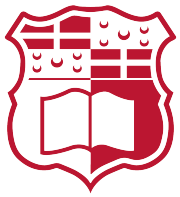

Anomaly Detection Using Machine Learning Techniques for Beam Injections from the SPS to the LHC at CERN

MARC FERRIGGI

Supervised by Dr. Gianluca Valentino

Department of Computer Science

Faculty of ICT

University of Malta

May, 2019

A FYP submitted in partial fulfilment of the requirements for the degree of B.Sc. (Hons.) Computing Science AND Statistics and Operations Research.

Statement of Originality

I, the undersigned, declare that this is my own work unless where otherwise acknowledged and referenced.

Candidate Marc Ferriggi

Signed _____

Date May 21, 2019

Acknowledgements

I'd like to first and foremost thank my supervisor Dr. Gianluca Valentino, this work wouldn't have been possible without his constant patience and support. I'd also like to thank my parents and family who have always believed in me and supported me financially to help me achieve my goals.

Abstract

Filling the LHC is a challenging task and anomalous injections are common when operating such a large machine. In this dissertation, unsupervised machine learning techniques for anomaly detection were used to analyse past data in order to understand the sources of the anomalies for improving the LHC machine availability and performance reach. Data from sensors around the moment of injection from the SPS to the LHC were analysed and different feature sets were chosen as input parameters for the anomaly detection algorithms. In this study, LOF was found to achieve the best performance when run on the full feature dataset with an accuracy of 92.76% for Beam 1 and 91.08% for Beam 2. The anomalous points are also visualised in 3D plots which serve to help researchers understand better the nature of these anomalies. Furthermore, the beam's position drift with time is also presented and is shown to also be a factor in impacting the injection quality.

Contents

List of Figures

List of Tables

List of Abbreviations

AGM Abort Gap Monitor 5
BCT Beam Current Transformers 5
BLM Beam Loss Monitors 4
BPM Beam Position Monitors 5
CERN European Organization for Nuclear Research 1
DBSCAN Density Based Spatial Clustering of Applications with Noise 7
FNR False Negative Rate 29
FPR False Positive Rate 29
GMM Gaussian Mixture Model 8
Gy/s Grays per Second 4
IQC Injection Quality Check 2
LHC Large Hadron Collider 1
LOF Local Outlier Factor 7
LS Logging Service 2
MJ Mega Joule 1
MKD horizontally deflecting extraction kicker magnets 5
MSE Mean Square Error 3
PCA Principal Component Analysis 5
RF Radiofrequency 5
SPS Super Proton Synchrotron 1
TDI Beam Absorber for Injection 4
TeV tera electron Volts 1
TIMBER the user interface to the LS 9

1

Introduction

The Large Hadron Collider (LHC) is a two-ring superconducting hadron accelerator and collider installed at the European Organization for Nuclear Research (CERN) and commenced operation in 2008 [1]. The collider is 26.7km long and its purpose is to accelerate and collide two heavy ion beams [2].

In order to operate the LHC with a centre-of-mass energy of 14 tera electron Volts (TeV), twelve injections from the Super Proton Synchrotron (SPS) consisting of a number of proton bunches of around 1 Mega Joule (MJ) of stored energy are required [3]. Thus, in order to fill the LHC, approximately 4 minutes per beam is required. Furthermore, the whole experiment process of filling the LHC, performing the required checks, running the tests and dumping the beam should take a theoretical minimum of 70 minutes [1]. However, from past experiences, this is expected to take longer, partly due to unsuccessful or anomalous proton injections to the LHC.

Clearly, filling the LHC is a challenging task given the high energy of the beam, the very small apertures and the delivery precision's tight tolerances. The beam must pass through many accelerators and transfer lines before reaching the LHC. During this process the beam must be monitored, thus multiple sensors are installed around the CERN particle accelerator complex [4] which gather readings and data that can be used to check the quality of the injected beam.

For this particular study, data generated from the sensors around the injection from the SPS to the LHC will be of particular interest (Figure 1.1). As the first beam (Beam 1) leaves the SPS, it must pass through the transfer line TI2, while the second beam (Beam 2) must pass through TI8. The data from sensors around these transfer lines as well

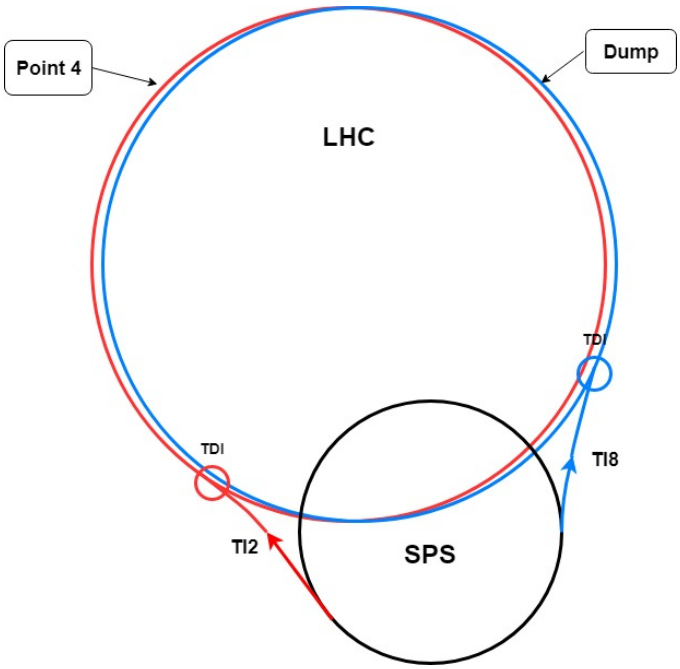

Figure 1.1: Diagram of the particular area of interest of the CERN Particle Accelerator Complex for this study

as at some points around the LHC and SPS will be used in this study and are stored using CERN's Logging Service (LS) [5]. While many studies have been made using this logged data and lots of statistical tests have been done with regards to injection quality checks for the LHC (such as [3] and [6]), no literature was uncovered where researchers used unsupervised machine learning methods to analyse this particular data.

Furthermore, the Injection Quality Check (IQC) software currently installed has a set of hard-coded rules for detecting anomalies in the SPS-LHC injection [3], however there are documented cases in the past where situations occurred which were outside the originally foreseen rules and were therefore not caught as anomalies. Apart from causing experiments to fail, these anomalous injections could be very costly as a lot of data must be examined after such failures which wastes time that could be used to run more experiments [7].

The major cause of these anomalies is due to the fact that the machine is so large, and needs to be so precise, that minor ground motions over time affect the tilts in the quadrupole magnets which thus affect the orbit of the beam. Figure 1.2 highlights two possible cases of anomalous injections. The first case shows what happens to the beam

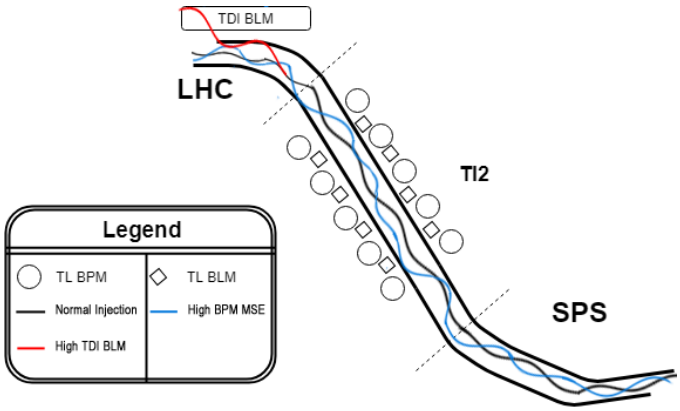

Figure 1.2: Examples of Anomalous Beam Injections showing the locations of the BPMs and BLMs in the transfer line

when the BPM gives a high Mean Square Error (MSE) reading with respect to its original position in the first injection of the season. The second case shows what happens to the beam when there is a high loss recorded by the TDI BLM.

The purpose of this study is to apply unsupervised anomaly detection algorithms to try solve the problem of detecting anomalous injections with the hopes of finding a technique that will detect the anomalies not being picked up by the IQC. This can help researchers understand the source of these anomalies and improve the LHC machine availability and performance reach in terms of beam lifetime, beam stability and luminosity.

In Chapter 2, the background to the problem domain is explained in further detail with the intention of providing the reader with information needed to fully understand this study. Recent published material on anomaly detection in particle accelerators is also presented in this chapter. The methodology used to analyse the provided dataset can be found in Chapter 3 where work done on the entire data science process is presented. The results of this study are then presented in Chapter 4 and a detailed evaluation on the performance of the anomaly detection algorithms can be found in Chapter 5. Some final conclusions and suggestions for further work are then presented in Chapter 6.

2

Background and Literature Review

2.1 Beam Instrumentation

Throughout this study, data recorded as the beam leaves the SPS and enters the LHC will be used as input parameters to the chosen anomaly detection algorithms. This data was recorded using different sensors located in different parts of the injection life cycle. This section describes the different types of sensors that were used to collect the data, highlighting the particular details which need to be considered when analysing this data.

The Beam Loss Monitors (BLM) are some of the most safety critical modules of the LHC because a loss of a very small fraction of this beam may damage parts of the machine or cause a quench in the superconducting magnets [8]. A high beam loss reading could also indicate over-injection. In fact, an injection of a high intensity beam into the LHC is only allowed if there is a low intensity bunch circulating the LHC in order to avoid settings errors [6]. The BLM module is the mostly used module in the current IQC software checks [3]. The BLMs must be reliable; the probability of not detecting a dangerous loss was found to be 5×10^{-6} per channel and they are only expected to generate 20 false dumps per year [8]. The BLMs are extensively logged to a database for offline analysis [8].

The LHC has beam absorbers installed around the machine in order to absorb residual beam. For this particular study, the readings logged for the Beam Absorber for Injection (TDI) BLMs and the Transfer Line (TL) BLMs in TI2 and TI8 will be used (refer to Figure 1.1). These readings come in 10 second windows around the injection of a bunch in Grays per Second (Gy/s).

The Beam Position Monitors (BPM) were installed as a system for fast monitoring of the beam's position with respect to its orbit drift [9]. The trajectory offsets recorded by the BPMs in the transfer lines must be minimised in order to reduce losses [3]. In fact, if the change in orbit substantially exceeds its provided boundary values then the beam should be dumped [9] so as to not cause any damage to the equipment. Unlike the TDI BLMs, the BPM system is independent of the collimator system. For this study, the readings from the transfer line BPMs around TI2 and TI8 will be used (refer to Figure 1.1). Raw values for these readings are stored by the LS in mm and are logged every 1 - 5 seconds on average.

When filling the LHC, it is necessary to keep an abort gap (i.e. an absence of particles) of at least $3\mu s$ (each turn of the LHC is $\approx 87\mu s$ long) in order to accommodate for the horizontally deflecting extraction kicker magnets (MKD) rise time [10]. As the LHC is filling to nominal intensity, this gap will be populated with un-trapped particles and particles leaking out of their Radiofrequency (RF) buckets [10]. The Abort Gap Monitor (AGM) was hence specifically designed to measure this particle population in the abort gap [11]. This monitor can be found in Point 4 (refer to Figure 1.1) in the LHC [11]. The raw values extracted for this study are stored in number of particles and come in 10 second groups around the moment of injection.

The actual intensities of the circulating beam are measured by Beam Current Transformers (BCT). For the LHC in particular, a fast BCT is used which is capable of monitoring a broad range of currents as it must be able to detect a single pilot bunch circulating the machine (of 10 μA) as well as the full nominal machine (over 0.5 mA) [12]. These readings are then converted from amps to number of protons per beam and stored for analysis. The logged intensities for the LHC come in 10 second time windows around the moment of injection into the LHC while the intensities for the SPS give a single value of the intensity at the time of extraction from the SPS into the transfer line.

2.2 Feature Scaling and Reduction Techniques

Feature Scaling and Feature Reduction are two important pre-processing steps that should be considered when using machine learning in the data science process. Standard Scaling in particular will be used in this study as a pre-processing step to Principal Component Analysis (PCA). Standard Scaling ensures that all the features have the properties of a standard normal distribution [13], which is especially important since PCA involves finding the components that maximise the variance [14]. This is achieved

in practise by ignoring the shape of the distribution and simply transforming the data to have a mean value of 0 and variance of 1 by subtracting the mean value for each feature and dividing each feature by its standard deviation [13].

Another feature scaling technique that will be used in this study in the data analysis part is MinMax Scaling. This is an alternative technique to Standard Scaling which is considered to be more robust to very small standard deviations of features where the features are scaled to lie between a given minimum and maximum value [13]. In this study, features will be scaled to lie between 0 and 1 and this is achieved using Equation 2.1. Where X represents the feature vector, $max(X)$ gives the largest value in X and $min(X)$ gives the smallest value in X.

$$Scaled_X = \frac{X - max(X)}{max(X) - min(X)} \tag{2.1}$$

Apart from scaling, another challenge for outlier detection algorithms is data involving high dimensions since the contrast between different points diminishes as the number of dimensions increases [15]. This phenomenon is known as 'The Curse of Dimensionality' and a technique to reduce the effect of this phenomenon is to use a dimension reduction technique and run the outlier detection algorithm on this new lower-dimensioned dataset. In this study, PCA will be used as a dimension reduction technique.

PCA uses statistical and mathematical techniques to reduce the dimension of large data sets, thus allowing a large data set to be interpreted in less variables called principal components [16]. This technique works with the hope that the variance explained by an acceptably small number of principal components is large enough to explain the underlying structure of the dataset reasonably [14]. In fact, this non-parametric method has been used as a means of revealing the simplified structures' underlying complex datasets with minimal effort. The fact that this technique is non-parametric gives it the advantage that each result is unique and only dependent on the provided data set since no parameter tweaking is required [14], however this is also a weakness of PCA as there is no way of exploiting prior expert knowledge on the data set.

2.3 Unsupervised Anomaly Detection Techniques

Unsupervised machine learning algorithms refer to the class of machine learning algorithms where only the input features are available to the learner as there is no access

to output labels corresponding to each input feature vector, or the aim of the algorithm is simply to observe or detect patterns in the available data. A. Hyvärinen states in [17] that some of the goals of unsupervised learning include data visualisation, noise reduction, feature extraction and finding interesting components; all of which are of particular interest in this study.

Density Based Spatial Clustering of Applications with Noise (DBSCAN) and Local Outlier Factor (LOF) will both be used as unsupervised anomaly detection algorithms to detect and classify anomalous injections of the past year. Furthermore when working in 3 dimensions or less, these points can also be visualised to help the reader understand better the cause of these anomalies.

DBSCAN was created out of the necessity of having a clustering algorithm with the following requirements:

1. "Minimal requirements of domain knowledge to determine the input parameters,"
2. "Discovery of clusters with arbitrary shape," and
3. "Good efficiency on large databases" [18]

DBSCAN manages to attain these requirements by viewing clusters as "areas of high density separated by areas of low density" [19]. The points with a lower density will thus be considered as anomalies when compared to the regular clusters which have a higher density. This algorithm also introduces the concept of *core samples* which was then used in the design of other unsupervised anomaly detection algorithms such as LOF.

The word *factor* in LOF refers to a "degree of outlier-ness" that this algorithm considers for each point in the data rather than using the concept that "being an outlier is binary" [20]. This algorithm uses a clustering technique which takes concepts from DBSCAN to measure the LOF of each point where a LOF value greater than 1 implies that the point has a lower density than its neighbours and is thus probably an outlier.

2.4 Anomaly Detection in Particle Accelerators

In the paper released entitled "Opportunities in Machine Learning for Particle Accelerators" [21], it was stated that due to the "large number of process variables, non-linear

behaviour, and many interacting subsystems," conventional analysis techniques on today's particle accelerator data is often insufficient and thus machine learning could be used as a means of anomaly detection. Furthermore, the authors also stated that these techniques could be used to detect "subtle behaviours of key variables prior to negative events" and they can also be used to "identify and throw away bad signals."

In his Master's Thesis, A. Halilovic used anomaly detection techniques solely on data obtained from the injection kicker magnets [7]. Halilovic made use of a Gaussian Mixture Model (GMM) and Isolation Forests to detect anomalies however found that the best performance achieved by his proposed pipeline "leaves something to be desired" as too many anomalies were not correctly classified. The author also goes on to suggest that analysing LHC data using the LOF class provided in '*scikit-learn*' could lead to interesting results.

Wielgosz, *et. al.* also wrote a scientific paper on using anomaly detection techniques on the LHC magnets [22]. This time, the authors went for a supervised approach and used Recurrent Neural Networks. They found that using adaptive quantisation to reduce 20-bit inputs into a 4-bit representation was an essential step in improving the algorithm's performance. The authors also stated that these anomaly detection techniques being proposed should not only be considered useful for CERN equipment but also useful in the broader field of anomaly detection on time series data.

In 2017, Valentino *et. al.* released a paper on using anomaly detection techniques "to detect minor changes in the loss maps over time due to collimator settings errors or orbit variations" [2]. The authors used PCA as a dimension reduction technique and then applied LOF on the resulting 2 dimensional data. Their proposed method was shown to positively identify these anomalous loss maps based solely on BPM and BLM readings. Furthermore, they proposed using this technique to monitor losses during fills of the LHC.

2.5 Software Implementation

Although performance of k-means and k-Nearest Neighbours is not as optimal as in other Python packages such as '*PyMVPA*' [23] or '*shogun*' [24] (see Table 1 in [25]), it was decided to use the '*scikit-learn*' machine learning package for this study due to its "state-of-the-art implementation" and "easy-to-use interface tightly integrated with the Python language" [25].

3

Methodology

3.1 Data Collection

The data used in this study was collected from the user interface to the LS (TIMBER) with the help of Dr. Gianluca Valentino who has access to the CERN Intranet. Data was collected from the instrumentation discussed in Section 2.1 and covers 1624 Injections over a time period of 3 months (from 17th August to 20th October 2018). During this time, approximately 65 LHC fills were performed.

The file sizes for the data gathered from each instrument ranged from 4 KB to 2 MB, these were initially individually analysed (refer to Section 3.2) and then merged to create the dataset used to run the anomaly detection algorithms on (refer to Section 3.4). The total size of the merged datasets were 231 KB and 324 KB for Beam 1 and Beam 2 respectively. Loading this data in memory was not an issue since the file size is rather small, thus the problem of dealing with Big Data was not encountered in this study.

3.2 Data Cleaning and Analysis

After Data Extraction, the provided datasets were analysed separately in order to understand their nature, remove any outliers and be able to aggregate the data correctly for further analysis. In this section the results of this analysis will be presented with the hopes that the reader will have a clearer understanding of later results. Note that all the steps mentioned here were repeated for both beams.

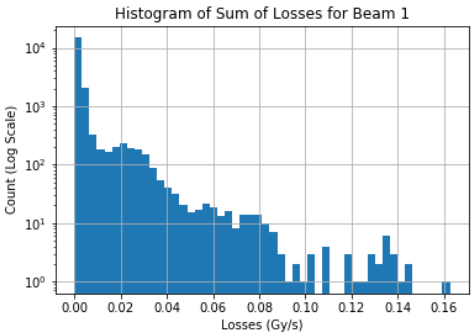

Figure 3.1: Histogram of Sum of Losses for Beam 1

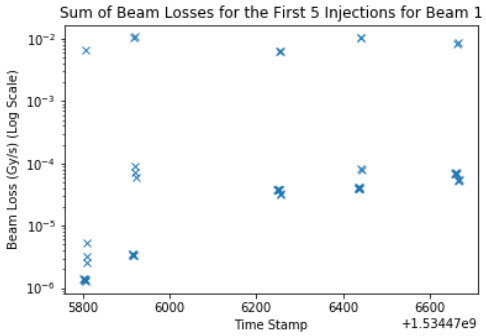

Figure 3.2: Time Series of Beam Loss Sum for the First 5 Injections

3.2.1 TDI BLMs

There are three BLMs in the TDI, each one giving 10 readings around the moment of injection. In order to get a total loss for each injection, the sum of each reading from the 3 monitors was taken (3.1). From the plot of this data (Figure 3.2) it was noted that at the exact moment of injection, there was a spike in the amount of beam lost. Thus, in order to then obtain a single reading corresponding to that particular injection, the maximum sum of losses for each 10 second window was kept.

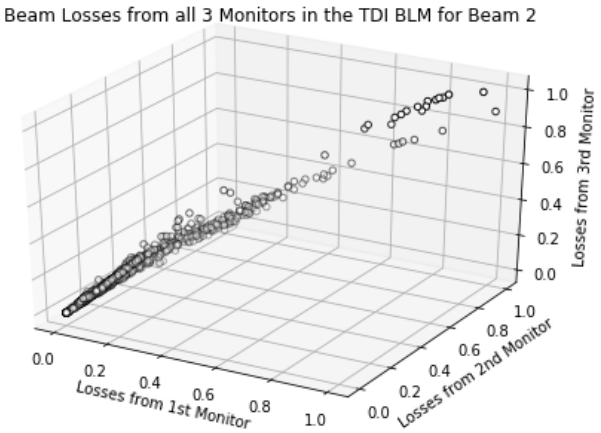

Figure 3.3: Beam Losses from all 3 Monitors in the TDI BLM for Beam 2 after MinMax Scaling, each point corresponds to one injection.

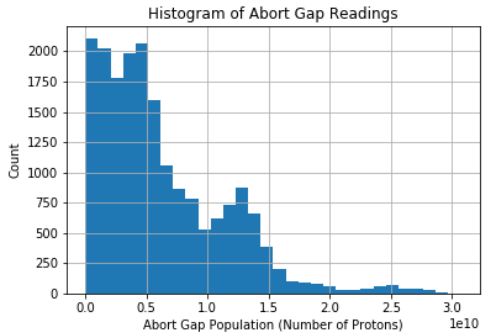

Figure 3.4: Histogram of Abort Gap Population for Beam 1

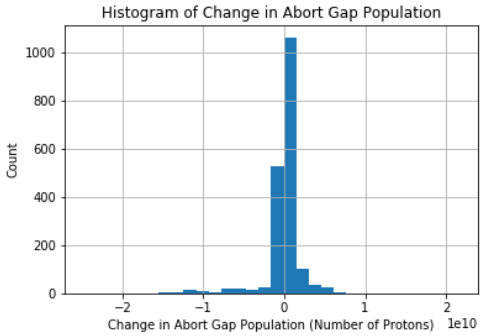

Figure 3.5: Histogram of Change in Abort Gap Population for Beam 1

Once the relevant readings were kept, the sum column was dropped and this data set was saved to be used for anomaly detection. Furthermore, after scaling these points using MinMax scaling, it was noted from Figure 3.3 that the readings from the 3 monitors are highly correlated. This was confirmed by computing the correlation matrix which gave a Pearson Correlation value > 0.98 for all pairwise comparisons.

3.2.2 Abort Gap

Similar to the TDI BLM readings, the Abort Gap readings also come in 10 second windows around the moment of injection. In this case however, the change in Abort Gap population is of interest for this study, thus the difference between every 10^{th} reading was kept and saved to be used for anomaly detection.

Figures 3.4 and 3.5 show the histograms of the Abort Gap Population and the Change in Abort Gap Population respectively. A time series plot of the Abort Gap Readings can be seen in Figure 3.6.

3.2.3 SPS and LHC Intensities

As mentioned in Section 2.1, the raw LHC intensity readings come in 10 readings around the moment of injection, while the SPS intensity readings give the value of the beam intensity as its leaving the SPS. Thus, it is expected that change in LHC intensity at the moment of injection (10^{th} reading - 1^{st} reading) should be approximately equal to the SPS intensity value. Some of the beam however is lost in the transfer line (which is picked up by the TL BLMs) and as it enters the LHC (which is picked up by the TDI

BLMs). Thus as an input parameter to the anomaly detection algorithm, the change in LHC intensities and the SPS intensities were extracted for each injection and saved.

The histograms of the change in LHC intensities and SPS intensities can be seen in Figures 3.7 and 3.8 respectively. Figure 3.9 shows the increase in the LHC reading and the corresponding SPS intensity.

3.2.4 TL BLMs

Each TL has 61 BLMs each recording the amount of losses separately. The issue with these readings however is that their readings are not consistently stored after the experiments have been performed. In fact, during the time of the study, 13 of these monitors didn't have any logged data at all. The data which was logged was either 0 or close to 0. Thus, it was decided to drop this feature from the study.

3.2.5 TL BPMs

There are 18 BPMs in each TL, each one giving a separate reading of the beam's deviation from its expected path at a different position in the TL. From data taken from 1624 injections, data corresponding to 1420 injections was left for Beam 1 after removing all missing values and 1455 injections for Beam 2. Figure 3.10 shows the histogram of the readings recorded by the first monitor in TI2.

In order to measure the beam drift over time, the first injection was assumed to be the expected path and the MSE of each injection with respect to the first injection was

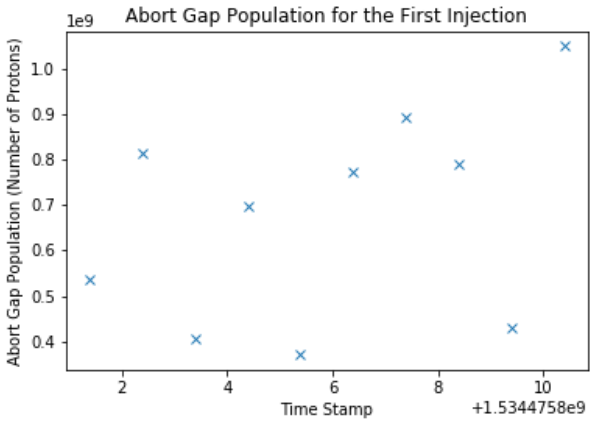

Figure 3.6: Time Series of Abort Gap Population for the First Injection

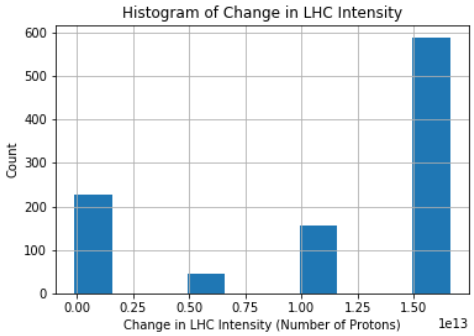

Figure 3.7: Histogram of Change in LHC Intensities for Beam 1

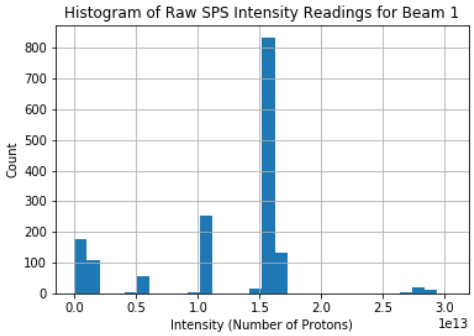

Figure 3.8: Histogram of SPS Intensities for Beam 1

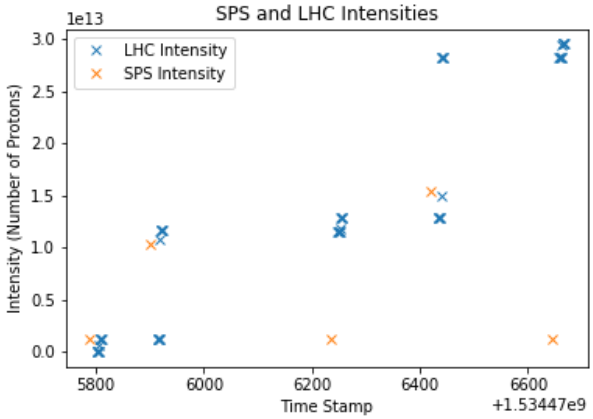

Figure 3.9: Time Series of SPS and LHC Intensities for the first 5 Injections for Beam 1

taken. The first injection was used as the ideal expected path of the beam and any drift from this path would result in possible anomalous injections. This was worked out by taking the average of the squared differences in the readings of each monitor for each injection. This led to an interesting result which is presented in Chapter 4.

3.2.6 Number of Bunches

The number of bunches injected into the LHC was another useful feature that needed to be extracted for normalisation. The beam losses, change in intensities and change in abort gap population are all relative to the number of bunches injected into the LHC as for example, a large loss may only appear to be large because more bunches were injected into the LHC for that injection than previous injections. In this case, that large

loss should not be considered an anomaly.

After the first single bunch is injected, the number of bunches injected in one go are typically multiples of 12, e.g. 12, 96, 144 which derives from how they are set up in the injector accelerators.

The number of bunches circulating the LHC was hence extracted around the time of each injection and the change in number of bunches was then worked out. Figure 3.11 shows the nature of this data and should give the reader an idea of how many bunches are injected at one time.

3.3 Feature Selection

The anomaly detection algorithms were first run on a three dimensional subset of the chosen features so as to be able to properly visualise the anomalies and understand the main features that are causing them. Due to the high correlation between the 3 TDI BLM readings, it was decided to use one of the TDI BLMs, the change in LHC intensity minus the SPS intensity (i.e. the amount of beam intensity lost in the transfer line) and the MSE of the BPM readings as the three dimensional features. Note that all readings were appropriately normalised by the number of bunches.

When performing the study on the full set of data, once again only one of the 3 TDI BLM readings would be needed as a feature. Furthermore, as mentioned in Section 3.2.4,

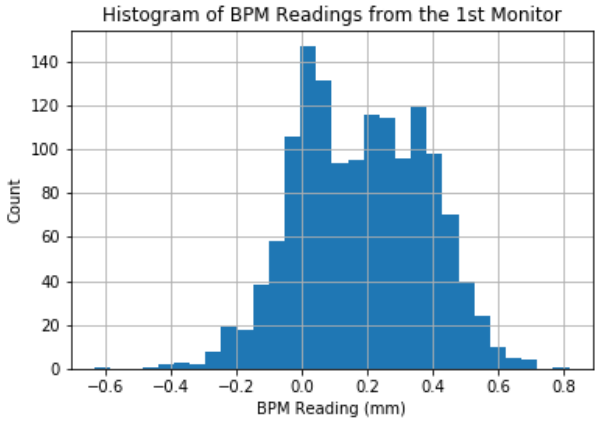

Figure 3.10: Histogram of the readings recorded by the first monitor in TI2

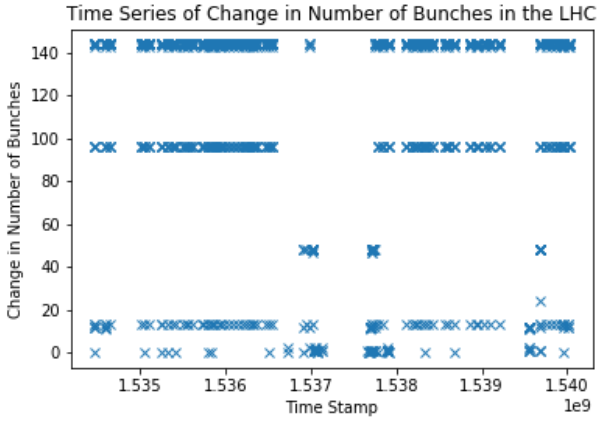

Figure 3.11: Number of Bunches Injected into the LHC for Beam 1

the readings from the TL BLMs will not be used. Thus, the total number of dimensions used in the full model was 21 dimensions and are listed below:

1. Normalised Abort Gap Population
2. Normalised LHC - SPS Intensities
3. Normalised BLM Reading
4. BPM 1 Reading
5. BPM 2 Reading
6. BPM 3 Reading
7. BPM 4 Reading
8. BPM 5 Reading
9. BPM 6 Reading
10. BPM 7 Reading
11. BPM 8 Reading
12. BPM 9 Reading
13. BPM 10 Reading
14. BPM 11 Reading
15. BPM 12 Reading
16. BPM 13 Reading
17. BPM 14 Reading
18. BPM 15 Reading
19. BPM 16 Reading
20. BPM 17 Reading
21. BPM 18 Reading
22. Number of Bunches Injected into the LHC

3.4 Merging the Dataset

Since there's no *'injection number'* or a similar index for the data points, correctly merging the data proved to be quite a challenge. The only available information to be able to tell what injection a particular data point corresponded to was the time stamp.

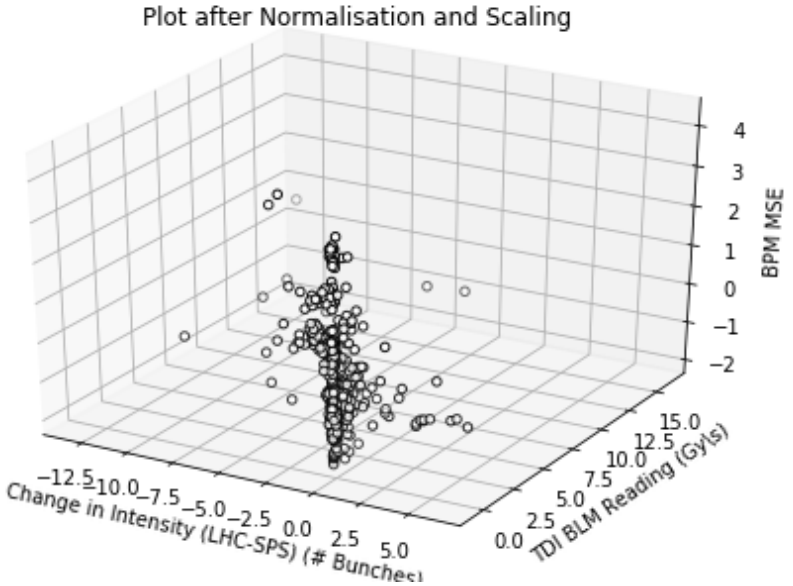

Figure 3.12: 3 Features chosen as parameters for LOF after Normalisation and Scaling

However, the readings are taken at different points in the LHC machine cycle thus care had to be taken when matching these time intervals to ensure correctly matching data.

After cleaning and merging the datasets, from an original 1624 injections collected, 859 injections were kept for Beam 1 and 1212 injections were kept for Beam 2. Although this might seem like a large loss of data, it was vital to ensure that the data being used was accurate.

3.5 Anomaly Detection

After all the data cleaning and preparation, the resultant consistent data was ready to be used for anomaly detection. Each feature was first scaled using standard scaling, a plot of the features chosen as parameters for the 3D anomaly detection after scaling and normalisation by the change in number of bunches can be seen in Figure 3.12. From this plot alone, its already clear to start noticing some of the points which could be anomalous injections. LOF and DBSCAN were then run for both Beam 1 and Beam 2 given this scaled and normalised data, the results are presented in Chapter 4.

LOF was then used on the full dataset (whose features are listed in Section 3.3) and those results are also presented in Chapter 4. Finally, PCA was performed on the full-

feature dataset and it was chosen to keep the components that explain at least 80% of the data's variation. Thus, 5 components were left for both Beam 1 and Beam 2. LOF was once again performed on this lower dimension dataset and the results are presented in Chapter 4.

When fitting the algorithms, care was taken to ensure the correct parameters were used in the models. Since this is an unsupervised approach, fitting the correct parameters is crucial to the overall performance of anomaly detection algorithms. However, since we do not have any training sets or guidelines on what points are actual anomalies, the parameters were tweaked by visual inspection of the resultant 3D plots derived after running the algorithm.

For LOF, the number of neighbours parameter was set to 50. According to the guidelines in [19], this parameter should be greater than the minimum number of points a cluster can contain and less than the maximum number of close by objects that could potentially be outliers. For DBSCAN, the minimum samples parameter was set to be 10. This parameter indicates the minimum number of samples in a neighbourhood for a point to be considered as a core point and hence not an outlier.

In order to compare the performance of the different algorithms, the anomalous points detected were then manually checked on TIMBER by Prof. Valentino and labelled 1 if the injection was actually anomalous and 0 otherwise. When performing this analysis, Prof. Valentino discovered that from the 14th to the 16th of September, the LHC was running some tests, thus all the injections that were found to be anomalous on these dates by the algorithms were not actually anomalous injections. These points were therefore removed from the study to ensure accuracy.

4

Results

In this chapter all results obtained from analysing the data and running the anomaly detection algorithms will be presented.

4.1 Beam Displacement Over Time

When performing the initial analysis on the provided BPM data, the MSE of the Beam's Position with respect to its initial position in the first injection was calculated (refer to Section 3.2). A rolling average of the time series points was taken to visualise the trend component. The window size was taken to be 12 since 12 injections are needed to fill the LHC. Figures 4.1 and 4.2 show the trend component for Beam 1 and Beam 2 respectively. Although there's some noise in the data, it is clear from both plots that the Beam's position drifts with time.

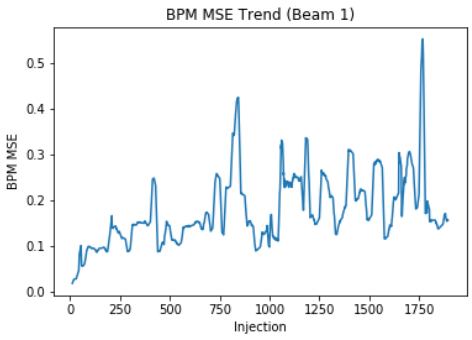

Figure 4.1: Trend Component of the BPM MSE for Beam 1

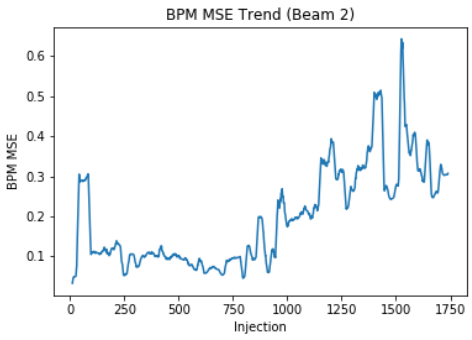

Figure 4.2: Trend Component of the BPM MSE for Beam 2

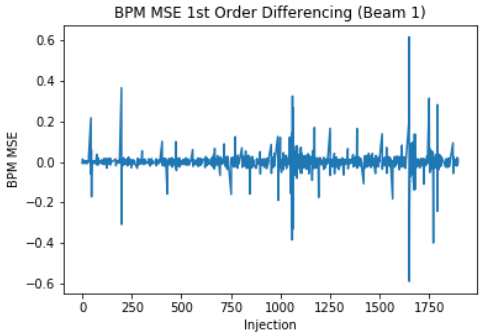

Figure 4.3: 1st Order Differencing for the BPM MSE for Beam 1

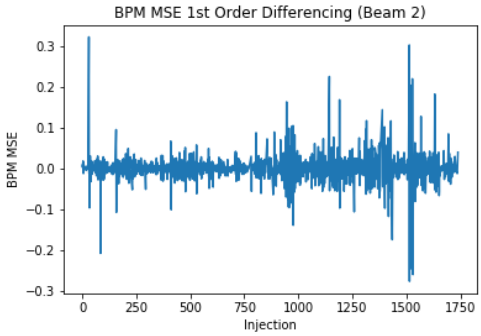

Figure 4.4: 1st Order Differencing for the BPM MSE for Beam 2

This result confirms the suspicion that the beam position in the transfer line changes over time, which could have an impact on the injection quality. The cause of this phenomenon is due to ground motion as the slightest movement in one of the LHC's quadrupole magnets can throw off the beam's precision. Thus regular servicing and maintenance of the LHC is important to reduce the number of anomalous injections.

First order differencing was then done on the data to remove the trend component. The resulting plots can be seen in Figures 4.3 and 4.4. From these plots we can conclude that the spikes in the MSE values are not cyclic but merely random.

4.2 Anomaly Detection

After running all the algorithms as described in Section 3.5, a total of 131 unique anomalous injections were detected for Beam 1. After manual inspection of each injection by Prof. Valentino, 90 (68.70%) of these points were found to be actual anomalies. Furthermore, a total of 232 unique anomalous injections were detected for Beam 2, where 134 (57.76%) of these were found to be actual anomalies. Since we do not know the actual number of anomalous injections that happened over the analysed time period, it was assumed in Chapter 5 that these are all the anomalous injections that happened in this time period and thus the performance of the different algorithms was compared based off of these points.

4.2.1 3D LOF

Of the 60 injections detected as anomalies by the 3D LOF algorithm for Beam 1, 39 of these were found to be actual anomalies (65%). These detected points can be seen in Figure 4.5 where the green points represent the correctly classified injections and the red points represent the false-positives generated. The 2D plots in 4.7 and 4.8 highlight the nature of these anomalies more clearly.

Of the 98 injections detected as anomalies by the 3D LOF algorithm for Beam 2, 45 of these were found to be actual anomalies (45.91%). These detected points can be seen in Figure 4.6.

As can be clearly seen from these plots, performing LOF on this 3 dimensional dataset is not sufficient to explain the causes for anomalies as many points that are clearly outliers are not anomalous injections after all. The algorithm still performed reasonably well however and adding more dimensions might improve its performance.

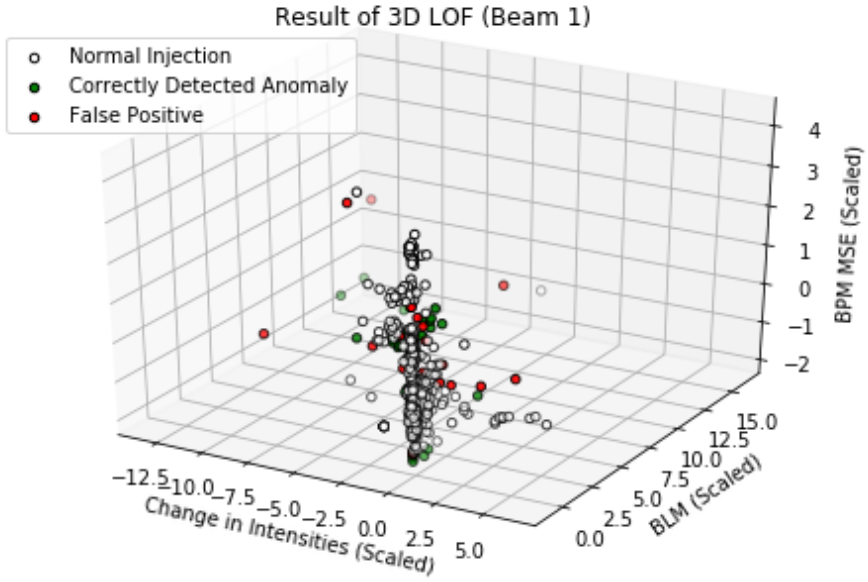

Figure 4.5: 3D Plot Highlighting Anomalies Detected by the 3D LOF Algorithm for Beam 1

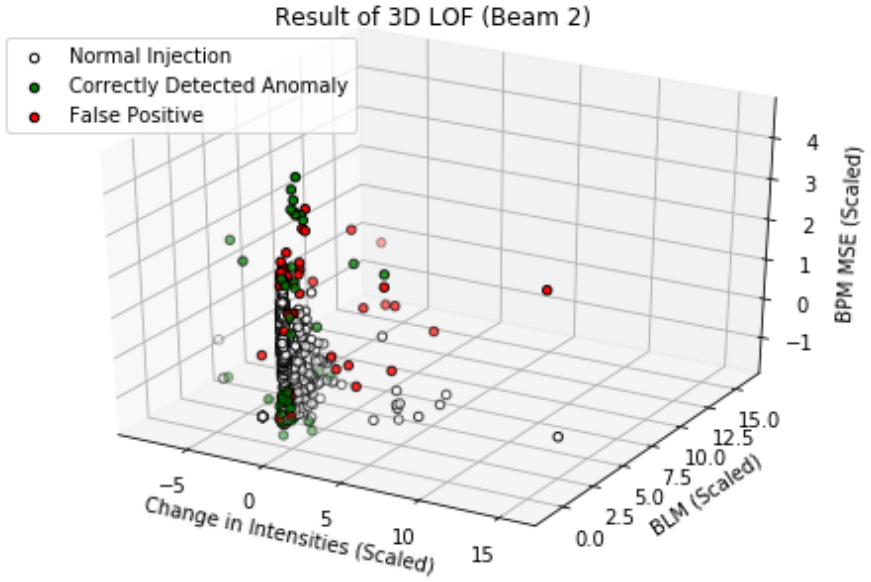

Figure 4.6: 3D Plot Highlighting Anomalies Detected by the 3D LOF Algorithm for Beam 2

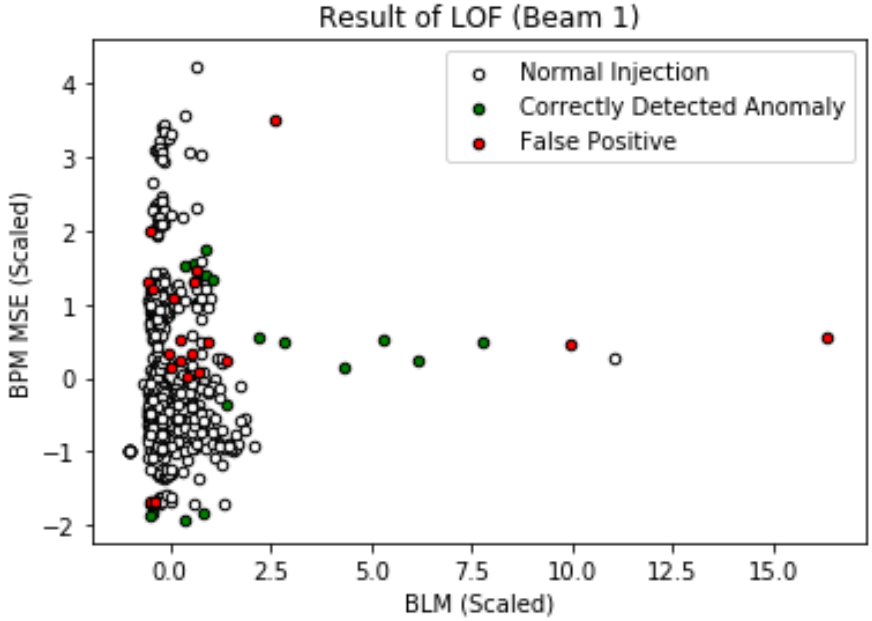

Figure 4.7: Anomalies Detected by the 3D LOF Algorithm

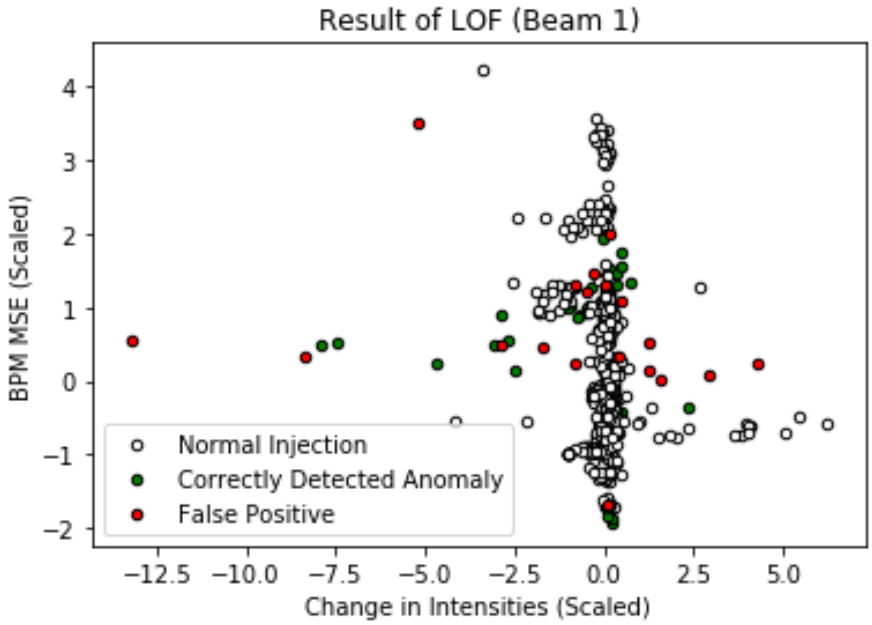

Figure 4.8: Anomalies Detected by the 3D LOF Algorithm

4.2.2 3D DBSCAN

The DBSCAN algorithm seems to have performed considerably worse than the LOF algorithm for 3D data. In fact, only 40 points were detected as anomalous by this algorithm for Beam 1 and only 23 (57.5%) points were found to be actual anomalous injections. Furthermore, 60 points were detected for Beam 2 where only 28 (46.67%) were actual anomalies.

These detected points can be seen in Figures 4.9 and 4.10 respectively where once again, the green points represent the correctly classified injections and the red points represent the false-positives generated. From these results alone, it was concluded that LOF works better for this particular application, thus the full model and the model after performing PCA were analysed using LOF only.

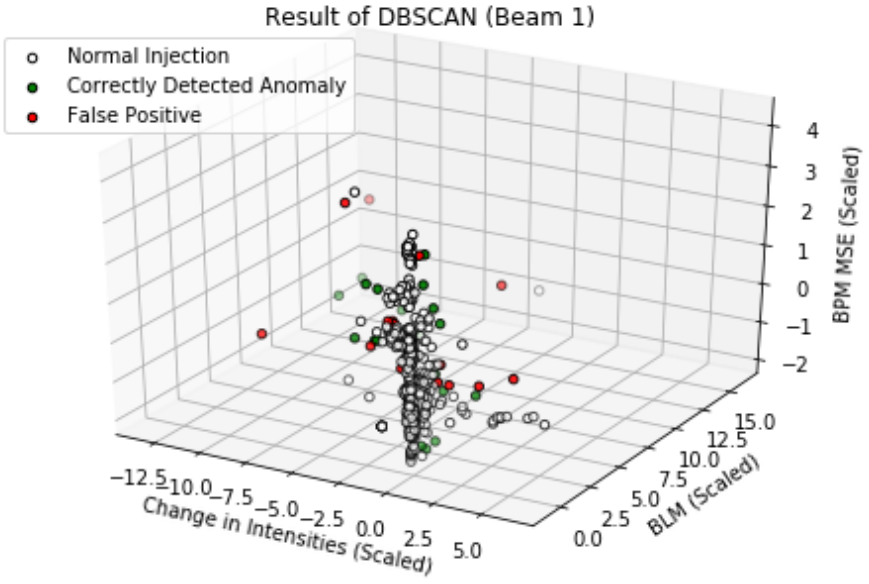

Figure 4.9: 3D Plot Highlighting Anomalies Detected by the 3D DBSCAN Algorithm for Beam 1

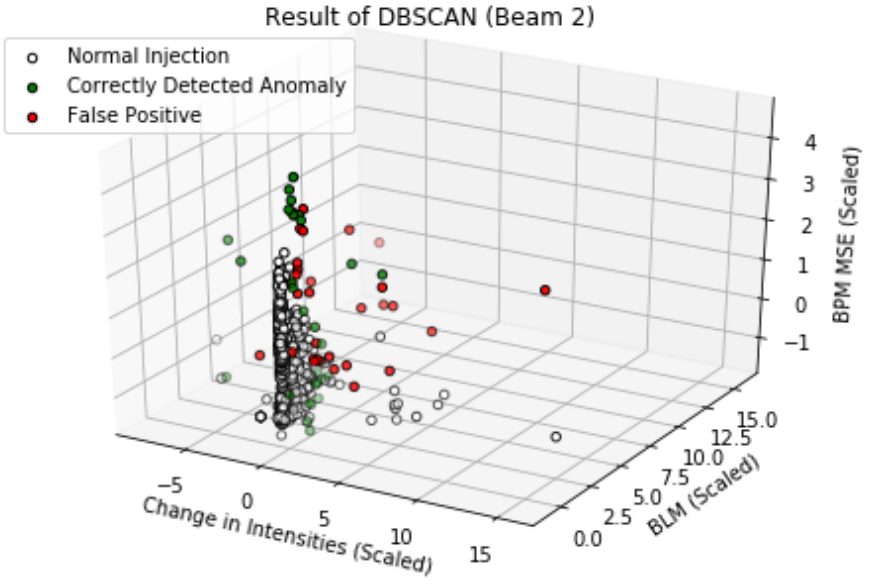

Figure 4.10: 3D Plot Highlighting Anomalies Detected by the 3D DBSCAN Algorithm for Beam 2

4.2.3 Full Model

As expected, running LOF on the full dataset seems to have provided better results than when the 3D dataset was used. In particular, of the 64 injections detected as anomalous for Beam 1, 46 (71.88%) of these were actual anomalies. 96 injections were detected to be anomalous for Beam 2 and 61 (63.54%) of these were actual anomalies. These results are presented in Figures 4.11 and 4.12 for Beam 1 and Beam 2 respectively where the same parameters as the 3D data were used in order to be able to visually compare the algorithms' performances.

Figures 4.13 and 4.14 are 2D plots which show the nature of these anomalies more clearly, this time for Beam 2. Once again it can be noted that some points which are clearly outliers for the provided dimensions are actually not anomalies. This can be seen in particular for high beam losses. There must be a parameter that's not captured in this model that's causing this phenomenon. Thus, the data was transformed using PCA to check if this would improve the performance even more.

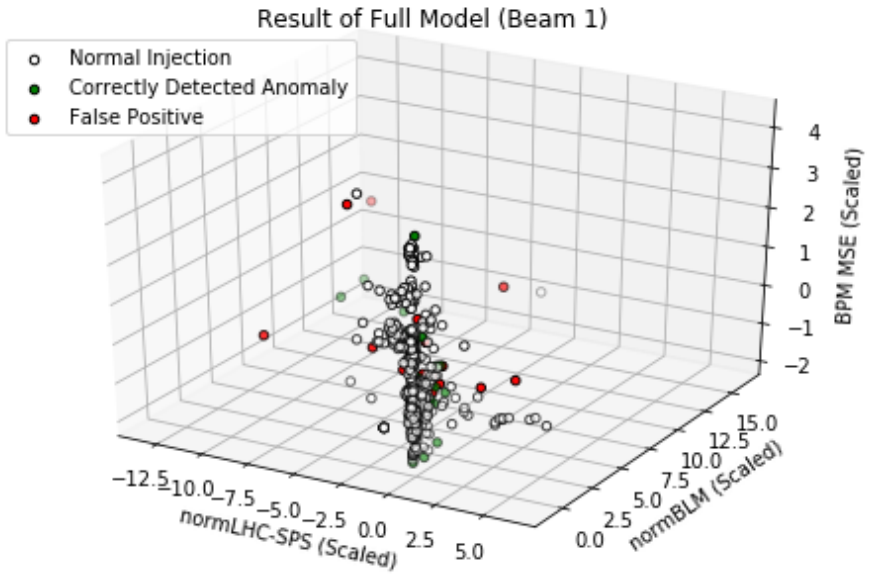

Figure 4.11: 3D Plot Highlighting Anomalies Detected by the Full LOF Algorithm for Beam 1

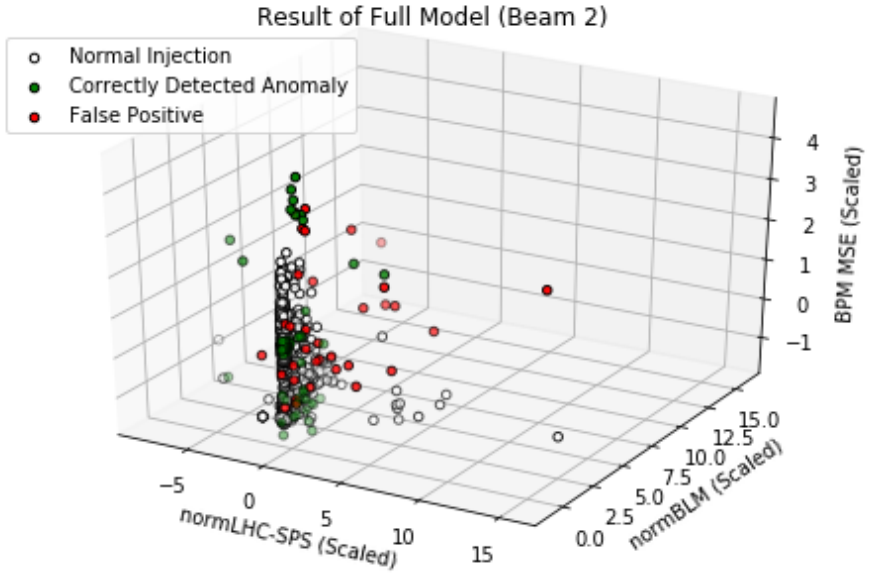

Figure 4.12: 3D Plot Highlighting Anomalies Detected by the Full LOF Algorithm for Beam 2

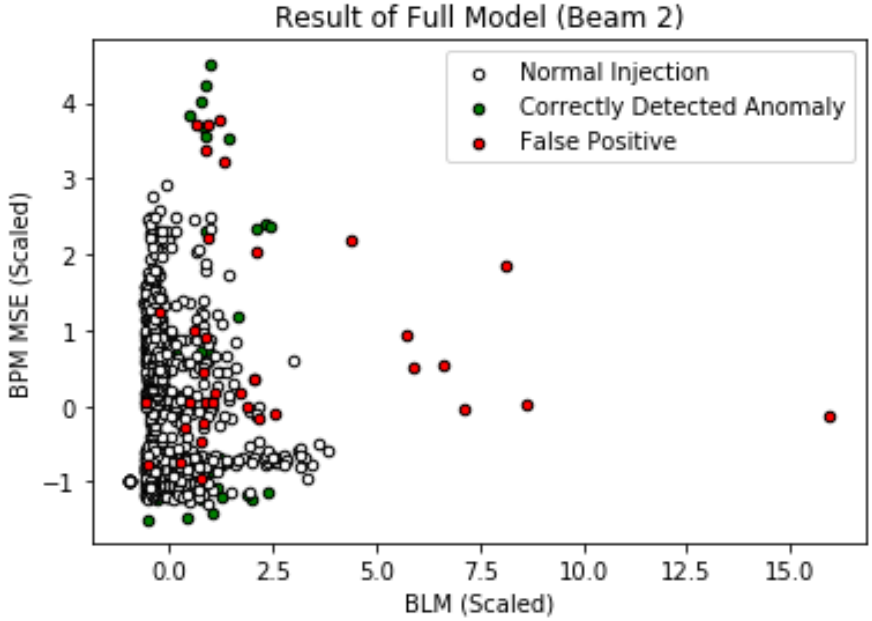

Figure 4.13: Anomalies Detected by the Full LOF Algorithm

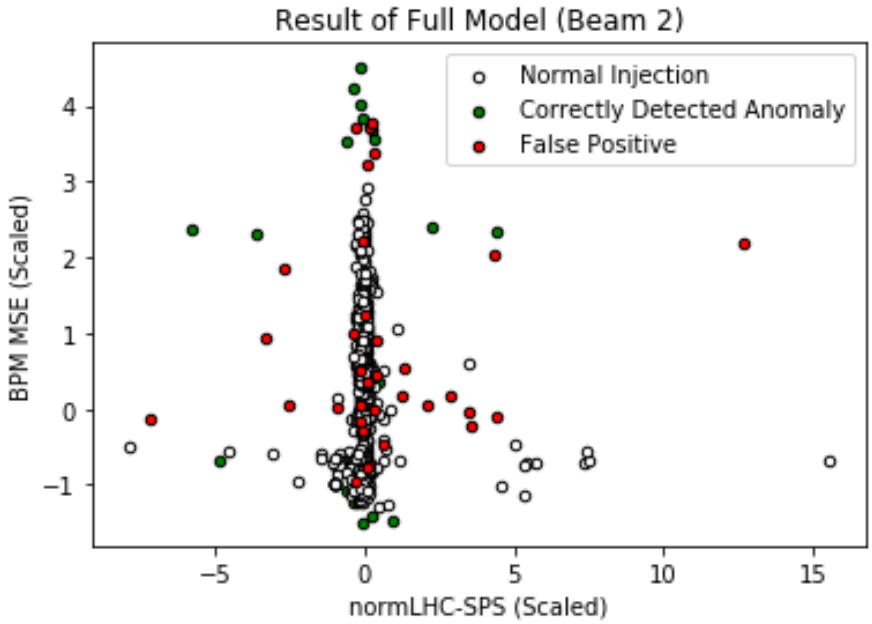

Figure 4.14: Anomalies Detected by the Full LOF Algorithm

4.2.4 PCA Model

Although through PCA, the transformation of the data led to certain injections (such as those with high losses but still not anomalous) to not be incorrectly classified, the overall performance was similar to that of the full model. Of the 63 injections that were classified as anomalies, 43 (68.25%) of them were actual anomalies for Beam 1. For Beam 2 however, 115 points were classified as anomalies and 65 (56.52%) of them were actual anomalies. Figures 4.15 and 4.16 show the nature of the detected anomalies plotted in the same 3 dimensions for Beam 1 and Beam 2 respectively.

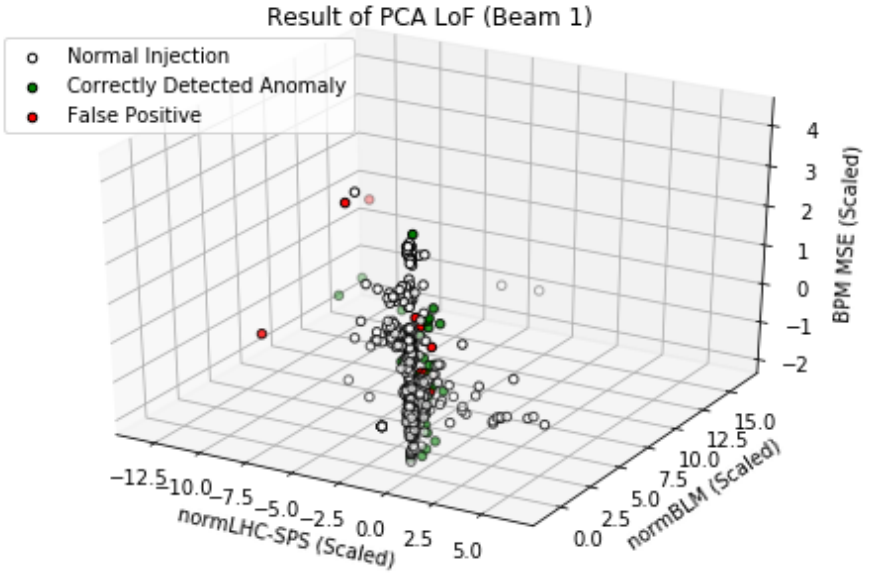

Figure 4.15: 3D Plot Highlighting Anomalies Detected by the Full LOF Algorithm after PCA for Beam 1

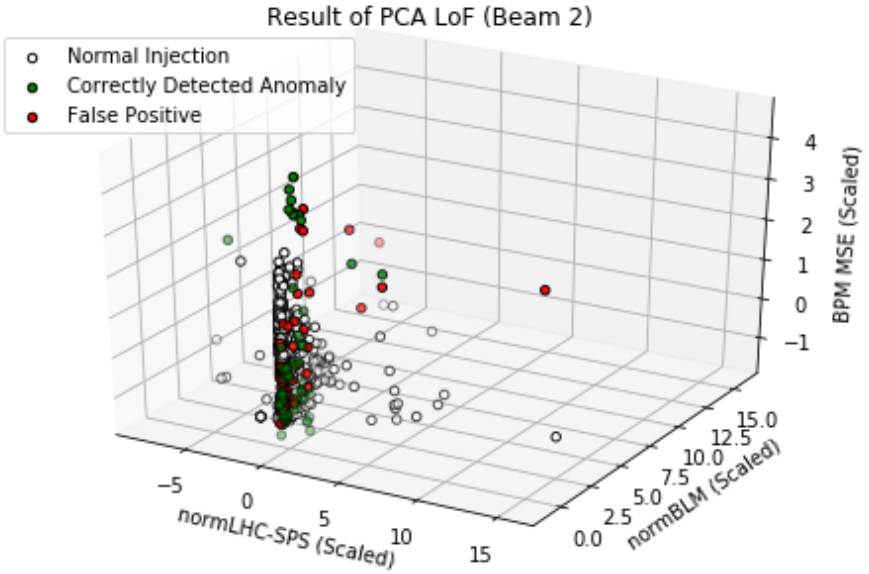

Figure 4.16: 3D Plot Highlighting Anomalies Detected by the Full LOF Algorithm after PCA for Beam 2

4.3 Nature of Anomalous Injections

One of the purposes of this study is to provide clear visual representations of the anomalies in order for experts to understand more clearly the nature of these anomalies with respect to the studied parameters. Figures 4.17 and 4.18 highlight these points.

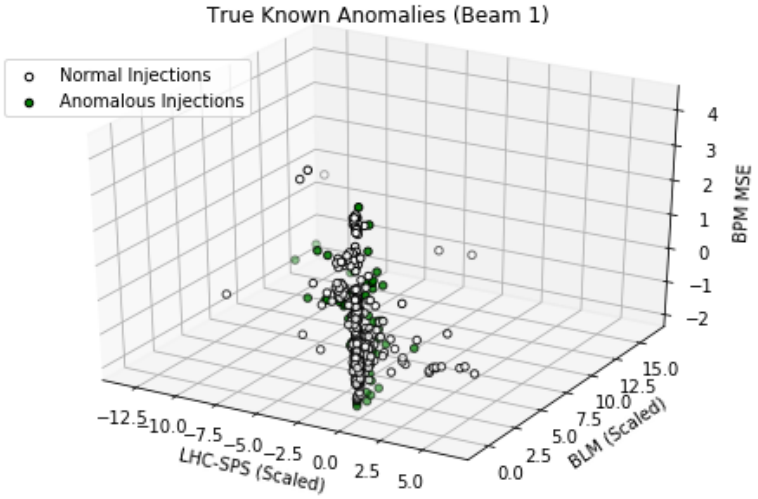

Figure 4.17: 3D Plot Highlighting the Known Anomalies for Beam 1

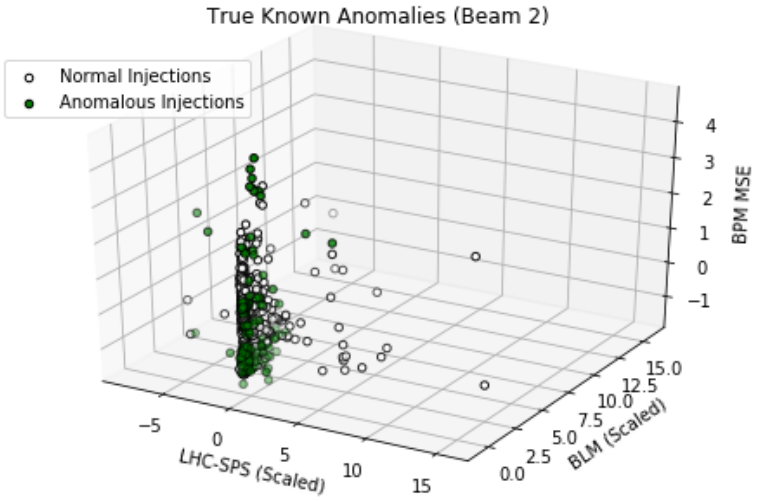

Figure 4.18: 3D Plot Highlighting the Known Anomalies for Beam 2

5

Evaluation

5.1 Performance Metrics

Comparing the performance of unsupervised anomaly detection algorithms is not a straight forward task and can be trickier than the supervised classification case since no ground truth is available [26]. There are many metrics that can be used as performance measures, the most common ones being efficiency measures such as True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR) and False Negative Rate (FNR). FNR is also known as 'Sensitivity', this measures the chances of a true anomaly to be detected by the algorithm and is calculated as shown in Equation 5.1 [27]. This measure is also known as a Type II error in statistics.

$$Sensitivity = \frac{TP}{TP + FN} 100\% \tag{5.1}$$

FPR is also known as 'Specificity', this measures the probability of classifying an anomaly when the injection is actually an anomaly. This measure is defined in Equation 5.2 [27].

$$Specificity = \frac{TN}{TN + FP} 100\% \tag{5.2}$$

The accuracy of the algorithm is defined in Equation 5.3 and gives an overall evaluation of the algorithm's performance [27].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} 100\% \tag{5.3}$$

Other metrics that are useful in evaluating the performance of the algorithms include the Precision, which is defined in Equation 5.4 and the F-measure, which is the harmonic mean of the Sensitivity and Precision and is defined in Equation 5.5 [27].

$$Precision = \frac{TP}{TP + FP} \tag{5.4}$$

$$F - measure = \frac{2 * Precision * Sensitivity}{Precision + Sensitivity} \tag{5.5}$$

5.2 Performance of Algorithms

The overall results for Beam 1 of all the algorithms are summarised in Table 5.1. Table 5.2 shows the efficiency metrics for Beam 1 and Table 5.3 shows the performance metrics for Beam 1.

Method	Total Anomalies	True Positives	True Negatives	False Positives	False Negatives
3D LoF	60	39	745	21	51
3D DBSCAN	40	23	749	17	67
Full Model	64	46	748	18	44
PCA Model	63	43	746	20	47

Table 5.1: Summary of Results for Beam 1

Method	Accuracy	Sensitivity	Specificity
3D LoF	91.59%	43.33%	97.26%
3D DBSCAN	90.19%	25.56%	97.78%
Full Model	92.76%	51.11%	97.65%
PCA Model	92.17%	47.78%	97.39%

Table 5.2: Efficiency of Algorithms for Beam 1

Method	Precision	F-measure
3D LoF	0.65	0.52
3D DBSCAN	0.575	0.3538461538
Full Model	0.71875	0.5974025974
PCA Model	0.6825396825	0.5620915033

Table 5.3: Performance of Algorithms for Beam 1

Similarly for Beam 2, the overall results are summarised in Table 5.4. Table 5.5 shows the efficiency metrics for Beam 2 and Table 5.6 shows the performance metrics for Beam 2.

Method	Total Anomalies	True Positives	True Negatives	False Positives	False Negatives
3D LoF	98	45	1024	53	89
3D DBSCAN	60	28	1045	32	106
Full Model	96	61	1042	35	73
PCA Model	115	65	1027	50	69

Table 5.4: Summary of Results for Beam 2

Method	Accuracy	Sensitivity	Specificity
3D LoF	88.27%	33.58%	95.08%
3D DBSCAN	88.60%	20.90%	97.03%
Full Model	91.08%	45.52%	96.75%
PCA Model	90.17%	48.51%	95.36%

Table 5.5: Efficiency of Algorithms for Beam 2

Method	Precision	F-measure
3D LoF	0.4591836735	0.3879310345
3D DBSCAN	0.4666666667	0.2886597938
Full Model	0.6354166667	0.5304347826
PCA Model	0.5652173913	0.5220883534

Table 5.6: Performance of Algorithms for Beam 2

In both cases, the Full Model has outperformed all the other models, achieving the highest accuracy, precision and F-measure. An accuracy of 92.76% and 91.08% is quite high and is one of the positive outcomes of this study.

The number of anomalies detected in beam 2 was higher than that of Beam 1, however with this increase in anomalies there seems to be a decrease in precision. It is also worth noting that the sensitivity is rather low, having a value of 51.11% for Beam 1 and 45.52% for Beam 2. The fact that the probability of a true anomaly being detected by the algorithm is so low means that further work still could be done to improve the overall performance. The F-measure also highlights this fact as these values are less than expected.

6

Conclusions and Further Work

In this study, LOF and DBSCAN were applied to data gathered from sensors around the moment of injection from the SPS to the LHC as unsupervised anomaly detection algorithms on various datasets. These anomalous points were then presented and it was concluded that the best performing algorithm in this case was LOF when applied to the full dataset with all the provided parameters. Furthermore, the beam displacement over time was shown and the nature of the anomalies were presented in 3D plots.

Similar to Valentino *et. al.*'s study of 2017 [2], this proposed method can positively identify anomalous injections. However the method could use some tweaking as just like in Halilovic's thesis [7], the best performance still "leaves something to be desired" as there were still a large number of false positives being detected.

One limitation in this study came from the loss in performance of this algorithm on the PCA Model. This could stem from the fact that when performing PCA, only 80% of the true variance of the data was kept. It would be interesting to see if the results would improve if more principal components were kept.

As anomaly detection in particle accelerators using unsupervised machine learning has not been explored much as of yet, further research in the area could prove to be beneficial not just for particle accelerator research but also for time series research in general. Further studies on properly fitting the parameters and their effect on the performance of the algorithms would be interesting. Also, as these techniques have the potential of detecting anomalies not being picked up by the IQC, a study on implementing such a method in real time to work in conjunction with the IQC could prove beneficial to reduce the number of anomalous injections in the LHC and save researchers time and

money when running these tests. Furthermore, now that a labelled dataset was created for this study, future work on using supervised learning on this data could result in some interesting results, possibly leading to a more accurate model.

References

[1] L. Evans and P. Bryant, "LHC Machine," *JINST 3*, pp. 1–7, 2008.

[2] G. Valentino *et al.*, "Anomaly Detection for Beam Loss Maps in the Large Hadron Collider," presented at the 8th Int. Particle Accelerator Conference, 2017.

[3] L. N. Drosdal *et al.*, "Automatic Injection Quality Checks for the LHC," in *Proc. ICALEPCS*, 2011, pp. 1077–1080.

[4] C. Lefevre, "The CERN Accelerator Complex," CERN, Tech. Rep., 2008.

[5] C. Roderick, L. Burdzanowski, and G. Kruk, "The CERN Accelerator Logging Service - 10 Years in Operation: A Look at the Past, Present and Future," presented at the 14th Int. Conf. Accelerator & Large Experimental Physics Control Systems, 2013.

[6] V. Kain *et al.*, "Injection Beam Loss and Beam Quality Checks for the LHC," in *Proc. of IPAC*, 2010, pp. 1671–1673.

[7] A. Halilovic, "Anomaly Detection for the CERN Large Hadron Collider Injection Magnets," Master's thesis, KU Leuven, 2018.

[8] E. Holzer *et al.*, "Beam Loss Monitoring System for the LHC," presented at the IEEE NSS, 2006.

[9] *Protection of the CERN Large Hadron Collider*, ser. New Journal of Physics, vol. 8, no. 290, CERN, 2006. [Online]. Available: http://www.njp.org/

[10] M. Meddahi *et al.*, "LHC Abort Gap Monitoring and Cleaning," presented at the IPAC, 2010.

[11] T. Lefevre *et al.*, "First Operation of the Abort Gap Monitors for LHC," CERN, Tech. Rep., 2010.

[12] O. R. Jones, "LHC Beam Instrumentation," in *Proc. of PAC*, 2007, pp. 2630–2634.

[13] (2019, April) Importance of Feature Scaling. [Online]. Available: scikit-learn.org

[14] J. Shlens, "A Tutorial on Principal Component Analysis," April 2014.

[15] A. Zimek, E. Schubert, and H.-P. Kriegel, "A Survey on Unsupervised Outlier Detection in High-Dimensional Numerical Data," *Statistical Analysis and Data Mining 5*, 2012.

[16] M. Richardson, "Principal Component Analysis," May 2009, Class Lecture.

[17] A. Hyvärinen, "Unsupervised Machine Learning," Lecture Notes, University of Helsinki.

[18] M. Ester *et al.*, "A Density Based Algorithm for Discovering Clusters," in *Proc. KDD-96*, 1996, pp. 226–231.

[19] (2018, November) Clustering. [Online]. Available: scikit-learn.org

[20] M. Breunig, H. Kriegel, and J. Sander, "LOF: Identifying Density-Based Local Outliers," in *Proc. ACM SIGMOD 2000 Int. Conf. On Management of Data*, 2000.

[21] A. Edelen, C. Mayes, and D. Bowring, "Opportunities in Machine Learning for Particle Accelerators," *ArXiv*, 2018.

[22] M. Wielgosz *et al.*, "The Model of an Anomaly Detector for HiLumi LHC magnets based on Recurrent Neural Networks and Adaptive Quantization," *ArXiv*, 2017.

[23] (2018, November) PyMVPA Developer Guidelines. [Online]. Available: pymvpa.org

[24] (2018, November) The Shogun Machine Learning Toolbox. [Online]. Available: pypi.org/project/shogun-ml

[25] *Scikit-learn: Machine Learning in Python*, ser. Journal of Machine Learning Research, vol. 12, 2011.

[26] M. Goldstein and S. Uchida, "A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data," *PLoS ONE*, pp. 1–31, April 2016.

[27] K. J. Danjuma, "Performance Evaluation of Machine Learning Algorithms in Post-operative Life Expectancy in the Lung Cancer Patients," Master's thesis, Modibbo Adama University of Technology, 2015.

Appendix

Media Content and Running the Code

The CD that was handed in with this document contains the following files:

1. **Data:** This folder contains all the data used for this study, as well as *.pkl* files created during the course of this work.
2. **Code:** All the code can be found here and is ordered accordingly.
3. **'main.pdf':** This is a soft copy of this document.
4. **'thesis.yaml':** The Anaconda environment needed to properly run the code.

This code was run and created on a Linux System (Ubuntu 16.04.5 LTS) using Python 3 and the Anaconda Environment provided with the code ('thesis.yaml'). In order to run the code make sure you have Python 3 installed and Anaconda, then make sure you're in the correct directory and run the following in the terminal:

```
$ conda env create --file thesis.yaml
$ source activate thesis
$ jupyter notebook
```

Other Plots

This appendix entry contains some plots that didn't make it into the document due to the page limit.

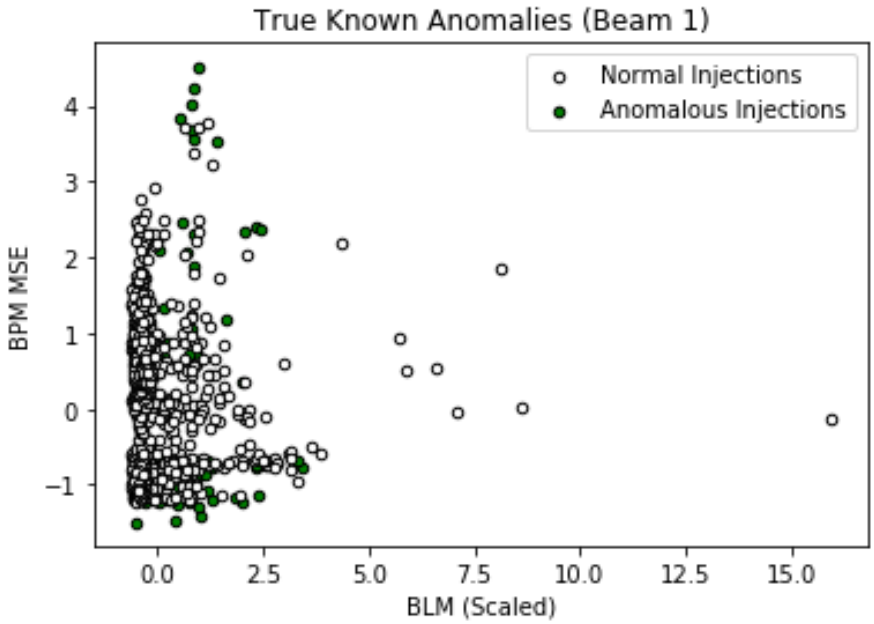

Figure .1: True Anomalies for Beam 1

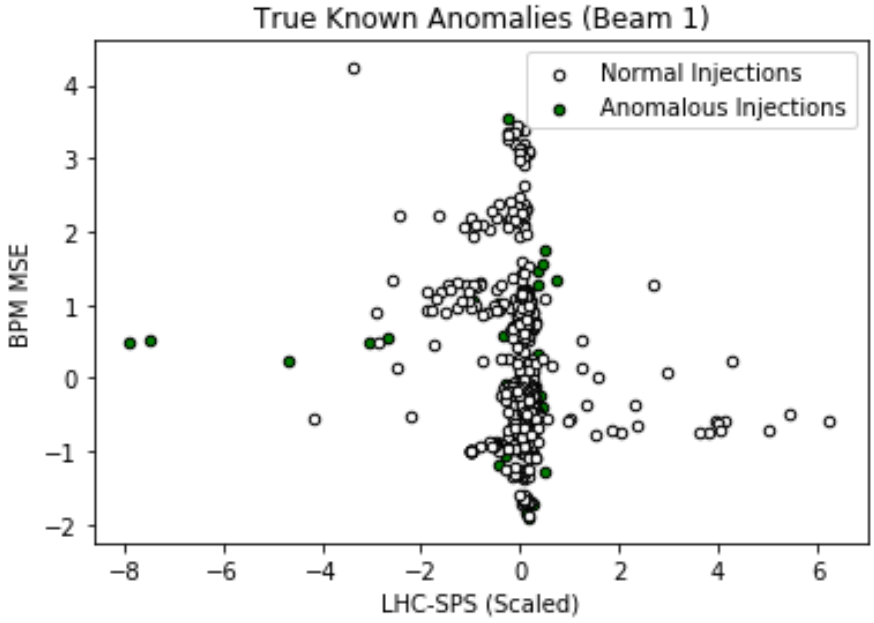

Figure .2: True Anomalies for Beam 1

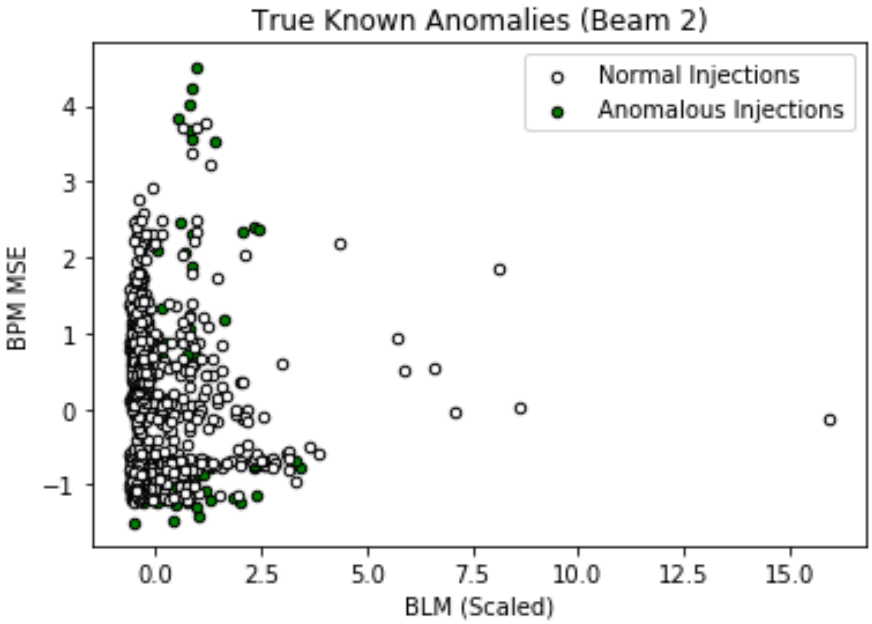

Figure .3: True Anomalies for Beam 2

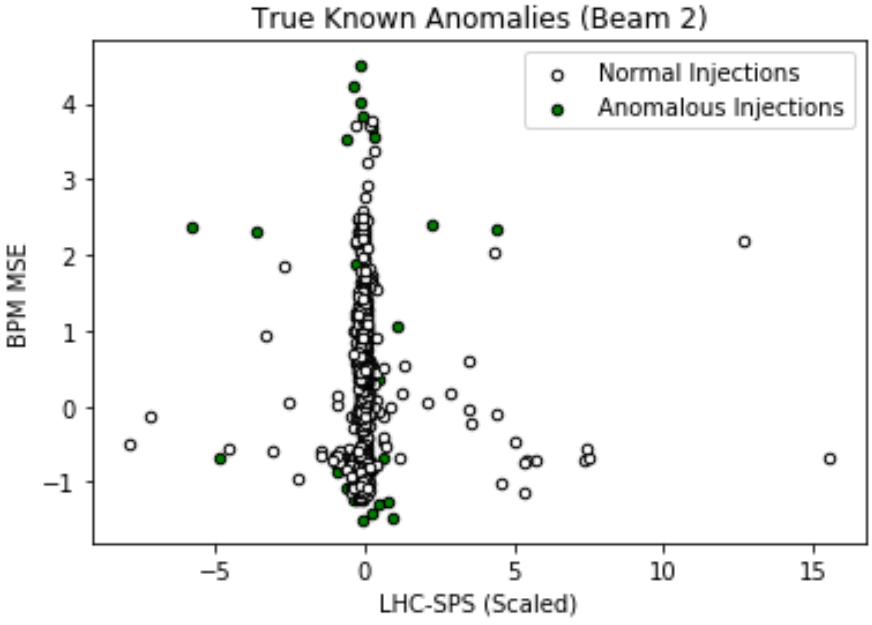

Figure .4: True Anomalies for Beam 2