applied sciences

MDPI

Article

Hybrid Machine Learning–Statistical Method for Anomaly Detection in Flight Data

Sameer Kumar Jasra [1,*], Gianluca Valentino [2], Alan Muscat [3] and Robert Camilleri [1]

[1] Institute of Aerospace Technologies, University of Malta, MSD2080 Msida, Malta
[2] Department of Communications and Computer Engineering, Faculty of ICT, University of Malta, MSD2080 Msida, Malta
[3] QuAero Limited, MST3503 Mosta, Malta
* Correspondence: sameer.jasra@um.edu.mt

Abstract: This paper investigates the use of an unsupervised hybrid statistical–local outlier factor algorithm to detect anomalies in time-series flight data. Flight data analysis is an activity carried out by airlines primarily as a means of improving the safety and operation of their fleet. Traditionally, this is performed by checking exceedances in pre-set limits to the flight data parameters. However, this method highlights single events during a flight, making this analysis laborious. The process also fails to establish trends or reflect potential unknown hazards. This research took advantage of machine learning techniques to recognize patterns in large datasets by implementing the local outlier factor (LOF). In order to minimize human input, a statistical approach was adopted to establish the threshold value above which the flights are considered to be anomalous and interpret the scores. This paper shows that LOF quantifies the degree of outlier-ness of an outlier rather than binary categorizing a point into inlier or outlier, as in the case of clustering algorithms. Thus, with LOF, for the first time, we demonstrated that in the aviation industry, anomalous flights could not only be identified but also be given an anomaly score to compare two anomalous flights in an unsupervised manner. Furthermore, LOF helps to track anomalous behavior in time during the flight. This is insightful when a flight is abnormal, only for some seconds or short duration. For the first time, we attempted to detect flight parameters responsible for anomalous behavior or at least give direction to human experts looking for the cause of abnormal behavior. This was all analyzed with real-life flight data in an unsupervised manner in contrast to simulated data.

Keywords: unsupervised anomaly detection; time series anomaly detection; local outlier factor; Tukey's method; flight data analysis

Citation: Jasra, S.K.; Valentino, G.; Muscat, A.; Camilleri, R. Hybrid Machine Learning–Statistical Method for Anomaly Detection in Flight Data. *Appl. Sci.* **2022**, *12*, 10261. https://doi.org/10.3390/app122010261

Academic Editor: Markus Goldstein

Received: 18 August 2022
Accepted: 10 October 2022
Published: 12 October 2022

1. Introduction

Flight data monitoring (FDM) is the analysis of flight data recorded by the digital flight data recorder (DFDR) onboard an aircraft. This activity is generally performed by airlines to enhance safety and improve the performance of their flight crew. Furthermore, this activity also helps to improve aircraft operating procedures, crew training, aircraft maintenance repairs and designs [1]. With the advent of big data and the machine learning (ML) era, flight data analysis has gained importance in the aviation world for safety management and efficiency improvement [2]. As the aviation industry operates at a tight financial margin [3], therefore, FDM can also help to increase profit margins through improvements in flight operations [4].

Currently, FDM involves the use of statistical methods to highlight exceedances in-flight data. These exceedances are expert-defined thresholds. The occurrence of such exceedance is termed an event. Events are typically analyzed by safety officers with the aim of learning lessons and improving standard operating procedures and good airmanship. The current technique has severe limitations. The state-of-the-art software requires the

human operator to check each exceedance from every individual flight of the airline fleet, making flight data analysis laborious.

Flight data analysis involves the processing of massive amounts of high-dimensional or unstructured data. While the minimum requirements of flight data recorders are to be able to store 88 flight parameters for 25 h, modern aircraft manufacturers have been taking advantage of data storage technology to capture more parameter data, which can then be fed into the safety and design of their product. For example, the digital flight data recorder onboard the Boeing 787 can record approximately 2000 parameters for 50 h. Therefore, while the data recording aspect has improved dramatically, the data analysis aspect has been lagging, making it very difficult for a human to establish patterns in exceedances and thus receive further benefit from such events. Moreover, the parameter pre-defined maxvals have been based on historical lessons. Therefore, they also impose an intrinsic limitation in the system, which is unable to discover new anomalies.

This paper addressed this by adapting a modern machine learning algorithm to analyze flight data more effectively and efficiently and making it suitable to establish useful trends. The paper made use of an unsupervised technique—the local outlier factor to establish anomalies in flight data. The technique is added with a hybrid statistical element, thereby reducing the continuous reliance on a human expert to decide if an outlier is anomalous or not. In order to achieve this, the paper is organized in the following manner: Section 2 provides a brief literature review of some machine learning techniques that have been applied to flight data analysis. Section 3 introduces the local outlier factor technique, which is the focus of this paper. This section describes the various tweaks that were required for the successful implementation of the technique in this domain. Section 4 provides the results for a scenario of flights with the same tail number landing at three different airports, and a discussion follows. A conclusion is finally provided in Section 5.

2. Literature Review

In the past decade, several advanced machine learning-based analytical methods have been proposed to find anomalies and new hidden patterns within large sets of flight data. Broadly, all of these methods can be categorized into three categories: supervised, unsupervised and semi-supervised. Supervised techniques make use of the correct solution known in advance for a learning set of data, and this solution is used to train the models. On the other hand, unsupervised techniques do not know any solution for the given dataset, and therefore learning algorithms try to find the hidden structure of the given dataset. Semi-supervised techniques generally use a small amount of data with correct answers and a large amount of data without any solution.

The labeling of data in anomaly detection problems is time-consuming and involves very high costs. It requires human expertise, and thus it may not be free from human bias. Furthermore, labeled datasets with anomalies tend to be highly skewed, e.g., 95% normal and 5% anomalous. This may have issues such as challenges in training a classifier in a supervised way. In civil aviation, finding or generating examples based on specific anomalies is highly dependent on human expertise. It is not possible to acquire all the examples of anomalous flights (outliers), and therefore it is important to select the correct examples of labels in order to perform accurate classification with the least number of labels. This is known as the problem of label acquisition [5]. Furthermore, unsupervised learning helps to model the underlying or hidden structure or distribution in the data to learn more about the data. The following paragraphs discuss major anomaly detection methods that have been proposed for flight data analysis in the past decade.

Based on the work of Srivastava [6], which combined discrete and continuous data, Multiple Kernel Anomaly Detection (MKAD) developed by Das et al. [7] is based on kernel functions and a one-class support vector machine. It is still one of the state-of-the-art methods to detect anomalies in flight data containing both continuous as well as discrete data; both works were one class problem. The ClusterAD-Flight [8] and ClusterAD-DataSample [9] were two cluster-based anomaly detection algorithms developed to address

the issue of multiple classes of flight data. Apart from the binary classification of the flights, these two algorithms identified the norms of flight data and detected any outliers. There were no exceedance-based criteria used in these algorithms. The ClusterAD-Flight, proposed by Li et al. [8], was based on a clustering method called density-based spatial clustering of applications with noise (DBSCAN). In this method, the multivariate time series of each flight was transformed to a high dimensional vector space, which resulted in performance issues when applied to large datasets. A Gaussian mix model (GMM) can be used to detect multiple patterns in flight data. This model was used in ClusterAD-DataSample [9], where clustering was performed at each time point during a flight. Probability-based models were used to characterize the results. The performance of three methods, ClusterAD (CAD), MKAD and exceedance-based detection, was discussed in the work of Li et al. [8]. Surpassing the exceedance-based detection method, both CAD and MKAD were much more capable of identifying operationally significant anomalies. For continuous flight parameters such as airspeed and for earlier known safety issues, CAD performed better as compared to MKAD. MKAD, on the other hand, was more sensitive towards discrete parameters such as landing gear on or off. More heterogeneous datasets were suitable for MKAD, and both MKAD and CAD detected anomalies across a group of flights. Semi-Markov switching vector autoregressive (SMS-VAR) model used by Melnyk et al. [10] detected anomalous flights by measuring the difference between the predicted value and the actual data value.

Das et al. [11] used a real-world dataset of a commercial passenger jet, and their work was focused on the descent phase of the flight. This work, based on iOrca [12], which is the scalable or indexed version of Orca [13], was developed by (Bay and Schwabacher, 2003). Similar to our work, Orca is also based on the k nearest neighbor algorithm. Pairwise distance is calculated in the Orca, whereas in the method proposed in this paper, distance is measured between a data point and its neighborhood to calculate the density of that data point. The performance of iOrca and MKAD was studied by Matthews et al. [14]. An anomaly score was given by both methods. Scoring anomalies helps to distinguish between two anomalous data points. The whole flight was labeled as normal or abnormal by MKAD, whereas iOrca was able to point out the location of the anomaly as well. MKAD performed better for a group of flights, whereas iOrca was better for individual flights

The work conducted by Oehling and Barry [15] used a similar algorithm as proposed in this paper. Local outlier probability (LoOP) is a method derived from the local outlier factor algorithm, which is the basis of this paper. LoOP is less dependent on parameter k of the k nearest neighbor. However, this paper discussed a method to decide the optimal value of parameter k. Moreover, the work conducted by Oehling and Barry [15] differs in the way raw data are processed and how the flight data are input into the algorithm.

Anomaly detection in aircraft operations using a self-organizing map neural network (SOM NN) was performed by Megatroika et al. [16]. It is a neural network-based unsupervised method with two layers known as the input layer and output layer. The input layer is a vector representation of the data, and the output layer presents the input data in a self-organized manner. SOM NN transforms data points in high dimensions into two-dimensional space. This reduction in dimensionality helps in further analysis of the data as the structure of the dataset is better visualized. Thus, SOM clusters similar data points, and the anomalous data points are discovered easily.

Deep Learning algorithms such as recurrent neural networks (RNNs) can handle long sequences of multivariate time series data without any dimensionality reduction and can detect anomalies in hidden features as well [17]. RNNs can have long short-term memory (LSTM) cells or gated recurrent units (GRUs). A key attribute of RNNs is their ability to persevere information for later use in the network. This makes RNN more suitable for the analysis of time series data, such as flight data that changes over time.

After reviewing the existing work on flight data analysis, the following section focuses on the local outlier factor technique and its applications to flight data analysis.

3. Local Outlier Factor

3.1. Basic Concept

The local outlier factor (LOF) models outliers as the points in the data space that are isolated from the remaining dataset based on the density of data points. The dense regions in the dataset are found, and outliers are defined as those points that do not lie in these dense regions or are far away from the dense regions [18]. Conceptually LOF is very similar to clustering and distance-based methods for outlier detection but combines the strength of all three-proximity based outlier detection methods, namely clustering, distance-based and density-based. LOF segments the data space instead of segmenting data points, as in the case of clustering. Instead of finding sparse data points, LOF finds sparse regions in the data space. With LOF, sparse regions in the data space can be easily identified with the combination of many original features (flight parameters). Thus, LOF is suitable for high-dimensional data such as flight data. LOF divides the data space into small local regions (defined as distance-based regions). The number of other data points within this small local region is used to define local density. In LOF, these local density values are converted into outlier or anomaly scores. The runtime and the memory requirements of the LOF algorithm scale by $n \times 2$ [15]. Thus, a dataset of thrice the size requires six times as much memory and calculation time.

3.2. Implementation

Figure 1 summarises all the steps of analyzing flight data using the hybrid LOF–Tukey's method, and the following subsections discuss each step mentioned in Figure 1 independently.

3.2.1. Flight Data

Flight data, a type of multivariate time series used for this analysis, are available in the public domain and are provided by NASA [19]. The entire flight data were disidentified to protect the identities of the airlines as well as the flight crew involved during the actual flights. The flight data used for this study are from three specific aircrafts (identified by the tail number). These aircrafts were flown over different destinations over a given period. In total, there were 186 flight parameters recorded by the various onboard sensors.

3.2.2. Pre-Processing

One hundred and thirty-two relevant flight parameters out of a total of one hundred and eighty-six parameters recorded by the aircraft sensors were selected for the analysis of flight data. The selection of these flight parameters was based on discussions with industrial experts. Dynamic parameters relating to weather were removed so all flights may be treated on the same grounds and independent of any environmental influence. Any event related to weather influences the pilot's behavior, which is captured by other flight parameters. Three airports located in the USA were randomly selected for the analysis. These airports are Detroit (DTW), Minneapolis (MSP) and Memphis (MEM). Flights with the same tail number were grouped together. Therefore, each airport has flights from the same group or tail number.

As presented in Figure 2, accidents during the approach and landing phase account for more than 50% of all the accidents, even though this phase is only 16% of the flight time [20]. Moreover, as per a report by Airbus [21], most of the accidents (fatal and nonfatal hull-loss accidents) over the last 20 years occurred during the approach and landing phases. In 2020, all three fatal accidents occurred in the approach and landing phases [21]. Therefore, for this study, the approach and landing phase was selected. In this phase, only three minutes before touchdown are studied to detect anomalous flights. The touchdown point for each aircraft was identified using phase (PH), weight on wheels (WOW) and latitude (LAT) and longitude (LONG) flight parameters. In order to avoid negative values of flight parameter altitude (ALT) for a given airport, the minimum altitude value for that airport was subtracted from all readings of the ALT flight parameter. Therefore, all the flight

landings at a given airport were synchronized in time with each other to touch down at the same time.

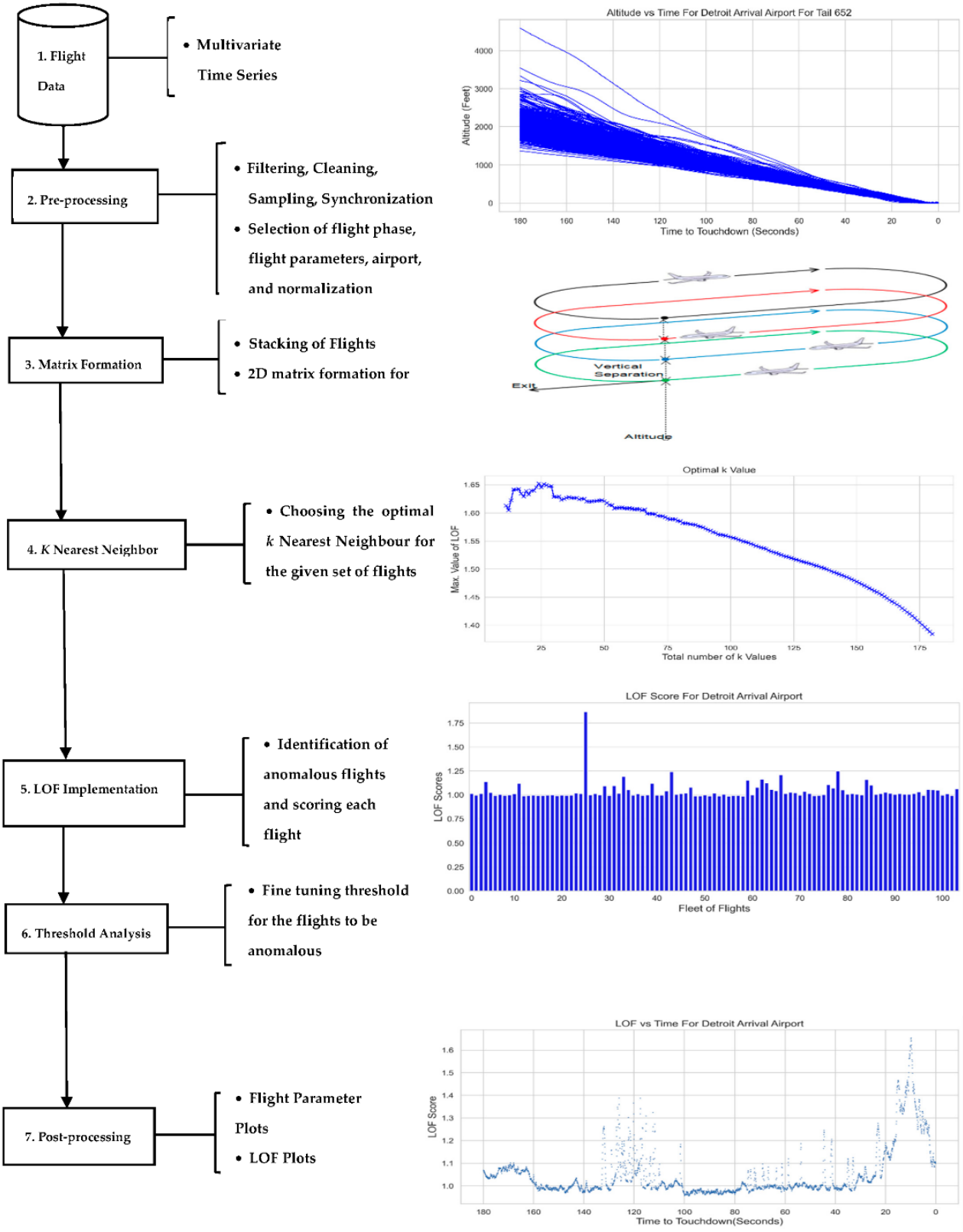

Figure 1. Step-by-step implementation of FDM using hybrid machine learning–statistical method (Image created using BioRender.com).

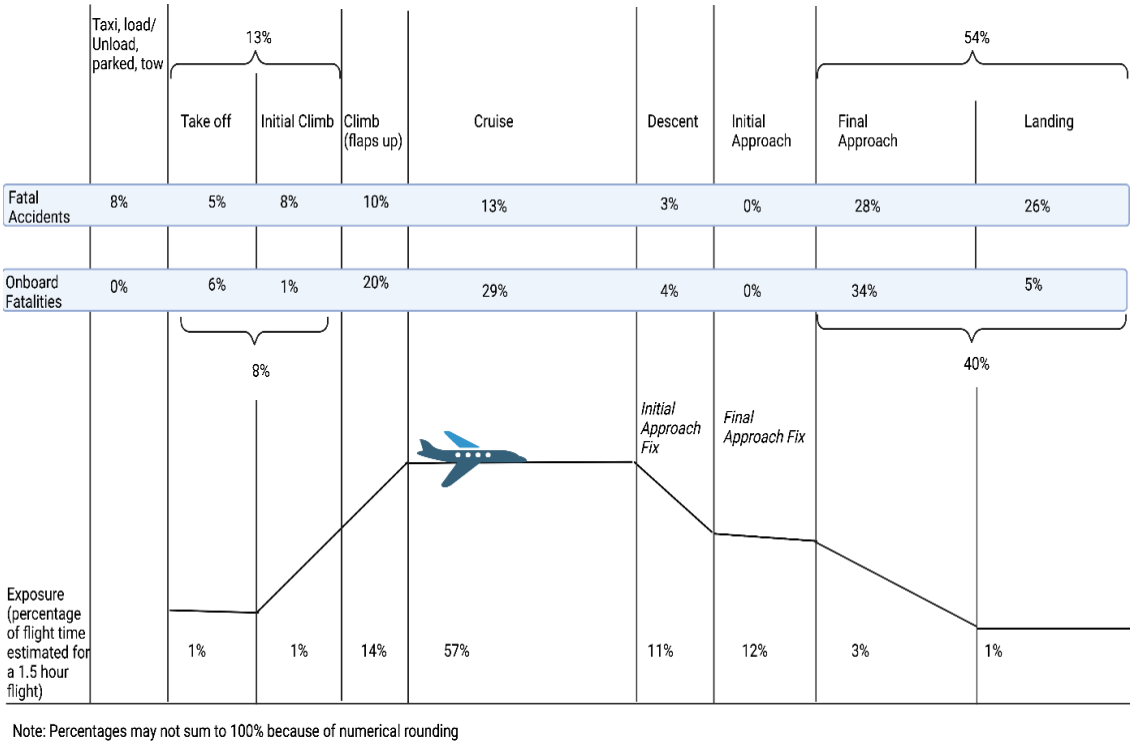

Figure 2. Phase wise fatal accidents (2011–2020) [20] (Image created using BioRender.com).

The dataset was cleaned of any noise or missing data by the onboard sensor. All flight parameters were normalized to accommodate different units and ranges of values of flight parameters.

The data files were converted from MAT files to SQL tables for better access and analysis. Flight parameters were recorded at sampling rates ranging from 1 Hertz to 16 Hertz. LOF algorithm requires flight data to have the same number of data points for each flight. Therefore, flight parameters recorded at lower sampling rates were all converted to 16 Hertz by interpolating the values. This helped to synchronize data from all the flights. Figure 3 shows a plot of flight parameter altitude (ALT) plotted against time for 600 flights from one tail (652) arriving at Detroit airport after all data pre-processing was completed.

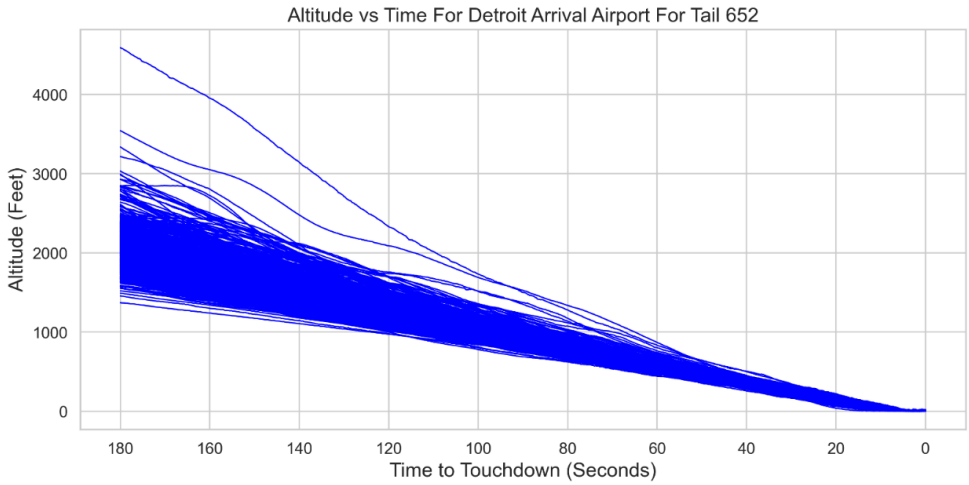

Figure 3. Altitude vs. time plot for synchronized flights after data pre-processing.

3.2.3. Matrix Formation

Flight data consist of various flight parameters recorded frequently. The analysis of flight data is challenging because of the high dimensions in the dataset. In order to compare various flight parameters across the number of aircrafts, a matrix M was designed. The two-dimensional matrix consists of all the flight parameters of aircrafts under study stacked together. The matrix M is defined as [m × n], where m is the total number of flights under study and n is the total number of features. This feature space is given by flight parameters p recorded at d time-steps for each flight. Therefore, there were p^*d points in the feature space for each of the flights under study. For each flight, the feature space was defined as:

$$[p_1d_1, p_2d_2 \cdots p_id_j] \quad (1)$$

where p_id_j is the value of the ith parameter from p total parameters recorded at jth time-step, and as there are m total number of flights, the matrix M is given by

$$\mathrm{M} = \mathrm{m} \times [p_1d_1, p_2d_2 \ldots \ p_id_j] \quad (2)$$

3.2.4. k Nearest Neighbor

In the LOF algorithm, parameter k is the neighborhood size, which defines the neighborhood of a data point for the computation of its local density. The original LOF paper [18] proposed some guidelines for determining the range of value for parameter k. In principle, the value of k should be lower bound by the minimum number of points in a cluster while the upper bound should be the maximum number of nearest points that can potentially be anomalies. However, such information is generally not available and is highly domain dependent. Even if such information is available, the optimal k value between the lower bound and upper bound is still undefined. A range of k values is suggested as one value of k cannot be generalized over various datasets with diverse underlying data distribution. Aviation data are time-series-based multidimensional data. The dimensions typically run into thousands. This nature of data being of higher dimensions, heterogeneously unstructured and having diverse underlying distributions makes the process of determining the optimal value of k more challenging. The following paragraphs explain the technique used to determine the optimal k value while using the LOF algorithm to identify anomalous flights.

A locality or local density is defined by k Nearest Neighbors (kNN). The distance between point A and its neighbors is calculated to estimate the local density. Then the local density of point A is compared with the local densities of its neighbors. If point A has a lower density than its neighbors, then it is considered an outlier. In Figure 4, point A has a much lower density than its neighbors 1, 2 and 3. Therefore, point A is an outlier compared to its three nearest neighbors (k = 3).

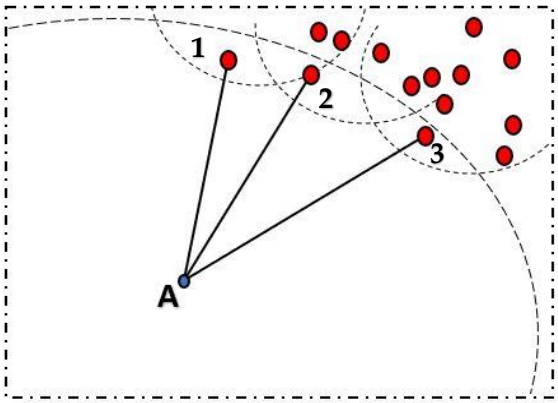

Figure 4. Illustrating basic concept of LOF.

The local density is measured by the typical distance at which point A can be "reached" from its neighbors. This defines "reachability distance" in the LOF algorithm. The formal definition of the LOF algorithm interpreted by Breunig et al. [18] is discussed in the following paragraphs.

Let *k* distance (A) be the distance of point A to its *k*th nearest neighbor. The set of the *k*NN includes all the points at this distance, which can, in the case of a tie, be more than *k* points. We denote the set of *k*NN as Nk(A).

The reachability distance of point A from point B is the distance between these two points, which is calculated as the maximum value of the two distance measures, namely *k* distance (B), which is the distance of the point B to its *k*th nearest neighbor and d(A, B), which is either Euclidean, Manhattan, etc. distance measure. Thus, mathematically reachability distance can be defined as:

$$\text{reachability distance}_k(A, B) = \max\{\text{k distance (B)}, \ d(A, B)\} \tag{3}$$

The local reachability density of point A is defined as

$$\text{LRD}_k(A) = 1/\left(\frac{\sum_{B \in N_k(A)} \text{reachability distance}_k(A, B)}{|N_k(A)|}\right) \tag{4}$$

LRD_k (A) is the estimated distance at which point *A* can be found by its neighbors. If a neighbor were to reach out LRD value distance in any direction, then it would be most likely to find that individual point *A*. LRD is the count of the items in the kNN set of each flight over the reach distance of the point to all the values in its set, which is the kNN set. Equation (4) summarizes LRD_k (A) as the inverse of the average reachability distance of point *A* from its neighbors.

The local reachability densities are then compared with those of the neighbors using

$$\text{LOF}_k(A) = \frac{\sum_{B \in N_k(A)} \frac{\text{LRD}_k(B)}{\text{LRD}_k(A)}}{|N_k(A)|} = \frac{\sum_{B \in N_k(A)} \text{LRD}_k(B)}{|N_k(A)| * \text{LRD}_k(A)} \tag{5}$$

LOF is the average local reachability density of the neighbors divided by the object's own local reachability density. LOF values of point A can be interpreted as follow:

1. LOF value approximately equal to one means that the density of point A is comparable to its neighbors, and thus A is not an outlier;
2. LOF value less than one means that point A has a higher density than its neighbors, and thus A is an inlier;
3. LOF value greater than one means that point A has a lower density than its neighbors, and thus A is an outlier.

A small value of *k* is not preferred as the algorithm becomes sensitive to noise, and a too-large value does not detect local anomalies. There cannot be one definite value for *k* in finding the anomalous flights, as each dataset given is unique in the number of total flights (samples) and the number of flight parameters. As such, there are no pre-defined statistical methods to find the most optimal value of *k*. In this present technique, before implementing the LOF algorithm, the optimal *k* value was determined by calculating the LOF score values for each possible *k* value. As discussed in detail in the original paper [18], the least value for parameter *k* should be 10. This helped to remove unwanted statistical fluctuations. In the aviation domain, we could not fix the lower bound for *k* as we did not know how many minimum similar objects a cluster had (other objects can be outliers relative to this cluster), or we did not know the exact number of normal flights. Similarly, we could also not decide the total number of anomalous flights in each dataset, and hence the upper bound for *k* could not be fixed. Since the algorithm uses unsupervised learning and we did not have labeled data or know in advance the normal or abnormal flights, *k* can be any number. For every *k* value, we observed the LOF score calculated. The *k* value,

which gives the highest LOF score, is the optimal k value. The k value corresponding to the highest LOF score was chosen to catch the instance at which the object is the most outlying. The lowest LOF score was not chosen as it erases the outlying nature of an object completely. Figure 5 highlights three values for parameter k taken at three instances. The k value, which gives the highest LOF score, is 13. The k value of 49 gives the mean LOF score, and the lowest LOF score is represented by a k value of 80. In each of these three instances, the LOF score was calculated, and the results for the three flights are presented in Table 1.

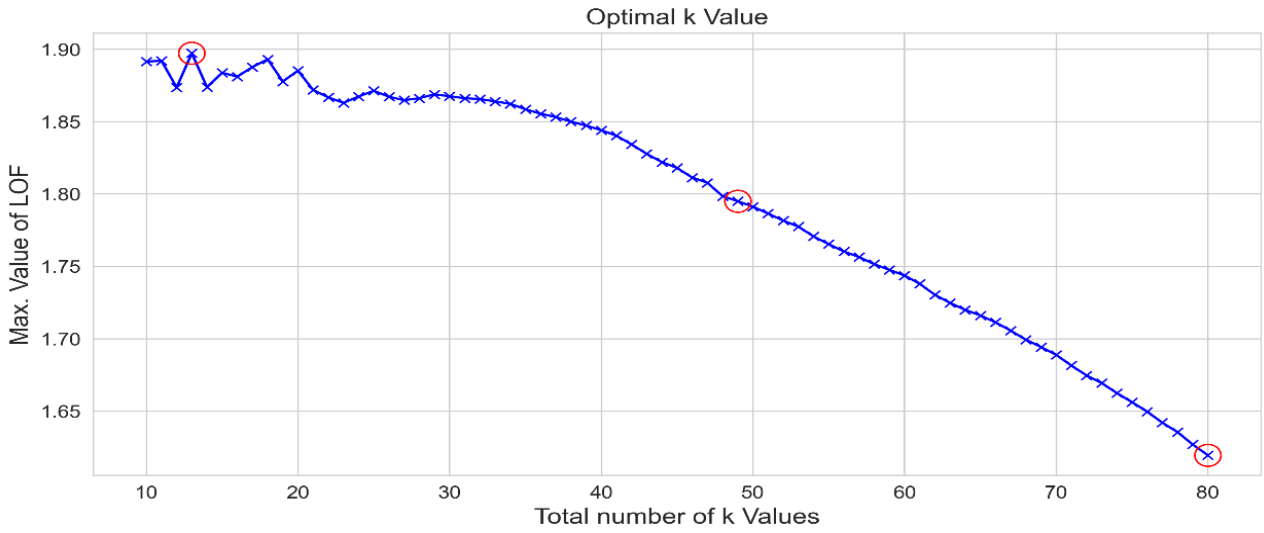

Figure 5. Selection of k values at different instances of LOF score.

Table 1 shows that all three flights are most outlying for a k value of 13, which corresponds to the highest LOF score in the dataset. Flight 2 and Flight 3 even tend to become normal (with respect to Tukey's threshold of LOF score value of 1.1.69) when LOF is calculated at k value corresponding to the mean LOF score or the lowest LOF score in the dataset. Thus, the optimal value for k is at the instance which captures the maximum LOF score in the dataset.

Table 1. Outlying nature of flights for different k values.

Flights	LOF Score (k = 13)	LOF Score (k = 49)	LOF Score (k = 80)
Flight 1	1.89719894601340	1.794999571605941	1.61952094645704
Flight 2	1.24792810898067	1.09357548151929	1.00903988539139
Flight 3	1.20925468652125	1.12810045075146	1.07174161543744

The distance measure is another important parameter of the K nearest neighbor algorithm. Choosing the right type of distance measure is another challenge involved. Euclidean distance was not chosen as the distance measure because as the dimensionality increases, the curse of dimensionality impacts Euclidean distance measure [22]. In the case of flight data, there are 132 parameters recorded at different time steps. The dimensions run into thousands. Cosine similarity as the distance measure is suitable for high-dimensional datasets. However, the disadvantage of this measure is that it ignores the magnitude of vectors and only considers their direction. Therefore, the difference in values may not be considered, which is important for detecting anomalous flights.

The Manhattan distance was chosen to find the optimal value of k as it calculates the distance between real-valued vectors. This helps to capture the k value where a data point is most outlying. More importantly, flight data have both discrete and binary attributes. For instance, flight parameters such as landing gear and weight on wheels have only values as 0 or 1. Using Manhattan as the distance measure helps in this scenario as it considers the paths that realistically could be taken within values of such attributes. Manhattan distance measure is fast compared to Euclidean distance for each pair of distance calculations; there is no need for squaring or taking square roots to obtain absolute values. Moreover, squaring of the vector components can skew results by giving more weight to outliers that are of our interest.

Minneapolis airport was taken as an example. There are, in total, 200 flights in a dataset arriving at this airport. We calculate the LOF score for all possible k values between 10 and 180. The k value, which gives the highest LOF score, is the optimal one. In this case, the k value of 53 is the most optimal as it has the highest LOF score of 1.87. The left side of Figure 6 gives the plot for the LOF score and possible k values between 10 and 180. From the plot, it can be seen that the highest LOF value is between the k value of 40 and 60. Moreover, in between this range, only the k value stabilizes before decreasing. The highest LOF score is between k values 50 and 55. The highlighted box of Figure 6 shows that the highest LOF score is at a k value of 53. Therefore, 53 is the optimal k value for this case.

For the purpose of comparison, Figure 7 shows the line plot comparing the optimal k values for three airports Detroit, Minneapolis and Memphis. The optimal k value is for the different total number of flights arriving at each airport. From the work of Lall and Sharama, it was concluded that the suitable setting of the k value should satisfy $k = \sqrt{N}$ for the datasets with a sample size N which is larger than 100 [9]. However, such a setting has been proven to be not suitable for flight data in this study. In the case of the detection of anomalous flights, this general rule is not valid due to the nature of aviation data, and the method discussed above gives a dynamic value of k depending on the total number of fights and the arrival airport.

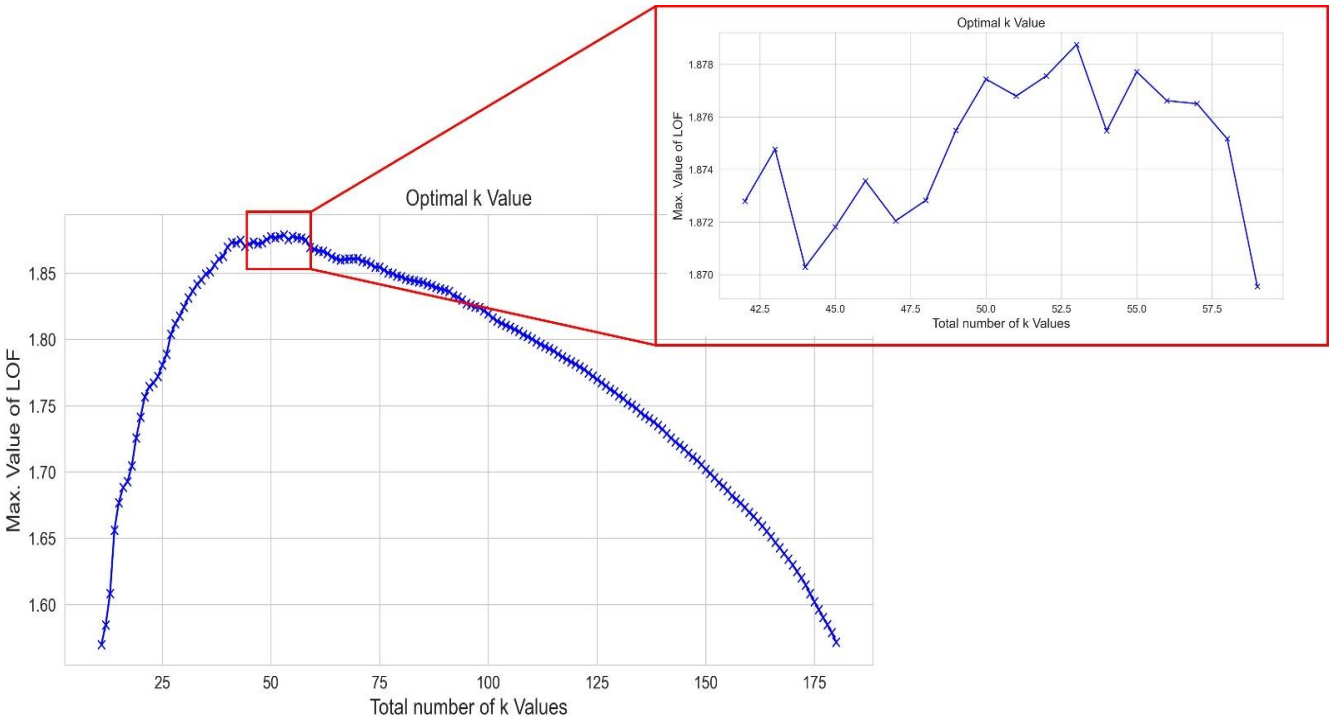

Figure 6. Maximum LOF scores across all flights at Minneapolis airport for different k values.

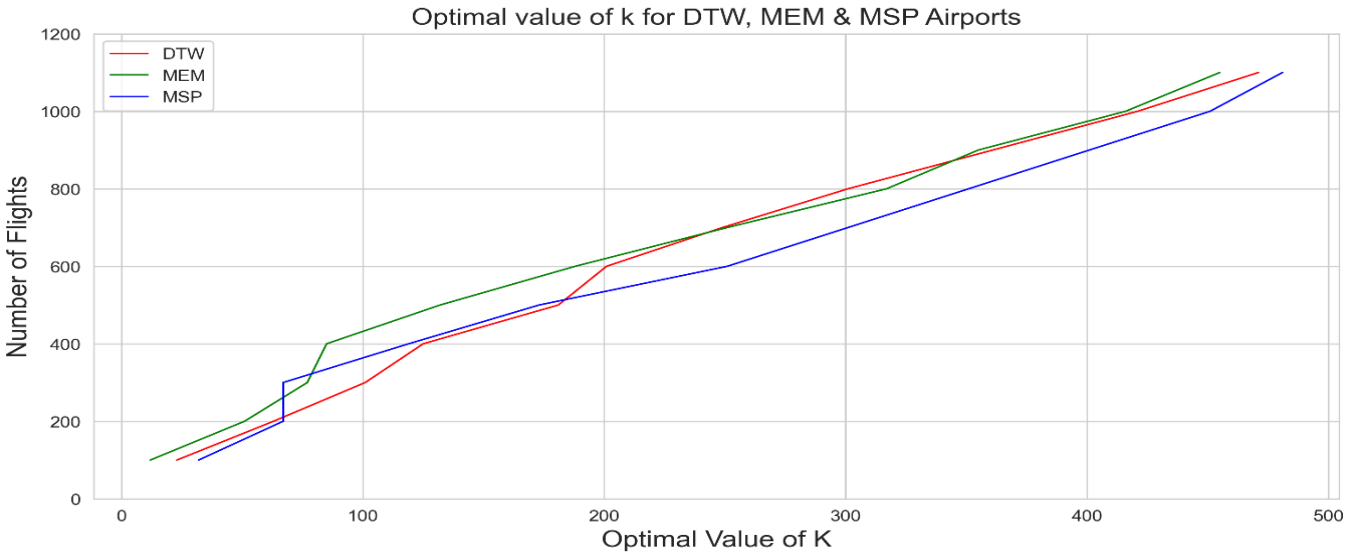

Figure 7. Optimal *k* vs. number of flights for three airports.

3.2.5. LOF Implementation

Based on the optimal *k* value identified and Manhattan as the distance measure, the LOF score for each flight with the same tail number approaching the same airport was calculated. As a rule, any flight with LOF score greater than one is anomalous. All such flights need to be investigated further. Investigating all those flights is still a laborious process and requires a lot of human effort. Setting up a threshold can help to further filter the flights that have scores greater than one but are still normal. The following sub-section explains how the threshold is set.

3.2.6. A Hybrid Statistical Threshold Analysis

As discussed in the original paper by Breunig et al. [18], any data point "A" deep inside a cluster (dense region) with a density comparable to its neighbors or higher than its neighbors is an inlier. The LOF value of such point A will be =1 or <1. The paper [18] also gives a detailed discussion on points with LOF value > 1 and labeled as outliers. Such points may be near the periphery of the cluster or dense region (LOF value slightly greater than 1) or far outside the cluster (LOF value significantly greater than 1). The original LOF paper [18] defines very well the lower and upper bound of an outlier in terms of LOF values. Whilst, in theory, any point with a LOF value > 1 is an outlier, in practice, there are often peripheral points that are not necessarily anomalous. Often these can be explained by experts in a particular field. Therefore, our aim here is to set a threshold that defines how far a point should be from a cluster of points (dense region) to be categorized as an outlier.

The value of this threshold is, to an extent, dependent on the field of application. In the flight data analysis, while it may be acceptable to have false positives, which would then be further reviewed by experts, false negatives are not tolerated as these may provide a false sense of safety. In order to achieve this, the LOF results were discussed with aviation experts. In our research, we came across several false positives whereby flights with LOF values greater than 1, despite deviating from the rest of the fleet, would still fit under the standard operating procedures and therefore posed no threat to safety. In order to decrease false positives, a method for fine-tuning the LOF threshold value was required. The upper limit of LOF value greater than 1, which could safely be a normal flight, was to be determined as a threshold. Deciding this threshold arbitrarily and based on feedback from a human expert is not only a laborious process but also introduces human bias. Furthermore, it was also found that this threshold changes for each group of flights and for each arrival airport. In order to decide this threshold, statistical methods such as the z-score method, bell curve method, Tukey's method and median method were explored.

Statistical analysis helps to find important parameters of the dataset. Some of these parameters, such as median, lower quartile, upper quartile and outliers, can easily be represented by constructing a boxplot for continuous univariate data by using Tukey's method. Tukey's test was chosen as it is distribution independent. The rules of the method are as follow:

1. The Inter Quartile Range (IQR) is the difference between the upper quartile (Q3) and the lower quartile (Q1);
2. Inner fences are fixed at a distance of 1.5 times IQR below Q1 and above Q3. They are given by:

$$[\text{Q1} - (1.5 \times \text{IQR}), \text{Q3} + (1.5 \times \text{IQR})]; \quad (6)$$

3. Outer fences are fixed at a distance of 3 times IQR below Q1 and above Q3. They are given by:

$$[\text{Q1} - (3 \times \text{IQR}), \text{Q3} + (3 \times \text{IQR})]; \quad (7)$$

Any data point lying between these fences is a possible outlier. Any data point beyond the outer fence is a probable outlier. For the detection of anomalous flights, inner fences are considered.

Figure 8 shows a group of 103 flights from the same tail number arriving at the same airport. LOF scores are plotted on the x-axis, and flight count is plotted on the y-axis. Out of 103 flights, 63 flights have a LOF score greater than 1. Analyzing all these 63 flights is very time-consuming. As per Tukey's rule of the inner fence (Equation (6)), the threshold comes to a LOF value of 1.1690. There are now only five flights that are greater than this threshold. These are the real outliers.

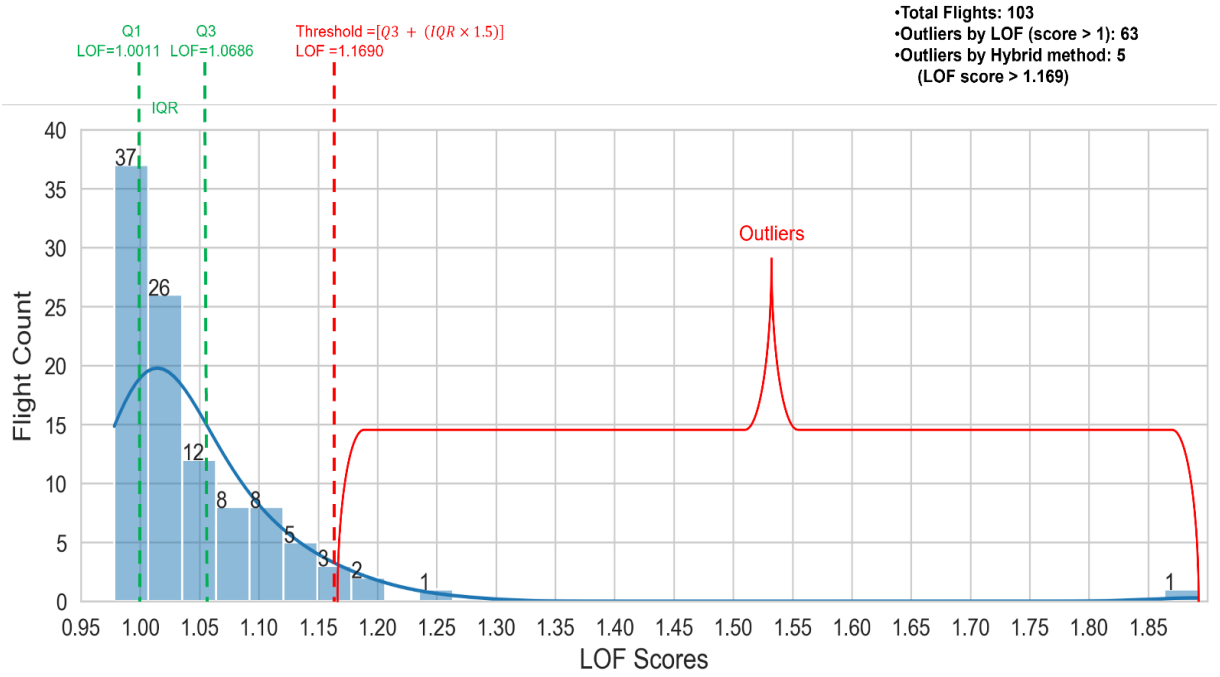

Figure 8. Fixing LOF threshold using Tukey's method.

While validating these results, the human expert (experienced pilot) found that four out of these five outliers were true positives. One flight was found to be safe from the parameters related to aviation, but it was categorized as anomalous as it followed a very short approach, which may be a decision taken by Air Traffic Control (ATC). Since it behaved abnormally as compared to another set of normal flights and had a very high LOF score of 1.87, therefore it was considered to be an outlier. Therefore, there was just one false positive in this case. Furthermore, the remaining 58 flights with a LOF score greater than 1 were examined to detect any false negatives, as these flights were labeled normal by the

hybrid method. Initially, the top 10 flights based on LOF score were picked from these 58 flights. Each flight was examined to find if the flight was anomalous or not. None of these 10 flights were found to be false negative. Since the remaining 48 flights have even lower LOF scores, therefore, they were also assumed to be normal.

Figure 9 shows the threshold analysis of three different arrival airports. For each airport, flights from the same tail number were taken, and then after calculating LOF scores, the threshold was fixed, as explained above. For each case, a threshold of LOF score was established as per Tukey's rule. Figure 9 shows that for each airport, the threshold is dynamic and stabilizes after a certain value. Thus, Tukey's rule helped to eliminate these 58 flights with LOF scores greater than 1 and highlighted real anomalous flights.

After establishing the threshold for our flight data, the following section shows how additional information may be inferred.

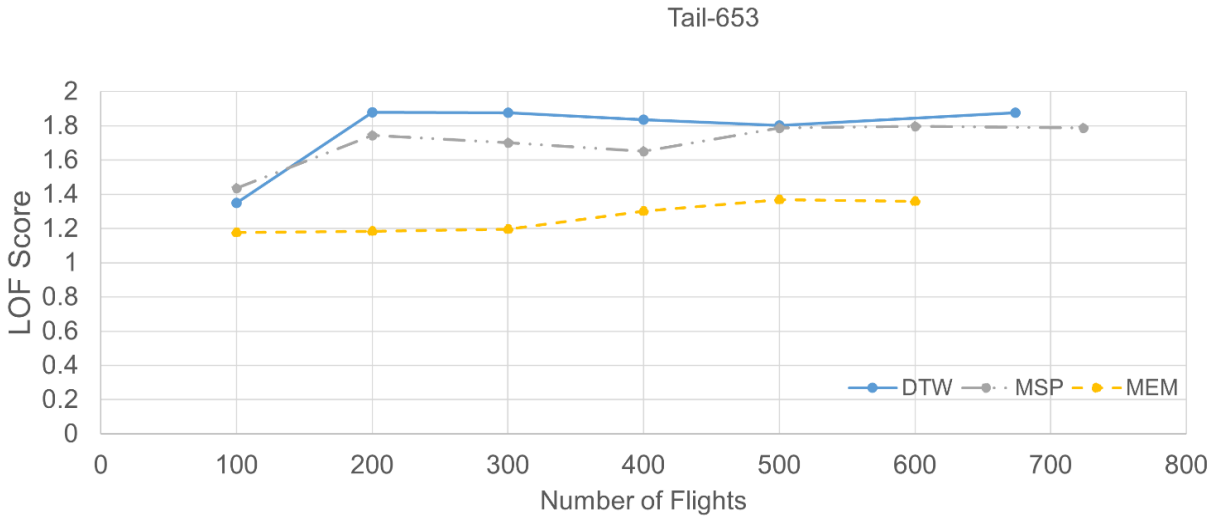

Figure 9. LOF score analysis for three airports.

3.2.7. Post Processing

In this step, further plots were plotted to obtain more insight into the flights. These plots included LOF scores and flight parameters at each time stamp during the flight. Furthermore, flight parameters responsible for anomalous behavior were identified and plotted. In order to find parameters responsible for the anomalous behavior of any anomalous flight, the mean of all flight parameters for all normal flights was calculated. Then for each flight parameter of the anomalous flight, the value of each flight parameter was subtracted from the mean value of that parameter of all normal flights. This was performed to obtain the flight parameter that was furthest away from the mean value of the same flight parameter of normal flights. In this way, top n flight parameters that may be responsible for the anomalous behavior can be found, which are further away from the mean values of those n parameters of normal flights. These plots are discussed in detail in the following section.

4. Results and Discussion

A set of flights with the same tail number was analyzed for three different airports, and the LOF score of each set is plotted in Figure 10. From Figure 10, the LOF score of each flight in the set can be seen for three different airports, and highly abnormal flights can be seen from the bar chart. Figure 11 combines the LOF algorithm and Tukey's method. It shows how for each of the three different airports, the LOF threshold is different and dynamic. This plot also shows the median LOF values for each airport and the anomalous flights as outliers for each airport. Detroit (DTW) airport is taken as a case study. From Figures 8 and 11, we can see that DTW has five flights as an outlier. One flight with a LOF score of 1.87 is an extreme outlier. Figure 12 shows the top 10 probable flight parameters responsible for the anomalous behavior. In this plot, flight duration is plotted on the x-axis,

and abnormal flight parameters are plotted on the *y*-axis in the order of how far they were from normal flight parameters. This plot gives a starting point for any investigation or to know what might have caused a flight to be abnormal. From discussions with the human expert, it was concluded that this flight was highly anomalous may be due to flight parameter ABRK (air brakes). Frequent use of air brakes to slow down the aircraft just before touchdown may have caused this anomalous behavior. Similar analyses were performed for the remaining four anomalous flights, and many flight parameters such as engine vibrations, high speed and pitch were found to be some of the contributing factors behind the anomalous behavior.

Figure 13 shows changing LOF value at each time stamp during the last three minutes of the landing phase. This figure shows how the LOF values change in time as the flight is about to touchdown. The figure shows that the flight became anomalous just 20 s before the touchdown. This plot for any flight can be used to find the time window when this flight was anomalous and to what extent the flight was anomalous. We found that many flights were anomalous during the approach and landing phase, but these flights became normal while touching down. Therefore, any normal flights can also be investigated to check for any instances where these flights might have become anomalous.

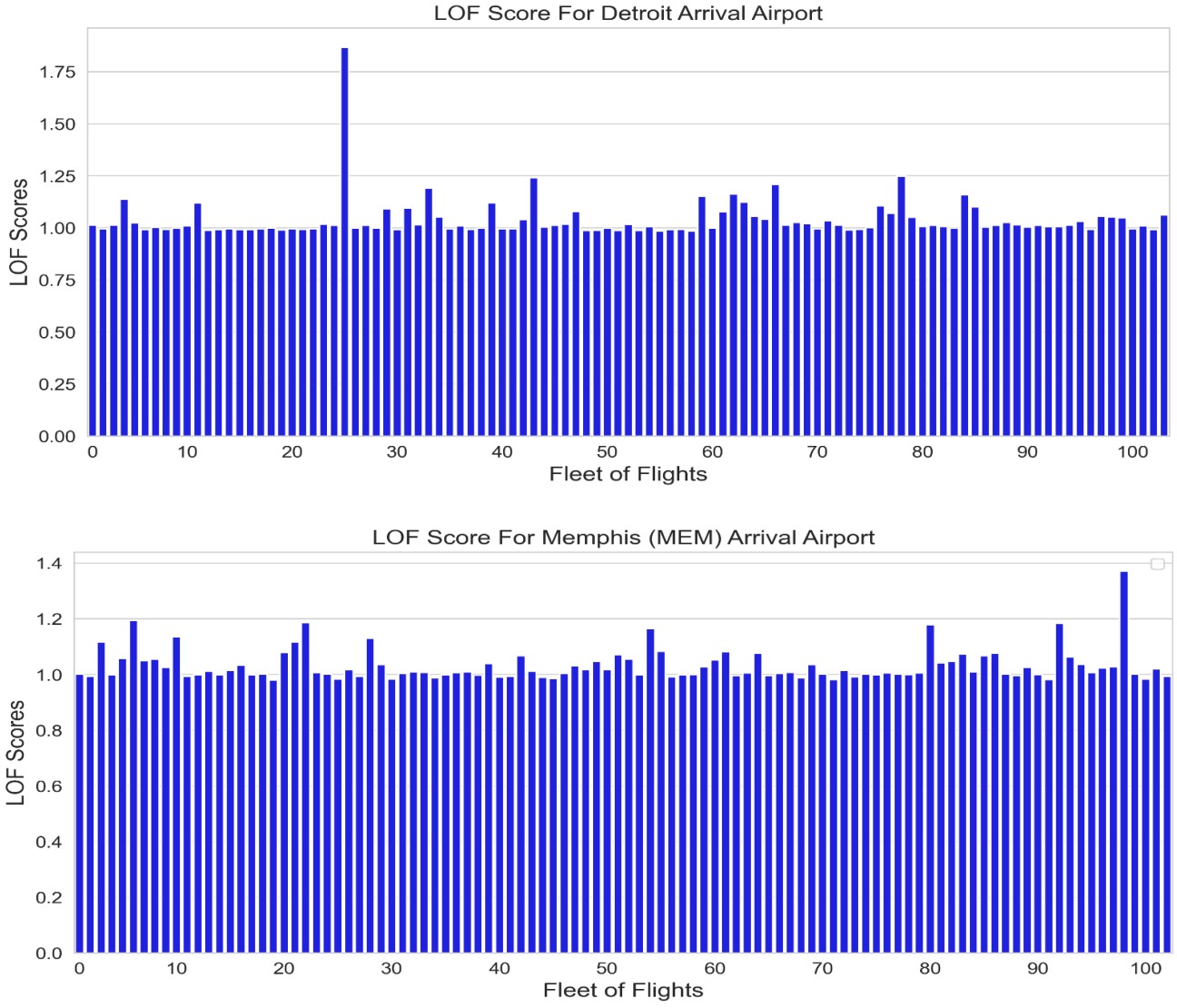

Figure 10. *Cont.*

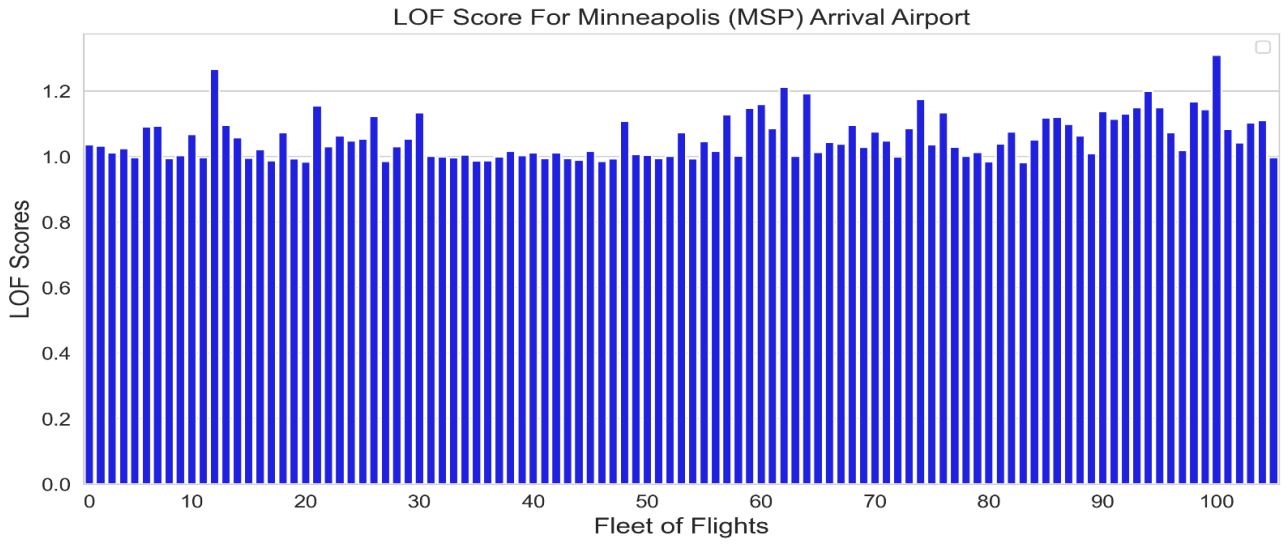

Figure 10. LOF analysis for three different airports.

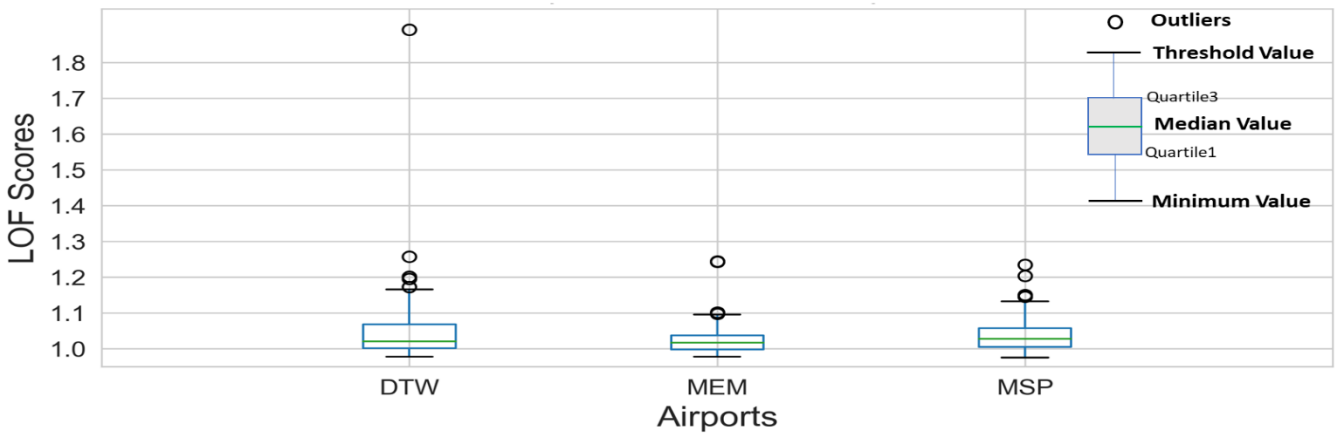

Figure 11. Boxplots for 3 different airports showing outliers using hybrid LOF–Tukey method.

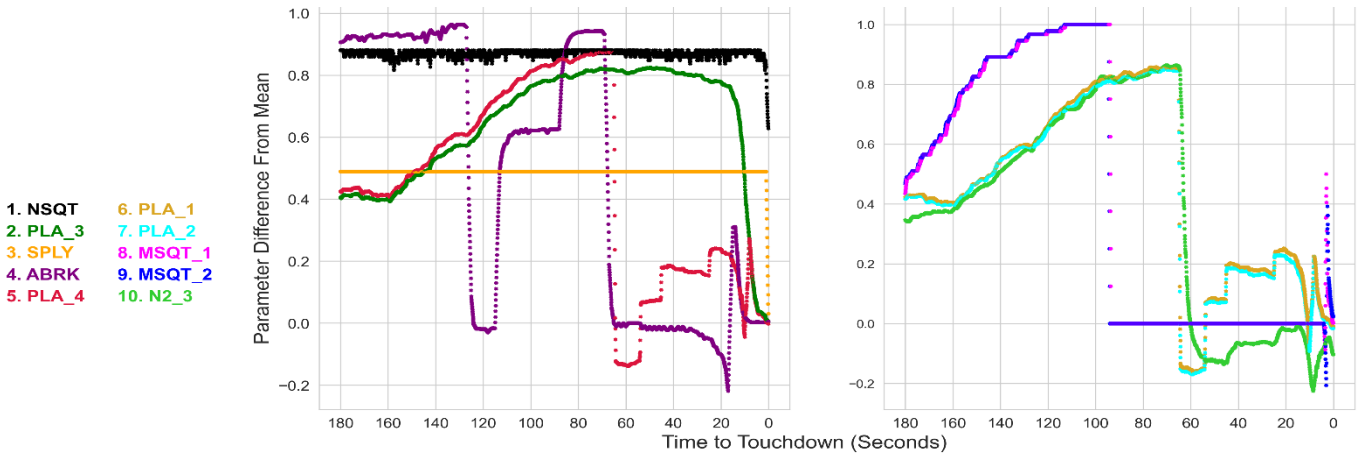

Figure 12. Top 10 flight parameters responsible for an anomalous flight during last 3 min before touchdown.

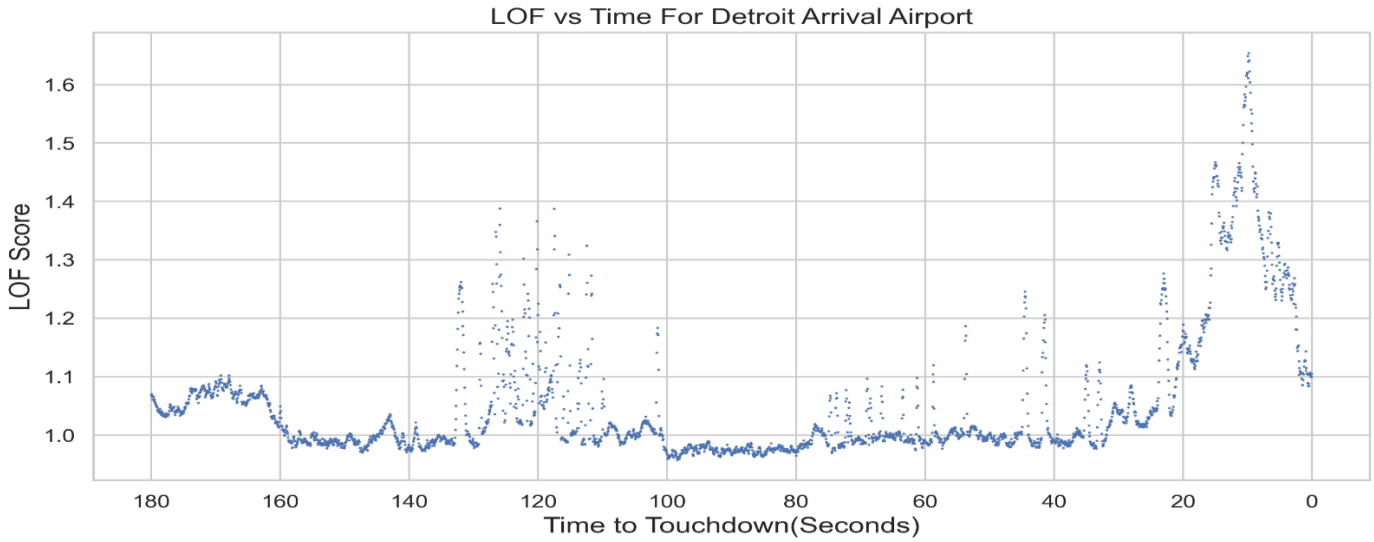

Figure 13. Changing LOF score of a flight during last 3 min before touchdown.

These results were verified and validated by the industry expert. Flights labeled anomalous for each airport were also anomalous, as per the human expert. Tukey's method reduced the number of false positives given by the LOF algorithm. All the points exceeded Tukey's threshold, but one was found to be anomalous and exerting an unstabilized approach, which is of safety concern. The one flight which was established to be anomalous but was later found to be a false positive was due to the aircraft taking a completely different approach route from the rest of the dataset under investigation. Therefore, while still anomalous, this was not found to be of any safety concerns.

5. Conclusions

This paper provides a hybrid statistical—local outlier factor for anomaly detection for flight data analysis by combining the local outlier factor technique with Tukey's method of anomaly detection

The paper shows that LOF quantifies the degree of an outlier by giving each flight a score. This is a development in machine learning techniques applied to flight data so far, which have produced a binary categorizing, as in the case of clustering algorithms. Thus, with LOF, for the first time, we demonstrated that in the aviation industry, anomalous flights can not only be identified but also be given an anomaly score to compare two anomalous flights in an unsupervised manner. Furthermore, LOF helps to track anomalous behavior in time during the flight and identifies the time window during the flight duration when a flight was anomalous. This is insightful when a flight is only abnormal for some seconds or a short duration.

For the first time, we detected flight parameters responsible for anomalous behavior or at least gave direction to human experts looking for the cause of abnormal behavior. By combining the LOF algorithm with Tukey's method, a dynamic threshold is fixed for the set of flights arriving at a given airport, which helps to identify real outliers and filters false positives given by the LOF algorithm. The paper also shows how the optimal value of k is decided for the implementation of the LOF algorithm. The analysis was conducted on real-life flight data in an unsupervised manner, as opposed to simulated data used in many works discussed earlier, thus giving an insight into the difficulties when applying such techniques to real data. The same methodology can be implemented on any time-series-based multidimensional dataset to detect anomalous data points.

Author Contributions: Conceptualization, S.K.J., G.V. and R.C.; data curation, A.M.; formal analysis, S.K.J., G.V. and R.C.; funding acquisition, R.C.; investigation, S.K.J., G.V. and R.C., methodology, S.K.J. and G.V.; project administration, R.C.; supervision, R.C. and G.V.; resources, R.C.; visualization, S.K.J.; validation, A.M.; writing—original draft preparation, S.K.J.; writing—review and editing, R.C. and G.V. All authors have read and agreed to the published version of the manuscript.

Funding: The findings presented in this paper are a result of the Project WAGE, financed by the Malta Council for Science and Technology, for and on behalf of the Foundation for Science and Technology, through the FUSION: R&I Technology Development Programme.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available in the public domain and can be accessed at https://c3.ndc.nasa.gov/dashlink/projects/85/ (accessed on 18 February 2018).

Acknowledgments: The authors would like to thank all reviewers and editors for their helpful suggestions for the improvement of this paper.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Avions de Transport Régional. *Flight Data Monitoring on Atr Aircraft 2016*; Avions de Transport Régional: Paris, France, 2016; p. 1.
2. Zhao, W.; Li, L.; Alam, S.; Wang, Y. An Incremental Clustering Method For Anomaly Detection In Flight Data. *Transp. Res. Part C Emerg. Technol.* **2021**, *132*, 103406. [CrossRef]
3. Mazareanu, E. EBIT Margin of Airlines Worldwide 2010–2022 | Statista. Available online: https://www.statista.com/statistics/225856/ebit-margin-of-commercial-airlines-worldwide/ (accessed on 28 June 2022).
4. Smart, E. Detecting Abnormalities in Aircraft Flight Data and Ranking Their Impact on the Flight. Ph.D. Thesis, Institute of Industrial Research, University of Portsmouth, Portsmouth, UK, 2011.
5. Pelleg, D.; Moore, A. Active Learning for Anomaly and Rare Category Detection. In Proceedings of the Advances in Neural Information Processing Systems 17 (NIPS 2004), Vancouver, BC, Canada, 13–18 December 2004.
6. Srivastava, A.N. Discovering system health anomalies using data mining techniques. In Proceedings of the 2005 Joint Army Navy NASA Airforce Conference on Propulsion, Monterey, CA, USA, 5–8 December 2005.
7. Das, S.; Matthews, B.L.; Srivastava, A.N.; Oza, N.C. Multiple kernel learning for heterogeneous anomaly detection. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD, Washington, DC, USA, 25–28 July 2010. [CrossRef]
8. Li, L.; Das, S.; John Hansman, R.; Palacios, R.; Srivastava, A. Analysis of Flight Data Using Clustering Techniques for Detecting Abnormal Operations. *J. Aerosp. Inf. Syst.* **2015**, *12*, 587–598. [CrossRef]
9. Li, L.; Hansman, R.; Palacios, R.; Welsch, R. Anomaly detection via a Gaussian Mixture Model for flight operation & safety monitoring. *Transp. Res. Part C Emerg. Technol.* **2015**, *64*, 45–57.
10. Melnyk, I.; Banerjee, A.; Matthews, B.; Oza, N. Semi-Markov switching vector autoregressive model-based anomaly detection in aviation systems. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016.
11. Das, S.; Sarkar, S.; Ray, A.; Srivastava, A.; Simon, D. Anomaly detection in flight recorder data: A dynamic data-driven approach. In Proceedings of the 2013 American Control Conference, Washington, DC, USA, 17–19 June 2013. [CrossRef]
12. Bhaduri, K.; Matthews, B.L.; Giannella, C.R. Algorithms for speeding up distance-based outlier detection. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD, San Diego, CA, USA, 21–24 August 2011. [CrossRef]
13. Bay, S.; Schwabacher, M. Mining Distance-Based Outliers in Near Linear Time with Randomization and A Simple Pruning Rule. In Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD, Washington, DC, USA, 24–27 August 2003.
14. Matthews, B.; Das, S.; Bhaduri, K.; Das, K.; Martin, R.; Oza, N. Discovering Anomalous Aviation Safety Events Using Scalable Data Mining Algorithms. *J. Aerosp. Inf. Syst.* **2014**, *11*, 482. [CrossRef]
15. Oehling, J.; Barry, D. Using Machine Learning Methods in Airline Flight Data Monitoring To Generate New Operational Safety Knowledge From Existing Data. *Saf. Sci.* **2019**, *114*, 89–104. [CrossRef]
16. Megatroika, A.; Galinium, M.; Mahendra, A.; Ruseno, N. Aircraft anomaly detection using algorithmic model and data model trained on FOQA data. In Proceedings of the 2015 International Conference on Data and Software Engineering (Icodse), Yogyakarta, Indonesia, 25–26 November 2015. [CrossRef]

17. Nanduri, A.; Sherry, L. Anomaly detection in aircraft data using Recurrent Neural Networks (RNN). In Proceedings of the 2016 Integrated Communications Navigation and Surveillance (ICNS), Herndon, VA, USA, 19–21 April 2016. [CrossRef]
18. Breunig, M.; Kriegel, H.; Ng, R.; Sander, J. LOF: Identifying density-based local outliers. In Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data—SIGMOD, Dallas, TX, USA, 15–18 May 2000. [CrossRef]
19. DASHlink—Sample Flight Data. Available online: https://c3.nasa.gov/dashlink/projects/85/ (accessed on 18 February 2018).
20. Boeing. *Statistical Summary of Commercial Jet Airplane Accidents*; Boeing: Seattle, WA, USA, 2021; p. 14.
21. Airbus. *A Statistical Analysis of Commercial Aviation Accidents 1958–2021*; Airbus: Blagnac, France, 2022; p. 27.
22. Aggarwal, C.; Hinneburg, A.; Keim, D. On The Surprising Behavior of Distance Metrics in High Dimensional Space. In *Database Theory—ICDT*; Van den Bussche, J., Vianu, V., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2001; Volume 1973, pp. 420–434. [CrossRef]