Yu. Random-forest-based failure prediction for hard disk drives. *International Journal of Distributed Sensor Networks*, 14(11): 1550147718806480, 2018. doi:10.1177/1550147718806480.

[31] Jing Shen, Jian Wan, Se-Jung Lim, and Lifeng Yu. Random-forest-based failure prediction for hard disk drives. *International Journal of Distributed Sensor Networks*, 14(11): 1550147718806480, 2018. doi:10.1177/1550147718806480.

[32] James H Stock and Mark W Watson. Vector autoregressions. *Journal of Economic perspectives*, 15(4):101–115, 2001.

[33] Ji Wang, Weidong Bao, Lei Zheng, Xiaomin Zhu, and Philip S. Yu. An attention-augmented deep architecture for hard drive status monitoring in large-scale storage systems. *ACM Trans. Storage*, 15(3), August 2019. ISSN: 1553-3077. doi:10.1145/3340290.

[34] Ying Zhao, Xiang Liu, Siqing Gan, and Weimin Zheng. Predicting disk failures with hmm- and hsmm-based approaches. volume 6171, pages 390–404, 07 2010. ISBN: 978-3-642-14399-1. doi:10.1007/978-3-642-14400-4$_3$0.