Aircraft Anomaly Detection

Benchmarking unsupervised machine learning algorithms for anomaly detection

Word count: 4, 026

Abstract

The United States Navy uses outdated techniques, such as the Magnetic Anomaly Detection (MAD) and the Consolidated Afloat Networks and Enterprise Services (CANES) system to detect anomalous aircraft flight behavior. Both these methods used by the Navy rely on a checklist organization and are extremely time consuming and tedious. With the recent crashes of the Ethiopian Airlines Flight 302 and the Lion Air Flight 610, which were both Boeing 737 Max 8 aircrafts (a manufacturer used by the Navy), the need for more accurate and real-time anomaly detection is even more crucial. Timely and accurate flight pattern anomaly detection can help reduce terrorist attacks using airplanes and prevent incidents like 9/11 from reoccurring. Additionally, this real time anomaly detection can also discourage surveillance activities through spy crafts or drones that may use aircraft masking or excessive autopilot mode. This study proposes that implementing machine learning (ML) and pattern recognition techniques can take these traditional capabilities one step further and detect anomalous behavior and isolate outliers in flight patterns in real time by analyzing huge volumes of data. A research dataset from the ADS-B Exchange (the world's largest co-op of unfiltered data comprised of 10,000 aircrafts) was prepared for this study using the Gaussian model by eliminating extreme outliers or anomalies for more reliable results. Subsequently, the dataset was divided into groups based on country, aircraft type, and engine type. Feature pairs were identified from the aircraft data and analyzed using three machine learning algorithms. These included K-Means Clustering (ii) Linear Regression, and (iii) Density-Based Spatial Clustering of Applications with Noise (DBSCAN), all of which are unsupervised ML algorithms that are particularly suitable to evaluate and provide structure to clustered aircraft data. The study identified certain aircraft

flight patterns as anomalous and also revealed the importance of classifying the nature of the anomalies as “deep” or “normal” to help define the threshold for alarm based on the nature of the violation. The study concluded that real-time anomaly detection could be automated and verified against protocols and anomalies that were the result of approved stealth and espionage missions need not be deemed serious. The study proved that machine learning techniques would be extremely effective in identifying accurately potential safety and security threats through anomaly detection in real time.

Introduction

The “mission” of the United States Navy is “to maintain, train, and equip combat-ready naval forces capable of winning wars, deterring aggression, and maintaining freedom of the seas” (Surface Warfare Magazine, vol. 56, 2017). To conduct such operations, the Navy utilizes aircraft carriers, fighter jets, and planes like the Grumman F6F Hellcat. The most common technology used for anomaly detection in naval aircrafts, such as the Grumman F6F Hellcat is Magnetic Anomaly Detection (MAD). MADs reduce disturbances from the electrical or metal equipment on an aircraft (including the magnetic field around an aircraft) and have been traditionally incorporated into aircraft detection systems to detect foreign submarines, aircrafts, and other vehicles (Fundamentals of Naval Weapons Systems, Chapter 9, 2010). Furthermore, MADs have been used in geomagnetic and aeromagnetic surveying instruments to search for minerals in various terrains. Recently, the Navy has experimented with implementing autonomous detection systems, such as the Consolidated Afloat Networks and Enterprise Services (CANES), which identify anomalies or disturbances with its “cyber-hardened, flexible architecture” (Northrop Grumman “CANES”, 2015). The CANES system of anomaly detection

is currently used on ships, or aircraft carriers and creates a map of the flight data based on feature dependency and uses the map to detect anomalous behavior by checking every feature on a list to provide a secure "common computing environment" (Thie, Harrell, Jenkins, 2009).

However, both of these technology initiatives used by the United States Navy involve somewhat time-consuming and tedious programming techniques for anomaly detection that are based on a "checklist" method. Additionally, identification of aircrafts can be challenging because "aircrafts move quickly", may not provide "cooperative identification", or may "not have Identify Friend or Foe (IFF) systems installed or malfunctioning". While "commercial aircrafts may be easier to identify because of expected routes and altitudes, nowadays they are often selected dynamically and opt for fuel saving options" (Rowe, Das, Zhou, 2018). The key problem with identifying combat or unfriendly aircrafts is the availability of excessive volumes of unreliable data. Therefore, in order for the U.S. Navy to become a more defensive militia, these outdated hard-coded programming techniques must be improved with machine learning techniques.

Literature Review

According to Daveed Gartenstein-Ross (2018), terrorists are going to increasingly use machine learning (ML) and artificial intelligence (AI) to subvert our defense systems. Thus it is imperative to recognize the relevance of machine learning for anomaly detection, in order to achieve a higher level of safety and security.

What is machine learning and why use it for anomaly detection?

Anomalous features within aircraft flight data can be isolated in real time within huge volumes of data through ML techniques that can search for dependent features and recognize patterns to help evaluate anomalies and identify associated threat levels. However, since machine learning has become a buzzword in today's world, it is important for this study to define ML in the context of this research.

Machine learning can be "broadly defined as computational methods (that uses) experience to make accurate predictions, which typically take the form of electronic data collected and made available for analysis in the form of digitized human-labeled training sets" (Mohri, Mehryar. 2012).

The aviation industry has increasingly turned to adopting ML techniques that "can compare flight data parameters from a large number of flights and identify new or unknown patterns (and) these patterns may show abnormal or inconsistent behavior with respect to most of the flights" (Jasra, Sameer, Gauci, Jason, et al. 2018). According to Dr. Jasra's team, "the outliers are of interest and require further investigation". An example of "an abnormal flight pattern is during the descent phase where the aircraft is not following the standard procedure for a stable approach (and) the landing gear is not down by 1000 feet of altitude" (Jasra, Sameer, Gaucci, Jason, et al. 2018). Though in this study, the researchers explored how the aviation industry can conserve oil usage using machine learning, the same principles for anomaly detection can be utilized for anomaly detection and tracking threats. Thus, it is necessary to further explore the utility of machine learning techniques for anomaly detection to identify potential security threats for the purpose of our national defense.

Which machine learning algorithms are used in the study and why?

The purpose of the research was to identify patterns for outliers within the datasets. Therefore, supervised machine learning techniques were not used because datasets with outliers usually appear skewed and outliers or anomalies occur at a much lower rate compared to regular datasets. Also, clustered data like aircraft flight patterns usually responds best with unsupervised algorithms as the algorithm provides structure to the data and need not be trained.

Thus, the following three unsupervised machine learning algorithms were created and tested for this study: (i) K-Means Clustering (ii) Linear Regression, and (iii) Density-Based Spatial Clustering of Applications with Noise (DBSCAN).

K-Means Clustering clusters data points based on similarities and makes it easier to identify if an aircraft is not following the flight plan that its unit is following, even if the unit is categorized under "special circumstances". Essentially, "K-means stores *k* centroids that it uses to define clusters. A point is considered to be in a particular cluster if it is close in distance to the clusters centroid, or center of mass." (Piech, Chris, Ng, Andrew. 2013). To do so, "K-means finds the best centroid by assigning data points to clusters based on the current centroids or choosing centroids based on the current assignment of data points to clusters" (Piech, Chris, Ng, Andrew. 2013). In both cases, K-Means clustering is an efficient machine learning algorithm for grouping together similar data points, which is effective since the majority of aircrafts follow one set of rules unless the aircraft is anomalous.

Linear Regression plots data points along a line of "best fit" based on what the model predicts is an "ideal" correlation between a feature pair. Linear Regression is helpful for identifying whether a datapoint follows government policies because it can predict where the

data points should be if the majority of data points are following the restrictions. These data points are used to create the line of best fit. However, if a group of aircrafts are not following government policies, due to special circumstances, this algorithm would identify the whole unit as anomalous. Due to this, it was necessary to meticulously test the outliers identified using using the other algorithms as well.

Finally, **DBSCAN** identifies similarities in data points by grouping together similar aircrafts into higher density regions and outlier data points into low density regions. DBSCAN is helpful because unlike K-Means, this model uses specific parameter constraints to identify outliers. Thus, inputting the constraints that each government policy has into the models makes it significantly easier to find a "deep" anomaly. In the paper "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", the authors discusses how "DBSCAN, or density-based spatial clustering application with noise, requires only one input parameter and supports the user in determining an appropriate value." (Ester, Martin, Kriegel, Hans-Peter, et al. 1996).

The next step was to find a valid repository that contained raw, unfiltered, open-source, free faircraft data and after researching several potential sources, the pilot and experimental datasets were taken from the open source ADS-B Exchange dataset.

What is the ADS-B Exchange Dataset and why it is used?

The ADS-B Exchange is a free repository of aircraft flight data for approximately 10,000 aircrafts (each with 67 features), ranging from transport vehicles to commercial planes and fighter jets. The Automatic Dependent Surveillance-Broadcast (ADS-B) Exchange is the

"world's largest co-op of unfiltered flight data" that serves "the flight tracking enthusiast." It contains a significant number of features (approximately 67 features) that is associated with various aircrafts and is a "reliable source for tracking" flight patterns (Ryan Salcido, Anthony Kendall, Ying Zhao. 2017). It is open-source aircraft flight data that is contributed by the community through ADS-B receivers available from Amazon at a cost of 100 dollars. Additionally, ADS-B data is used extensively in developing decision support systems and providing Business Intelligence (BI) to assist airport management. ADS-B does not include data that is manually entered by the pilot, but only relies on reliable features and data points. The features in the ADS-B exchange dataset identified characteristics like flight patterns, server usage, and ICAO registration numbers for each aircraft that can increase situational awareness for decision makers and help identify critical airborne objects as friendly, hostile, or neutral with high precision.

Method

The methodology used in this study was quantitative data analysis. ADS-B Exchange aircraft data was downloaded as JSON files from the months of August and September, 2017 and 2018. The data amounted to approximately 16.8 GB and represented approximately 10,000 aircrafts ranging from transport vehicles to commercial planes and fighter jets. The JSON files were converted to CSV files and the aircraft features that identified characteristics like flight patterns, server usage, and ICAO registration numbers for each aircraft were organized into columns in a spreadsheet as part of the cleaning process to make the data more readable.

The raw data using the older datasets was filtered to identify preliminary patterns by fitting each feature on a normal distribution. The data points outside the two sigma were

discarded using the Gaussian model, as they were likely outliers resulting from data collection errors or other special circumstances. This data scrubbing or the process of detecting corrupt or inaccurate records was necessary to ensure that the research dataset was more meaningful and accurate for the purpose of the research. The final research dataset was a collection of data representing 9,500 aircrafts.

This research dataset was further categorized for testing by (i) country, (ii) aircraft type, and (iii) engine type. Interestingly, the majority of the data available on the ADS-B Exchange website came from three countries: (i) United States (2,500 planes) (ii) Turkey (3,200 planes), and (iii) England (3,600 planes). Other countries in the raw data were disregarded because I needed a sample size that was adequately representative of each country.

For each country, the data was further categorized by engine type and aircraft type. The engines were classified as follows: (i) Jet engine: Spins around to suck air into the engine as the plane flies through (ii) Turbo jet engine: Uses the mechanical energy from a turbine or a fan to push air rearwards. The aircraft types were classified as follows: (i) Fighter aircraft (military/training aircrafts) (ii) Commercial planes (transporting passengers, troops, supplies) (iii) Small, private aircrafts. The following diagram indicates the six research datasets that I used in the study:

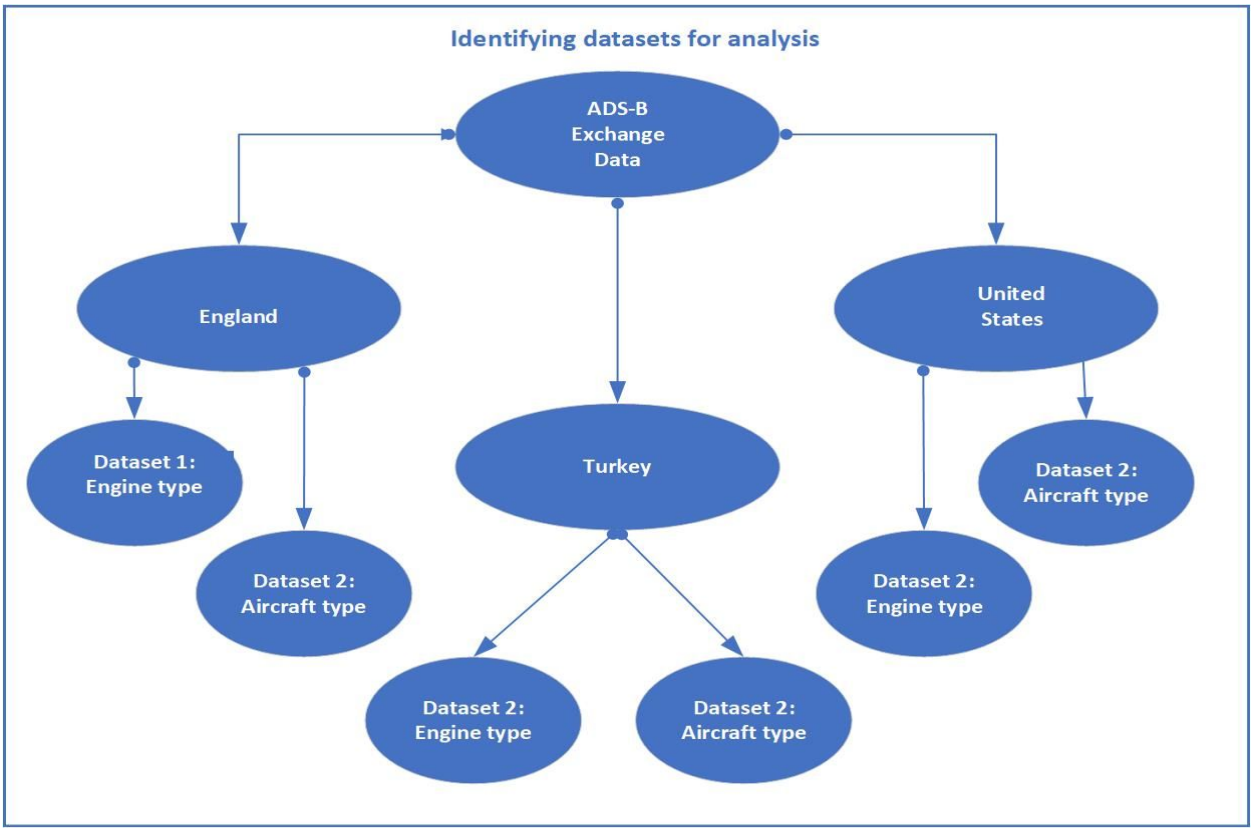

Figure 1: Research datasets used in the study

Using my pilot experiment, I identified three feature pairs that correctly fit the expected patterns indicated on the website, and used these pairs to test the machine learning algorithms that I wanted to use. The three feature pairs were as follows: (i) position vs time of each aircraft (ii) the amount of time each aircraft was tracked for vs the number of messages sent to Air Traffic Control (iii) speed vs altitude. The first two feature pairs helped to identify any potential aircrafts that were suspected to be foreign spy crafts, as well as pilots who used excessive autopilot mode, which can be a safety concern. However, the third feature pair was used merely as a foundation since most previous studies that were investigated tested speed vs altitude as well.

Machine learning algorithms like K-Means Clustering, Linear Regression, and DBSCAN (Density based Spatial Clustering Applications with Noise) were used for attribute analysis on the three-selected aircraft feature pairs within the ADS-B Exchange dataset. Data was constantly fed into the algorithm, forcing the algorithm to make a model based on the patterns that were detected. The dataset was cleaned through a normal distribution model to take out the extreme outliers that may have have resulted from errors in the data collection. Balancing the cases of "right" versus "wrong" was essential for using unsupervised learning because like any model, having a disproportionate amount of "right" or "wrong" cases could make it difficult for a model to detect an accurate pattern.

However, before reaching to conclusions, a CSV file, or spreadsheet, containing the original dataset was used to ensure that the respective algorithms did not unintentionally alter the data or find incorrect patterns. Once each algorithm was validated by the reference spreadsheet, each outlier detected by the three algorithms were investigated by contacting the ADS-B Exchange and online resources. This validation was done to ensure that there were no special circumstances that would invalidate the anomalous behavior of the outlier.

Results

The study proved that machine learning algorithms were useful in identifying four aircrafts whose flight patterns were anomalous. It helped to classify the nature of the anomalies as "deep" or "normal", which helped to define the seriousness of the violation. The study concluded that real-time anomaly detection could be automated and verified against protocols. Additionally, anomalies that were the result of approved stealth and espionage missions were not

to be considered serious. In short, machine learning techniques were proved to be extremely effective in accurately identifying potential safety and security threats through anomaly detection in real time.

The following table outlines each significant anomaly found in the aircraft data using the three machine learning algorithms.

Feature Pairs	Normal Anomaly	Deep Anomaly
Position of aircraft **vs** Time during which aircraft was at that position	**(i) Aircraft manufacturer:** Piper **Country:** United States **Engine:** P28A (Turbo engine, used for flight training) **Unit Group:** Part of a unit of three aircrafts	**(iv) Aircraft manufacturer:** Boeing **Country:** United States **Engine:** B737 (Jet engine) **Unit Group:** Part of a unit of five aircrafts
Amount of time aircraft was tracked for **vs** Number of messages the aircraft sent to Air Traffic Control (ATC)	**(ii) Aircraft manufacturer:** Airbus **Country:** Turkey **Engine:** A319 (Jet engine) **Unit Group:** Solo commercial airplane **(iii) Aircraft manufacturer:** Piper **Country:** England **Engine:** M600 (turbo engine) **Unit Group:** Solo private aircraft (4 seater)	No anomalies found
Speed **vs** Altitude	No anomalies found	No anomalies found

Table 1: Deep and normal anomaly identification for feature pairs on aircraft data

Three aircrafts displayed normal anomalies, while one aircraft flight pattern displayed deep anomaly. The Piper aircraft from the United States with a P28A Turbo engine that was flying as part of a unit of three airplanes was found anomalous using the first feature pair, **Position vs Time**, against the **K-means clustering** algorithm. The flight patterns of the aircrafts flying within this unit were grouped together due to their relative similarity to each other in position and the flights occuring during the same time interval. However, this specific aircraft was plotted significantly far from the cluster that displayed the flight patterns for the other aircrafts within its unit. This indicated that the aircraft was not following the flight plan, yet was still within range of the government policy radius. As the entire unit was following the government regulations and a predetermined flight plan, the whole unit was not considered as anomalous.

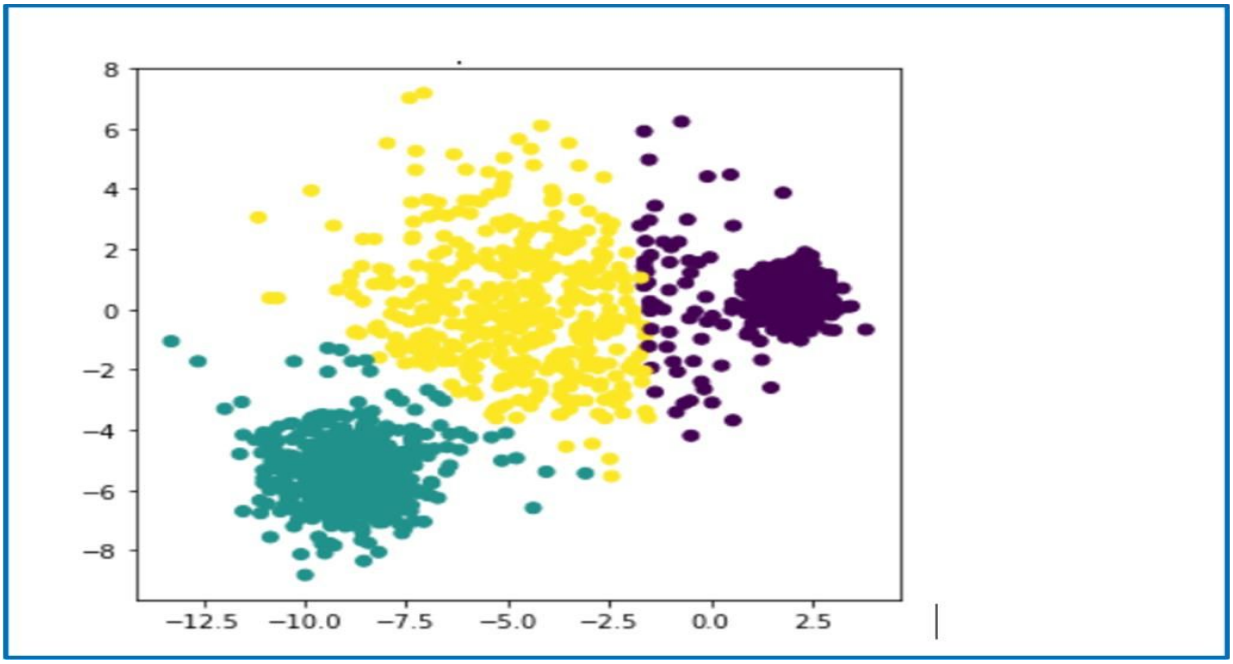

Figure 2: K-means displays the initial outliers indicated by the scattered dots in the plot

Using the **DBSCAN** algorithm also proved that only this aircraft was anomalous because its flight pattern was the only outlier in the entire dataset.

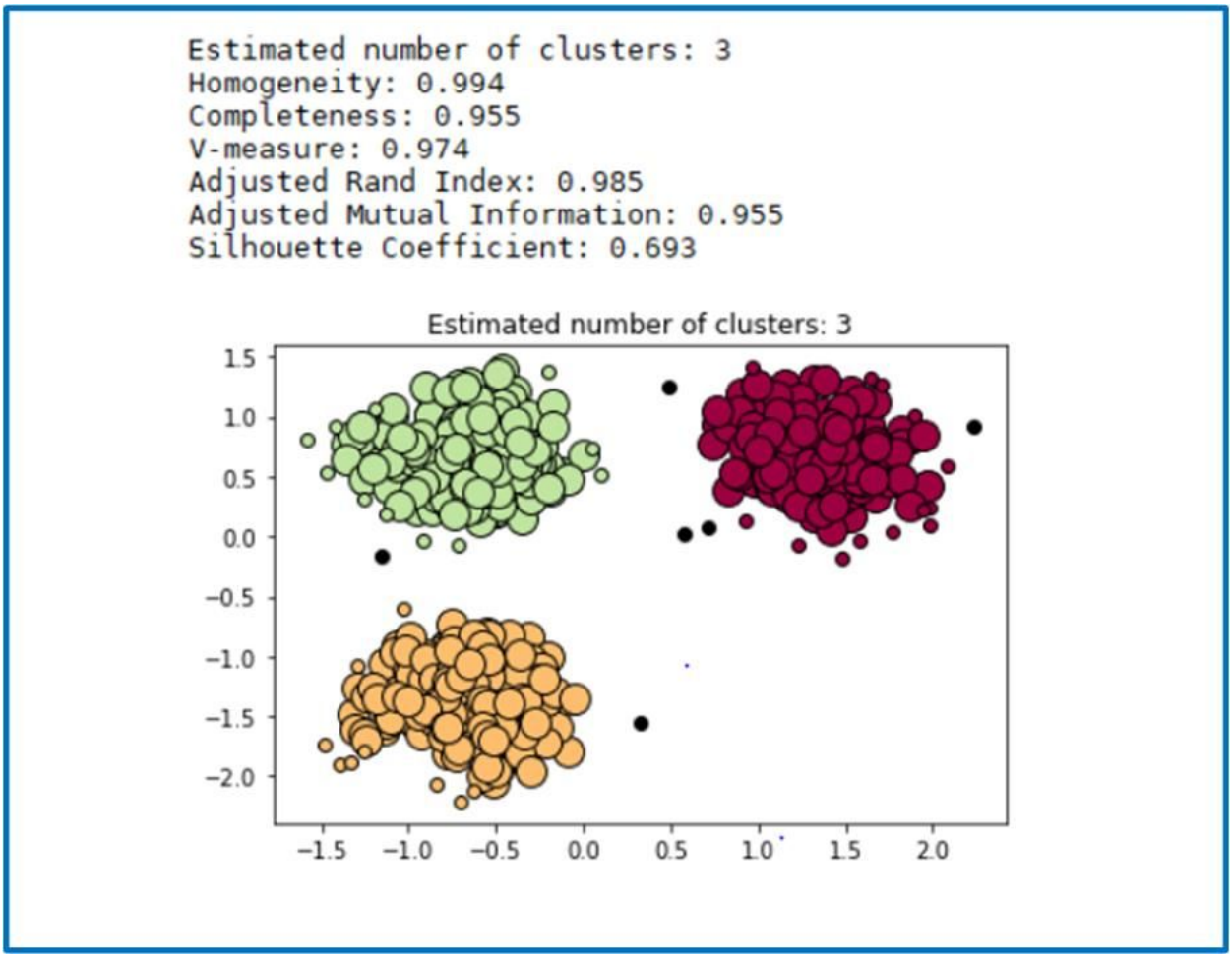

Figure 3: DBSCAN displays the outliers using the scattered dots

The **Linear Regression** algorithm also proved that the entire aircraft unit was following government regulations, whereas this particular aircraft was not following the flight plan. This indicated that the aircraft was an outlier and displayed “normal” anomaly.

The United States Boeing B737 Jet engine aircraft, which was a part of a unit of five aircrafts displayed “deep anomaly”. Using **K-means clustering** algorithm indicated the possibility of a special circumstance mission, due to the group clustering method. The clustering

on the plot showed that inside the non-compliant cluster, this aircraft was extremely anomalous. The **Linear Regression** algorithm created a correlation between aircrafts that followed similar flight patterns.

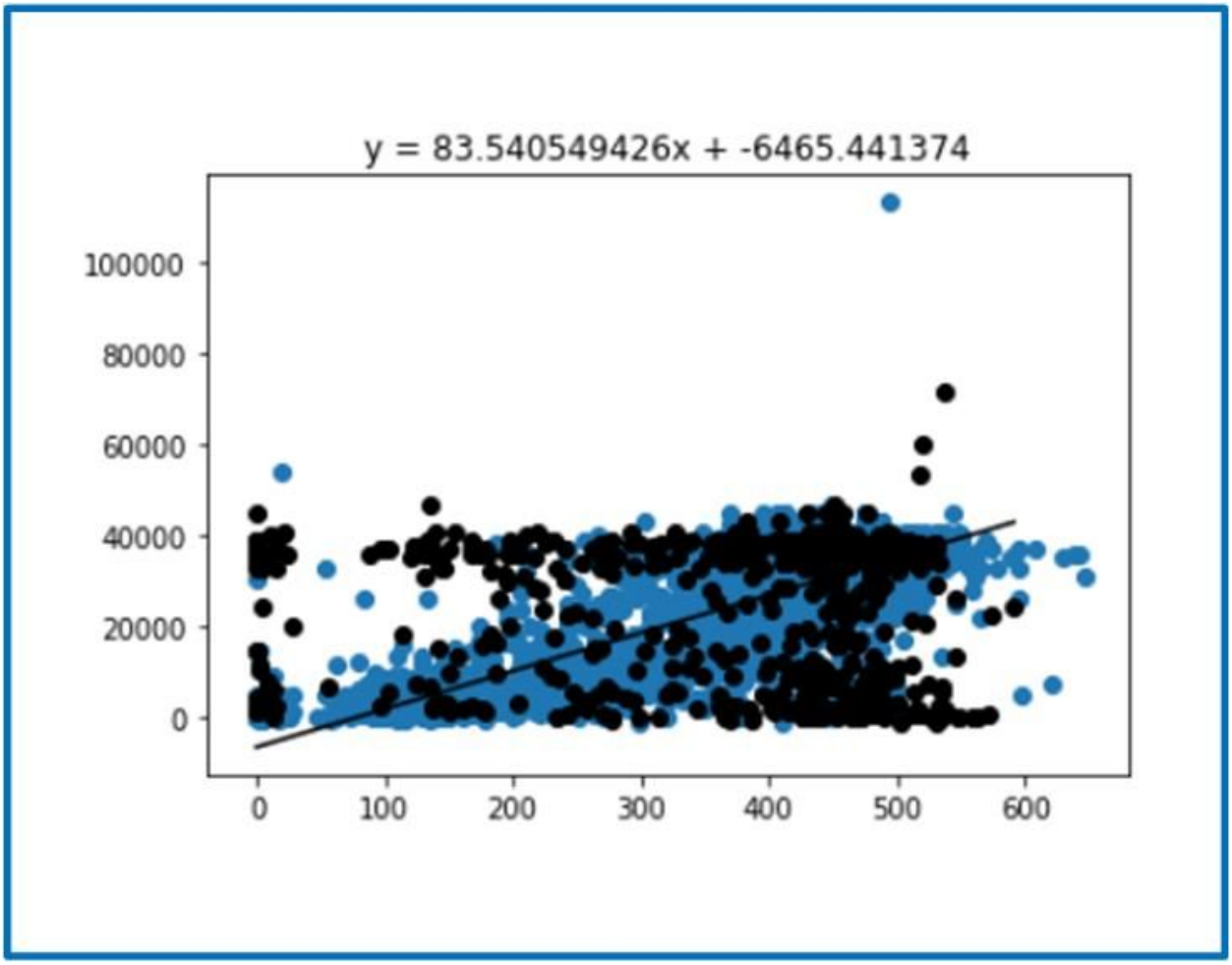

Figure 4: Linear Regression creates a correlation in the flight patterns

DBSCAN uses densities of overall flights to find anomalies and is unable to account for special circumstance flights. In this case, DBSCAN showed that the whole unit was an anomaly because it was a special circumstance mission.

The second feature pair **Time tracked vs Number of messages sent to Air Traffic Control (ATC)** displayed normal anomalies for two aircrafts. The Airbus A319 from Turkey

with the jet engine (a solo commercial airplane) and the Piper aircraft from England with the M600 Turbo engine (a four-seater solo private aircraft) were analyzed with **DBSCAN.** The aircrafts followed the minimum message requirements that government policy dictated, but were in an extremely low density region. This implied that the aircraft could have been under excessive autopilot usage, yet still relayed enough messages to fulfil the requirements of government policy and other aircrafts.

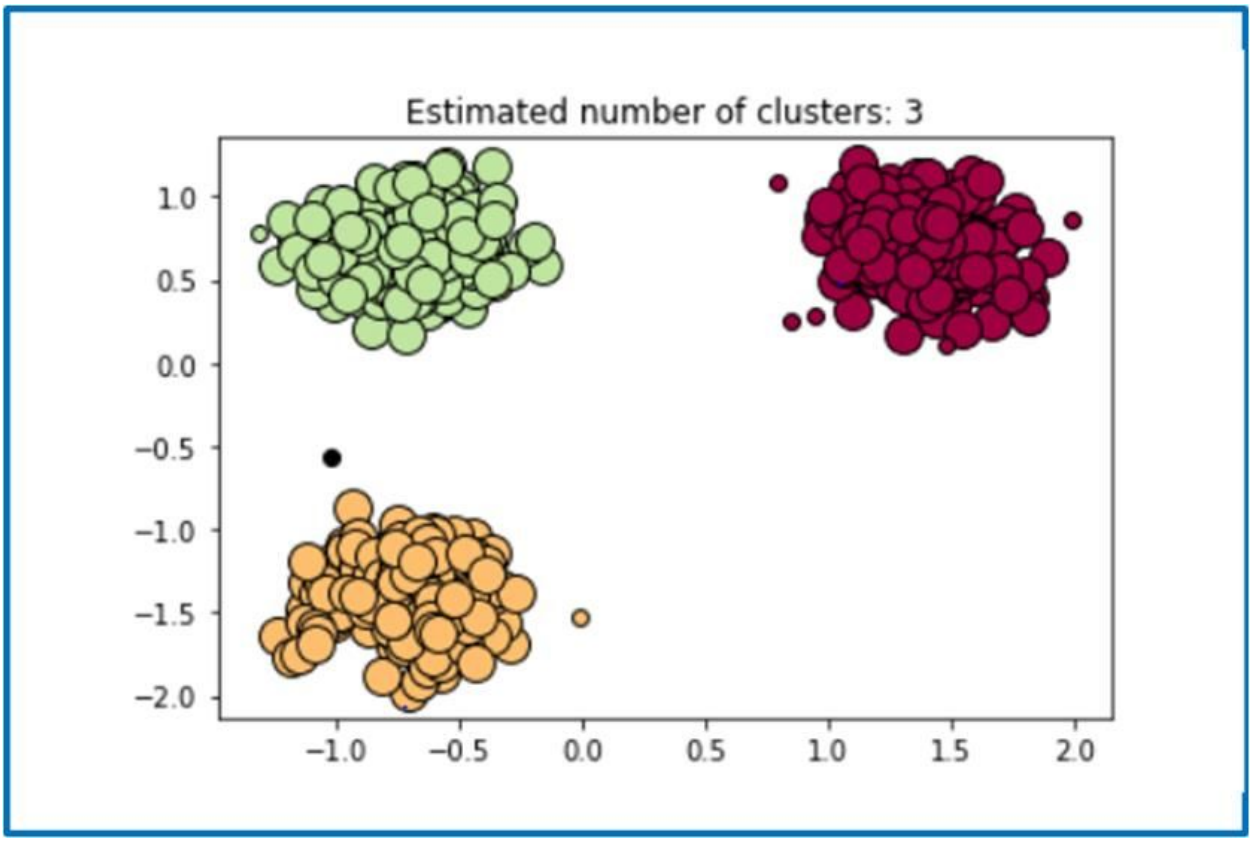

Figure 5: DBSCAN plot displaying the anomaly shown by the Airbus from Turkey

The **Linear Regression** algorithm indicated that the aircraft followed a general pattern that was estimated by the line of best fit and was created according to airtime codes. However, the point had the largest digression from the line, compared to other aircrafts in the experiment batch. As the aircrafts were solo aircrafts with an individual flight plan that followed government

policy, they could not be clustered with other aircrafts and created their own cluster. Even though the **K-Means Clustering** algorithm showed that there was an anomaly, it was obvious that the anomaly resulted not from the fact that it was deviating from its flight plan, but because there were no similar flights available for comparison. Solo flights differ in the amount of time they are in the air, compared to an overall dataset. Thus, clustering a solo flight would not be plausible due to the "uniqueness" of the flight.

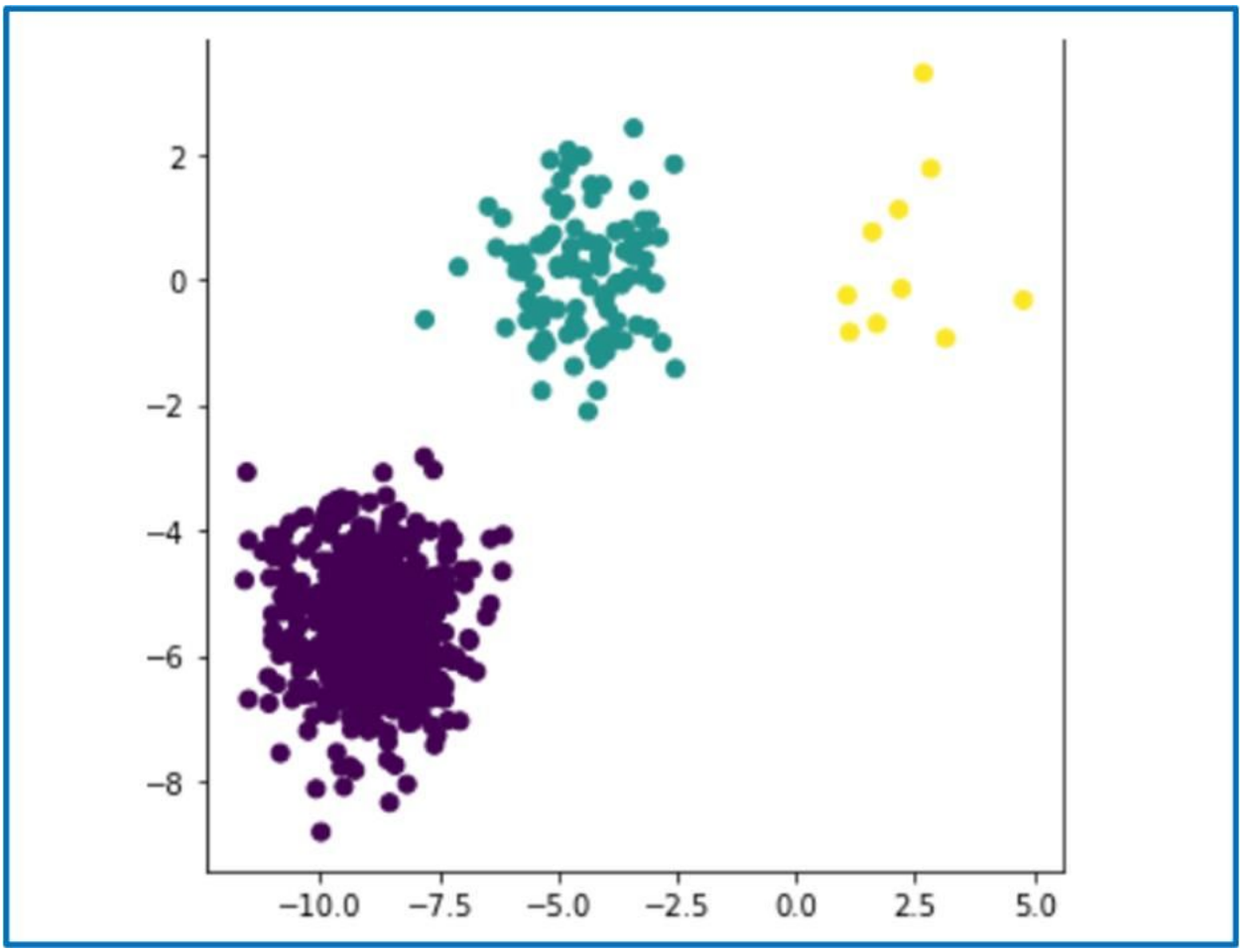

Figure 6: K-Means clustering showing initial outliers for the Airbus from Turkey

No deep anomalies were discovered for the second feature pair because no aircrafts had violated government policy regulations that were in place specifying the minimum number of

messages aircrafts had to send to ATC. All anomalies found were specific to designated flight plan violations.

When analyzing the third feature pair of **Speed vs Altitude,** no "normal" anomalies were detected. This was due to the fact that most weather conditions were likely predicted in advance when the flight plan was created. Yet, a few flight plans were updated during mid-flight, most likely due to unexpected or miscalculated weather disruptions, including turbulence from storms and so on. These points were not considered anomalous because the updated flight plans confirmed that action had to be taken, and the action was authorized.

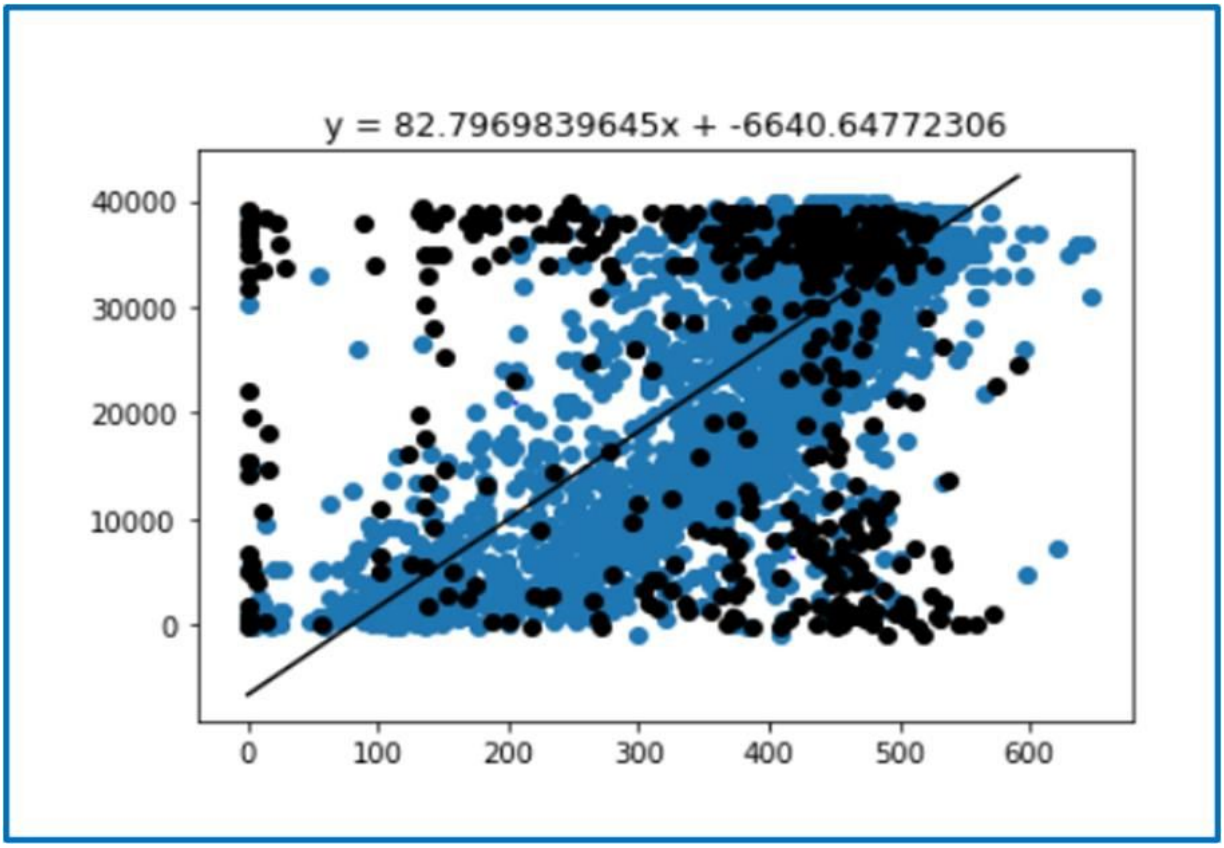

Figure 7: Linear regression showing no outliers for third feature pair

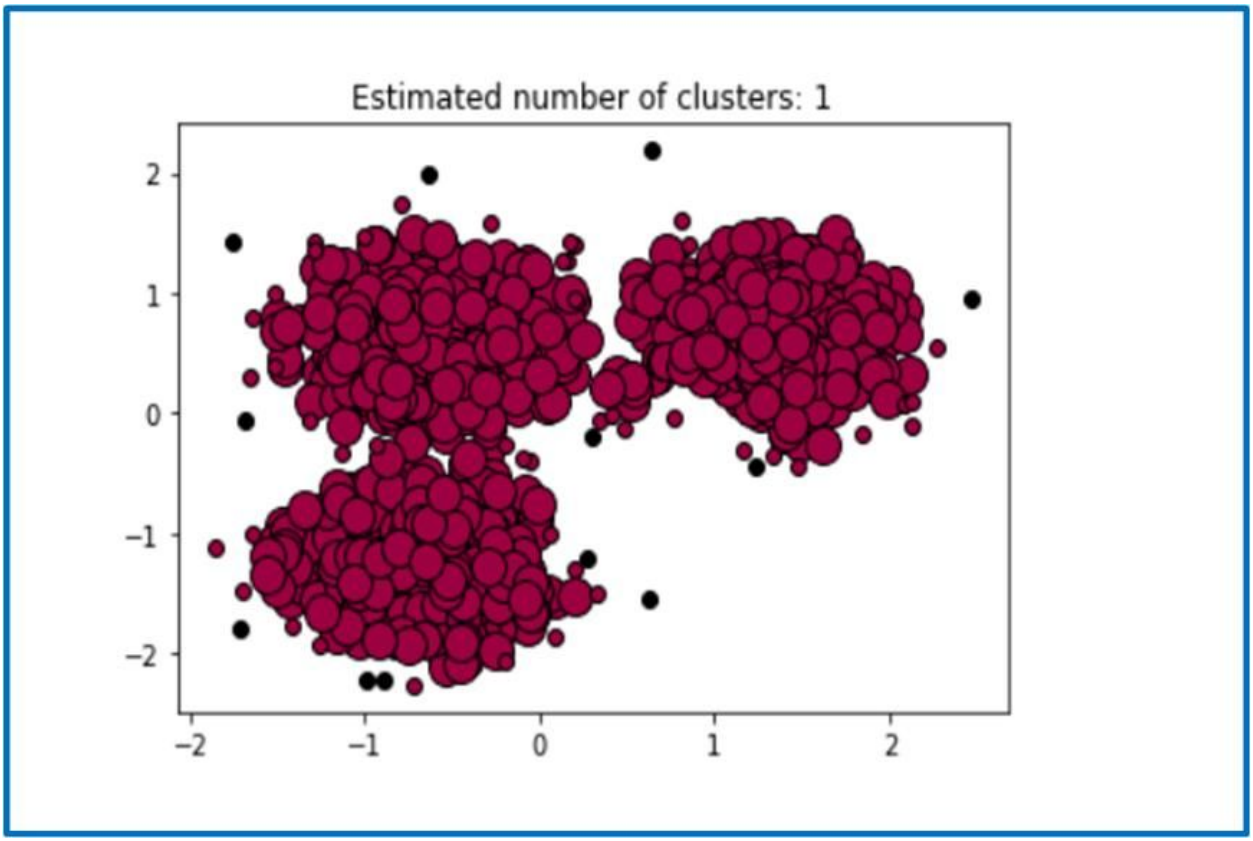

Figure 8: DBSCAN plot showing no outliers for the third feature pair

No deep anomalies were detected for the third feature pair for any of the aircrafts. All deviations from flight plans were within the regulations of government policy. Thus deviations were authorized and planned using government policies as constraints.

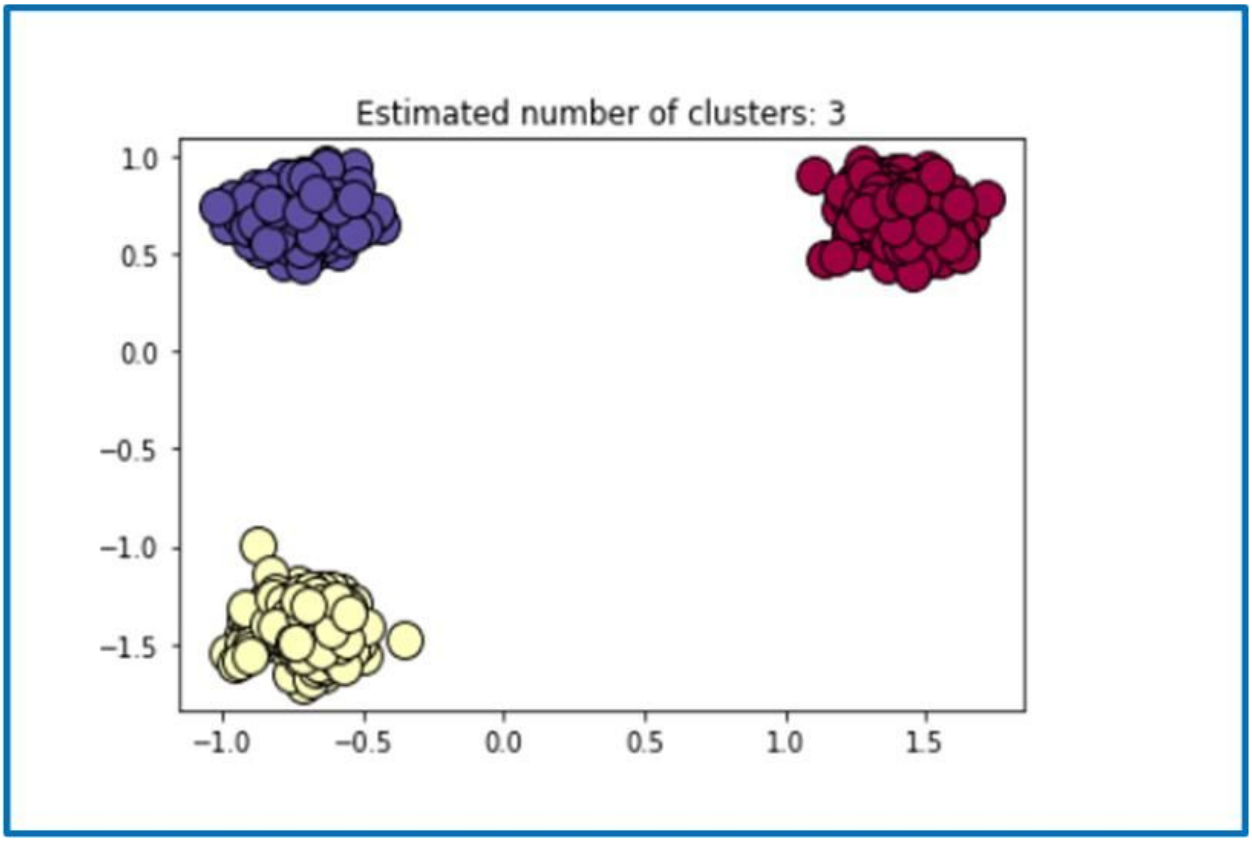

Figure 9: DBSCAN displays no outliers and no deep anomalies for the third feature pair

Analysis

After analyzing the outliers that were isolated by the three algorithms, I found that out of the 10,000 aircrafts in my dataset, four of the aircrafts, including two solo aircrafts and two units of aircrafts were accurately labelled as anomalous. After investigating the type of aircrafts, flight plans, and government affiliations of each anomaly, it was decided that I needed to define two additional parameters: “normal” anomaly and “deep” anomaly.

Normal anomalies were anomalies found in the data that was collected during pre-flight checks and the flight plan of a solo aircraft or a unit of aircrafts. Violating the flight plan implied breaking low-level decision making, that was less impactful and accounted for special circumstance flights and missions. **Deep anomalies** were anomalies found in data collected during flight that violated the country’s specific government policies for aircrafts, with regards to the feature pair. An aircraft could have a “deep” anomaly but no “normal” anomaly if the unit or aircraft followed its individualized flight plan, but did not adhere to government policies. This implied that they were involved in stealth and espionage missions. The following table explains how each feature pair tested could produce a “normal” or “deep” anomaly.

Feature pairs	**Normal Anomaly**	**Deep Anomaly**
Position of aircraft **vs** Time aircraft was at that position	If the location of the aircraft at a given time does not follow the pre-flight check and flight plan	If the location of the aircraft at that time is in violation of government or security policy for the specific type of aircraft

Amount of time aircraft was tracked for **vs** Number of messages the aircraft sent to ATC	In a pre-flight plan, there is an estimated number of messages that can be relayed to ATC. Solo flight anomalies could be detected by finding points that were grouped with the overall dataset, yet were on the very edge. This was because deviations from their flight plan could not be compared with any other aircrafts due to their individualized flight plan.	Most governments have regulations regarding the minimum or maximum number of messages that must be sent within a time period. The United States is lenient about their policies compared to other countries. Therefore, true outliers would have to significantly deviant.
Speed vs Altitude	Weather is a factor when determining pre-flight plan and non-compliance can cause accidents due to airspace violation and weather conditions. Some deviations may occur due to unexpected weather changes,	Policies of countries regarding type of aircraft, engine types, and aircraft types can vary. A true anomaly would have to be an aircraft or unit of aircrafts that performed extremely dangerous or deviant

	thus outliers can only be found by looking for an updated flight plan.	behavior by not responding appropriately according to airtime protocols.

The need to classify anomalies as “normal” and “deep” anomalies arose when I learned that a small portion of my dataset included planes authorized by the government to fly in areas that would normally be considered outside the jurisdiction of these planes. Furthermore, a key factor in isolating the anomalies was the fact that while the plane may have followed requirements for relaying messages back to ATC, the plane deviated from its flight plan, due to a change in orders from the pilots superiors. Thus, though the three machine learning algorithms were successful in isolating planes and units that could be considered anomalous given a certain set of parameters and as dictated by a country’s aerospace codes, special circumstances were not always considered because of the limited parameter capabilities of these algorithms.

Conclusion and Future Direction

To summarize, if machine learning algorithms were implemented in systems that were not subject to “special circumstances” conditions outside of the country’s aerospace policies, it can be concluded that these algorithms can have a significant impact in real-time anomaly detection. A future step for truly solving the issue of real-time anomaly detection would be isolating anomalies and immediately checking the required documents to ensure an anomaly is legitimate. However, in the real world, not all missions or flights can be predicted by a set of parameters, as seen by the four potential anomalies revealed in my study. Furthermore, when evaluating the number of aircrafts under the United States Navy’s jurisdiction, it would be

interesting to consider how these ML algorithms could increase their respective processing powers. Instead of processing a data of only 10,000 aircrafts, the impact of the ML models would be much higher if these algorithms could analyze one million or two million aircrafts at the same rate and accuracy. According to Dr. Matei Zaharia and his team “As data sizes have outpaced the capabilities of single machines, users have needed new systems to scale out computations to multiple nodes”. Applications including MapReduce and Hadoop, which are packages in the Apache Spark software, can aid in completing large scale processing. Thus, for any researchers interested in taking the abilities of these ML algorithms to the next step, using these applications would aid in achieving large scale data processing.

References:

1. Naval Surface Force. "Official U.S. Navy Website - Commander Naval Surface Force, U.S. Pacific Fleet." *Affinity Groups*, Surface Warfare Magazine, 2017, vol 56. www.public.navy.mil/surfor/swmag/Pages/Our-Navy's-Mission-How-the-surface-forces-fit-in.aspx.
2. Federation of American Scientists. Fundamentals of Naval Weapons Systems. "Underwater Detection and Tracking Systems", 2010. https://fas.org/man/dod-101/navy/docs/fun/part09.htm.
3. Northrop Grumman "CANES". 2015. http://www.northropgrumman.com/Capabilities/CANES/Pages/default.aspx?utm_source=PrintAd&utm_medium=Redirect&utm_campaign=CANES+Redirect
4. Harry Thie, Margaret C. Harrell, Joseph Jenkins, Aine Seitz McCarthy. Consolidated Afloat Networks and Enterprise Services (CANES): Manpower, Personnel, and Training Implications. United States Navy, Rand Corporation, National Defense Research Institute (U.S.), 2009.
5. Neil C. Rowe, Arijit Das, and James Z. Zhou. Distributed Combat Identification of Interesting Aircraft. Presented at the 23rd International Command and Control Research & Technology Symposium, Pensacola, FL, US, November 2018. Retrieved on February 2, 2019 from: https://faculty.nps.edu/ncrowe/ncrowe_iccrts18_distributed_combat_id.htm.
6. Daveed Gartenstein-Ross. "Terrorists are going to use Artificial Intelligence" Defense One. May 3, 2018.

https://www.defenseone.com/ideas/2018/05/terrorists-are-going-use-artificial-intelligence/147944/

7. "Introduction." *Foundations of Machine Learning*, by Mehryar Mohri et al., MIT Press, 2012, pp. 1–10. *JSTOR*, www.jstor.org/stable/j.ctt5hhcw1.4. Retrieved from https://www.jstor.org/stable/j.ctt5hhcw1
8. Jasra, Sameer & Gauci, Jason & Muscat, Alan & Valentino, Gianluca & Zammit-Mangion, David & Camilleri, Robert. (2018). Literature review of machine learning techniques to analyse flight data. Retrieved from: https://www.researchgate.net/publication/328725531_Literature_review_of_machine_learning_techniques_to_analyse_flight_data
9. Salcido, R., Kendall, A., and Zhao, Y., Analysis of Automatic Dependent Surveillance-Broadcast Data. Proc. Deep Models and Artificial Intelligence for Military Applications, AAAI Technical Report FS-17-03, 2017.
10. Liu, Bo. Parallel Maritime Traffic Clustering Based on Apache Spark. Retrieved from https://pdfs.semanticscholar.org/ee57/da09d549d6d074de29bf03aa2b33948295ad.pdf
11. Scikitlearn (DBSCAN): https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html
12. Jasra, Sameer & Gauci, Jason & Muscat, Alan & Valentino, Gianluca & Zammit-Mangion, David & Camilleri, Robert. (2018). Literature review of machine learning techniques to analyse flight data.
13. Piech, Chris, and Andrew Ng. "K-Means." CS221, Stanford University, 2013. <http://stanford.edu/~cpiech/cs221/handouts/kmeans.html>.

14. Ester, Martin, Kriegel, Hans-Peter, Sander, Jorg, and Xu Xiaowei. "A Density-Based Algorithm for discovering clusters in large spatial databases with noise". (1996). <https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf>.

15. Matei, Zaharia, and Reynold S Xin. “Apache Spark: A Unified Engine for Big Data Processing.” *Data Bricks*, 8 Apr. 2019, <https://pages.databricks.com/rs/094-YMS-629/images/p56-zaharia.pdf?_ga=2.26889277.1223871922.1554742728-1207322235.1554742728>.