FACULTY OF SCIENCE & TECHNOLOGY

MSc Data Science and Artificial Intelligence
Jan 2021

Exploring the Effect of Feature Selection and Sampling Techniques on Current HDD Methods Using Hard Drive Samples of 2022

by

Sanskar Behl

Faculty of Science & Technology

Department of Computing and Informatics

Individual Masters Project

Abstract

Hard disk drives are the most used component in storage drives; however, as with any other device, they are also prone to failure. Hard disk failure prediction has become an active research area. Recent trends have shifted towards using machine learning techniques using disk SMART attributes to make predictions about disk failure. The extremely imbalanced Backblaze dataset has been studied extensively with many methods, including feature selection and sampling techniques. None of the previous studies analysed the effect of sampling ratios on the predictive model's performance. Based on some of the best performing HDD analysis methods, our study intends to verify their effectiveness on the newer Backblaze 2022 dataset and analyse the effect of sampling ratios on the performance of predictive models. Bagging methods and new state-of-the-art methods were studied and discussed in the study, along with the use of feature selection and sampling techniques to address the challenges posed by highly imbalanced datasets. Bagging classifiers were found to be more effective and consistent across different experiments. Sampling techniques were found to be more effective in dealing with class imbalances than feature selection methods.

Dissertation Declaration

I agree that, should the University wish to retain it for reference purposes, a copy of my dissertation may be held by Bournemouth University normally for a period of 3 academic years. I understand that once the retention period has expired my dissertation will be destroyed.

Confidentiality

I confirm that this dissertation does not contain information of a commercial or confidential nature or include personal information other than that which would normally be in the public domain unless the relevant permissions have been obtained. Any information which identifies a particular individual's religious or political beliefs, information relating to their health, ethnicity, criminal history, or sex life has been anonymised unless permission has been granted for its publication from the person to whom it relates.

Copyright

The copyright for this dissertation remains with me.

Requests for Information

I agree that this dissertation may be made available as the result of a request for information under the Freedom of Information Act.

Signed: Sanskar Behl .

Name: Sanskar Behl

Date: 7-May-2023

Programme: MSc Data Science and Artificial Intelligence

Original Work Declaration

This dissertation and the project that it is based on are my own work, except where stated, in accordance with University regulations.

Signed: Sanskar Behl .

Name: Sanskar Behl

Date: 7-May-2023

Acknowledgments

First of all, I am thankful to my God who made me able to complete my dissertation successfully. I would also like to show gratitude to my parents for providing me the chance and support to study in Bournemouth University. This dissertation wouldn’t be successful without great and complete support from my supervisor Dr. Lai Xu. This dissertation was a great experience for me as I was able to analyse how theoretical concepts have been utilised in real life.

TABLE OF CONTENTS

LIST OF FIGURES

Terminology

BRF: Balanced Random Forest

BBC: Balanced Bagging Classifier

BBHG: Balanced Bag of Histogram Boosted Gradient Decision Trees

CMRR: Centre for Magnetic Recording Research

DT: Decision Tree

EE: Easy Ensemble

FAR: False Alarm Rate

FDR: False Detection Rate

FMMEA: failure modes, mechanisms, and effects analysis

G-mean / Gmean: Geometric Mean

GBDT: Gradient Boosted Decision Tree

GMM: Gaussian mixture model

HDD: Hard Disk Drives

IR: Imbalance Ratio

LSTM: Long Short-Term Memory

LSTM: Long Short-Term Memory

MD: Mahalanobis distance

PCA: Principal Component Analysis

RAT: Reverse Arrangement Test

RF: Random Forest

RFE: Recursive Feature Elimination

RNN: Recurrent Neural Network

ROS: Random Over Sampling

RUS: Random Under Sampling

SMART: Self-Monitoring, Analysis and Reporting Technology

SMOTE: Synthetic Minority Over Sampling Technique

SVM: Support Vector Machine

TIA: Time in Advance

TLDFP: Transfer Learning for Minority Disk Failure Prediction

WLR: Weighted Logistic Regression

1 INTRODUCTION

1.1 Background

Hard disk drives (HDD) have come a long way since their development around 70 years ago. Initially, manufacturers never anticipated the immense success that these devices would achieve in modern times, as now they are used to store critical data almost everywhere you look. One key reason for HDDs' surge in popularity and wide-spread usage is due to their impressive durability compared to other hardware components like RAM or flash memory which can easily succumb electrostatic discharge whereas HDDs are resistant to electromagnetic interference ensuring no loss of stored information. While hard disk drives (HDDs) are generally considered reliable, they have earned a reputation as the most frequently replaced hardware component in personal computers (PCs) (Li et al. 2014). Despite having a history of failure, the incidents of HDD malfunctions have risen significantly (Schroeder and Gibson 2007b). As per reports, a staggering 78% of Microsoft data centres' hardware replacements were attributed to hard drives (Vishwanath and Nagappan 2010). Therefore, it is a vital and active research area to improve the reliability and availability of data storage systems.

A hard disk drive (HDD) is a series of disks that use one or more rigid, rapidly rotating disks to store data. The drive is responsible for data storage. There are four main units of an HDD: One of the components of an HDD is the Head disk interface. The other three include head stack assembly, spindle motors/bearings, and the electronics module (Wang et al. 2011; Shen et al. 2018). A hard disk's mechanical and electrical parts will each have different early-warning signals for when a failure is about to happen. For example, the head–disk interface is an excellent place to look for early warning, as is the quality of the motor or bearings. The types of problems that will affect these parts are also different, so each aspect of the disk will give you some information to pick out from. A head–magnetization problem will be very different to a read error. By using these separate pieces of information and separating them out, you get a much clearer picture of the health of the drive (Shen et al. 2018).

There are three major categories of algorithms and techniques that have been applied by past research into hard drive failure prediction: The first one is to try to predict the amount of remaining time until failures of the chosen drives occur as a forecasting problem (Lima et al. 2018; Züfle et al. 2020). An approach to HDD failure prediction is as a series of events outside of the control of the drive and then to predict the probability of such events based on current and historical information about the drive. Specifically, historical data about the average lifetime of a given disk is used to build a regression model, which can be used to predict the expected lifetime of a given disk.

However, both the difficulties involved in obtaining sufficient failure of the disk examples and the fact that their parameters depend on the operating conditions and environment make it challenging to learn a distinctive model (Ircio et al. 2022b). Still, other algorithmic approaches include machine learning and clustering.

The second strategy involves modelling the drive's health condition as a multi-class classification issue (Liu et al. 2020; Züfle et al. 2020). Consequently, the multiple classes are defined as varying health statuses to differ in consequence of the failure over time. The highest degree indicates explicitly that the disc is functioning properly, and for the other grades, the lower the grade, the closer failure is, with the lowest grade signifying impending failure. It should be noted that the previously recommended technique is equivalent to discretizing a previously described regression model and therefore possesses the same difficulties. Additionally, because failures are rare, the percentage of successful and unsuccessful discs is quite imbalanced, and by producing additional classes, this issue is even worse (Ircio et al. 2022a).

The third and most popular method models the hard drive state binary classification problem as the failure prediction problem (correct or failed) (Aussel et al. 2017; Queiroz et al. 2017; Ahmed and Green 2022). A disk is classified as failed if it exhibits signs of deterioration, indicating that it will fail within a short period. We will be using this approach in our study. While this is the most preferred approach, it does have its flaws. This is because the failure lead time is known in advance. It is assumed that all drives will show signs of malfunction with exactly the same anticipation. The problem is that there is no one lead-time that works for all hard drives. This is because the environment in which the drive operates, and its life expectancy is different from each other. Therefore, an automated method is developed to detect every disk's first signs and symptoms of malfunction. It does not require that each disk be pre-set with a lead time.

In this research, we reviewed some of the best-performing HDD failure analysis methods that have been used in earlier research. Our study focuses on verifying whether existing HDD methods are still a good fit for predicting HDD failures, as the dataset is getting imbalanced every year. We compared the methods using the latest 2022 Backblaze dataset and various class imbalance techniques to verify and in the hope of further improving performance. We also measured the effect of different sampling ratios on the machine learning models.

We thoroughly review existing HDD failure prediction methods, highlighting their strengths and limitations. Next, we verified those methods on the newer dataset that leverages advanced machine learning techniques, such as ensemble methods and hybrid methods, to improve the accuracy and reliability of HDD failure predictions. To further improve those existing methods, we used techniques such as feature selection and sampling methods.

Finally, we compared the effect of different sampling ratios on the performance of ML models. Based on our analysis of the Backblaze HDD dataset, sampling techniques are more important than feature selection when it comes to resolving the issue of class imbalance in the dataset.

1.2 SMART

The SMART (Self-Monitoring, Analysis and Reporting Technology) system is a monitoring system included in computer hard disk drives (HDD) and solid-state drives (SSD) that detects and analyses various indicators of reliability in the hope of anticipating failures. The SMART system was implemented on all ATA and SATA drives and is also supported by SCSI and SAS interfaces. Using SMART, built-in functions for HDDs that collect data that correspond to records or physical units by sensors or counters (Hughes et al. 2002; Pinheiro et al. 2007). Up to thirty internal drive attributes such as relocated sector count (RSC), spin-up time (SUT), seek error rate (SER), temperature Celsius (TC), and power-on hours (POH) are included in SMART data (Shen et al. 2018). These attributes are related to the hard disk drive's health and can be used to determine its internal state. The value of relocated sector count (RSC) indicates the number of bad sectors. A change in the Temperature Celsius (TC) and spin-up time (SUT) strongly corresponds with the health condition of the spindle motor.

The five fields that make up each attribute are raw data, value, threshold, worst value, and status. The values measured by a sensor, or a counter are raw data. The value is the current raw data's normalized value. Failure is detected using the threshold. HDD manufacturers specify the threshold value and the algorithm for calculating the value. A warning is issued when the normalized value goes above the threshold. Additionally, SMART will sound a failure alarm when any attribute's status changes to a warning (Shen et al. 2018).
Drive failures are significantly associated with SMART characteristics. Three attributes—grown defect count, read soft errors, and seek errors—were combined in a prediction technique provided by (Hamerly and Elkan 2001). This technique produced predictions with higher accuracy when all three attributes were combined. According to this, not all SMART qualities are equally helpful for predicting HDD failure. (Pinheiro et al. 2007) Found that relocated sector count, scan error and offline relocated sector count errors correlate highly with HDD failures. According to (Ma et al. 2015), RSC is the most crucial factor for identifying impending failures, with latent sector faults being the primary cause of HDD failure. (Wang et al. 2013) identified different indicators of failure in inner units of the HDD and determined the priority of SMART attributes to be used in a prediction system based on the severity and occurrence of relevant failures. (Huang 2017) showed that failure-correlated attributes of SMART for each type of HDD failure are pretty different. The failure prediction of HDDs is therefore based on strong-correlation attributes rather than a single combination.

1.3 Problem definition

Data centres commonly use hard disk drives as data storage devices, but as the number of hard drives used in the storage systems increases, it becomes more challenging to maintain their reliability (Schroeder and Gibson 2007a; Li et al. 2014; Shen et al. 2018). When hard disk drives fail, it can lead to information losses, which can be a considerable problem for the users. Even though multiple copies of data can be stored in the system as backup, it can also increase costs at the same time (Huang 2017). According to Backblaze, the Annual Failure Rate (AFR) for all the drives in 2021 was 1.01% which is around 1820 drives out of 202,759 (Klein 2022). Hard disk drive failure is a prevalent problem. This may happen in any area of the disk drive, and therefore it is not possible to predict with certainty where a failure will occur. In the case of a single drive, a replacement is usually sufficient. A disk mirroring, disk duplexing, or disk striping system can be used as a contingency measure in case of failure of one of two or more drives respectively. In such cases, the mechanisms provided by the operating system can be used to replace one of the drives without requiring any manual configuration changes. However, this solution requires buying a second hard disk drive. This expense might be prohibitive for small businesses, and a more straightforward solution might be desirable. Consequently, it seems practical to develop a model that can predict hard drive failure and then the operators can use the prediction result to improve system reliability and reduce cost.

The most common cause of hard drive failure is the accumulation of errors on the magnetic media. Hard drives are designed to compensate for these errors by writing data to a different location on the disk. However, when the number of errors exceeds a certain threshold, the drive will no longer be able to compensate, and data will be lost (Schroeder and Gibson 2007a). The number of errors that a hard drive can compensate is limited by the number of spare sectors available on the disk. Spare sectors are used to replace sectors that have become unusable due to excessive errors. When a hard drive is new, it has many spare sectors available. Hard disk drives have a finite number of read and write operations, after which failure is inevitable. A hard disk drive is built to withstand a single component's failure but not multiple components within the same component category. If the head fails, spare heads are on the drive to take their place. However, if a head and a platter both fail, the drive is rendered inoperable (Aussel et al. 2017). Many predictive models have been proposed to mitigate hard drive failures (discussed in chapter 2.4) but they all have their own limitations. For example, some models need to be more accurate to accurately predict the time of failure, while others cannot be used in real-time systems because they require too much computation power. Based on what has been presented so far, the diagnosis of hard disk drive failure should be well on its way to being solved. However, it remains a problem. Why is this the

case? The best explanation for this is that hard disk drive failure has been addressed as a problem dictated by a single failure mode (e.g., head crash). However, more is needed to explain why the high performance of the failure prediction models that appear in the literature has yet to mitigate the problem. Therefore, the specifics of hard drives need to be better taken into account: First, because hard drives are assembled from components with different performances, it is necessary to quantify their individual performances to have a final performance expressed in a single value. Second, as hard drive failure only results from a single cause, failure prediction models must be based on something other than event counts per sector. Instead, they should be based on events weighted by their relevance. Furthermore, A class imbalance is a common issue in the research domain, where the number of healthy hard disks greatly exceeds the number of failing ones. Therefore, this study will investigate and evaluate methods for handling this imbalance.

1.4 Aims and Research Questions

Our project aims to investigate the possibility of applying machine learning techniques to improve prediction accuracy over baseline HDD methods in hard disk drives. Our main objectives are:

1. Review existing HDD methods: pre-processing, feature selection methods and ML methods.
2. Evaluate one HDD manufacturer using the latest Backblaze dataset, utilizing machine learning to predict failure with imbalanced data.
3. Compare and evaluate different models using the latest Backblaze dataset, which includes hard drives from various manufacturers.
4. Compare the ML models while using different sampling ratios to see how they affect the given model.
5. Discover which potential pre-processing methods may further improve the performance.

In order to narrow the project's scope, three research questions have been formulated:

1. Which data pre-processing technique can enhance performance on analysing Backblaze 2022 datasets, and how do they compare to existing methods?
2. What are the differences in performance when analysing Backblaze 2022 datasets using the Recursive Feature Elimination (RFE), Random Under-Sampling (RUS), and Balanced Random Forest (BRF) methods on imbalanced datasets?
3. How does the sampling ratio affect the performance of proposed method when analysing Backblaze 2022 datasets?

1.5 The structure of the thesis

Chapter 1: Introduction – This chapter considers prior HDD failure prediction options, defines the issue, and provides background information. We also examine SMART attributes within HDDs and describe our research's objectives and goals.

Chapter 2: Literature Review – This chapter examines numerous techniques for handling imbalanced datasets including sampling methods, feature selection methodologies, plus performance measures alongside relevant literature overviews covering these areas extensively.

Chapter 3: Methodology – This chapter defines our methodology, evaluates potential risks involved, and identifies necessary resources for successfully completing this work utilizing adaptations from CRISP-DM frameworks according to what fits best within current requirements.

Chapter 4: Experiments – This chapter discusses about conducting experiments utilizing Backblaze data to provide an overview of results with comparisons between different tested methodologies.

Chapter 5: Discussion – This chapter discusses the experiments of this research, and the objectives will be critically analysed. This chapter also discusses limitations plus possibilities for future research beyond its current focus.

Chapter 6: Conclusion – This chapter concludes this study by summarizing key findings and then outlining their implications for HDD failure prediction.

2 LITERATURE REVIEW AND RELATED WORK

In this chapter, we discuss about different methods for dealing with the imbalanced dataset. Section 2.1 discusses about different sampling methods that have been used in earlier studies, such as under-sampling, oversampling, and SMOTE. Section 2.2 focuses on discussing about the various feature methods such as RFE, using Backblaze recommendation, and others. While in section 2.3, we discuss about the performance measures like precision, recall, F1-score, FDR, FAR, and Geometric mean. And then finally, we make a summary of the literature in section 2.4.

2.1 Data Sampling Methods

One approach to address class imbalance is the creation of one or more data sets, each of which has a different class distribution from the initial. In order to achieve this, two primary types of data sampling are employed: oversampling and undersampling. Undersampling removes the instances of the majority of classes, and when the process is random, this method is described as *random undersampling* (RUS) (Batista et al. 2004). Oversampling can add instances of the minority class. If it is random, then the technique is referred to by the name of *Random Oversampling* (ROS) (Batista et al. 2004). *Synthetic Minority Oversampling Technique* (SMOTE) can be described as an oversampling technique that creates artificial instances that are created between minorities that are close to one another (Chawla et al. 2002). Between ROS, RUS, SMOTE, and variants of SMOTE, it has been proven that RUS is the one that imposes the least computational burden and has the fastest training time (Hasanin et al. 2019).

Table 1 shows the data sampling methods used in previous studies with different datasets.

Table 1: Sampling methods

Paper	Dataset	Data sampling methods
(Murray et al. 2005)	CMRR	Double resampling
(Aussel et al. 2017)	Backblaze 2014	SMOTE
(Zhang et al. 2020)	Backblaze and Tencent	Undersampling
(Hu et al. 2020)	Backblaze	Downsampling

The majority of research studies focused on hard disk failure have identified imbalanced data as a challenge to be addressed. However, a critical evaluation of the existing literature reveals that several problematic approaches have been employed. For instance, (Zhang et al. 2020) utilized an undersampling technique to improve training with an imbalanced ratio of 3:1. Nonetheless, their study raises concerns as the undersampling was performed on the entire dataset before the train-test split, potentially leading to data leakage. (Ahmed and Green 2022)

To reduce the problems caused by the imbalanced learning, the application of the SMOTE method was studied by (Aussel et al. 2017). The researchers did not provide the sampling ratio that was used in SMOTE even though a higher ratio of sampling could enhance general performance models (Ahmed and Green 2022). SMOTE could have improved prediction performance which did not work contrary to what was expected. This highlights the problems caused by the Backblaze dataset's extreme imbalanced. Further work on sampling techniques is necessary to balance the data. According to (Aussel et al. 2017) Ensemble-based Hybrid Sampling techniques such as SMOTEBagging are an improvement to the SMOTE sampling method. This could improve the learning models and allow to use of learning strategies that are more sensitive to class imbalances like logistic regression.

In the paper of research "A disk failure prediction method based on LSTM network due to its individual specificity" (Hu et al. 2020) The authors employed a down-sampling method to solve the problem of class imbalances when it comes to failure data from hard drives. The procedure involved reducing number of samples within the majority group to obtain an even distribution of classes, with an inverse ratio of 4:1 between the minority and majority classes. One drawback to this technique is that it doesn't adequately explain how to keep the relationship between consecutive values and preserve the primary covariance structure of timing series (Ahmed and Green 2022). That is, cutting down on the number of samples within the major class could cause a loss of information as well as patterns crucial to make accurate predictions.

To prevent overfitting in machine learning models, (Aussel et al. 2017; Ahmed and Green 2022) utilized a k-fold cross-validation (CV) approach. This method randomly divides the dataset into k equally sized groups or folds. One of the folds is held out as a validation set, while the remaining k-1 folds are used to train a classifier. To further reduce the risk of overfitting due to selection bias and data leakage, feature transformation and data augmentation should be performed within CV on each independent training set. This helps to ensure that the model is not learning from the validation set or any other external data. By applying k-fold CV with feature transformation and data augmentation, (Ahmed and Green 2022) were able to achieve better generalization performance of their machine learning models. The issue of feature scaling was solved by scaling the features using the absolute maximum value. To avoid any risk of leakage, all the data processing was carried out in a stratified 5-fold cross-validation framework. This method ensures the authenticity of information is protected and the results are high quality.

2.2 Feature Selection

The conventional wisdom is that the more you use it, the more likely it will fail. The study (Pinheiro et al. 2007) shows that this is not necessarily true and that there is not necessarily a relationship between temperature and failure rates. Several articles in the popular press suggested that higher temperature drives were more likely to fail (Yang and Sun 1999; Cole 2000).However, the analysis shows no clear patterns related to either temperature or utilization levels.

Table 2 shows the feature selection methods used in previous studies with different datasets.

Table 2: Feature Selection Methods

Paper	Dataset	Feature selection
(Murray et al. 2005)	CMRR	RAT, Z-scores
(Wang et al. 2013)	CMRR	FMMEA, mRA
(Queiroz et al. 2017)	CMRR	RFE
(Xu et al. 2016)	Baidu W Baidu S Baidu M	RAT, rank-sum test
(Li et al. 2017)	Baidu W Baidu Q_all Baidu Q_s	Quantile function
(Zhang et al. 2017)	Backblaze 2015	Smart 5,183,184,18,188,193, 197
(Aussel et al. 2017)	Backblaze 2014	Smart 12,187,188, 189,190,198,199,200
(Huang 2017)	Backblaze 2016	Correlation matrix
(Zhang et al. 2020)		PCA
(Hu et al. 2020)	BackBlaze	Pearson correlation
(Ahmed and Green 2022)	Backblaze	Smart 5 features: 5, 187, 188, 197, 198

(Murray et al. 2005) employed the reverse arrangement test (RAT) and z-scores for feature selection in their study. RAT is a nonparametric test for trend, commonly utilized in data analysis, where it is applied to each attribute present in the dataset (Mann 1945). The researchers employed this method as they posited that an increasing trend of drive errors is indicative of failure. Therefore, they applied RAT as a means of identifying relevant features within the dataset.
In accordance with previous research of (Murray et al. 2005; Xu et al. 2016) utilized three non-parametric techniques, namely the reverse arrangement test, rank-sum test, and z-scores, to conduct feature selection from a pool of 23 significant attributes within the Self-Monitoring, Analysis, and Reporting Technology (SMART) dataset. Through this process, they identified 10 attributes as being particularly relevant for their analysis.

(Wang et al. 2013) described a stage of data preparation in which key parameters were identified via feature selection. Fault Mode and Effect Analysis (FMMEA) and minimum Redundancy maximum relevance (mRMR) were used in a two-step process for feature selection. Particularly, FMMEA was used for identifying relevant features that could impact the outcome. mRMR was used for selecting highly relevant features. This two-step process allowed data analysis to be more accurate and reliable.
(Li et al. 2017) used a novel approach to choose features over the three previously used statistical methods. They used quantile functions to quantitatively assess each feature for healthy and failing drives. This allowed for a deeper analysis of each feature. Additionally, critical features that are highly indicative of drive health were chosen for further analysis. (Li et al. 2017) identified features that could help predict failures of drives.

In (Queiroz et al. 2017) study, they utilized Recursive Feature Elimination (RFE) as their feature selection method. To ensure robustness of their approach, ten iterations of RFE were performed with bootstrapped samples of the attribute selection dataset. Through RFE, the authors were able to identify the attributes that exhibited strong correlation with the failure mechanisms of a hard disk drive (HDD). While a direct causal relationship between the selected attributes and HDD failures cannot be definitively established, the identified attributes do provide some indication of a potentially reasonable correspondence.
In the study by (Huang 2017), a correlation matrix was used to select features. The study found that four S.M.A.R.T. attributes (4, 9, 190, 198) had a correlation higher than 0.9 and were therefore removed as redundant features. After this selection process, 17 S.M.A.R.T. attributes remained for analysis.
In the study of (Hu et al. 2020), each disk has 24 Self-Monitoring, Analysing and Reporting Technology (SMART) attributes. For feature selection, the Pearson Correlation test is applied to each attribute in order to test its ability to differentiate between negative and positive examples. The test revealed 14 attributes that could not discern between positive and negative samples, which is why they were eliminated. Because of this selection process, 10 attributes were picked as the learning features of the model of learning.

One of the (Pinheiro et al. 2007) study's key findings is that after HDDs first scan error, drives are 39 times more likely to fail within 60 days than drives with no such errors. First errors in re-allocations, offline re-allocations, and probational counts are also strongly correlated to higher failure probabilities. They also find a strong correlation between the reported S.M.A.R.T error rate of the drive and its failure probability. An examination of failure trends in a large disk drive population of over 100000 enterprise HDDs at a Google data centre found that specific SMART parameters (scan errors, re-allocation counts, offline re-allocation counts, and probational counts)

had a significant impact on failure probability (Li 2017). It is unlikely to achieve an accurate predictive failure model that can be built based on SMART signals alone, most notably because a significant fraction of failed drives showed no signs of failure in any of the monitored SMART features (Pinheiro et al. 2007). The result of the study showed that SMART models are more accurate in predicting trends for large groups of people rather than individuals. It was also found that more reliable predictive models need to consider factors that SMART models do not consider (Pinheiro et al. 2007).

Numerous research studies (Aussel et al. 2017; Huang 2017; Queiroz et al. 2017; Zhang et al. 2017; Hu et al. 2020; Zhang et al. 2020; Ahmed and Green 2022) have employed SMART parameters to predict the failure of hard disk drives. Backblaze, a cloud backup and storage company, has identified five SMART stats that assist in assessing the probability of a drive failure. These include SMART 5, which records the number of reallocated sectors, SMART 187, which indicates reported uncorrectable errors, SMART 188, which measures command timeout, SMART 197, which records the current pending sector count, and SMART 198, which records the number of uncorrectable sectors. These parameters have proven to be effective in predicting hard disk drive failure, making them crucial in the realm of data storage and backup.

2.3 Performance Measures

Various performance metrics have been employed in assessing the effectiveness of classifiers across different studies. These metrics can be grouped according to the specific evaluation measures used, which include precision, recall, F1 score, false discovery rate (FDR), false alarm rate (FAR), area under the receiver operating characteristic curve (AUC-ROC), and geometric mean (G-mean).

Table 3 shows the evaluation metrics used in previous studies with different datasets.

Table 3

Paper	Dataset	Evaluation metrics
(Murray et al. 2005)	CMRR	FDR, FAR
(Wang et al. 2013)	CMRR	Prediction accuracy, ROC curve, and time before failure
(Queiroz et al. 2017)	CMRR	ROC
(Xu et al. 2016)	Baidu W Baidu S Baidu M	FDR, FAR

(Li et al. 2017)	Baidu W Baidu Q_all Baidu Q_s	FDR, FAR, TIA
(Zhang et al. 2017)	Backblaze 2015	AUC of ROC
(Aussel et al. 2017)	Backblaze 2014	Precision, Recall
(Huang 2017)	Backblaze 2016	Precision, recall, F1-score and FPR
(Zhang et al. 2020)	Backblaze, Tencent	FDR, FAR, f-score, AOC-ROC
(Hu et al. 2020)	BackBlaze	FDR, FAR, Precision, F1
(Ahmed and Green 2022)	Backblaze	G-mean

In terms of precision and recall, (Aussel et al. 2017; Huang 2017) utilized these metrics in their evaluation of a classifier. Precision refers to the proportion of true positive predictions made by the classifier, while recall is the ratio of true positives correctly identified by the classifier to the total number of actual positive instances. The formula to calculate them are as follows:

$$\left(Precision = \frac{TP}{TP + FP}\right)$$

$$\left(Recall = \frac{TP}{TP + FN}\right)$$

According to (Powers 2020), using recall and precision as evaluation metrics can overlook the ability of the model to handle the majority class and can also perpetuate intrinsic bias. However, it is essential to note that more than precision or recall alone may be needed to provide a complete understanding of the model's performance. Both metrics fail to account for the number of true majority examples, which can result in a model with high precision but low recall or a model with high recall but low precision (Ahmed and Green 2022).

FDR and FAR were used by (Murray et al. 2005; Xu et al. 2016; Li et al. 2017; Hu et al. 2020; Zhang et al. 2020) in their respective studies. FDR measures the proportion of false positives in relation to the total number of positive predictions, while FAR represents the ratio of false positives to the total number of negative instances. The formula to calculate them is given below:

$$False\ Discovery\ Rate\ (FDR) = \frac{FP}{(TP + FP)}$$

$$False\ Alarm\ Rate\ (FAR) = \frac{FP}{(TN + FP)}$$

Although false discovery rate (FDR) and false alarm rate (FAR) are frequently used as measures to assess inequitable classification problems, they do not provide a comprehensive evaluation of a model's performance. By merely utilizing FDR or FAR, faulty conclusions can be drawn regarding the efficiency of the algorithm (Ahmed and Green 2022). The algorithm's performance in learning imbalances should therefore be assessed based on other appropriate measures.

To evaluate the performance of their classifiers, (Wang et al. 2013; Zhang et al. 2017; Zhang et al. 2020) used the AUC-ROC method. In this metric, the trade-off between false positives and true positives is evaluated across a range of threshold values. As a result of its treatment of false positives and negatives equally, AUC-ROC has a number of significant drawbacks. The costs associated with false positives and negatives could differ in significant ways for many scenarios in the real world, such as diagnostics for medical conditions (Halligan et al. 2015) and HDD failures where the data is highly imbalanced. Therefore, AUC-ROC may not accurately represent the model's performance in such cases.

In the field of imbalanced learning, the geometric mean (Gmean) was proposed as a performance indicator. As a measure of both minorities and majorities, Gmean is particularly useful when data is imbalanced (Kubat and Matwin 1997; Liu et al. 2009). While other metrics, such as precision, accuracy, and recall, tend to favour the majority group, Gmean assigns the same importance to all classes (Branco et al. 2016). Other metrics may be misleading if the number of instances in one class is much smaller than in the others due to imbalanced learning. The reason for this is that classification models that always predict most classes can reach high accuracy in these circumstances. The formula to calculate G-mean is given below:

$$G - mean = \sqrt{((TP / (TP + FN)) * (TN / (TN + FP)))}$$

Numerous studies have shown that Gmean is more efficient when evaluating classifiers in learning imbalances when the minority class is more important (Kubat and Matwin 1997; Chawla et al. 2004; Liu et al. 2009; Ahmed and Green 2022) evaluated different algorithms for learning imbalances using Gmean with other metrics and showed that Gmean could be used to identify the most effective classifiers. Similarly, (Barua et al. 2014) showed Gmean to be superior to other metrics when it comes to identifying the most effective classifier for imbalanced data.

2.4 Machine Learning Methods

Failure prediction of hard disk drives (HDDs) is an important area for research. Machine learning (ML) has been used extensively in this context. Random forests are able to handle large datasets with multiple features. They also have the ability to deal with nonlinear relationships between features. Table 4 summarizes the ML methods used in previous studies with different datasets.

Table 4: ML methods used

Paper, Year	Dataset, Year	Best ML Method
(Murray et al. 2005)	CMRR	SVM
(Wang et al. 2013)	CMRR	MD
(Queiroz et al. 2017)	CMRR	GMM
(Xu et al. 2016)	Baidu W Baidu S Baidu M	RNN
(Li et al. 2017)	Baidu W Baidu Q_all Baidu Q_s	DT, GBDT
(Zhang et al. 2017)	Backblaze 2015	LSTM
(Aussel et al. 2017)	Backblaze 2014	RF
(Huang 2017)	Backblaze 2016	XGBoost
(Zhang et al. 2020)	Backblaze, Tencent	TLDFP
(Hu et al. 2020)	BackBlaze	LSTM
(Ahmed and Green 2022)	Backblaze	BRF, WLR, EE

Support vector machines (SVM) is one such method that has been used in HDD failure prediction. (Murray et al. 2005) employed SVM with recursive feature elimination (RFE) to select the most relevant features for prediction. Another ML method that has been used in this context is decision trees (DT). (Li 2017) used the quantile function to normalize the data, followed by DT for prediction. Gradient boosting decision trees (GBDT) is another popular ML method that has been employed in HDD failure prediction. (Li 2017) also used GBDT for prediction.

Gaussian mixture models (GMM) have also been employed to predict HDD failure. (Queiroz et al. 2017) used GMM with RFE for feature selection in their prediction model. Recurrent neural networks (RNNs) have also been used in this context. (Xu et al. 2016) utilized RNN to predict HDD failure with feature selection by rank-sum test.

Long short-term memory networks (LSTM) are another popular ML method that has been used in HDD failure prediction. (Zhang et al. 2017) employed LSTM for prediction with feature selection of the SMART attributes 5, 183, 184, 18, 188, 193, 197. (Hu et al. 2020) also used LSTM to predict with Pearson correlation for feature selection.

Extreme gradient boosting (XGBoost) is another method that has been employed in HDD failure prediction. (Huang 2017) used a correlation matrix for feature selection and XGBoost for

prediction. In addition, principal component analysis (PCA) has also been used for feature reduction and prediction. (Zhang et al. 2020) utilized PCA for this purpose.

(Aussel et al. 2017) used RF to predict HDD failure using the Backblaze 2014 dataset. They selected the SMART attributes 12, 187, 188, 189, 190, 198, 199, and 200 as features and applied mutual information for feature selection. Their study showed that RF was effective in predicting HDD failures and outperformed other popular ML methods, such as support vector machines and decision trees. However, an imbalanced dataset can pose a challenge to the effectiveness of random forest.

(Ahmed and Green 2022) proposed a variant of random forest called Balanced Random Forest (BRF). They used a subset of five SMART attributes and applied feature scaling with MaxAbs. They also compared the performance of BRF with weighted logistic regression (WLR) and easy ensemble (EE). Their study showed that BRF outperformed both WLR and EE in terms of predicting HDD failures, indicating the effectiveness of this method in handling imbalanced datasets.

2.5 Summary

Literature shows that SMART attributes contain health information about the disk drive (Pinheiro et al. 2007; Aussel et al. 2017; Shen et al. 2018). Researchers in this field are, therefore, optimistic about the predictive ability of SMART parameters. Backblaze (Klein 2016), identified these SMART metrics to be good predictors for imminent failure.

- SMART 5: Count of reallocated sectors
- SMART 187: Count of errors
- SMART 188: Count of interrupted operations due to HDD timeout
- SMART 197: Count of unstable sectors
- SMART 198: Count of uncorrectable sectors which were identified from offline hard drive’s scanning

By analysing Google's operational hard drives, (Pinheiro et al. 2007) also identified SMART parameters 5, 187, and 197 as critical predictors of failed disks. Many manufacturers believed that disk failure could be linked to temperature. However, (Pinheiro et al. 2007) found a lower correlation between temperature and disk failures. Failure rates are strongly correlated with the drive model and SMART attributes. Most past studies have focused on analysing one or more models of hard drives (Pinheiro et al. 2007; Yang et al. 2021).

Some studies (Piramuthu 2004; Čehovin and Bosnić 2010; A. Aziz et al. 2017) examined the impact of feature choice methods on classification. These studies showed that appropriate feature selection methods could improve performance in terms of accuracy and complexity. While these studies were focused on accuracy evaluations, in reality, there are other factors that must be taken into accounts, such as efficiency and generalization. One example is that an algorithm for anomaly detection runs very slowly and should be used sparingly. Also, the algorithm cannot be extended to new data at high performance (Yang et al. 2021).

Table 5: Summary of background

Paper, Year	Dataset, Year	Feature Selection	Best ML Method	Evaluation Method	Sampling Method	Sample Size	Good	Failed	IR
(Murray et al. 2005)	CMRR	RAT, Z-scores	SVM	FDR, FAR	Double-resampling	369 (drives)	178	191	0.93:1
(Wang et al. 2013)	CMRR	FMMEA, mRA	MD	prediction accuracy, ROC curve, and time before failure	N/A	369 (drives)	178	191	0.93:1
(Queiroz et al. 2017)	CMRR	RFE	GMM	ROC	N/A	369 (drives)	178	191	0.93:1
(Xu et al. 2016)	Baidu W Baidu S Baidu M	RAT, rank-sum test	RNN	FDR, FAR	N/A	71,619 (drives)	70,841	778	91.05:1
(Li 2017)	Baidu W Baidu Q_all Baidu Q_s	Quantile function	DT, GBDT	FDR, FAR, TIA	N/A	27,406,782 (samples)	26,809,112	597,670	44.85:1
(Zhang et al. 2017)	Backblaze 2015	Smart 5,183,184,18, 188,193, 197	LSTM	AUC of ROC	N/A	59,340 (drives)	58,754	586	100.26:1

(Aussel et al. 2017)	Backblaze 2014	Smart 12,187,188, 189,190,198,199,200	RF	Precision, Recall	SMOTE	12,582,414 (samples)	12,580,208	2,206	5702.72:1
(Huang 2017)	Backblaze 2016	Correlation matrix	XGBoost	Precision, recall, F1-score and FPR	N/A	3,196,55 (samples), 34,970 (drives)	34,736 (drives)	234 (drives)	148.44:1 (drives)
(Zhang et al. 2020)		PCA	TLDFP	FDR, FAR, f-score, AOC-ROC	Under-sampling	1281 (drives)	1143 (drives)	138 (drives)	9.282 (drives)
(Hu et al. 2020)	BackBlaze	Pearson correlation	LSTM	FDR, FAR, Precision, F1	Down-sampling	46,321 (drives)	45164 (drives)	1157 (drives)	39.03:1
(Ahmed and Green 2022)	Backblaze	Smart 5 features and Feature Scaling: MaxAbs	BRF, WLR, EE	G-mean	N/A	12237899 (samples)	12,236,835 (samples)	1064	11500.78:1

Table 5 provides the summary of the existing HDD methods used in past with different ML methods, datasets, feature selection techniques, sampling methods, and evaluation metrics.

(Aussel et al. 2017) tested SVM and RF classifiers for failure prediction in the presence of severely imbalanced HDD data. To achieve 95% precision and 67% recall with RF and all the features, approximately 12 million samples were used with 2,586 failure cases from the (Ahmed and Green 2022; Klein 2022).

(Zhang et al. 2020) proposed a technique called transfer learning for minor disk failure prediction (TLDFP). This involves integrating transfer learning into failure prediction. The threshold of 1,550 was used to define minority disks. The dataset with less than 1,500 hard drives is considered as minority disks. Instead of calculating the proportion of failed and healthy disks, they are classified as minorities. The authors reported 96% FDR and 0.5% FAR for TLDFP models. The authors also claimed that overfitting was due to the limited number of training samples (Ahmed and Green 2022).

(Yang et al. 2021) examined the effect of feature selections on anomaly detection algorithms' performance. Recursive feature elimination (RFE), reverse arrangement test (RAT), Z-scores, and minimum redundancy maximum relevance (mRMR), were all used in their study to evaluate

Logistic Regression (LR), multilayer perception (MLP), Random Forest (RF) Support Vector Machine (SVM), Gradient Boosted Decision Trees (GBDT) GMM and Mahalanobis separation (MD) algorithms (Ahmed and Green 2022). Gradient Boosted Decision Tree (GBDT) was the most efficient among anomaly detection algorithms. Regularized greedy forest (RGF), Random Forest (RF), and Support Vector Machine (SVM) were robust against redundant features. RGF, SVM, and Logistic Regression (LR) compute slowly, and Logistic Regression (LR) has a good generalization. Recursive Feature Elimination (RFE) was recommended among feature selection methods due to its high performance and high resistance against redundant features.

(Ahmed and Green 2022) pointed out the significance of the most reliable metrics (Gmean) to accurately identify disk failures by making use of the predictive power that the five SMART attributes in Backblaze data sets (Klein 2022). Because algorithm-level approaches such as cost-sensitive or hybrid/ensemble methods have been extremely successful in the domain of imbalance classification for moderate imbalance ratios and have been applied to EasyEnsemble, BRF, and WLR to develop model of binary classification for HDDs data that have an extremely high imbalance ratio. They find that BRF and Easy Ensemble are pretty consistent among different datasets while WLR performed good on some datasets only and suffers from inherent complexity of data. They have also tried to demonstrate that the traditional methods of ML can be modified in order to construct an efficient classifier to deal with the heavily imbalanced HDDs data.

- They have implemented hybrid methods: EasyEnsemble and BRF (Balanced Random Forest), together with the cost-sensitive algorithm called WLR (Weighted Logistic Regression), to solve the imbalanced learning problem that is causing the determination of failures on hard drives. They have also implemented the traditional methods of RF (Random Forest) as well as DT (Decision Tree) techniques as the base methods to facilitate comparison.
- The issue with scaling of features is solved with the use of maximum absolute value for scaling features.
- All data processing was conducted within a stratified 5-fold CV and sklearn’s pipeline in order to avoid leakage of data.
- The classifiers' performance was evaluated in relation to the Gmean measurement. We also have reported FDR and FAR as well as cross-validated AUC to evaluate our results against the literature available.

3 METHODOLOGY

This chapter discusses about the methodology used in the project, requirements, potential risks, inventory of resources and other project planning methods used. We will be using CRISP-DM for our research study which we will alter some parts of it according of our research and needs.

CRISP-DM (Cross-Industry Standard Process for Data Mining) is a methodological framework that offers a thorough method of planning and executing the data mining process. It is a tried and tested method composed of six stages: Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation, and Deployment. However, it is essential to remember that this is a simulated sequence of events. In reality, many tasks are performed in different ways. Therefore, we will be following and altering some of the parts provided here: (Smart Vision 2017) for example the stage one business understanding would be our literature review which is discussed in more detail in section 2. Figure 1 (Smart Vision 2017) describes the general process of Crisp-DM.

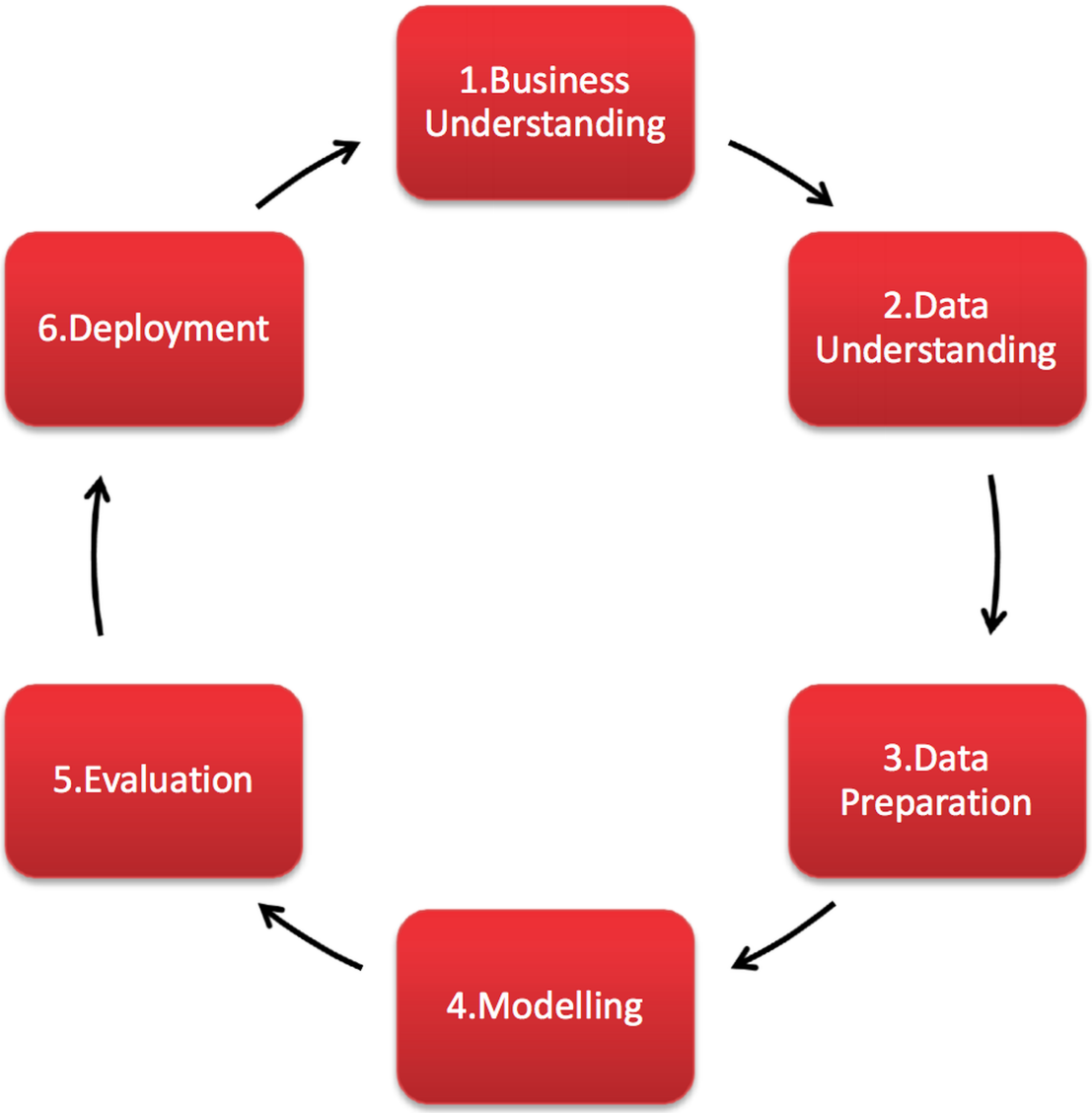


Figure 1: Crisp dm

We will be following the process with a little modification and the process is as follows:

Stage 1: Literature Review: We will review the existing HDD methods in this stage.

Stage 2: Data Understanding: We will have a detailed understanding of the data in this stage.

Stage 3: Data Preparation: We will clean and prepare the data for prediction in this stage.

Stage 4: Modelling: We will apply the considered machine learning models to prepare data in this stage.

Stage 5: Evaluation: We will evaluate the trained machine learning model for its effectiveness in this stage.

Stage 6: Deployment: This stage is not necessary for this project, so we will not be using this stage in our project

3.1 Stage One: Literature Review and Planning

The literature review is provided and discussed in more detail in Section 2. According to the literature and more recent studies, we will be using Backblaze (Backblaze 2022) dataset which is available publicly to download from their website. Data cleaning methods, data sampling methods, feature selection methods and ML models considered are also discussed in more detail in Section 2.

We will be using the project time planner template (Figure 2) which is available from here: https://templates.office.com/en-gb/gantt-project-planner-tm02887601

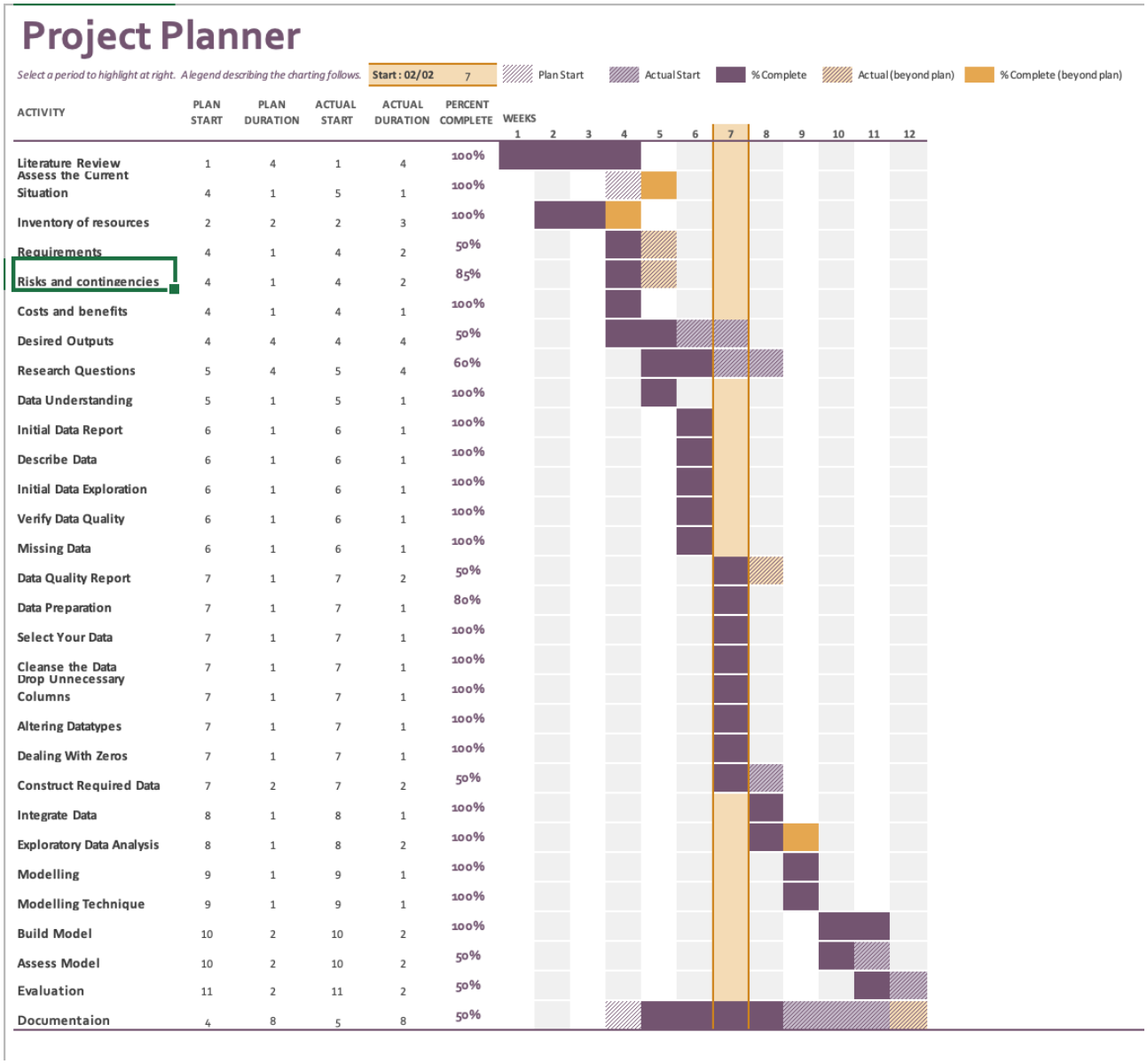


Figure 2: Project planner

3.1.1 Inventory Of Resources

Here we will be listing out the available list of resources we have to carry out our research project on predicting hard disk drive. At first, we only had access to personal laptop (Apple m1 MacBook with 8 gigs of ram) to carry out the research. Most of the research was carried out with the same however, later we got access to larger RAM computer which were provided by Microsoft Azure. A big shoutout to the Microsoft team for providing those resources. The list of resources are as follows:

1. Literature
2. Backblaze 2022 data
3. Computing resources: Apple m1 MacBook (8 cores + 8gb ram), Azure Computers (8 cores + 64gb ram)
4. Software: JupyterLab environment, Sklearn, Python, Pandas, SKLearn, imbLearn

3.1.2 Requirements, Constraints and Assumptions

The basic requirement to complete this project is a high performing computer which is required to perform analysis and dealing with very large dataset. We need to finish the project in timely manner and set realistic timelines for which we used GANTT chart and regular supervisory meetings. Also, we must consider the quality and clarity of our findings, as this can affect how effective and applicable our findings are. There are no assumptions made other than assuming the data provided by BackBlaze is accurate and up to date. The constraints to the project are another crucial aspect to consider. We are limited on resources, time and data which could impact our ability to conduct a thorough analysis and testing, additionally, technical limitations can also affect our methodology, for instance, the accuracy and quantity of available data and the computing power.

3.1.3 Risks and Contingencies

There are few risks involved in this project as with any other project that could cause the project to delay or fail as mentioned below:

1. Time crunch: It is possible that project might take longer than expected. I will overestimate the time needed to complete this project and schedule accordingly.
2. Stretched resources: I do not have enough resources to complete this project. For example, A high performance computer to train the models, Time required to complete the project

and extensive skills for time series analysis. I will be using the cloud computing available in the Microsoft Azure to overcome the risk of resources.

3. Data quality issues: The data provided by Backblaze is very messy and one might consider it to be of a poor quality and inconsistent which might negatively impact the accuracy of the predictions. To mitigate this risk, proper data cleaning and validation will be performed backed by the literature to ensure the quality of the predictions.

3.2 Stage 2: Data Understanding

3.2.1 Data Collection

The dataset used in this study was collected from Backblaze (Backblaze 2022). This data includes basic drive information along with the S.M.A.R.T. statistics reported by each drive and is collected daily from the hard drives in Backblaze's data centre. The data is accessible to download and is used to reproduce Backblaze's published statistics and insight. The data can be used to conduct research like analysing the failure rate of hard drives and trends. Backblaze releases quarterly and annual dataset on their hard disks and SSDs they examine at their data centres, which contain information about the failure rates for various drives.

3.2.2 Data Description

Backblaze takes a snapshot of each operational hard drive in their data centre every day, which includes basic drive information along with the S.M.A.R.T. statistics reported by that drive. Backblaze uses both SSDs and HDDs as boot drives in their storage servers, and the workload for each type of drive is similar. The snapshots of drives are put together into one file that is then consists of a distinct row that indicate the current status of the drive. The complete description of the data is given in table 7:

Table 6: Data description

Column	Column Name	Data Type	Description
1	date	timestamp	Date of file created in YYYY-MM-DD format
2	serial_number	string	Serial number of the drive assigned by the manufacturer

3	model	string	Model number of the drive assigned by the manufacturer
4	capacity_bytes	long	The capacity of the drive (in bytes)
5	failure	integer	Whether the drive has failed or not with '0' meaning still working and '1' meaning that the drive has failed
6-179	smart_x_raw / smart_x_normalized	integer	Smart stats of the drive with raw and normalized values

3.2.3 Initial Data Report

The dataset comprises of ~ 235,608 drives with **80357762** number of samples over the span of 01/Jan/2022 to 31/12/2022. The figure 3 shows the number of rows/samples in each quarter of 2022.

```
: print("Number of rows in Q1: ", df1.count())
  print("Number of rows in Q2: ", df2.count())
  print("Number of rows in Q3: ", df3.count())
  print("Number of rows in Q4: ", df4.count())

Number of rows in Q1:  18845260

Number of rows in Q2:  19424436

Number of rows in Q3:  20591757
[Stage 21:=====================================
Number of rows in Q4:  21496309
```

Figure 3: Number of rows in each quarter

According to backblaze (Klein 2023), the annual failure rate in 2022 in increased at 1.37% from 1.01% and 0.93% in 2021 and 2020 respectively. The figure 4 shows the number of failures in each quarter of 2022. Q3 2022 has the most number of failures in 2022.

quarter	not_failed	failed
1	18844596	664
2	19423640	796
3	20590794	963
4	21259959	735

Figure 4: Num of failures in each quarter

Our study shows that the model “ST4000DM000” had the most number of failures in each quarter with 117 failure in Q1, 156 in Q2, 202 in Q3 and 158 failures in Q4. The figure 5 shows the number of failures by each quarter in 2022.

model	quarter	num_failures
ST4000DM000	1	117
TOSHIBA MG07ACA14TA	1	88
ST8000NM0055	1	86
ST12000NM0008	1	73
ST8000DM002	1	38
ST4000DM000	2	156
ST12000NM0008	2	108
TOSHIBA MG07ACA14TA	2	97
ST8000NM0055	2	79
ST8000DM002	2	48
ST4000DM000	3	202
ST12000NM0008	3	124
TOSHIBA MG07ACA14TA	3	117
ST8000NM0055	3	107
ST8000DM002	3	62
ST4000DM000	4	158
ST12000NM0008	4	99
TOSHIBA MG07ACA14TA	4	83
ST8000NM0055	4	76
ST16000NM001G	4	45

Figure 5: Failures by each drive

After carefully considering all the models and drives, we found that that there were **3158** total number of failures in 2022 out of **80122147** number of samples that makes **0.0039%** failed drives out of total samples with IR (imbalance ratio) of over **25371.16 : 1**.

3.3 Stage 3: Data Preparation

3.3.1 Data Selection

The original data is extremely large, and it is not possible for us to work on the whole dataset due to time constraints and a limited number of resources. There are different ways to deal with this situation, and one of them is to use big data architecture such as Apache Spark. We used Apache Spark in our study to deal with big data and then to select only the data that we required. This solved the issue of ‘out-of-memory’ errors. Figure 6 shows the approach used to deal with large dataset using PySpark.

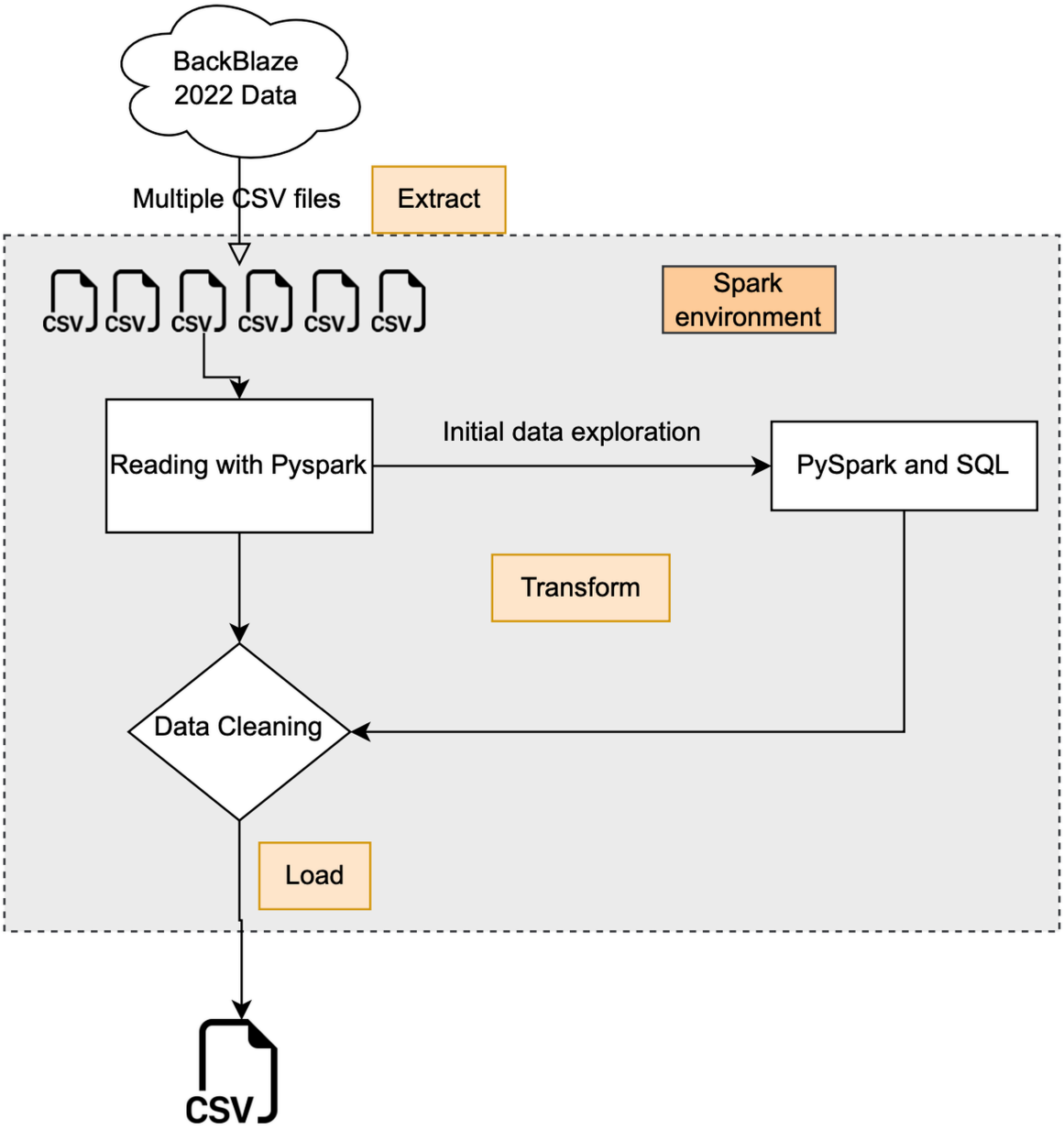


Figure 6: Data engineering

As seen in our study in chapter 3.2, we will be selecting the model "**ST4000DM000**" only as it had the most number of failures (**633**) in 2022. That being said, out of the total number of samples (**6681069**), **633** drives failed, which makes **0.0095%** of total samples with an IR ratio of **10553.61: 1**. We managed to bring down the dataset to 1.36GB with all features and 80MB with only 5 features.

3.3.2 Data Cleaning

Data cleaning is really important for before feeding to any machine learning model. The steps followed in cleaning the data are as follows:

1. Selecting only raw values from the dataset: The dataset consists of raw and normalised values of SMART attributes. We only considered the raw values to get the accurate results.
2. Dropping unnecessary columns: We dropped the columns date, serial_number, model, and capacity_bytes as these were not necessary to train the model.
3. Dropping duplicates: We dropped the duplicate values so that the hard drive that has failed already will not show multiple times in our dataset.
4. Filtering: We filtered our dataset by only selecting the model ST4000DM000
5. Dealing with null values: All the null values were replaced by ‘0’.

3.3.3 Feature Selection

Feature selection is an important technique in the machine learning domain; however, Backblaze has recommended 5 features to be extremely useful in the field of hard drive failure predictions. (Yang et al. 2021) analysed different feature selection techniques and found RFE (Recursive Feature Elimination) super useful and most accurate. We will be using the same set of feature selection methods to maximise the performance of considered models. (Johnson and Kjell 2019) explains RFE as a reverse choice of predictors. This method starts by building models on the whole collection of predictors and then calculating the importance scores for every predictor. The least important predictor(s) is then eliminated, the model is rebuilt, and scores for importance are calculated again. In the real world, the analyst will specify how many predictors they want to examine as well as the subset's size. Thus, the size of the subset is an important *adjustment element* that can be utilised to tune RFE. The subset size that meets its performance requirements is used to pick the predictors based on the importance ranking. The best subset is utilised for training an ultimate model.

3.3.4 Exploratory Data Analysis

Due to limited time and resources, we were only able to perform EDA (Exploratory Data Analysis) on 10 selected columns. There were 10+ columns which were selected by RFE, to get to know the data better, exploratory data analysis is required. Thus, an in-depth examination of data was carried out to find the relevant columns to the study. The exploratory data analysis was used to enhance the understanding of the data's distribution, as well as uncover patterns or relationships that could be present between columns.

A boxplot was used to check the distributions of numeric values, and as expected, we only had 0s and 1s. This means the dataset does not contain any other value and is suitable for our study as we want to perform binary classification. Figure 7 shows the distributions of number in the dataset.

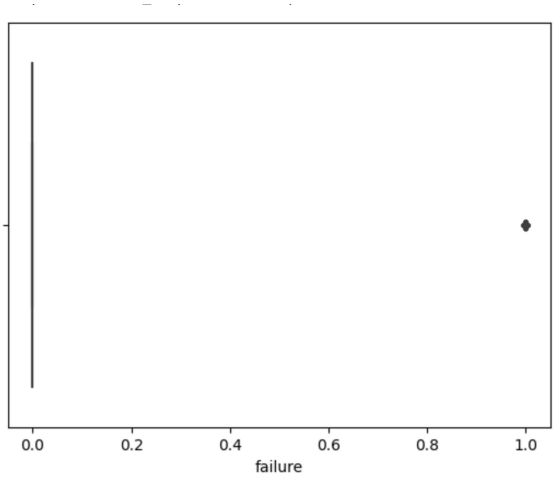


Figure 7: Boxplot

Seaborn's Countplot was employed to compare values within the dataset, and by utilizing this method, we were able to visually compare the distribution of values. Figure 8 shows the countplot

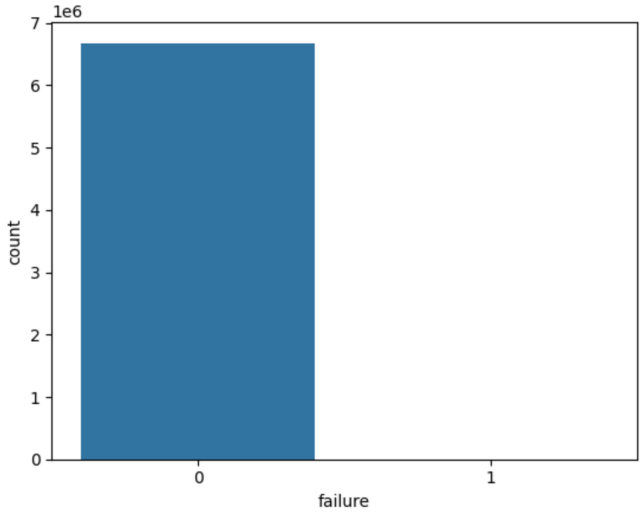


Figure 8: Countplot

We can see from the countplot that the distribution between good and failed drives is quite high which means the dataset is very imbalanced.

We then used correlation analysis which is a statistical method used to determine if there is a relationship between two or more columns and how strong that relationship is. Seaborn was used to create a correlation heatmap, which is a graphical representation of the correlation matrix.

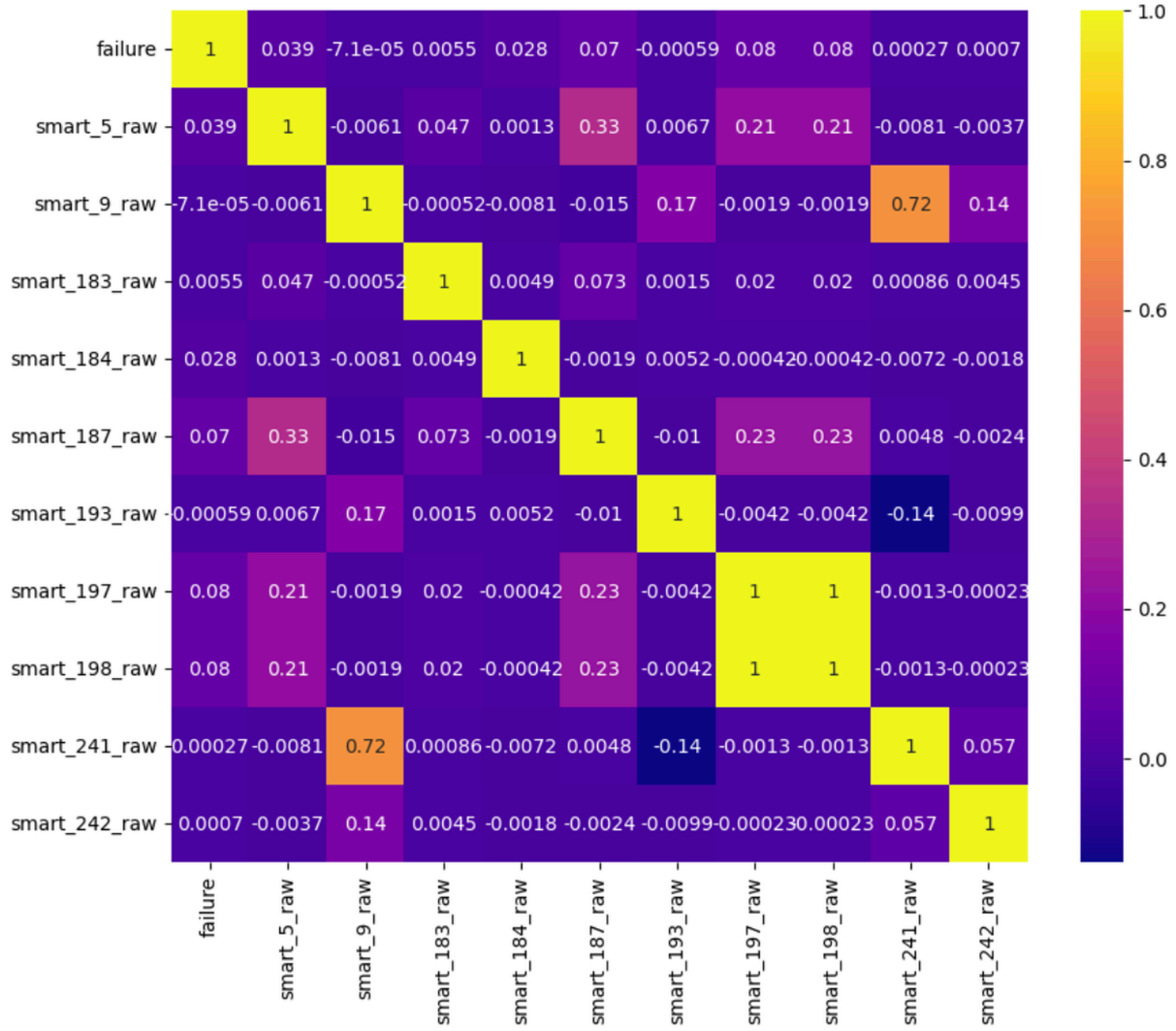


Figure 9: Correlation heatmap

From the heatmap (Figure 9), we can visually see that only two columns (smart_241_raw and smart_9_raw) are highly correlated with 72% confidence. Smart_5 was correlated with 3 columns (smart_187, smart_197 and smart_198) with low confidence of 33%, 21% and 21% respectively. Smart_187 was also corelated with smart_197 and smart_198.

To get a better understanding of correlations of those columns, a paiplot was used (Figure 10). Pairplot is basically a scatter plot used to find the correlations visually. We can see the same results of columns correlations from pairplot as we saw in from correlation heatmap but with more focused details for example, the relation of smart_5 with smart _187 and smart _197.

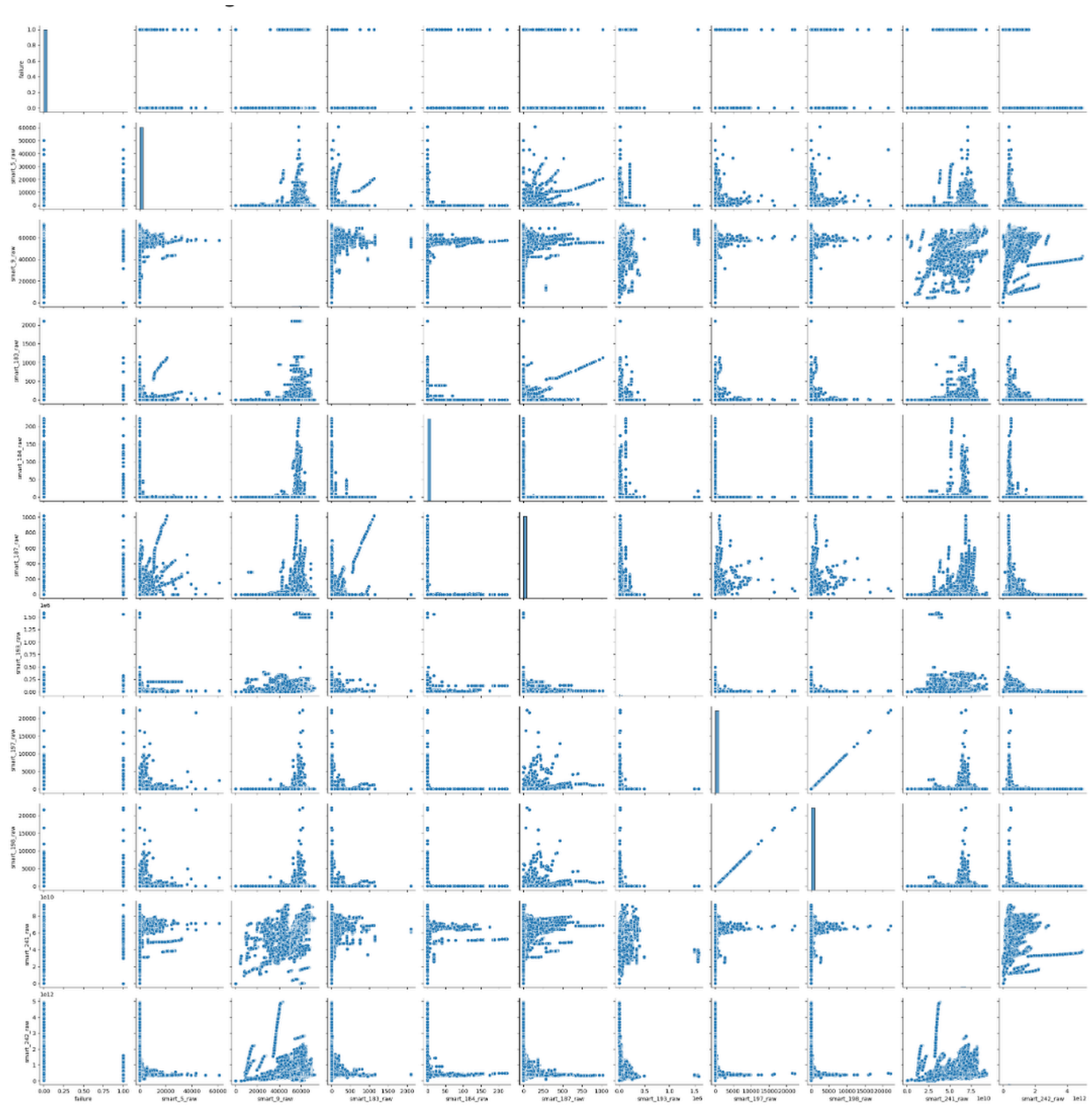


Figure 10: Pairplot

However, we did not get high correlations between the columns which could be because of the backblaze dataset which is highly imbalanced.

3.4 Stage 4: Modelling or ML models Considered

We evaluated a range of machine-learning models to predict HDD failures, such as the Balanced Random Forests (BRF) as well as decisions trees, Gradient Boosted Decision Tress along with random forests. We assessed the models on their geometric mean score. As discussed in section 2.4, many studies have utilised these ML models in the past have seen quite good results. In this study, we will focus more on only two ML models which have been considered to help predict HDD

failure by (Aussel et al. 2017) and (Ahmed and Green 2022). We will try to further boost the prediction accuracy by tuning with different hyper-parameters and feature selection techniques.

1. Random Forest (RF)
2. Balanced Random Forest (BRF)

3.4.1 Random Forest

Random forest (Breiman 2001) is an ensemble of unpruned regression or classification trees, generated by bootstrap samples from the training data using random feature selection during the process of tree induction. Prediction is achieved by adding (a majority vote for classification or the averaging of regression) the ensemble results. Random forest typically shows a significant improvement in performance over a single tree classifiers like CART as well as C4.5. It has a generalization error ratio which is comparable with Adaboost; however, it is more resistant to noise. Nevertheless, like many classification algorithms, RF can also suffer from the curse of learning by utilizing a highly imbalanced set of training data. Because it is designed to reduce the error rate overall, it tends to concentrate more on the accuracy of predictions for the majority of classes which can result in inadequate accuracy for the minority group. (Chen and Liaw 2004) To address this issue, one could use class weights with Random Forest. According to (Thakur 2020) One of the simplest methods to combat class imbalance is by using class weights, where we assign different weightage for different classes. The number of samples within the class is taken into consideration when calculating the class weights. We assign more weight to the minority class that places greater emphasis on the class. The classifier therefore gains the same information from both classes. Class weights regulate losses. When a minority class is misclassified and causing a greater loss, more losses are caused by the model as the minority group has more weight. This makes the model discover representations for the minorities. This results in somewhat lower performance for the majority of people. The easiest method of calculating the appropriate weights for classes is to utilize the Sklearn utility.

3.4.2 Balanced Random Forest

According to (Breiman 2001), a random forest generates every component of the tree out of a bootstrap sample of training data. In the case of learning data with extreme imbalance, there is a substantial possibility that a sample of the bootstrap is comprised of a few or none of the minority group and results in a tree that has low performance in predicting the class of minority. A simple method of solving this issue is to employ stratified bootstraps, i.e., sampling with the possibility of replacing each class within. However, this does not eliminate the imbalance issue completely. For

this classifier to artificially creating the class priors equal by down-sampling the majority of the class or over-sampling a minority class is generally more efficient for a specific performance measure, and that down-sampling appears to be more effective than the over-sampling. However, down-sampling a majority class could cause the loss of information because a substantial portion of the class that is majority-based is not utilized. Random forest was the inspiration for us to create group trees that were derived from balanced down-sampled data. Its Balanced Random Forest (BRF) algorithm is illustrated below:

1. For each iteration of the random forest, make a sample of bootstrap for the minority class. Then, randomly draw the exact number of instances in replacement from that class with the largest number of students.
2. Create a classification tree using the data in the maximum size, with no cutting. The tree is created using CART, the CART algorithm, but with the following modifications at each node: instead of going through all variables in search of the best split, just go through a list of randomly selected variables.
3. Repeat the steps mentioned above to the desired amount of times you want. Combine the predictions of the ensemble and create the final prediction. (Chen and Liaw 2004)

3.5 Evaluation

This section will discuss the criteria used to assess the predictive performance of the ML models we have considered. We evaluated the experimental results by the comparison of BRF and RF against various algorithms for ML and two advanced classifiers (weighted logistic regression and bagging classifier with Gradient boosting estimator). Gmean is particularly useful when data are imbalanced (Kubat and Matwin 1997; Liu et al. 2009). While other metrics, such as precision, accuracy, and recall, tend to favour the majority group.

3.5.1 Evaluation Metrics

We evaluated the performance of ML models by focusing on maximising the G-mean. Gmean assigns the same importance to all classes. Higher the G-mean, the better the model. G-mean is given by:

$$G - mean = \sqrt{((TP / (TP + FN)) * (TN / (TN + FP)))}$$

Where TP, stands for True Positive, FP stands for False Positive, TN stands for True Negative, and FN stands for False Negative. We also provided other related evaluation metrics for comparision, such as:

FDR. Failure Detection Rate $\left(FDR = \frac{TP}{TP+FN}\right)$:

Also known as recall rates. It measures the proportion of failed disks which are accurately predicted as failed. The greater the FDR is, the more accurate the model.

FAR. False Alarm Rate $(FAR = FP/FP + TN)$:

It is the percentage of good disks that are incorrectly predicted as failed. The less the FAR is, the more accurate the model.

F1-score. $(\frac{2\times FDR\times PP}{FDR+PP})$:

The F1 score takes the harmonic mean of precision and recall, which gives equal weight to both measures. The PP is the percentage of failed predictive disks which are accurately predicted to be failed. The higher the f1-score is, the better the model.

Precision. $(\frac{TP}{TP+FP})$:

It is defined as the proportion of the positive class predictions that were actually correct. A high precision score indicates that the model makes fewer false positive predictions. The higher the precision is, the more accurate the model is.

4 EXPERIMENTS

This section describes the experiments conducted in accordance with our study using data from Backblaze. The goal of this section is to give an overview on our results and comparison of different methods used.

2022 Hard drive data was collected from official BackBlaze website. PySpark, a distributed processing framework was used to load and describe data which fixed the issue of out memory issues Pandas could face as Pandas is not optimized for large dataset. Data selection and cleaning was performed as described in more detail in section 3.3.1 and 3.3.2. The selected dataset was then exported to CSV using PySpark. The loading time to read a CSV in Pandas was then drastically reduced which boosted the productivity and we managed to run the experiment locally using Pandas library.

The pre-processing was done in a pipeline which prevented data-leakage, pre-processing steps included:

(i) Imputing the features using Simple Imputer at 'constant' strategy and 'fill_value = 0' to improve the accuracy of data analysis and decision-making.

(ii) Scaling the features using maximum absolute value to fix the problem of feature scaling (Ahmed and Green 2022). To scale the data using the maximum absolute scaling method, each observation in the variable is divided by the maximum value of that variable, thereby scaling the data to its maximum value.

$$x_{scaled} = \frac{x}{\max(x)}$$

To reduce the time, it took for the model to train, eight cpu cores were used concurrently. Passing a simple parameter n_jobs=8 to the training classifier let the model to be trained parallelly on 8 cpu cores. This fixes slow runtime and slow training of models on machines with multiple cpu cores. The time taken by a model to train drops by increasing the number of CPUs. This can be done by using the parameter n_jobs = 8 or n_jobs = -1 (to use all cpu available). Inspired by (Ahmed and Green 2022) we select Cross validation strategy at 5 times and Gmean as a part of evaluation matrix.

After introducing our data pre-processing, the following sections, we will explain different experiments. Experiment 1 verifies the existing ML methods on new but limited dataset. Experiment 2 verifies the ML methods with different sampling methods and Experiment 3 analyses different sampling ratio on selected models.

4.1 Experiment 1: Using new data sets to verify existing HDD methods

The goal of this experiment was to get an overview of methods we used on a small dataset consisting of hard drives data in quarter 1 of 2022. We were constrained by limited time and computing resources, so we solely utilized the backblaze 2022 Q1 dataset (1/Jan/2022 to 31/Mar/2022) for this experiment. For our first experiment, we conducted two tests: t1, which includes all features, and t2, which only included five features (Smart 5, Smart 187, Smart 188, Smart 197, and Smart 198) recommended by Backblaze (Klein 2016). Both tests had identical datasets with the same sample size, positive samples, and IR (Imbalance Ratio).

t1: Dataset of 2022 Q1 backblaze data and all features (only raw values)
t2: Dataset of 2022 Q1 backblaze data and only 5 features (5, 187, 188, 197, and 198)

Table 6 shows the dataset used in experiment 1 with sample size and IR (Imbalance Ratio)

Table 7: Experiment 1 dataset

Test	Sample size	Positive samples	IR	Number of attributes
t1	1670395	117	14275.88 : 1	75
t2	1670395	117	14275.88 : 1	5

Our methodology for experiment 1 is shown in Figure 11.

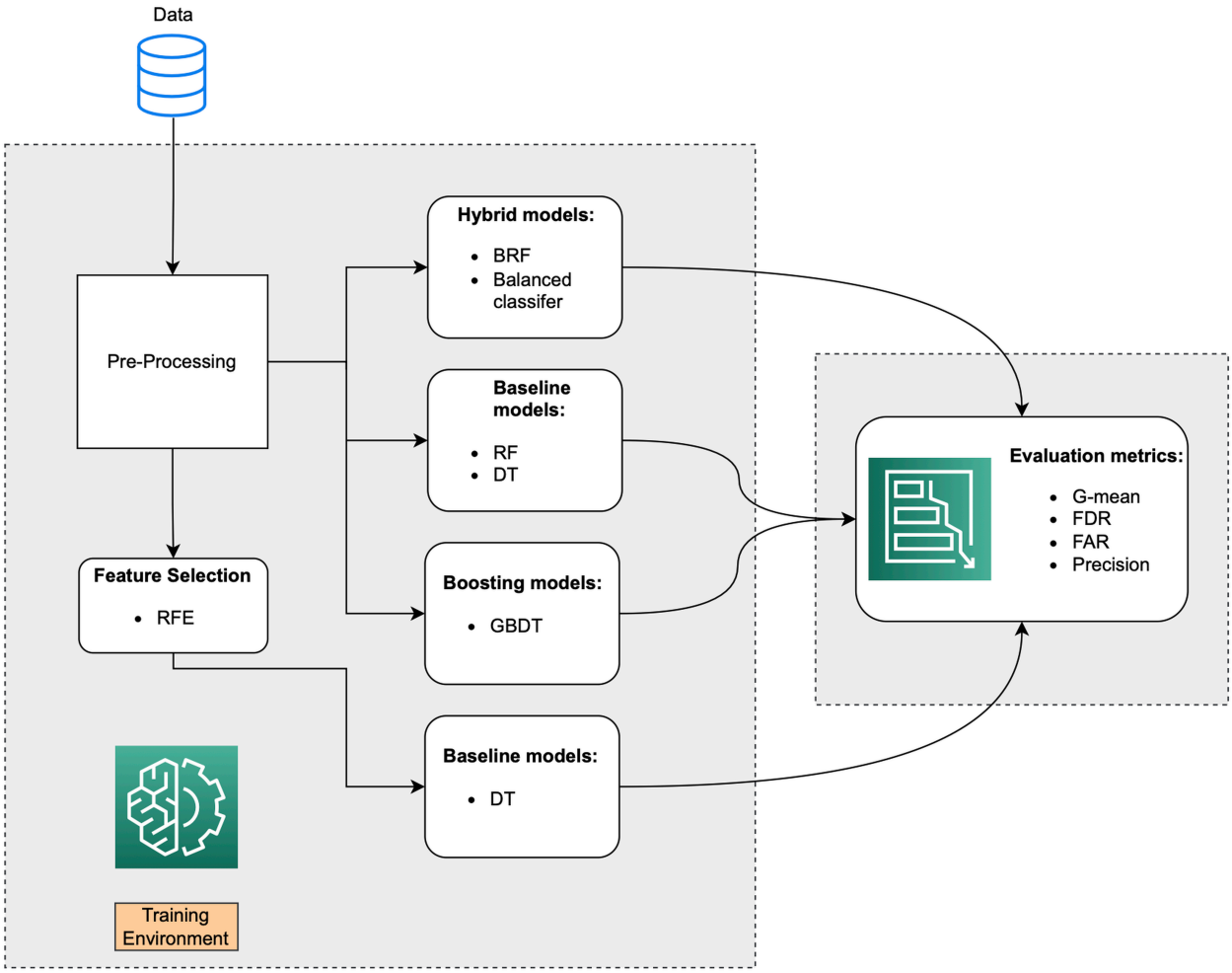


Figure 11: Overview of Methods on Limited HDD Dataset.

This experiment is inspired by the previous HDD analysis research (Aussel et al. 2017; Yang et al. 2021; Ahmed and Green 2022). We used newer dataset (2022 vs 2014) for verifying whether the methods are still valid. Four machine learning methods were used in this experiment RF (Aussel et al. 2017), BRF (Ahmed and Green 2022), DT (Li et al. 2017), GBDT (Yang et al. 2021) and RFE (Yang et al. 2021) was used as a feature selection on one machine learning method (DT). We also experimented by using Balanced Bag of Decision Trees (BBC) (Lemaître et al. 2017).

Table 7 describes the performance of different ML methods used in terms of $G_{mean,}$ FDR, and FAR at CV=5

Table 8: Overview of Methods in a Limited Dataset

	G_{mean}		FDR or Recall		FAR	
Method	t1	t2	t1	t2	t1	t2

BRF	**0.887**	**0.907**	0.8217	0.8728	0.03981	0.0560
RF	0.000	0.101	0.0	0.0260	4.789621 4889462 21e-06	1.4967
RFE (10 features) + DT	0.284	N/A	0.0855	N/A	0.00012	N/A
DT	0.254	0.258	0.0684	0.0684	0.0001	6.5258
GBDT	0.376	0.306	0.1539	0.1021	0.00012	8.8608
BBC	**0.881**	**0.907**	0.8217	0.8391	0.0527	0.0446

In terms of Gmean in both tests, the BRF method had the highest performance 0.887 and 0.907 for t1 and t2 tests respectively. BRF at both tests had the highest FDR (or Recall) at CV=5. Followed by BBC classifier at 0.881 and 0.907. BBC was only behind to BRF suggesting the balanced bag of decision trees and balanced bag of Random Forests are not really far from one another with BRF taking a slight improvement. RF could not perform well with all the features but with 5 features only. RFE+DT was at 0.284 and GBDT could only get to 0.376 in terms of G-mean.

Table 9: Experiment 1 training time

Model Name	**5 Features**	**All Features (75)**
BRF	2min 48s	3min 18s
Random Forest	37.6 s	4min 9s
DT	**5.22 s**	4min 3s
RFE + RF	N/A	12min 31s
GBDT	32.2 s	**58min 40s**
BBC	4.69 s	2min 59s

The table 8 provides information about the training times of six different machine learning models for two different feature sets: a small one with five features and a larger one with 75 features.
All models take longer time to train on larger dataset which was expected since it had a greater number of features to train the model. GBDT was extremely affected by the increase in number of features, it took almost an hour to train GBDT on 75 features on Q1 2022 dataset. Random Forest took 11 times more time to train on larger feature set. DT was the fastest to train on 5 features with

just 5.22s. Another interesting finding of this experiment was that BRF was not much affected by the number of features (2min 48s and 3min18s), highlighting the excellence of performance with greater feature set.

Based on the experimental results (Table 7), the BRF method was found to be the best for predicting hard drive failure as we expected and suggested by (Ahmed and Green 2022). Random Forest performed worst compared to others which was against the suggestion made by (Aussel et al. 2017). This could be due to the fact that the dataset is very imbalanced. To improve the accuracy of the predictions, further experimentation and refinement may be required.

It was evident from Experiment 1 that the BRF was way ahead of other classifiers in predicting the hard drive failure. The outcome was straightforward and anticipated, as BRF used a balanced bag of bootstrap samples while training. However, it is to note that it is not fair to compare BRF with other classifiers by default, which was done here and by (Ahmed and Green 2022) in their paper, as BRF uses the Random Under Sampler sampling strategy to sample the majority class (Lemaître et al. 2017; Agusta and Adiwijaya 2018) while the other classifiers were facing the curse of the imbalanced dataset. BBC gave comparable results with BRF with only slightly low G-mean than BRF.
It is also to be noted that the 5 smart feature column recommendations by Backblaze (Klein 2016) gave comparable results as with all features and 10 selected features by RFE; however, there was some information in other smart columns as well which is very similar case with (Aussel et al. 2017). This demonstrates that SMART functions are constructed in different ways on different drives, meaning the conclusion on features that can be used to predict failures specific to a certain kind of hard drive can't be easily generalized to all types of a hard disk (Aussel et al. 2017).
From the results it is verified that the balanced sampling could give us better results. In the further experiments we could check with different size of data i.e. the full 2022 dataset with different IR and sampling to check how do they perform on the methods. We could also check using feature selection method. This motivates us to go to experiment 2 where we apply full 2022 dataset and feature selection method.

4.2 Experiment 2: Using feature selection and sampling methods

In this experiment, the objective was to further refine and gain a deeper understanding of the predictive models for hard drive failures, utilizing whole 2022 data collected from BackBlaze. All pre-processing tasks from previous experiment, such as Simple Imputer and Maximum Absolute Scaler (Ahmed and Green 2022), were applied. RFE (Yang et al. 2021) was utilized as the feature selection method due to the belief that there may be relevant information in columns beyond the 5 recommendations provided by BackBlaze (Klein 2016; Aussel et al. 2017) . The full 2022

BackBlaze dataset, spanning from 1/Jan/2022 to 31/Dec/2022, was utilized for this experiment, with RFE set up to select the top 10 features among all. Table 9 shows the features that RFE selected:

Table 10: RFE selected features

smart_5_raw	smart_9_raw	smart_183_raw	smart_184_raw	smart_187_raw	smart_193_raw	smart_197_raw	smart_198_raw	smart_241_raw	smart_242_raw

We used two sampling methods (Random Under Sampling (Zhang et al. 2020) and Smote Bagging (Aussel et al. 2017)) for comparison and to have a fair comparison with BRF as it uses random under sampling by default (Lemaître et al. 2017; Agusta and Adiwijaya 2018).
Based on the results from Experiment 1 and previous literature, only the classifiers previously discussed (RF (Aussel et al. 2017), BRF (Ahmed and Green 2022), WLR (Ahmed and Green 2022)) were considered for Experiment 2. We also experimented by using a Balanced Bag of Histogram Gradient Boosting instead of decision trees as GBDT was performing better than DT in experiment. It is an ensemble method that uses histograms to boost GBDT training time.
Methodology for experiment 2 given in figure 12.

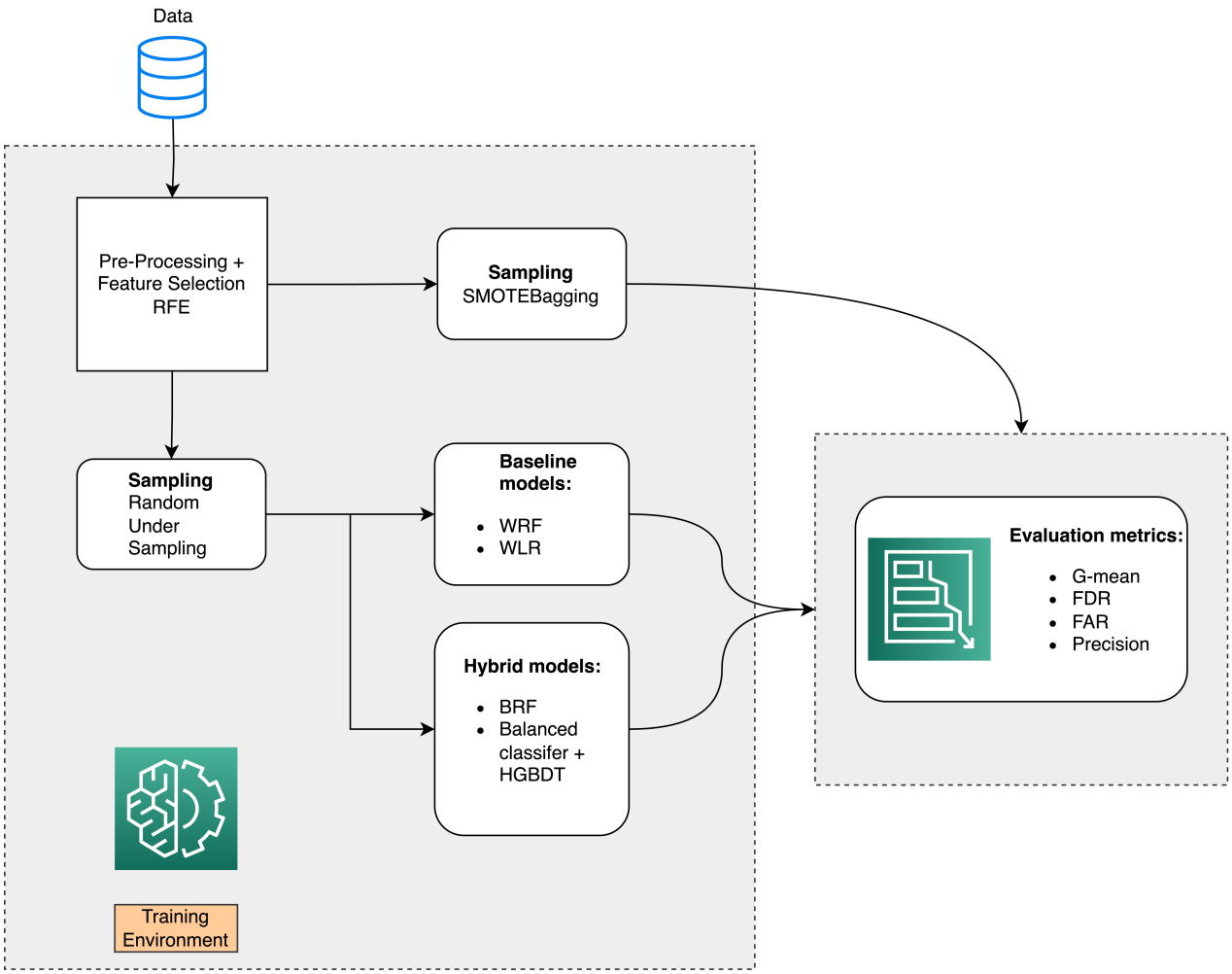


Figure 12: Experiment 2 setup

Each model was cross-validated 5 times, with G-mean, FDR/Recall, FAR, and Precision serving as the evaluation metrics. The dataset consisted of 6,678,738 non-failures and 632 failure samples. Table 10 displays the performance of each model with 10 features.

Table 11: Performance of ML models

Model	Gmean	F1	Precision	FDR	FAR	Recall
Balanced Random Forest	**0.913083**	0.003422	0.001714	0.876628	0.048611	0.876628
Weighted Random Forest	0.000000	0.000000	0.000000	0.000000	N/A	0.000000
Weighted Random Forest with Random under sampling	**0.915038**	0.003613	0.001810	0.878215	0.046236	0.878215

Model	Gmean	F1	Precision	FDR	FAR	Recall
Weighted Logistic Regression	0.901823	0.006883	0.003456	0.832321	0.022710	0.832321
Weighted Logistic Regression with random Undersampling	0.747562	0.014172	0.007177	0.563317	0.007444	0.563317
Balanced Bagging with Hist Gradient	**0.914767**	0.005266	0.002641	0.863992	0.031159	0.863992
Smote bagging	0.152879	0.034998	0.057117	0.025297	N/A	0.025297

According to the data from Table 10, some models (WRF, BRF, BBHG, WLR) performed better than others. The weighted RF combined with random undersampling showed exceptional performance and had the highest G-mean of 0.9150 (only slightly below BRF which was 0.9130) as seen in Table 11. Balanced Hist GBDT also exhibited strong overall performance of 0.9147 proving its success. WLR score a score of 0.9018 without any sampling method used. Interestingly, this experiment revealed that each model's performance was highly dependent on its sampling method used; for example, the random undersampling + weighted RF outperformed regular weighted RF across all metrics by a significant margin! It is worth noting that even an RF with only five features could make accurate predictions while SMOTE-Bagging (Aussel et al. 2017) performed poorly across all metrics indicating it may not be ideal for predicting hard drive failures accurately. To compare this study's results against those of (Ahmed and Green 2022) using RFE + BRF - we achieved higher accuracy rates at 0.91 compared to their score of only 0.86 - proves quite impressive indeed!

Figure 13 shows the performance of models with 10 features selected.

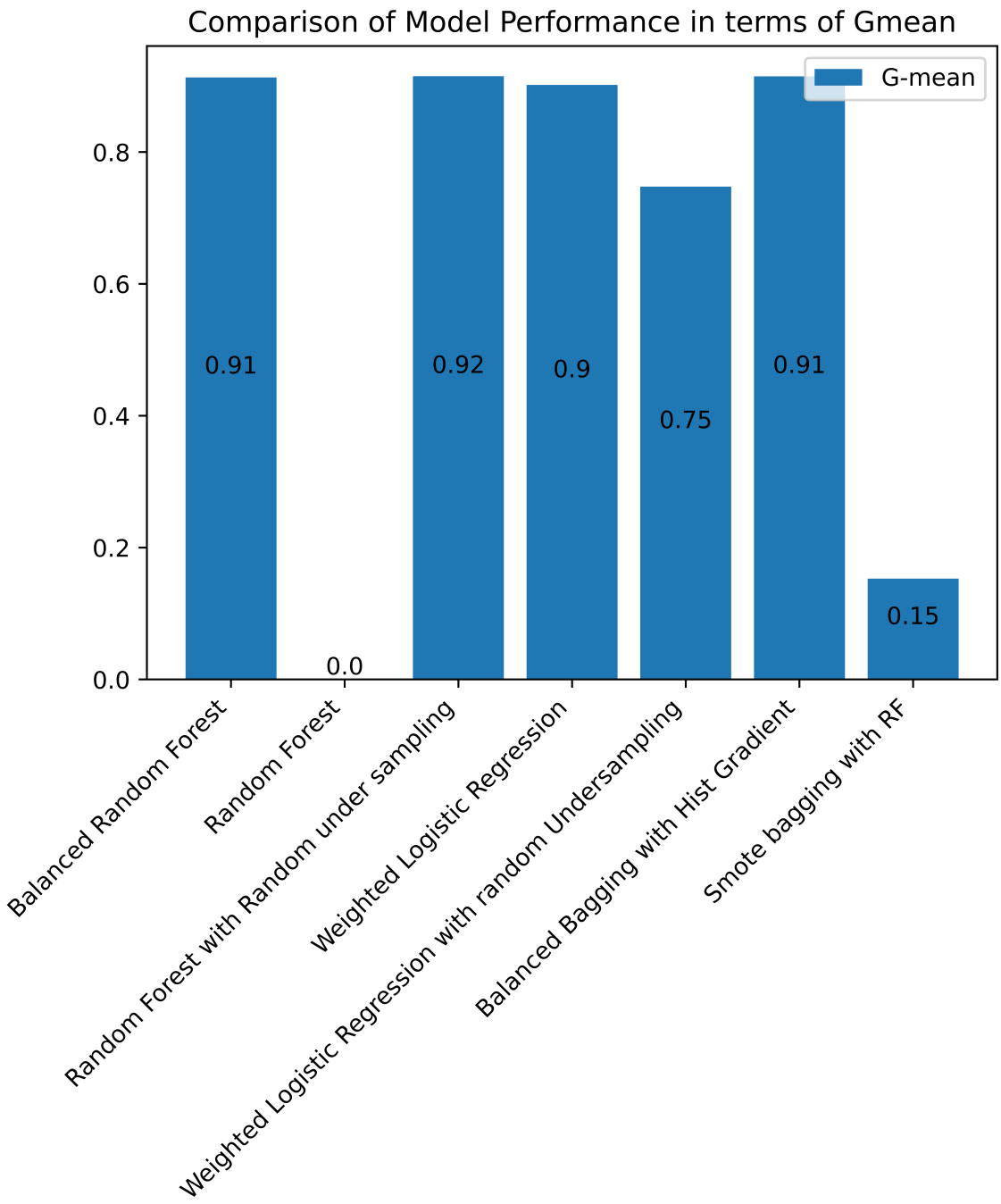


Figure 13: Performance of different models with 10 features

We compared the results obtained from two different feature selection techniques. Specifically, we evaluated the performance of Recursive Feature Elimination (RFE) and the 5 SMART backblaze recommended features, using the same dataset. The 5 SMART backblaze recommended features include Smart 5, Smart 187, Smart 188, Smart 197, and Smart 198. The table 11 below shows the results of same models with 5 Features:

Table 12: Performance of ML model for 5 feature

Model	G-mean	F1	Precision	FDR	FAR	Recall
Balanced Random Forest	**0.86659641**	0.00293457	0.00147001	0.79142607	0.05104547	0.79142607
Random Forest	0.21267328	0.00056254	0.00028304	0.04743157	0.01683543	0.04743157
Random Forest with Random under sampling	**0.86208142**	0.00282582	0.00141 55	0.78508936	0.05318201	0.78508936
Weighted Logistic Regression	**0.85175711**	0.00566345	0.00284255	0.74404449	0.02473222	0.74404449
Weighted Logistic Regression with random Undersampling	0.74409655	0.01476169	0.00748	0.55771779	0.00702 23	0.55771779
Balanced Bagging with Hist Gradient	**0.85841974**	0.00355655	0.00178251	0.76932883	0.04186718	0.76932883
Smote bagging	0.21914964	0.00568046	0.00314528	0.04896888	0.00422338	0.04896888

Results from the table 11 shows exact same pattern as of table 10 i.e the BRF, WRF, WLR and BBHG were among the top performing model with BRF being the highest in terms of Gmean. WLR again had the same inconsistency of the performance with balanced and imbalanced datasets. However this time WRF without any undersampling could make some predictions.

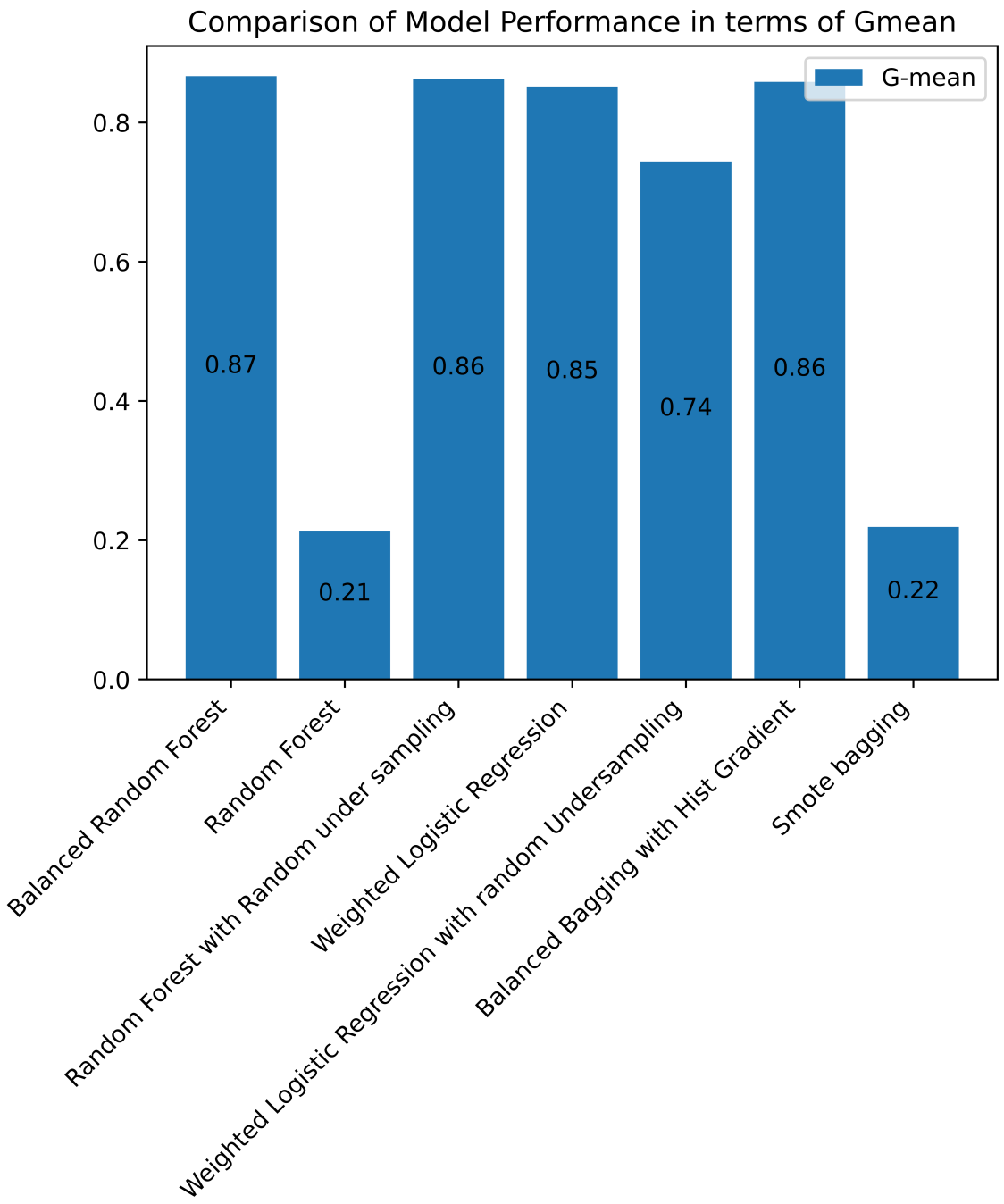


Figure 14: Performance of different models with 5 features

Our analysis revealed that RFE performed better than the 5 SMART backblaze recommended features up to some extent. Only one model (RF) performed better with only 5 features rather than 10. This finding suggests that both feature selection techniques can be effective for predicting HDD failure, with RFE having a slight advantage over the 5 SMART backblaze recommended features. These results also suggest that careful consideration must be given to the choice of sampling method when developing predictive models for hard drive failures.

Table 13: Experiment 2 training time

Model Name	5 Features	10 Features
BRF	3min 27sec	3min 36s
Random Forest	~13 mins	19min 51s
RF + Random Under sampling	28.3 s	33.5 s
Weighted Logistic Regression	27 s	7min 25s
WLR + RUS	7.14 s	24.4 s
BBHG	2min 50s	4min 6s
Smote Bagging	11min 35s	49min 32s

Table 12 displays the training time of machine learning models for two different datasets with different numbers of features (5 and 10). Two models (RF and WLR) were also trained with Random Under Sampling (RUS). WLR combined with Random Under Sampling was the fastest to train at 7.14s and 24.4s with both sets of features. WLR without sampling had the highest difference between training time between the feature set 27s and 7min25s, which is ~16 times extra. BRF, RF, and RF+RUS did not really have much of a difference in terms of training time with both feature sets. Smote Bagging was the slowest among all, with 11min 35s and 49min 32s for 5 and 10 features, respectively.

The results of our experiment 2 indicate that selecting the appropriate sampling techniques plays a more crucial role in the performance of predictive models for hard drive failures than choosing between 5 or 10 features. While our study showed that the RFE-selected features outperformed the 5 SMART backblaze recommended features, the difference in performance was not substantial. On the other hand, the choice of sampling method had a more significant impact on the models' performance. As the Backblaze data is highly imbalanced and getting more imbalanced than before, therefore selecting appropriate sampling methods will be crucial. This suggests investing time and effort into selecting the appropriate sampling method could lead to more accurate and reliable predictive models for hard drive failures.

4.3 Experiment 3: Analysing different sampling ratio on selected models

Experiment 3 goes further deeper into analysing the effect of different sampling strategy ratios on the models considered. The dataset remained the same from the previous experiment, i.e. 2022 Backblaze dataset and 10 SMART features. All the pre-processing was again all same from previous experiments. We only considered the Random Under Sampling strategy to have a fair

comparison of models as BRF uses Random Under Sampling by default. Due to limited computing resources and time, we only considered the three models in experiment 3: Random Forest, Balanced Random Forest and Weighted Logistic Regression. Figure 15 describes the design we used for this experiment.

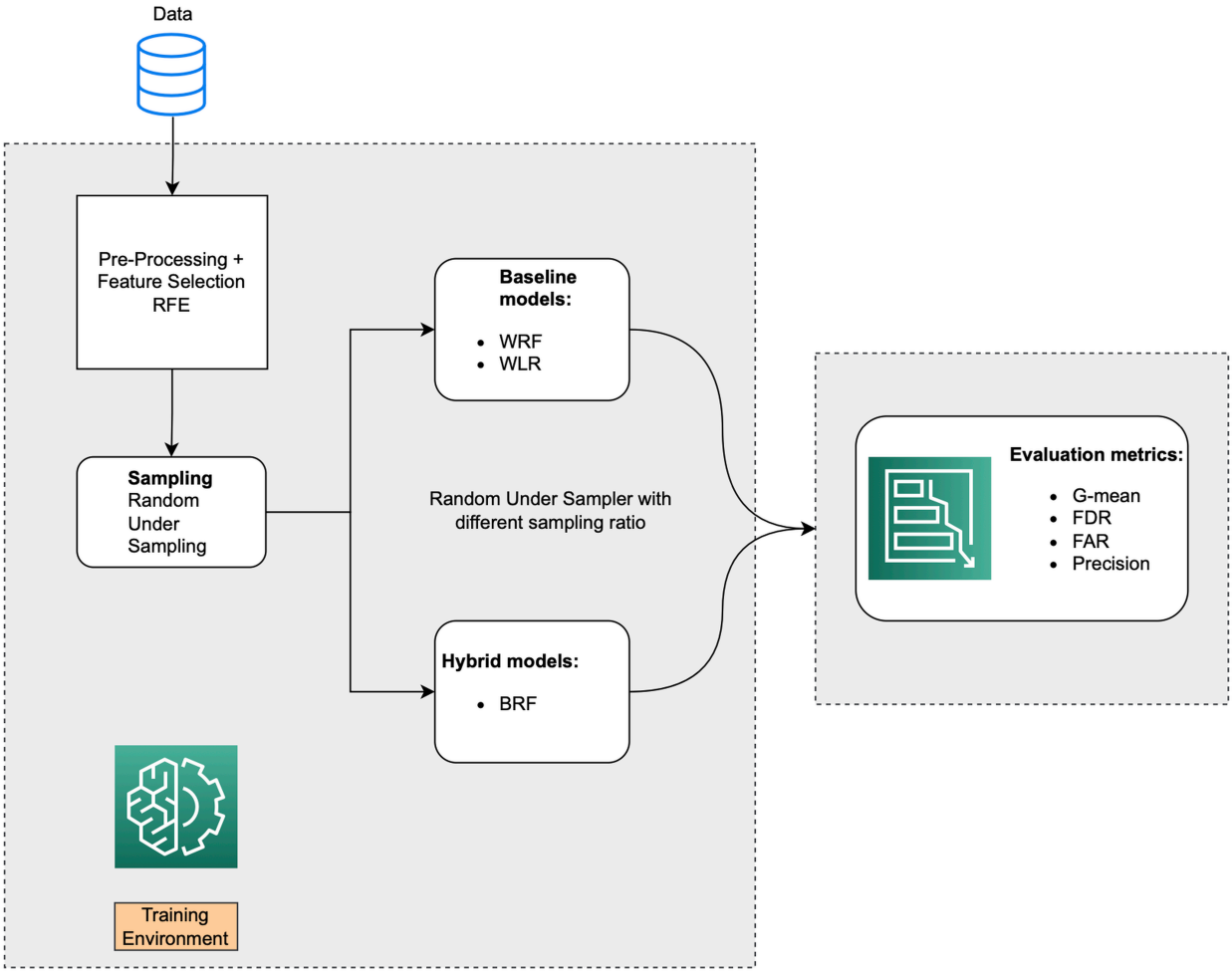


Figure 15: Experiment 3 Setup

The table 13 below summarises the results of experiment 3 with 3 machine learning models trained on different sampling ratios. We trained the model while cross-validating it 5 times and using different sampling ratios. The sampling ratios used in training range from 1000:1 (where the majority class is down-sampled to match 1000 with 1 of its minority class) to 1:1 (where both classes are evenly represented in the training data).

Table 14: Model and sampling performance

Model and sampling strategy	G-mean	F1	Precision	FDR	FAR	Recall
Balanced Random	0.2268360 2	0.0421288 3	0.0353544	0.0522184 7	0.0001338 6	0.0522184 7

Forest with 1000:1 sampling						
Random Forest with 1000:1 sampling	0.21756259	0.0315375	0.02367015	0.04748156	0.00018506	0.04748156
Weighted Logistic Regression with 1000:1	**0.88761602**	0.00705545	0.00354325	0.80548681	0.0214262	0.80548681
Balanced Random Forest with 100:1 sampling	0.62355555	0.03800467	0.01997457	0.39082615	0.00181516	0.39082615
Random Forest with 100:1 sampling	0.55844697	0.03117848	0.01640217	0.31478565	0.00179211	0.31478565
Weighted Logistic Regression with 100:1	**0.87086678**	0.00942773	0.0047429	0.77069116	0.01529436	0.77069116
Balanced Random Forest with 10:1 sampling	**0.89495121**	0.01223958	0.00616634	0.81179853	0.0123694	0.81179853
Random Forest with 10:1 sampling	0.86773249	0.01236172	0.00623166	0.76279215	0.01152673	0.76279215
Weighted Logistic Regression with 10:1	0.80736895	0.01275796	0.00644156	0.65826772	0.00962128	0.65826772

Balanced Random Forest with 4:1 sampling	**0.91256845**	0.0089937 8	0.0045209 5	0.8481814 8	0.0177166 7	0.8481814 8
Random Forest with 4:1 sampling	0.8980905 5	0.0089859 7	0.0045178 7	0.8213098 4	0.0171995 1	0.8213098 4
Weighted Logistic Regression with 4:1	0.7801775 6	0.0137841 6	0.0069704 7	0.6139357 6	0.0082837 5	0.6139357 6
Balanced Random Forest with 3:1 sampling	**0.91212412**	0.0077992 6	0.0039177 2	0.8497687 8	0.0205264 8	0.8497687 8
Random Forest with 3:1 sampling	0.9015097 7	0.0074479 6	0.0037409 4	0.8307711 5	0.0211499 5	0.8307711 5
Weighted Logistic Regression with 3:1	0.7691059 7	0.0138756 9	0.0070197 3	0.5965004 4	0.0080037 6	0.5965004 4
Balanced Random Forest with 2:1 sampling	**0.91333447**	0.0059890 4	0.0030050 7	0.8576677 9	0.0270092 9	0.8576677 9
Random Forest with 2:1 sampling	0.9111919 7	0.0061428 5	0.0030826 3	0.8529058 9	0.0263159	0.8529058 9
Weighted Logistic	0.755968	0.0141033 7	0.0071393 6	0.575978	0.0075957 5	0.575978

Regression with 2:1						
Balanced Random Forest with 4:3 sampling	**0.91547826**	0.0045033 9	0.0022575 9	0.8703037 1	0.0366925 6	0.8703037 1
Random Forest with 4:3 sampling	0.9148721 5	0.0045352 3	0.0022735 8	0.8687289 1	0.0361993 5	0.8687289 1
Weighted Logistic Regression with 4:3	0.7496677	0.0139941 9	0.0070852 4	0.5664916 9	0.0075493 3	0.5664916 9
Balanced Random Forest with 1:1 sampling	0.913083	0.0034217	0.0017142 2	0.8766279 2	0.0486105 3	0.8766279 2
Random Forest with 1:1 sampling	**0.91503836**	0.0036125 5	0.0018100 4	0.8782152 2	0.0462358 3	0.8782152 2
Weighted Logistic Regression with 1:1	0.7475618 7	0.0141721 7	0.0071774 2	0.5633170 9	0.0074442 2	0.5633170 9

The BRF model showcased its exceptional G-mean (0.91547826) with a 4:3 sampling ratio, suggesting excellent performance when the sampling is more balanced. However, the G-mean values decreased as the sampling ratio became more uneven (e.g., 100:1, 1000:1), indicating that the BRF model's performance declined when confronted with higher levels of class imbalance. Figure 16 shows the graph of the performance of balanced random forest with different sampling strategy.

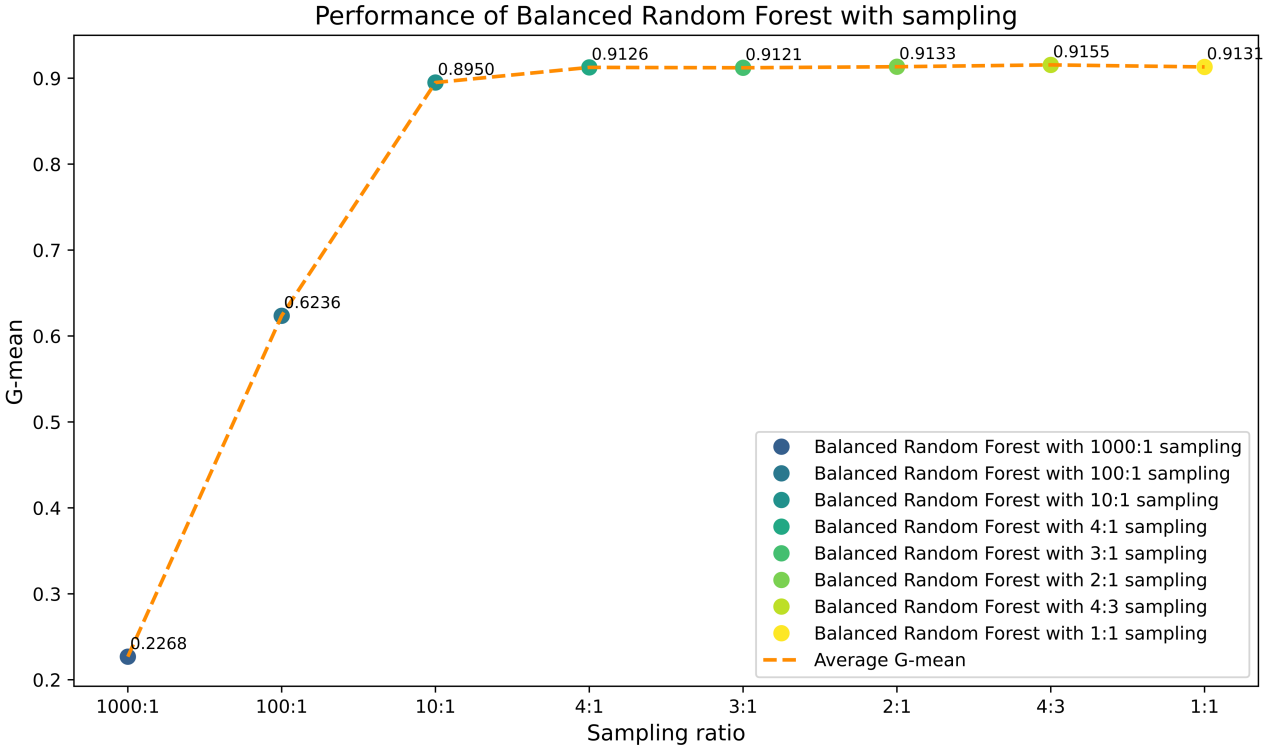


Figure 16: Performance of BRF with sampling

The RF model's G-mean values followed a similar pattern to BRF, with the highest G-mean (0.91503836) achieved at a 1:1 sampling ratio. As the sampling ratio became more imbalanced, the RF model's G-mean values declined, indicating that higher class imbalances did not result in better performance. Figure 17 shows the performance of RF with different sampling strategy

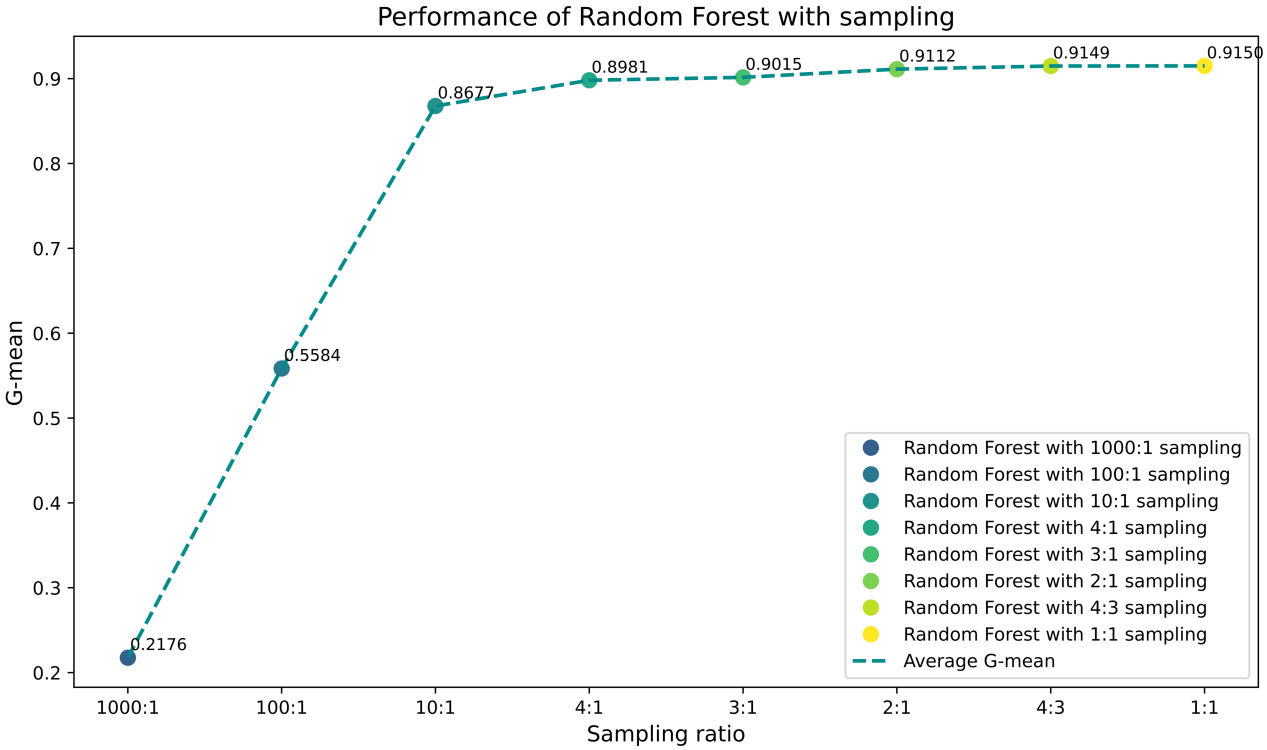


Figure 17: Performance of RF with sampling

On the other hand, Weighted Logistic Regression (WLR) displayed a different pattern. The results revealed that WLR's performance improved with an increase in the sampling ratio, with the model performing best at a high imbalance sampling ratio (0.88761602 at 1000:1). However, a more balanced dataset did not allow WLR to outperform BRF and RF models.
Figure 18 shows the performance of WLR at different sampling strategy.

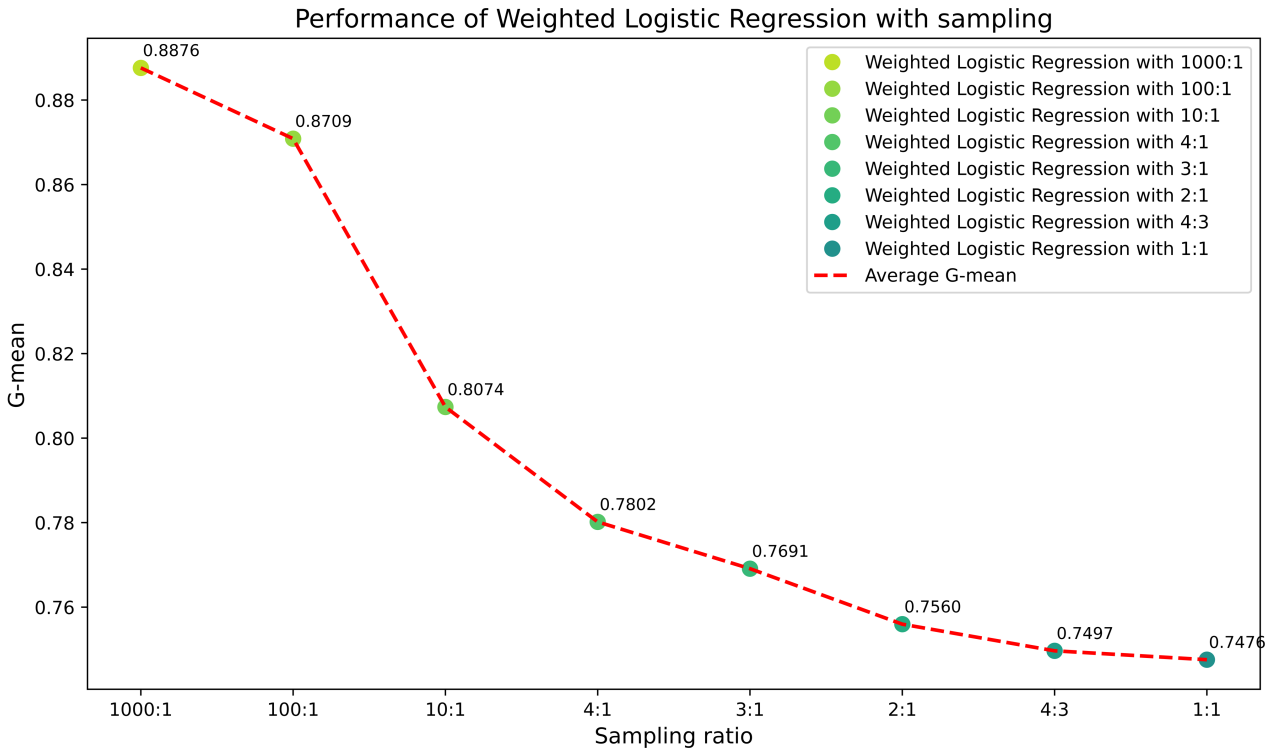


Figure 18: Performance of WLR with sampling

This inconsistent performance suggests that WLR is affected by the inherent data complexity. (Muchlinski et al. 2016; Ranganathan et al. 2017; Ahmed and Green 2022). The results for WLR were similar to those (Ahmed and Green 2022) ; however, BRF did not perform very well when the sampling was highly imbalanced.

Figure 19 shows the effect of different sampling strategy on WLR, RF and BRF. At around ~25:1, all three models were having similar G-mean performance ~0.82.

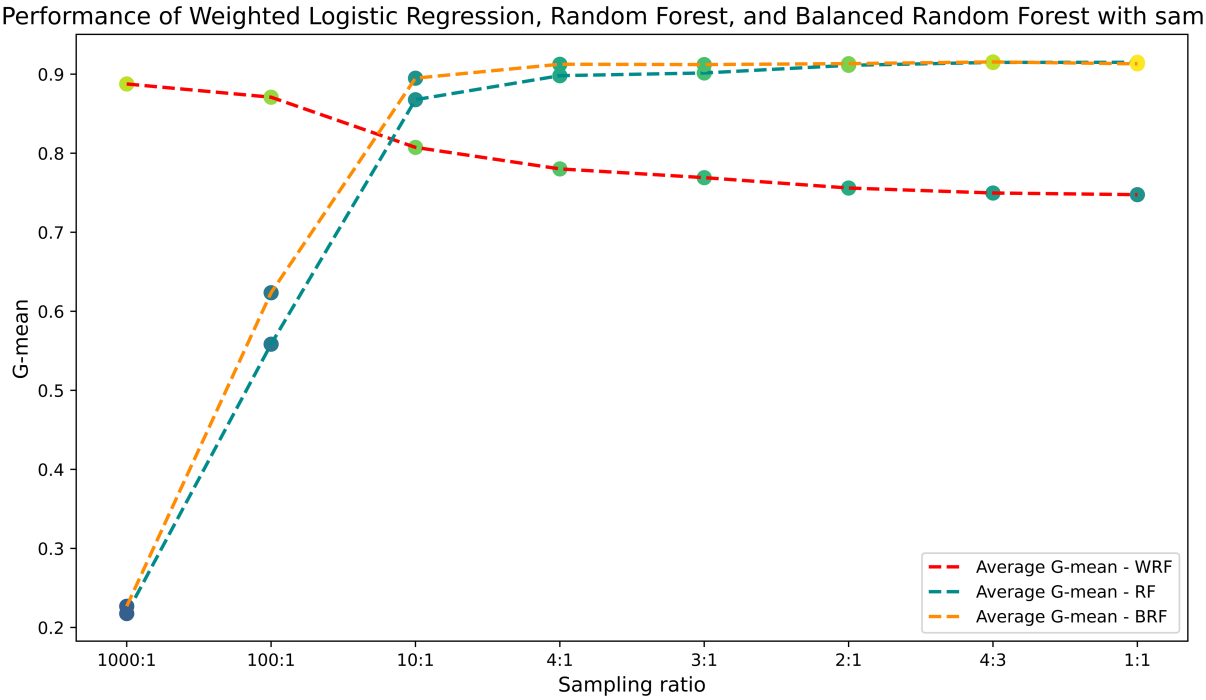


Figure 19: Performance of BRF, RF and WLR

The table 14 presents the training times of three machine learning models, Balanced Random Forest (BRF), Random Forest, and Weighted Logistic Regression (WLR), for various sampling ratios. The sampling ratios represent the imbalance between the positive and negative classes in the dataset.

Table 15: Experiment 3 training time

Sampling	BRF	Random Forest	WLR
1000:1	6min 22s	2min 12s	48.3 s
100:1	4min 2s	40.5 s	25.6 s
10:1	3min 51s	33.6 s	24.2 s
4:1	3min 50s	33.1 s	24.3 s
3:1	3min 49s	33.2 s	24.1 s
2:1	3min 49s	33.1 s	24 s
4:3	3min 49s	33.1 s	24 s
1:1	3min 49s	33.1 s	24 s

The table 14 reveals that the training time for each model does not change significantly with the sampling ratio. The model's training time is relatively stable for all the sampling ratios except for the extreme case of 1000:1, where the training time of BRF increases significantly to 6 minutes and 22 seconds. 2minutes and 12 seconds for RF and 48.3 seconds for the WLR.

Comparing the training times of the models, it is apparent that Random Forest and WLR have relatively similar training times across all sampling ratios. On the other hand, BRF has significantly

longer training times than the other two models, but it provides better performance in handling imbalanced datasets.

The findings from Experiment 3 in our study demonstrate that the sampling ratios substantially impact the classifier's performance. At around ~25:1 sampling ratio, all three methods were quite close to each other in terms of G-mean. The classifier's performance generally improved as the ratio approached a more balanced distribution. However, it is essential to note that the Weighted Logistic Regression (WLR) classifier served as an exception to this trend, as its performance was negatively affected by the balanced ratios. This suggests that while balancing the sampling ratios can enhance the performance of most classifiers, specific classifiers, such as WLR, may require additional considerations to address the challenges posed by complex data.

5 DISCUSSION

In this section we will discuss about the study, objectives, limitations and future work.

(Ahmed and Green 2022) pointed out the significance of the most reliable metrics (Gmean) to accurately identify disk failures which we implemented in our study as a evaluation metric. (Aussel et al. 2017) recommended the usage of full attributes rather than only five attributes. Following the suggestion by (Aussel et al. 2017), we used full 75 attributes in conjunction with feature selection method (RFE) that (Yang et al. 2021) suggested. Using different sampling ratios was seen as a blind spot during the literature review which we researched in our study. We started our research with first experiment, verifying the available HDD methods on new but limited datasets followed by second experiment, using feature selection and sampling method on full dataset and then third experiment, analysing the effect of different sampling ratios on the algorithms.

Table 16: Comparison with previous studies

Study	Dataset used	IR	Sampling	Feature Selection	ML methods	Evaluation metric	Performance
(Aussel et al. 2017)	Backblaze 2014	5702.72 : 1	SMOTE	SMART attributes 5, 12, 187, 188, 189, 190, 198, 199 and 200	RF	FDR (Recall) and Precision	0.67, FDR
(Yang et al. 2021)	Backblaze 2014-2018	Not provided	Custom sampling	RFE	GBDT, RF	FDR	0.77, FDR (CV)
(Ahmed and Green 2022)	Backblaze 2017	11500.78 : 1	None	SMART 5, 187, 188, 197, 198	BRF, WLR, EE	Gmean	0.78, BRF, Gmean
Our Study	Backblaze 2022	10567.62 : 1	Random Under Sampling	RFE	BRF, RF, WLR	Gmean	0.91 RF 0.91 BRF 0.90 WLR

Table 17 provides of previous studies on the HDD prediction domain with our study using different datasets, sampling techniques, feature selection methods, machine learning algorithms, and evaluation metrics. The study by (Aussel et al. 2017) used the Backblaze 2014 dataset and applied the SMOTE sampling technique and random forest (RF) machine learning algorithm. The selected SMART attributes for feature selection were 5, 12, 187, 188, 189, 190, 198, 199, and 200. The study achieved an FDR of 67% and 95% precision. The study by (Yang et al. 2021) used Backblaze 2014 to 2018 dataset and applied a custom sampling technique with gradient boosting decision trees (GBDT) and RF machine learning algorithms. The study used recursive feature elimination (RFE) for feature selection and achieved an FDR of 77% on cross-validation. The study by (Ahmed and Green 2022) used backblaze 2017 dataset and applied hybrid machine learning algorithms like Balanced Random Forest (BRF), Weighted Logistic Regression (WLR), and Easy Ensemble (EE). They used G-mean as their evaluation metric and achieved 78% G-mean accuracy on the 2017 dataset. Our study inspired by previous studies, implement RFE + Random Under Sampling to achieve G-mean accuracy of 91% for RF and BRF and 90% for WLR.

The primary goal of our research was to reproduce the results of existing HDD failure prediction methods on newer dataset and verify if they are still a good choice for predicting the hard drive failures. We also aimed to identify the effect of sampling methods and feature selection methods on available HDD failure prediction methods as it was identified as a blind spot during our research in literature review. We used G-mean as the evaluation for our study however, provided different metrics for comparison.

In pursuit of these objectives, we leveraged a large 2022 Backblaze dataset containing of over 6,678,738 non-failures and 632 failure samples with an imbalance ratio of 10567.62 : 1. The study successfully managed to verify the previous HDD methods employed in the past and get their performance. Unfortunately, not all algorithms were tested against whole datasets due to limited time and computing resources. Although there were various experiments conducted using a diverse set of classifiers, the results provided valuable insights into which algorithms are appropriate for predicting hard disk drive failure with increasingly imbalance ratio. The more advanced hybrid ensemble models underwent rigorous testing on new 2022 dataset and demonstrated exceptional performance consistently across them, indicating its potential generalizability. The results were pretty similar to the study (Ahmed and Green 2022) however, they did not study the effect of different sampling ratio on those classifiers which motivated us to study the effect of sampling ratios. The suggestion by (Aussel et al. 2017) that the backblaze dataset contains information in other columns than those of five recommendation of backblaze (Klein 2016) was also gave us an improvement in performance but did not change it significantly.

The study implemented and tested a selection of class imbalance techniques, such as feature selection methods and sampling methods. These techniques were evaluated to an extent that was not documented in existing literature. Our analysis found that using the sampling techniques plays a more crucial role in the class imbalance issue of backblaze HDD dataset than the feature selection method. Using SmoteBagging, as recommended by (Aussel et al. 2017) did not improve the performance of the classifiers, this could be due to extremely high class imbalance. RFE, a feature selection method used in our study found to be better than five backblaze recommendations. Hybrid ensemble methods like balanced bagging classifiers were more successful in predicting the failures of HDD across different sampling ratios. Random Forest was more successful when the dataset was more balanced. Weighted Logistic Regression found to be inconsistent with different sampling ratios and showed inherent data complexity. The results suggests that other algorithms may perform similarly but with slightly less success when used in conjunction with sampling techniques and feature selection methods on highly imbalanced datasets.

One limitation of the study is that not all algorithms were tested with all class imbalance techniques due to time constraints and limited computing power. This may have resulted in biased sample of algorithms and limited scope of study. Secondly, while a selection of class imbalance techniques was implemented and evaluated extensively, there may be other methods that were not considered in this research. Finally, it should be noted that the findings are specific to the Backblaze 2022 HDD dataset used in this study and may not necessarily generalize to other datasets or contexts without further investigation. As for future work, researchers could explore additional class imbalance techniques beyond those examined here as well as investigate how these approaches might perform on different types of data sets with varying degrees of class imbalance between classes. Another interesting future research could include the use of deep learning methods like auto-encoders (Pereira et al. 2020) or LSTM (Hu et al. 2020; Coursey et al. 2021) to see how they perform on this dataset.

5.1 Critical Evaluation

The project's success will be assessed based by its ability to achieve the objectives of the project set at the start of the program. The project had five objectives:

1) Review existing HDD methods: pre-processing, feature selection methods and ML methods.

Our project successfully achieved in this goal of reviewing existing HDD methods. We dealt with best performing HDD methods from existing literature. We conducted a thorough investigation into several popular ML methods mentioned in Chapter 2 of our thesis. Despite our limited resources

and time constraints, we were able to explore these methods extensively through rigorous testing and experimentation.

2) Evaluate one HDD manufacturer using the latest Backblaze dataset, utilizing machine learning to predict failure with imbalanced data.

Our project successfully achieved the goal of performing analysis and evaluation on one manufacturer using the latest Backblaze 2022 dataset. We employed machine learning algorithms to predict failure in the context of highly imbalanced datasets and provided a detailed discussion of one specific model ST4000DM000, demonstrating the project's success in meeting this objective.

3) Compare and evaluate different models using the latest Backblaze dataset, which includes hard drives from various manufacturers.

Our project failed to achieve the goal of comparing different models proposed and evaluate their performance on different hard drives from different manufacturers. We could only learn and compare the model’s performance on one specific hard drive model ST4000DM000, due to limited time and resources. The extremely large dataset did not allow us to complete this goal in given period of time. This could be an interesting future research to see how the proposed method deal with different hard drives and datasets.

4) Compare the ML models while using different sampling ratios to see how they affect the given model

Our project achieved this goal of comparing ML models while using different sampling ratios. We compared three ML models with 8 sampling ratios at varying imbalance levels, addressing a gap identified in the literature review. The findings from this objective revealed the impact of sampling ratios on the performance of the models, providing valuable insights for future research and practical applications. This comprehensive comparison contributes to a deeper understanding of the relationship between sampling ratios and model performance, ultimately enhancing the effectiveness of ML models in handling imbalanced datasets.

5) Discover which potential pre-processing methods may further improve the performance.

Our project successfully achieved the goal of discovering potential pre-processing methods that could further improve performance. Our findings suggest that feature selection could be an important factor rather than relying on five Backblaze recommended features as it could lead to better results. Our analysis also suggests that sampling methods play a crucial role in addressing

class imbalance. We effectively combined Recursive Feature Elimination (RFE) and sampling techniques like RUS, resulting in improved performance. This demonstrates our project's success in identifying and implementing pre-processing methods that enhance the performance of machine learning models.

6 CONCLUSION

Our research started by investigating the available literature and background of the HDD methods in predicting failures which are available in Chapter 2. The nature of the large dataset was next to impossible to continue the research on local machines. A data engineering methodology was developed using the Apache PySpark framework and discussed in Chapter 3 to overcome the limitations of time and resources. Microsoft Azure was used during the data preparation and data modelling stages. MaxAbs scaler and Simple Imputer were used as data pre-processing methods. Inspired by (Ahmed and Green 2022) All data pre-processing was performed within a stratified 5-fold CV to prevent data leakage. We used G-mean as the evaluation metric for our study, and we reported FDR, FAR, Precision, and Recall for comparison. To achieve our aims and objectives discussed in Chapter 1.5, we conducted three experiments discussed in Chapter 4:

1. To verify the available HDD methods.
2. Applying feature selection and sampling.
3. Analyse the effect of different sampling ratios on the algorithms.

We aimed to reproduce the results of existing HDD methods on a new 2022 Backblaze dataset in experiment 1. Due to time and resource constraints, we leveraged a subset of a dataset 2022 Q1 for experiment 1. Two tests were conducted, t1 and t2, with five and 75 features. BRF, Random Forest, DT, RFE + RF, GBDT, and BBC were trained and discussed. Bagging classifiers (BRF and BBC) were found to be outperforming other classifiers by a considerable margin, where we discovered two findings:

a. Bagging Classifiers use Random Under Sampling by default during training (Lemaître et al. 2017)

b. There could be more information in features other than five Backblaze recommendations

These findings motivated us to proceed with experiment 2, using the feature selection method (RFE) and sampling methods (RUS and SMOTEBagging) on the 2022 Backblaze dataset containing over 6,678,738 non-failures and 632 failure samples with an imbalance ratio of 10567.62:1. We leveraged seven algorithms including BRF, WRF, WRF with RUS, WLR, WLR with RUS BBHG and Smote Bagging with DT for this experiment. Our method, including feature selection (RFE) and sampling method, proved effective when dealing with class imbalances in the Backblaze dataset. RF with RUS, BRF and BBHG were among the top-performing methods, with 91.5, 91.3 and 91.5% in terms of G-mean, respectively. We discovered two findings with this experiment:

a. The role of the sampling method in addressing class imbalance is significant.

b. The classifier's performance was more significantly influenced by the sampling method than the feature selection technique.

These findings motivated us to proceed with experiment 3, where we analysed the effect of different sampling ratios on the algorithms. We leveraged the full 2022 dataset for this experiment with the same 10567.62:1 imbalance ratio, RFE as feature selection, and RUS as the sampling method. We used three algorithms, BRF, RF and WLR, for this experiment. BRF, with 91.54% G-mean, was found to be top performing at a 4:3 sampling ratio. All three classifiers were close to each at ~82% G-mean with a ~25:1 sampling ratio. Experiment 3 results show that sampling ratios significantly affect classifier performance. Balanced ratios improve performance for BRF and RF, except for WLR, which struggles with data complexity.

As discussed in Chapter 5, The results of our study showed that the more advanced bagging classifiers, such as Balanced Random Forest and Random Forest, when combined with sampling methods and feature selection methods, demonstrated exceptional performance consistently across the experiments, indicating their potential generalizability. A performance of ~91% G-mean was achieved using these methods by RF and BRF. Additionally, we found that the sampling techniques like Random Under Sampling played a more crucial role in the class imbalance issue of the Backblaze HDD dataset than the feature selection method.

As for future work, researchers could explore additional class imbalance techniques beyond those examined here as well as investigate how these approaches might perform on different types of data sets with varying degrees of class imbalance between classes. Another interesting future research could include the use of deep learning methods like auto-encoders or LSTM to see how they perform on this dataset.

Reflection on the process: My research study focused on “Exploring the Effect of Feature Selection and Sampling Techniques on Current HDD Methods Using Hard Drive Samples of 2022”. Apache Spark was used to handle the large dataset. I utilized the CRISP-DM (Cross-Industry Standard Process for Data Mining) framework to guide my research process and a Project Planner for time management.
The first step in the process was to identify the research question and goals. The main research question was: What is the effectiveness of current HDD analysis methods on imbalanced datasets? The goal was to verify whether the existing HDD methods were still a good fit in 2022. Next, I conducted a thorough literature review to identify existing research and best practices. This allowed me to identify key features to include in the model, such as feature selection (RFE) and sampling methods.
With knowledge of the domain and existing methods, I then collected and cleaned the data, consisting of a large Backblaze dataset of 2022. Apache Spark was used to handle the large

dataset, and data management and ETL (Extract, Transform and Load) skills were learned during this stage.

Once the data was prepared, I addressed the class imbalance issue using feature selection and undersampling techniques. I also utilized many machine learning algorithms, mainly Random Forest, Balanced Random Forest, and Weighted Logistic Regression.

I learned how to set up the experiments, tune the classifier, and evaluate model performance during the experimentation stage. I also learnt how to set up experiments on a cloud platform as we used Microsoft Azure due to the large nature of the dataset. I learnt how different algorithms perform with different class imbalances and how to deal with the problem of class imbalance.

The project went smoothly due to the clear research questions identified during literature reviews and over-estimating the required time, allowing for a well-planned and progress-oriented approach. The project planner was used to ensure that each stage was properly executed, and the use of Apache Spark allowed for the efficient handling of the large dataset. The knowledge gained from my master's degree through modules like Data Processing and Analytics, Artificial Intelligence, and Research Methods were really helpful throughout this project. I applied skills like Machine Learning, Data Analysis and Research Analysis learnt from the core units into this project. Through this project, I gained valuable research skills, machine learning knowledge, data and time management, and experience in dealing with class imbalance and large datasets, which will be very helpful for my future career.

Word count (main body of the report): ~15800

REFERENCES

A. Aziz, A. S., Hanafi, S. E. L. O. and Hassanien, A. E., 2017. Comparison of classification techniques applied for network intrusion detection and classification. *Journal of Applied Logic*, 24 (SI:SOCO14), 109-118.

Agusta, Z. P. and Adiwijaya, A., 2018. Modified balanced random forest for improving imbalanced data prediction. *International Journal of Advances in Intelligent Informatics*, 5 (1), 58.

Ahmed, J. and Green, R., 2022. Predicting severely imbalanced data disk drive failures with machine learning models. *Machine Learning with Applications*, 9 (2666-8270), 100361.

Aussel, N., Jaulin, S., Gandon, G., Petetin, Y., Fazli, E. and Chabridon, S., 2017. Predictive Models of Hard Drive Failures Based on Operational Data. *IEEE Xplore*, 619-625.

Backblaze, 2022. Backblaze Hard Drive Stats. *www.backblaze.com*.

Barua, S., Islam, M. M., Yao, X. and Murase, K., 2014. MWMOTE--Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning. *IEEE Transactions on Knowledge and Data Engineering*, 26 (2), 405-425.

Batista, G. E. A. P. A., Prati, R. C. and Monard, M. C., 2004. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, 6 (1), 20-29.

Branco, P., Torgo, L. and Ribeiro, R. P., 2016. A Survey of Predictive Modeling on Imbalanced Domains. *ACM Computing Surveys*, 49 (2), 1-50.

Breiman, L., 2001. Random Forests. *Machine Learning*, 45 (1), 5-32.

Chawla, N. V., Bowyer, K. W., Hall, L. O. and Kegelmeyer, W. P., 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16 (16), 321-357.

Chawla, N. V., Japkowicz, N. and Kotcz, A., 2004. Special Issue on Learning from Imbalanced Data Sets. *ACM SIGKDD Explorations Newsletter*, 6 (1), 1.

Chen, C. and Liaw, A., 2004. *Using Random Forest to Learn Imbalanced Data*.

Cole, G., 2000. *Estimating Drive Reliability in Desktop Computers and Consumer Electronics Systems*.

Coursey, A., Nath, G., Prabhu, S. and Sengupta, S., 2021. Remaining Useful Life Estimation of Hard Disk Drives using Bidirectional LSTM Networks, *2021 IEEE International Conference on Big Data (Big Data)* (pp. 4832-4841).

Halligan, S., Altman, D. G. and Mallett, S., 2015. Disadvantages of using the area under the receiver operating characteristic curve to assess imaging tests: A discussion and proposal for an alternative approach. *European Radiology*, 25 (4), 932-939.

Hamerly, G. and Elkan, C., 2001. Bayesian approaches to failure prediction for disk drives. *undefined*.

Hasanin, T., Khoshgoftaar, T. M., Leevy, J. L. and Bauder, R. A., 2019. Severely imbalanced Big Data challenges: investigating data sampling approaches. *Journal of Big Data*, 6 (1).

Hu, L., Han, L., Xu, Z., Jiang, T. and Qi, H., 2020. A disk failure prediction method based on LSTM network due to its individual specificity. *Procedia Computer Science*, 176, 791-799.

Huang, X., 2017. Hard Drive Failure Prediction for Large Scale Storage System. *escholarship.org*.

Hughes, G. F., Murray, J. F., Kreutz-Delgado, K. and Elkan, C., 2002. Improved disk-drive failure warnings. *IEEE Transactions on Reliability*, 51 (3), 350-357.

Ircio, J., Lojo, A., Lozano, J. A. and Mori, U., 2022a. A Multivariate Time Series Streaming Classifier for Predicting Hard Drive Failures [Application Notes]. *IEEE Computational Intelligence Magazine*, 17 (1), 102-114.

Ircio, J., Lojo, A., Lozano, J. A., Mori, U. and Lozano, J. A., 2022b. A Multivariate Time Series Streaming Classifier for Predicting Hard Drive Failures [Application Notes]. *IEEE Computational Intelligence Magazine*, 17 (1), 102-114.

Johnson, M. K. and Kjell, 2019. *11.3 Recursive Feature Elimination | Feature Engineering and Selection: A Practical Approach for Predictive Models*. Taylor & Francis Group.

Klein, A., 2016. What SMART Hard Disk Errors Actually Tell Us. *Backblaze Blog | Cloud Storage & Cloud Backup*.

Klein, A., 2022. Backblaze Drive Stats for 2021. *Backblaze Blog | Cloud Storage & Cloud Backup*.

Kubat, M. and Matwin, S., 1997. *Addressing the Curse of Imbalanced Training Sets: One-Sided Selection*.

Lemaître, G., Nogueira, F. and Aridas, C. K., 2017. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research*, 18 (17), 1-5.

Li, J., Ji, X., Jia, Y., Zhu, B., Wang, G., Li, Z. and Liu, X., 2014. Hard Drive Failure Prediction Using Classification and Regression Trees. *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*.

Li, J., Stones, R. J., Wang, G., Liu, X., Li, Z. and Xu, M., 2017. Hard drive failure prediction using Decision Trees. *Reliability Engineering & System Safety*, 164, 55-65.

Li, W., 2017. *Proactive Prediction of Hard Disk Drive Failure*.

Lima, F. D. S., Pereira, F. L. F., Chaves, I. C., Gomes, J. P. P. and Machado, J. C., 2018. Evaluation of Recurrent Neural Networks for Hard Disk Drives Failure Prediction. *IEEE Xplore*, 85-90.

Liu, D., Wang, B., Li, P., Stones, R. J., Marbach, T. G., Wang, G., Liu, X. and Li, Z., 2020. Predicting Hard Drive Failures for Cloud Storage Systems. *Algorithms and Architectures for Parallel Processing*, 373-388.

Liu, X. Y., Wu, J. and Zhou, Z.-H., 2009. Exploratory Undersampling for Class-Imbalance Learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39 (2), 539-550.

Ma, A., Traylor, R., Douglis, F., Chamness, M., Lu, G., Sawyer, D., Chandra, S. and Hsu, W., 2015. RAIDShield. *ACM Transactions on Storage*, 11 (4), 1-28.

Mann, H. B., 1945. Nonparametric Tests Against Trend. *Econometrica*, 13 (3), 245.

Muchlinski, D., Siroky, D., He, J. and Kocher, M., 2016. Comparing Random Forest with Logistic Regression for Predicting Class-Imbalanced Civil War Onset Data. *Political Analysis*, 24 (1), 87-103.

Murray, J., Org, J. j., Hughes, G., Edu, G. u. and Kreutz-Delgado, K., 2005. Machine Learning Methods for Predicting Failures in Hard Drives: A Multiple-Instance Application. *Journal of Machine Learning Research*, 6 (783–816), 783-816.

Pereira, F. L. F., Chaves, I. C., Gomes, J. P. P. and Machado, J. C., 2020. Using Autoencoders for Anomaly Detection in Hard Disk Drives, *2020 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-7).

Pinheiro, E., Weber, W.-D. and Barroso, L., 2007. *Failure Trends in a Large Disk Drive Population*.

Piramuthu, S., 2004. Evaluating feature selection methods for learning in data mining applications. *European Journal of Operational Research*, 156 (2), 483-494.

Powers, D. M., 2020. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*.

Queiroz, L. P., Rodrigues, F. C. M., Gomes, J. P. P., Brito, F. T., Chaves, I. C., Paula, M. R. P., Salvador, M. R. and Machado, J. C., 2017. A Fault Detection Method for Hard Disk Drives Based on Mixture of Gaussians and Nonparametric Statistics. *IEEE Transactions on Industrial Informatics*, 13 (2), 542-550.

Ranganathan, P., Pramesh, C. S. and Aggarwal, R., 2017. Common pitfalls in statistical analysis: Logistic regression. *Perspectives in Clinical Research*, 8 (3), 148-151.

Schroeder, B. and Gibson, G., 2007a. *Disk failures in the real world: What does an MTTF of 1,000,000 hours mean to you?*

Schroeder, B. and Gibson, G. A., 2007b. Understanding disk failure rates. *ACM Transactions on Storage*, 3 (3), 8.

Shen, J., Wan, J., Lim, S.-J. and Yu, L., 2018. Random-forest-based failure prediction for hard disk drives. *International Journal of Distributed Sensor Networks*, 14 (11), 155014771880648.

Smart Vision, E., 2017. Building and Applying Predictive Models in IBM SPSS Modeler training webinar. *Smart Vision - Europe*.

Thakur, A., 2020. Simple Ways to Tackle Class Imbalance. *W&B*.

Vishwanath, K. V. and Nagappan, N., 2010. Characterizing cloud computing hardware reliability. *Proceedings of the 1st ACM symposium on Cloud computing - SoCC '10*.

Wang, Y., Miao, Q., Ma, E. W. M., Tsui, K.-L. and Pecht, M. G., 2013. Online Anomaly Detection for Hard Disk Drives Based on Mahalanobis Distance. *IEEE Transactions on Reliability*, 62 (1), 136-145.

Wang, Y., Miao, Q. and Pecht, M., 2011. Health monitoring of hard disk drive based on Mahalanobis distance. *2011 Prognostics and System Health Managment Confernece*.

Xu, C., Wang, G., Liu, X., Guo, D. and Liu, T.-Y., 2016. Health Status Assessment and Failure Prediction for Hard Drives with Recurrent Neural Networks. *IEEE Transactions on Computers*, 65 (11), 3502-3508.

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

Yang, Q., Jia, X., Li, X., Feng, J., Li, W. and Lee, J., 2021. Evaluating Feature Selection and Anomaly Detection Methods of Hard Drive Failure Prediction. *IEEE Transactions on Reliability*, 70 (2), 749-760.

Zhang, J., Zhou, K., Huang, P., He, X., Xie, M., Cheng, B., Ji, Y. and Wang, Y., 2020. Minority Disk Failure Prediction Based on Transfer Learning in Large Data Centers of Heterogeneous Disk Systems. *IEEE Transactions on Parallel and Distributed Systems*, 31 (9), 2155-2169.

Zhang, S., Bahrampour, S., Ramakrishnan, N., Schott, L. and Shah, M., 2017. Deep learning on symbolic representations for large-scale heterogeneous time-series event prediction. *IEEE Xplore*, 5970-5974.

Züfle, M., Krupitzer, C., Erhard, F., Grohmann, J. and Kounev, S., 2020. To Fail or Not to Fail: Predicting Hard Disk Drive Failure Time Windows. *Lecture Notes in Computer Science*, 19-36.

Čehovin, L. and Bosnić, Z., 2010. Empirical evaluation of feature selection methods in classification. *Intelligent Data Analysis*, 14 (3), 265-281.

APPENDIX A: LARGE FILES

Large Files Zip contains following files:

A1: data_Q1_2022.zip
A2: data_Q2_2022.zip
A3: data_Q3_2022.zip
A4: data_Q4_2022.zip
A5: 2022_5_features_STM.csv
A6: 2022_allfeatures_STM.csv
A7: exp2_results_5features.csv
A8: exp2_results_10features.csv
A9: exp3_results_10features.csv
A10: HDD project plan.xlsx
A11: RFE_10_features.csv
A12: Masters HDD 2.docx
A13: Imgs:
- A13.1: Boxplot.png
- A13.2: correlation.png
- A13.3: Countplot.png
- A13.4: Exp_1.png
- A13.5: Exp_2.png
- A13.6: Exp_3.png
- A13.7: exp2_results_5features.png
- A13.8: exp2_results_10features.png
- A13.9: exp3_results_10features_all.png
- A13.10: exp3_results_10features_BRF.png
- A13.11: exp3_results_10features_RF.png
- A13.12: exp3_results_10features_WLR.png
- A13.13: Pairplot.png
- A13.14: Data Engineering.drawio.pdf

A14: Charts.ipynb
A15: HDD_eda.ipynb
A16: HDD_exp1_Q1.ipynb
A17: HDD_exp2-5features.ipynb
A18: HDD_exp2-10features.ipynb
A19: HDD_exp2-RFE and EDA.ipynb
A20: HDD_exp3_10features.ipynb

APPENDIX B: ETHICS APPROVAL

Bournemouth University

Research Ethics Checklist

About Your Checklist	
Ethics ID	46478
Date Created	21/11/2022 20:35:50
Status	Approved
Date Approved	29/11/2022 09:54:51
Date Submitted	21/11/2022 20:52:27
Risk	Low

Researcher Details	
Name	Sanskar Behl
Faculty	Faculty of Science & Technology
Status	Postgraduate Taught (Masters, MA, MSc, MBA, LLM)
Course	MSc Data Science & Artificial Intelligence

Project Details	
Title	Hard disk drive failure prediction
Start Date of Project	01/01/2023
End Date of Project	30/04/2023
Proposed Start Date of Data Collection	01/01/2022
Supervisor	Lai Xu
Approver	Lai Xu
Summary - no more than 600 words (including detail on background methodology, sample, outcomes, etc.)	
I will be analysing the study of hard disk drive failure prediction and will try to improve the current literature on the same. I will be taking the dataset of 2022 from (Jan-Dec) from backblaze and will apply the study on the dataset to check if the hard disk drive is going to fail.	

Filter Question: Does your study involve the use or re-use of data which will be obtained from a source other than directly from a Research Participant?

Additional Details	
Please describe the data, its source and how you are permitted to use it	The dataset is publicly available to use here. I am allowed to use the data as long as I fulfill the following conditions: You can download and use this data for free for your own purpose, all we ask is three things:

	• you cite Backblaze as the source if you use the data, • you accept that you are solely responsible for how you use the data, and • you do not sell this data to anyone, it is free

Research Data	
Will identifiable personal information be collected, i.e. at an individualised level in a form that identifies or could enable identification of the participant?	No
Will research outputs include any identifiable personal information i.e. data at an individualised level in a form which identifies or could enable identification of the individual?	No

Storage, Access and Disposal of Research Data	
Where will your research data be stored and who will have access during and after the study has finished.	
My research data will be stored online in a cloud platform and in my personal computer. Me and my supervisor will have access for the cloud platform.	
Once your project completes, will your dataset be added to an appropriate research data repository such as BORDaR, BU's Data Repository?	Yes

Final Review	
Are there any other ethical considerations relating to your project which have not been covered above?	No

Risk Assessment	
Have you undertaken an appropriate Risk Assessment?	No

APPENDIX C: PROJECT PROPOSAL

Bournemouth University

Department of Computing and Informatics

2022-23 Academic Year Individual Masters Project

Project Proposal Form

Please refer to the **Project Handbook Section 4** when completing this form. Note that your proposal should be your own original work and you must cite sources in line with university guidance on **referencing and plagiarism**[1].

Degree Title: Choose an item.	**Student's Name: Sanskar Behl**
	Supervisor's Name: Lai Xu
	Project Title/Area: Hard disk drive failure prediction

Section 1: Project Overview

1.1 Problem definition - use <u>one sentence</u> to summarise the problem:

There are many studies have done in past to predict hard drive prediction. Those previous studies were conducted on small datasets in controlled environment which lack implementation details and prevents comparison (Aussel et al. 2017). To overcome this issue, (Aussel et al. 2017) used publicly available dataset backblaze. However, the problem with this is that the dataset is highly unbalanced. He tried to use SMOTE sampling to overcome this issue but did not see any results. I will try to use more advanced sampling methods like SMOTEBagging which may enable us to get better results.

1.2 Project description - briefly explain your project:

Backblaze dataset of the recent year 2022 would be used in the project as it is publicly available. The main feature selection would be based on the pre-selected SMART features that are highly corelated to failure events such as relocated sector count, scan error and offline relocated sector count errors (Pinheiro et al. 2007). I will try using the SMOTEBagging for sampling of highly unbalanced dataset of backblaze as suggested by (Aussel et al. 2017) in his study. As suggested in (Ahmed and Green 2022) maximum absolute value might be good option for feature scaling and stratified cross validation might be used to prevent data leakage. I will try using the same algorithms used by (Aussel et al. 2017) to

[1] https://libguides.bournemouth.ac.uk/study-skills-referencing-plagiarism

reproduce the results. I may also use some more algorithms like Balanced Random Forest, Gradient boosting decision tree as suggested by (Ahmed and Green 2022). Later in the end, results will be evaluated based on the geometric mean as suggested by (Ahmed and Green 2022) which is a better evaluation metric for highly unbalanced datasets like backblaze.

1.3 Background - please provide brief background information, e.g., client, problem domain, and make reference to the literature (minimum 4-5 sources):

In data centre settings where hard disk drives (HDD) malfunctions cannot be common, but they can be expensive events. This is why HDD manufacturers are motivated to minimize the number of failures as a reduction measure. At present, HDD manufacturers use Self-Monitoring and Reporting Technology (SMART) attributes that are collected in regular operations to anticipate failures. These attributes provide HDD health indicators like the number of scanning errors, reallocation count, and probational counts for an HDD. If one of the attributes considered vital to HDD health is above its threshold, then the HDD is deemed more susceptible to failure (Pinheiro et al. 2007). (Aussel et al. 2017) used the backblaze dataset on Jan 2014-2015 to better understand the problem of HDD failure events and provided with better evaluation metrics. He used SMOTE sampling method however he did not see any improvements with the technique. The more advanced methods like SMOTEBagging will be used in our study on the recent 2022 dataset of blackblaze.

1.4 Aims and objectives – what are the aims and objectives of your project? should be specific and measurable:

The main objectives of this study are as follows:

1. Perform analysis and evaluation on one manufacturer on the latest backblaze dataset including the use of machine learning algorithms for predicting failure base on highly imbalanced datasets.
2. Compare different models proposed and evaluate their performance using the Backblaze dataset which includes hard drives from various manufacturers to see whether they can withstand the variations in SMART parameters.

1.5 Research Questions

Why many studies prior to this are not relevant in the context of the effects of class imbalance, heterogeneity in data and the volume of data that are based on machine learning algorithms to recognize patterns in failure of hard drives?
How are hard drives robust to the differences in Smart parameters
Which sampling method is actually useful in dealing with highly imbalanced dataset?
What features should be considered in order to optimize the prediction in drive failure?

Bournemouth University

Department of Computing and Informatics

2022-23 Academic Year Individual Masters Project

Section 2: Artefact

2.1 What is the artefact that you intend to produce?

I intend to reproduce the results by (Aussel et al. 2017) on backblaze 2022(Jan-Dec) dataset and use his suggestion of using SMOte Bagging sampling method to deal with imbalance dataset in order to improve the results. Feature selection methods like RFE (Recursive feature elimination) might be used because of its high performance, high robustness against redundant features (Yang et al. 2021). ML Methods like GBDT (Gradient Boosted Decision Tree), RF (Random Forest), LR (Logistic Regression) and EasyEnsemble will be used as they give better performance compared to other methods (Aussel et al. 2017; Yang et al. 2021; Ahmed and Green 2022)

2.2 How is your artefact actionable (i.e., routes to implementation and exploitation in the technology domain)?

The artefact can be used by data centres in predicting HDD failure prediction or It can help further studies in better understanding the problem domain of HDD failure prediction.

Section 3: Evaluation

3.1 How are you going to evaluate your project artefact?

As suggested by (Aussel et al. 2017), FDR (False Detection Rate) and FAR (False Alarm Rate) are unclear for HDD failure problem as the dataset is highly unbalanced. (Ahmed and Green 2022) suggested in his study that precision and recall are also shouldn't be used for the same reason and Gmean should be used for evaluation purpose. I will be using the same approach to evaluate the results in our study.

3.2 How does this project relate to your MSc Programme and your degree title outcomes?

I will be doing a thorough research on the domain and will try to reproduce the results from previous studies to compare and highlight the pros and cons of different methods. I will be utilising the research skills that are taught in the module Research Methods and will be using the knowledge from other modules to do the data analysis on backblaze 2022 dataset.

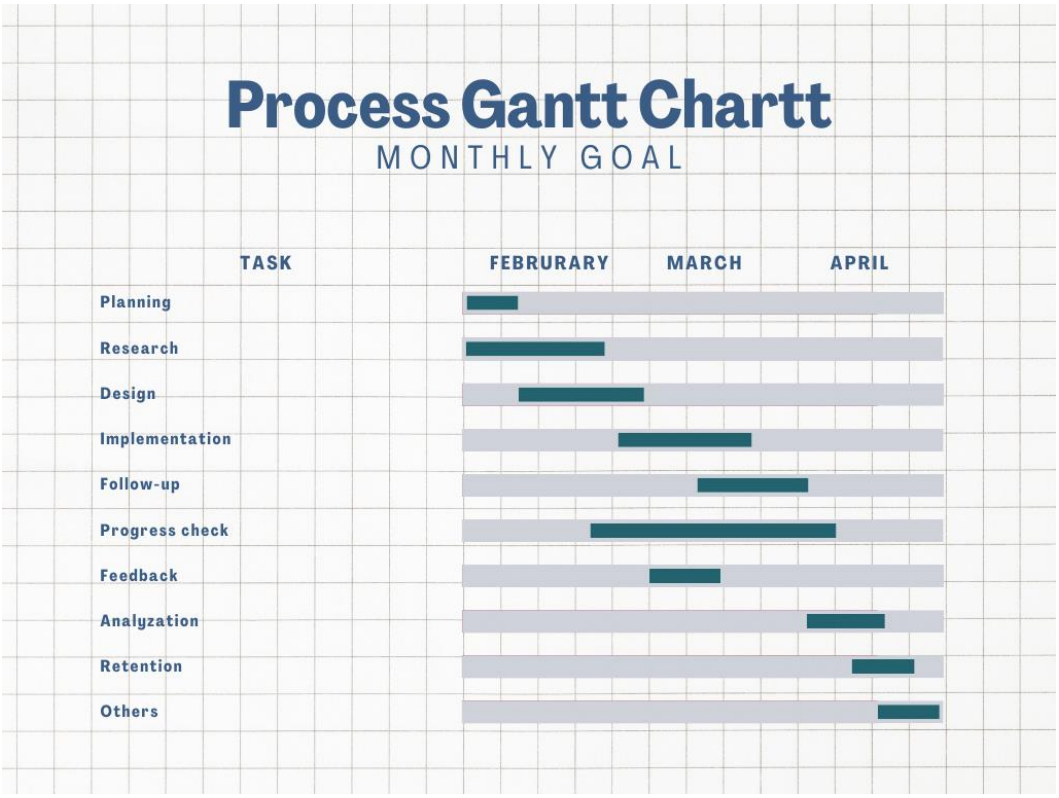
Bournemouth University

Department of Computing and Informatics

2022-23 Academic Year Individual Masters Project

Note: Please complete the research ethics checklist once the proposal has been approved by your supervisor.

Section 6: Proposed Plan (please attach your Gantt chart below)



Bournemouth University

Department of Computing and Informatics

2022-23 Academic Year Individual Masters Project

3.3 What are the risks in this project and how are you going to manage them?

There are few risks involved in this project as mentioned below:

1. Time crunch: It is possible that project might take longer than expected. I will overestimate the time needed to complete this project and schedule accordingly.
2. Stretched resources: I do not have enough resources to complete this project. For example, A high performance computer to train the models, Time required to complete the project and skills for time series analysis. I will be using cloud based jupyter notebooks like Azure, AWS or IBM to train the models and will be constantly taking care of the project.

Section 4: References

4.1 Please provide references if you have used any.

Ahmed, J. and Green, R., 2022. Predicting severely imbalanced data disk drive failures with machine learning models. *Machine Learning with Applications*, 100361.

Aussel, N., Jaulin, S., Gandon, G., Petetin, Y., Fazli, E. and Chabridon, S., 2017. *Predictive Models of Hard Drive Failures Based on Operational Data* [online]. IEEE Xplore. Available from: https://ieeexplore.ieee.org/document/8260700 [Accessed 1 Jan 2022].

Pinheiro, E., Weber, W.-D. and Barroso, L., 2007. *Failure Trends in a Large Disk Drive Population* [online]. Available from: https://static.googleusercontent.com/media/research.google.com/en//archive/disk_failures.pdf [Accessed 1 Jan 2020].

Yang, Q., Jia, X., Li, X., Feng, J., Li, W. and Lee, J., 2021. Evaluating Feature Selection and Anomaly Detection Methods of Hard Drive Failure Prediction. *IEEE Transactions on Reliability* [online], 70 (2), 749–760. Available from: https://ieeexplore.ieee.org/abstract/document/9112717 [Accessed 1 Jan 2022].

Section 5: Academic Practice and Ethics

Please delete as appropriate.

5.1 Have you made yourself familiar with, and understand, the University guidance on referencing and plagiarism?	**Yes**
5.2 Do you acknowledge that this project proposal is your own work and that it does not contravene any academic offence as specified in the University's regulations?	**Yes**

APPENDIX D: PROGRESS REVIEW FORM

Department of Computing and Informatics

Postgraduate Project Second Progress Review

To be completed and signed by the Supervisor and Student during week **commencing 3 April 2023**.

Student: Sanskar Behl	**Supervisor: Lai Xu**

Assessment

1. Definition of the problem *Has the problem, research aims, and questions been defined, has the artefact been identified and have objectives been set?*	Yes
Comments:	
2. Review of literature and related work *Is there evidence of appropriate research?*	Yes
Comments:	
3. Methodology and Artefact *Is there evidence of appropriate analysis of the problem and design of a solution and appropriate evaluation?*	Yes
Comments:	
4. Dissertation *Have sections of the dissertation been written and has the Supervisor seen these?*	To some extent
Comments: Sanskrit is busy with his experiments, but he needs to improve his writing about them.	
5. Planning & Progress *Is there an acceptable plan for this project and is it being followed?*	Yes
Comments:	
6. Proposal & Online Ethics Checklist *Are proposal and ethic checklist submitted? Are they approved?*	Yes, both are submitted and approved
7. Overall Assessment	Satisfactory
Signed: Supervisor: ……Lai Xu…….. Student: …… Sanskar Behl …….. Date: …19 April 2023……	

- Supervisor to retain the signed form and supply the student with a copy if required.
- Supervisor to upload the form on Brightspace and grade as *Satisfactory, Requires Major Improvement, Requires Minor Improvement, Unsatisfactory or Invalid*.
- Supervisor to notify the Project Coordinator if the student is at risk of failing the project or not engaging.