18th International Conference on Space Operations, Montreal, Canada, 26 - 30 May 2025.

SpaceOps-2025, ID # 408

Enhancing Machine Learning Analysis of Satellite HKTM through the Application of Domain Specific Knowledge

E Trollope[a]*, A. Van Der Steichel[b], A. Ahmad[c], H. Kraemer[d], G. Casonato[e]

[a] *Flight Operations Division, EUMETSAT, Germany, ed.trollope@eumetsat.int*
[b] *Faculty of Aerospace Engineering, TU Delft, Netherlands, a.m.n.vandersteichel@student.tudelft.nl*
[c] *Flight Operations Division, EUMETSAT, Germany, anees.ahmad@eumetsat.int*
[d] *Satellite and Service Operations, Telespazio Germany, Germany, holger.kraemer@external.eumetsat.int*
[e] *Technical and Scientific Support Division, EUMETSAT, Germany, gianni.casonato@eumetsat.int*
* Corresponding Author

Abstract

EUMETSAT is the European Organisation for the Exploitation of Meteorological Satellites, and is responsible for monitoring weather, climate and the environment from space. It is a key partner in the European Union's Copernicus Earth observation programme, which is the world's largest provider of Earth Observation data. In this context, the Copernicus Sentinel-3 mission provides crucial data for ocean and weather forecasting, environmental monitoring, and climate change research.

The flight operations division at EUMETSAT is responsible for ensuring the smooth operation of the satellites, monitoring and maintaining satellite health status, controlling the satellites, performing mission planning activities, and ensuring the collection of the scientific and housekeeping data from the satellites.

The EUMETSAT Copernicus Sentinel-3 Flight Control Team are assessing multiple Machine Learning tools for performing novelty detection on spacecraft telemetry data, with the aim of refining and integrating at least one of these tools into routine spacecraft health monitoring.

With thousands of parameters generated every few seconds, satellite housekeeping telemetry datasets are generally assumed to be well suited to the application of machine learning. However, expected ageing effects and well-established environmental trends/effects can lead to complex evolutions of the data over many years, which are typically not predictable by traditional machine learning (ML) applications. This contributes to the detection of large numbers of "novelties" that are at odds with the fundamental desire of operations teams to focus attention on key aspects and critical issues. The team at EUMETSAT are therefore looking at the introduction of domain-specific knowledge to further enhance the robustness of the ML application outputs and minimise the quantity of false positive or trivial detections. In particular, a specific use case has been running on the EUMETSAT ML Framework platform for validating that approach using ML applications with real operational spacecraft housekeeping telemetry (HKTM) data.

This paper includes an assessment of two outlier detection tools that have been running in parallel, using different methodologies but utilising the same satellite data, for integration into the routine monitoring concept of the Sentinel-3 mission. The assessment is given in terms of both effectiveness and usability, as well as describing efforts made to extend and enhance the tool chosen for integration into the routine monitoring concept. Real examples of detected satellite anomalies and micro-meteorite or debris impacts on the satellite are discussed. The results, lessons learned and recommendations for potential applicability to other missions are presented.

Keywords: machine learning, data analysis, satellite monitoring, spacecraft operations, mission operations concept, outlier detection

Nomenclature

FN	Total Number of False Negative Classifications
FP	Total Number of False Positive Classifications
k	Number of Nearest Neighbours (used by kNN algorithm)
TN	Total Number of True Negative Classifications
TP	Total Number of True Positive Classifications
t	Time
F	Scaling factors (aging, annual, solar)
λ	Periods (annual, solar)
δ	Tuneable offset
i	Intercept at t=0
a	Aging factor

Acronyms/Abbreviations

AI	Artificial Intelligence
AOCS	Attitude and Orbit Control System
ANN	Autoencoder Neural Network
CHART	Component Health Analysis & Reporting Tool
FCT	Flight Control Team
GUI	Graphical User Interface
HKTM	Housekeeping Telemetry
kNN	k-Nearest Neighbours
MAE	Mean Absolute Error
MCS	Mission Control System
MSE	Mean Square Error
ML	Machine Learning
MLF	Machine Learning Framework
MLOps	Automatized AI/ML model training, re-training, and serving
ODA	Outlier Detection Algorithm
OLCI	Ocean and Land Colour Instrument
PUS	Packet Utilisation Standard
ROC AUC	Receiver Operating Characteristic Area Under the Curve
STD	Standard Deviation

1. Introduction

With the established trends of increasing numbers of spacecraft telemetry parameters and decreasing team sizes to monitor them, combined with the advent of increasingly large constellations of spacecraft to monitor, many operators have turned to artificial intelligence (AI) algorithms, and machine learning (ML) in particular, to support the routine health monitoring of spacecraft [1-6]. At EUMETSAT, semi-supervised outlier detection algorithms (ODAs) have been incorporated into the routine operations reporting toolsets of multiple missions for a selected subset of parameters [2,3].

It has been established that such tools are sufficiently sensitive to bring improvements to the timeliness of anomaly detections over traditional monitoring concepts. However, there remains large gap between the use of these tools today, and the goal of having a tool that can be universally applied to a significant proportion of the parameters of a satellite mission, or for these tools to be promoted from novelty detection to anomaly detection, whether on ground or in orbit.

This work focuses on ways of mitigating the challenges experienced by the flight control teams (FCTs) at EUMETSAT and elsewhere when integrating such tools into their routine operations concepts. In particular, the challenges associated with minimising the workload associated with investigating detections of novel-but-not-anomalous behaviour observed in the data, distinguishing expected deviations in housekeeping telemetry (HKTM) values from anomalous or concerning trends, minimizing the time between launch and the operational readiness of the tool, and performing the maintenance of the tool in terms of re-training. Real satellite anomalies and debris impacts detected by the tool are discussed.

1.1 EUMETSAT Component Health Analysis & Reporting Tool (CHART) Framework

Originally created as a tool to analyse spacecraft HKTM on the Metop mission at EUMETSAT [7], the EUMETSAT Component Health Analysis & Reporting Tool (CHART) has evolved steadily over the past two decades [8] and continues to be enhanced continuously.

CHART is used by all missions operated at EUMETSAT and is implemented through a 'core' codebase common to all missions, and 'project' codebases to implement mission-specific features. Spacecraft HKTM is ingested and stored in timeseries database tables, while events (either generated by the spacecraft or auxiliary information coming from other ground sources) are ingested into a dedicated events database table. CHART enables a great deal of flexibility, as it allows for ingestion and synthesis of data coming from many different sources. For example, synthetic parameters or events can be defined using a combination of HKTM and ground-based data, which would not be possible in the mission control system (MCS) which relies on the spacecraft database. 4-dimensional statistics (min/max/mean/std) are automatically calculated and stored for HKTM data on a regular (orbital and daily) basis.

Data stored in the CHART database can be viewed in several different ways through a web interface:

- Timeseries data (either 'all-points' or statistical data) can be plotted via the CHART GUI. Various types of plots can be generated, e.g. line plots, correlation plots or geolocation plots
- Events can be visualised in a tabular representation, or overlaid on plots
- HTML reports present data as defined in an XML template.

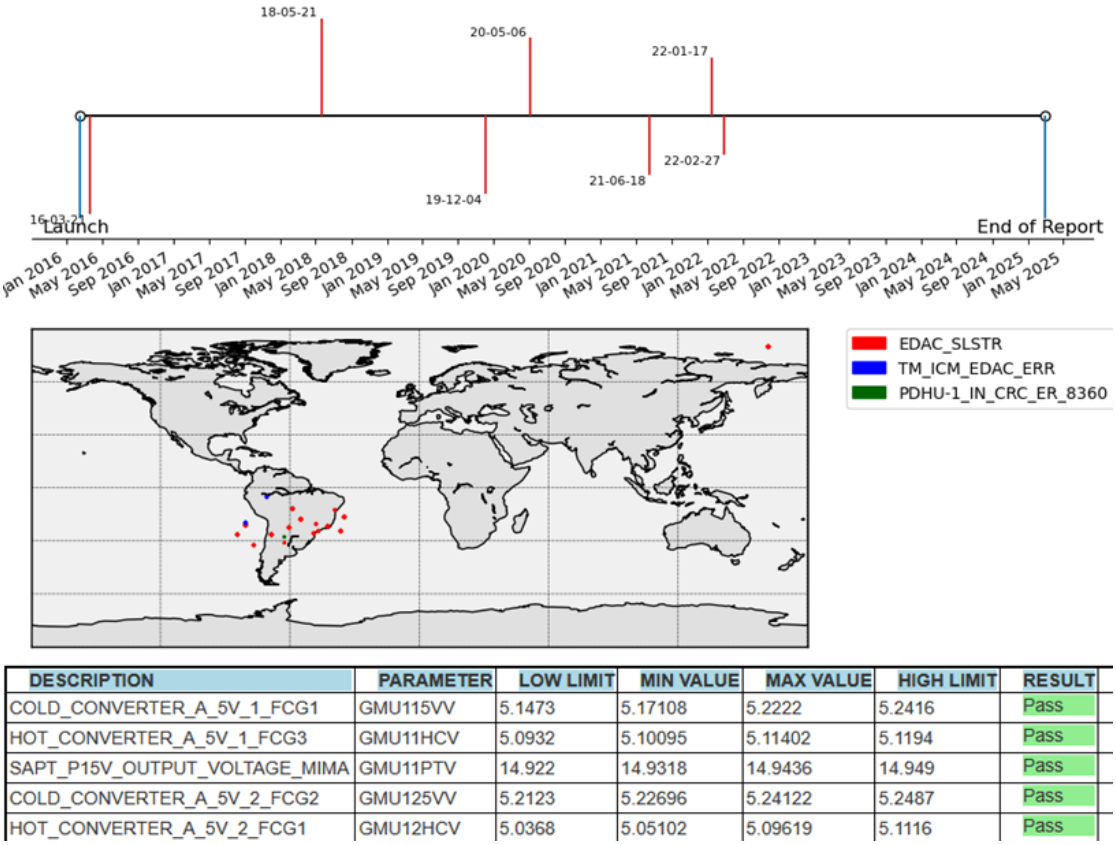
Report generation can be triggered through an automatic scheduler process or on an ad-hoc basis. Report contents are defined using Python 'widgets' to process and present data as desired with a great deal of flexibility. It should be noted, however, that the output of any processing performed by a widget is not stored permanently in the CHART database. Generated reports are accessed via the CHART GUI as shown in Figure 1 below. Some examples of widgets are shown in Figure 2.

Figure 1 Accessing reports via the CHART GUI – list of available reports (left) and a calendar view for choosing the report instance (right)

The Sentinel-3 Flight Control Team make extensive use of CHART for monitoring and analysis of spacecraft health. Some examples are:

- A daily report generated each morning gives the on-call Spacecraft Operations Engineer a high-level overview of any operations, unexpected events or data gaps observed during the previous day
- Ad-hoc offline HKTM analysis (both short and long-term) is generally performed via CHART rather than in the MCS due to ease of access and superior plotting functionality
- Monthly reports for each spacecraft subsystem provide an overview of subsystem health, including long-term HKTM trending and monitoring of specific trends of interest (e.g. generation rates of low-severity

events). These reports are also made available to external partners (e.g. the spacecraft manufacturer) through an external access portal



DESCRIPTION	PARAMETER	LOW LIMIT	MIN VALUE	MAX VALUE	HIGH LIMIT	RESULT
COLD_CONVERTER_A_5V_1_FCG1	GMU115VV	5.1473	5.17108	5.2222	5.2416	Pass
HOT_CONVERTER_A_5V_1_FCG3	GMU11HCV	5.0932	5.10095	5.11402	5.1194	Pass
SAPT_P15V_OUTPUT_VOLTAGE_MIMA	GMU11PTV	14.922	14.9318	14.9436	14.949	Pass
COLD_CONVERTER_A_5V_2_FCG2	GMU125VV	5.2123	5.22696	5.24122	5.2487	Pass
HOT_CONVERTER_A_5V_2_FCG1	GMU12HCV	5.0368	5.05102	5.09619	5.1116	Pass

Figure 2 Some examples of CHART widget functionalities – displaying a timeline of event generation (top), geolocating events (middle) and checking TM parameters against limits (bottom)

1.2 EUMETSAT Machine Learning Framework

The Machine Learning Framework (MLF) system is a platform developed in EUMETSAT for delivering centralised AI/ML application-controlled execution services, providing automatized AI/ML model training, re-training, and serving (so-called MLOps capabilities) for small-to-large scale scenarios and supporting a large variety of AI/ML models based on Deep Networks, Convolutional Networks and Large Language Models [9].

As illustrated in Figure 3, the MLF supports two main use cases:

- AI-as-a-service to Data Analyst/Consumer users interested in the AI/ML application results
- Sandpits to AI/ML Expert users for supporting hands-on model exploration and development taking benefit of the MLF platform compute capabilities.

For the first class of use cases, ad-hoc applications are developed and deployed in production providing a "black-box" AI/ML service to end users - typically as model prediction results in a certain format (e.g. as a file, in a WebUI, in a database). For the second class there is a Jupyter Notebook access granted on a sandpit node and the expert use is free to develop and experiments directly its AI/ML models. For both use cases MLF provides also a Python API library streamlining the AI/ML applications development covering direct access to EUMETSAT data sources, to a large range of predefined models, and to common data preprocessing and model scoring methods.

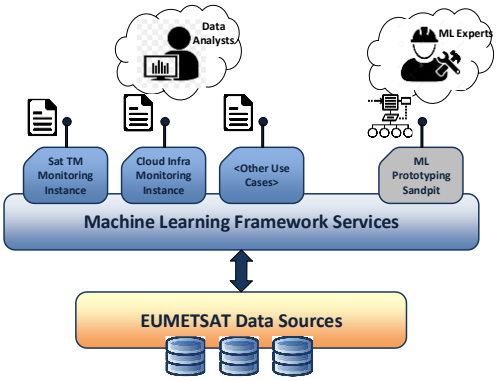


Figure 3 MLF Access Use Cases. [9]

From a deployment standpoint, the MLF is based on a Docker image-based deployment on a Kubernetes cluster which provides the capability of running practically and type of application wrapped in a Docker, and nodes connectivity with EUMETSAT network for access from office PCs. The cluster makes natively possible a basic level of production automatism (MLOps level 1 – see Figure 4 below) for applications execution and monitoring. A further step in automatism is in preparation for supporting MLOps level 2 and streamlining application development and deployment in production. Finally, this approach allows horizontal (separated multiple nodes for each AI/ML applications) and vertical (transparent CPU and GPU compute resources scaling) scalability, with no impact on running applications.

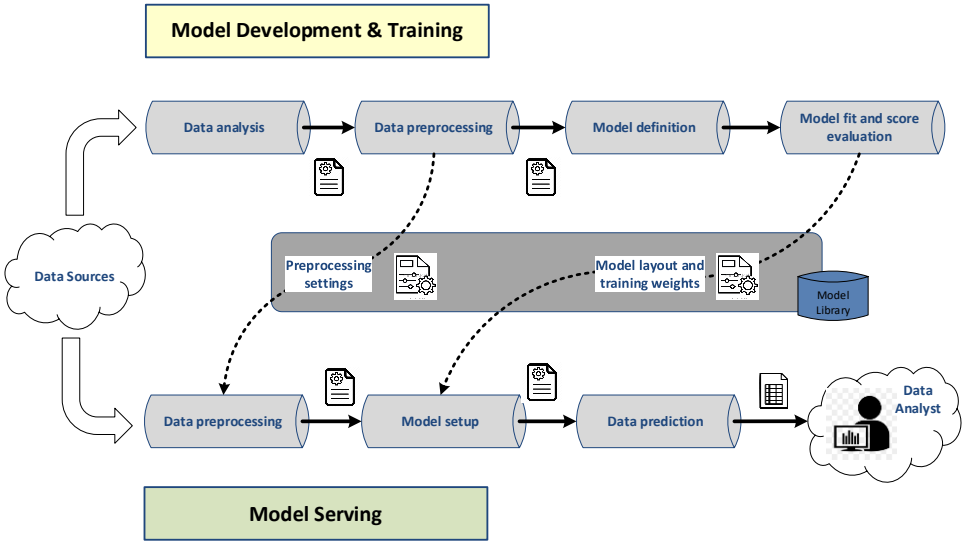


Figure 4 MLF MLOps level 1 capabilities. [9]

1.3 kNN Algorithm

A k-Nearest Neighbours (kNN) ODA identifies outliers by assessing the local density of data points in x dimensions, where x is the number of columns in the dataset. For each point, it calculates the distance to its k nearest neighbours and assigns an outlier distance score. Points located in sparse regions, indicated by large distances to their neighbours, receive higher outlier scores. A predefined threshold is then used to classify points with sufficiently high scores as outliers, effectively flagging them as deviations from the typical data distribution.

The kNN algorithm used in this work was developed building on the algorithm presented by Losco et al [3] which was originally developed for use on the Metop mission at EUMETSAT but for this project has been adapted for use with the Sentinel-3 mission data. This algorithm expands the generic kNN concept with the introduction of an "Influencer/Follower" concept for conditional outlier detection, which significantly reduces the volume of detections

that are reported to the users in the event of a spacecraft anomaly or special operation [3]. Parameters with one or more "influencers" that are flagged as outliers are excluded from the evaluation.

This algorithm was provided as input the 4-dimensional vectors of normalised orbital statistics (min, max, average and standard deviation) for each parameter being evaluated and returns an outlier score for each parameter per orbit. These scores are compared against a set of user-defined thresholds and converted into colour-coded categories of severity for presentation to the user [3]. Modifications to this process were explored and are discussed later in this paper, including additional sources of data to mitigate simultaneous detections of large numbers of parameters (see §4.1), the introduction of engineering expectations (see §4.2) and judgement (see §4.3), and experimentations with additional dimensions (see §4.4).

User-defined time ranges representing nominal operational values were provided as training data. During the evaluation period, all orbital statistics for the parameters under test were evaluated by the algorithm on a weekly basis, with an automated report being provided to the team. False positives due to gaps in data caused by e.g. RFI or special operations are prevented by discarding orbits with unexpected data volumes [2].

1.4 Autoencoder Neural Network Algorithm

The algorithm used was an Autoencoder Neural Network (ANN). Its basic idea for identifying outliers in a dataset is the concept of "distance" of an input data sample with respect to a "known" nominal dataset, by considering the measure of the distance of the data sample from the centroid of the distribution of the "known" dataset and evaluating it against a precalculated threshold for determining if the input sample is an outlier or not.

There are different methods for calculating the distance and for imposing statistical properties to the "known" dataset distribution, and one used in this paper was based on the Mahlanobis distance calculation, i.e. calculation of the distance of a sample from the distribution centroid for each principal component axis by using the covariance matrix between the sample and each axis [10], and this implies that distance measures have stochastic nature. ANNs can be effectively used for calculating the sample distance in a similar way noting that the ANN can be trained to provide an output as replica of the input with a certain accuracy using a "known" training distribution and then applying it to "unknown" samples for getting their distance from the trained one. Specifically, it can be shown that the deviation of the ANN output from corresponding input, calculated as Mean Absolute Error (MAE), corresponds to the Mahlanobis distance of the same sample with respect to the centroid of the same training dataset. The advantage of using ANN is a more efficient computation of the distance (within a certain accuracy), not requiring the calculation of the covariance matrix of the whole dataset.

Another advantage of the ANN method is that the definition of the threshold to be used for assessing if a calculated MAE distance is a regular or novel datapoint can be done as a byproduct of the ANN training, by applying a training dataset with a predefined ratio of regular and novelty samples. A typical split is 3-sigma, i.e. 99.7% regular and 0.3% novelties. That makes the ANN able to scale to problems with many features very easily, not requiring an ad-hoc feature specific threshold identification like other classical ML methods. ANN method has also some constraints and limitations, specifically given by the requirement of applying to datasets with a normal (or normal-like) distribution. This is not a hard constraint, and it is satisfied in a large number of cases for satellite HKTM telemetry, but there are still situations like state-based parameter values switching between different digital values (single or multi-modal parameters) where the distribution would be not suitable for ANN because of the zero standard deviation. In that case feature engineering can be applied, for example by applying controlled "noise" to single-mode parameters to increase the standard deviation and make the distribution normal-like, or to split the multi-modal parameters in elementary single-mode and apply noise as described. This would allow the ANN method still to work, but it would introduce a (limited) drop in the result accuracy.

This algorithm was provided as input the same 4-dimensional orbital statistics used by the kNN algorithm described above. The outputs of this tool are an outlier mean absolute error distance score, and min/max expected values that are unique to each parameter. Each of these outputs were ingested into the Sentinel-3 CHART tool, so that the FCT could plot the outputs of the ANN algorithm alongside the actual data from the spacecraft. One such parameter is illustrated in Figure 5 below.

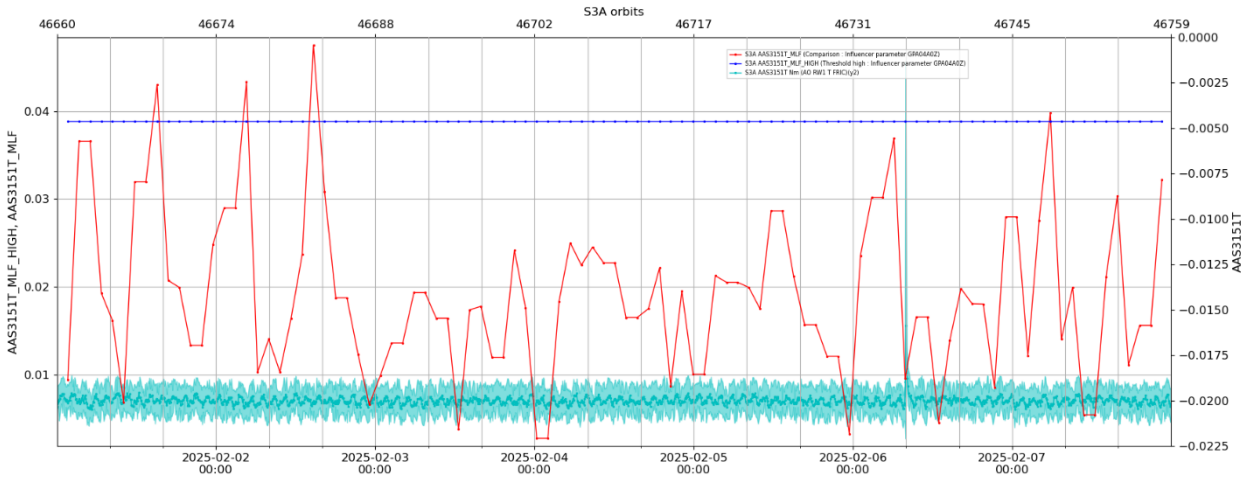


Figure 5 ANN Outlier Score (red, left axis) and max (dark blue, left axis) for AOCS reaction wheel friction (teal, right axis) over one week.

2. Balanced Accuracy, Paranoid Algorithms and Engineering Judgement

It has been well established that even relatively basic ODAs are highly sensitive and can detect anomalous behaviour in spacecraft HKTM much faster than relying on humans to monitor behaviour. However, despite the evidence showing these tools detect anomalies faster than engineers, such tools are not typically included in routine spacecraft operations concepts.

Traditionally, machine learning approaches are assessed using functions of the true positive rate and the true negative rate (sensitivity/recall, precision, specificity, balanced accuracy, F_1 score, ROC AUC). However, this approach only identifies datapoints that are mathematically distinct from the training dataset, and it can be a significant challenge to differentiate between novel and genuinely anomalous data points. Unlike anomalies, novelties are typically incorporated into the normal model after being detected [11], but this may rely on a categorisation being done manually. This is an accurate description of the outlier detection concepts currently in use within the EUMETSAT flight operations division, as the nature of satellite aging typically leads to the phenomena of domain-shift, which in turn results in the identification of many data points that are novel but not anomalous.

One of the payloads onboard the Copernicus Sentinel-3 Satellites is a cryogenic optical instrument, namely the Sea and Land Surface Temperature Radiometer. This instrument includes infra-red detectors that are cooled to below 80K, with correspondingly significant power demands. On occasions when that cooling is interrupted, ODAs have proven sensitive enough to detect non-standard temperatures across the entire spacecraft as a result. These changes in temperature impact on some heater control loops, leading to further parameters that can be identified as impacted by this change in operation of the unit. Even the AOCS reaction wheel speeds can be impacted, as the outgassing resulting from the warming imparts a torque on the satellite. Despite all these detections being interesting from an academic perspective, they are dire from an operations perspective. Such a large volume of non-nominal readings being detected at the same time would potentially create a lot of "information noise" that risks masking the root cause of all these changes – the change in behaviour of the instrument's cooling apparatus. A significant quantity of this can be overcome using the follower-leader concept described by Losco [3] – a concept that has been included in both the ANN- and kNN-based tools used in this work.

It is not sufficient for a tool to be assessed in terms of true/false positive & negative, when considering integration into an operations concept. It should be noted that a data point that is flagged as novel, but is not truly anomalous, must not be considered as a False Positive if in fact it is mathematically distinct from the training data. Reliance on engineering judgement to distinguish between novel and anomalous data points is something that must be minimised as a goal, to avoid the risk of user error (e.g. identifying anomalous data as novel and adding it to the training data), and to minimise workload demand on the team (and therefore cost). We therefore introduce the concept of "paranoid algorithms", where paranoia is defined as *the probability that an outlier will be considered as normal behaviour by an expert through the application of domain-specific knowledge.* i.e. a "paranoid" detection is one that is novel, but not

anomalous. For an application to be truly useful for purposes of spacecraft operations monitoring, it must have high accuracy ($\frac{TP+TN}{TP+FP+TN+FN}$), high precision ($\frac{TP}{TP+FP}$), high sensitivity ($\frac{TP}{TP+FN}$), and low paranoia ($\frac{TP_{novelty}+FP}{TP_{novelty}+TP_{anomaly}+FP}$).

3. kNN Vs ANN Assessment

In this section, the authors evaluate the outputs of the kNN and the ANN models, together with the methods of visualisation and data transfer used to present the information to the flight control team (categorised 2D scatter plots and time-stamped numeric weighting). The tools are evaluated and compared based on their effectiveness and usability. The authors emphasise that although the methods of visualisation used for the output of the two tools differs, this was done to avoid duplication of effort and that the visualisation methods are independent of the algorithm used.

3.1 Advantages and Lessons Learned from using the ANN Algorithm with Timestamped Numeric Weighting

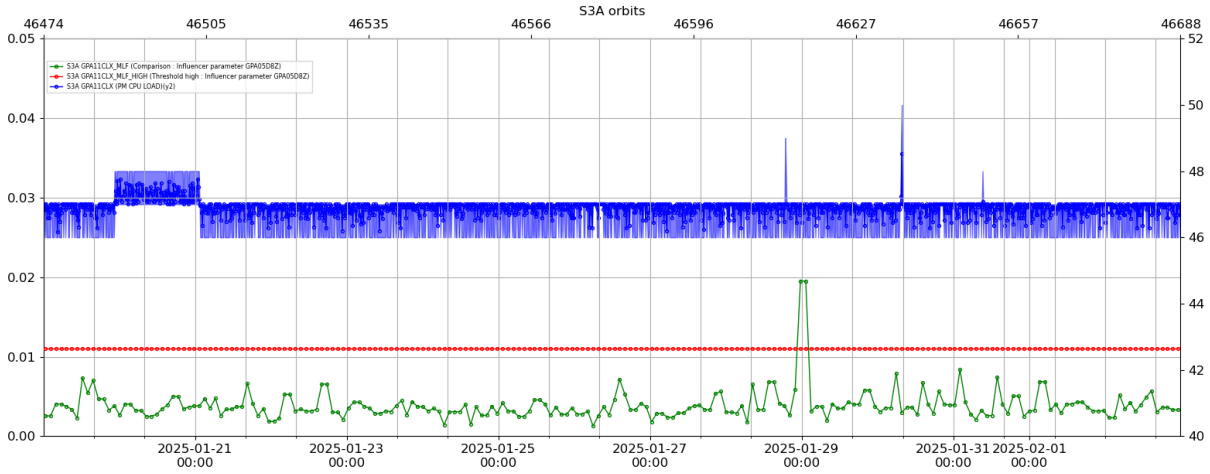


Figure 6 Spacecraft parameter (blue, right axis) compared to the output weighting of the ANN tool (green, left axis) for the parameter, and the outlier threshold (red, left axis) for the weighting over a period of 2 weeks

Figure 6 above shows a spacecraft parameter (blue, right axis) for a period of 2 weeks. Underneath, the weighting score output by the ANN tool (green, left axis) for this parameter can be seen to break the defined threshold (red, left axis) for a short period lasting 2 orbits. There are many features observable in the behaviour of the parameter being monitored, with step changes on the left of the plot and spikes on the right. None of these features are identified as novel by the ANN and presenting the ANN score alongside the data makes life a lot easier for the FCT who may otherwise need to check other time periods or the unit documentation to confirm whether such features represent nominal behaviour or not. This demonstrates how the assessment provided by such a tool can directly benefit the FCT and reduce workload even in situations where unit behaviour is nominal.

Two orbits within this period are identified as outliers by the ANN algorithm. From this plot however, there is no clear reason that can be seen for these two orbits to be highlighted as outliers. If the team is not to immediately dismiss such detections as *paranoid detections*, it is necessary for the team to be able to “dig into” the output of the tool and understand why they have been flagged as outliers in a manner that is not possible with a single numeric output alone, even if the original data for the parameter is also available.

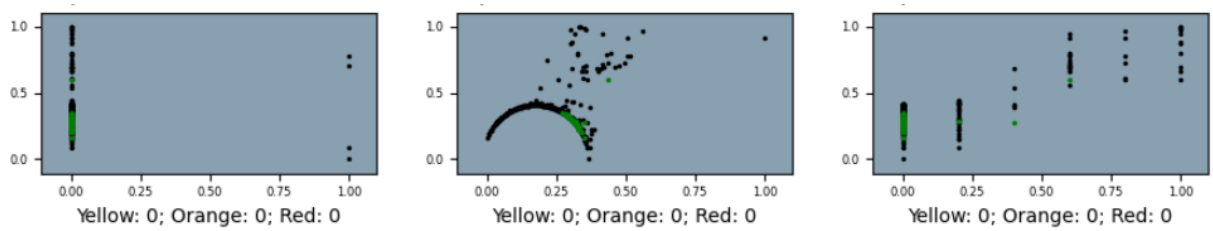


Figure 7 Series of scatter plots of the training data (black) and test data (green) for the same data shown in Figure 6.

Figure 7 shows both the training data and test data for the same parameter and timeframe as discussed for Figure 6 previously, but presented as a series of two-dimensional scatter plots with STD on the y-axis of each plot and the min/mean/max used for the x-axis sequentially. The training data is plotted as black dots, with test data plotted as green dots. Although the weighting value associated with each individual vector is not reflected in these plots, they are sufficient to allow the users to confirm that these two outliers can be classified as false positives as the input values were not significantly different from the training data. However, without such a method for visualising the data such an understanding would be very difficult to achieve.

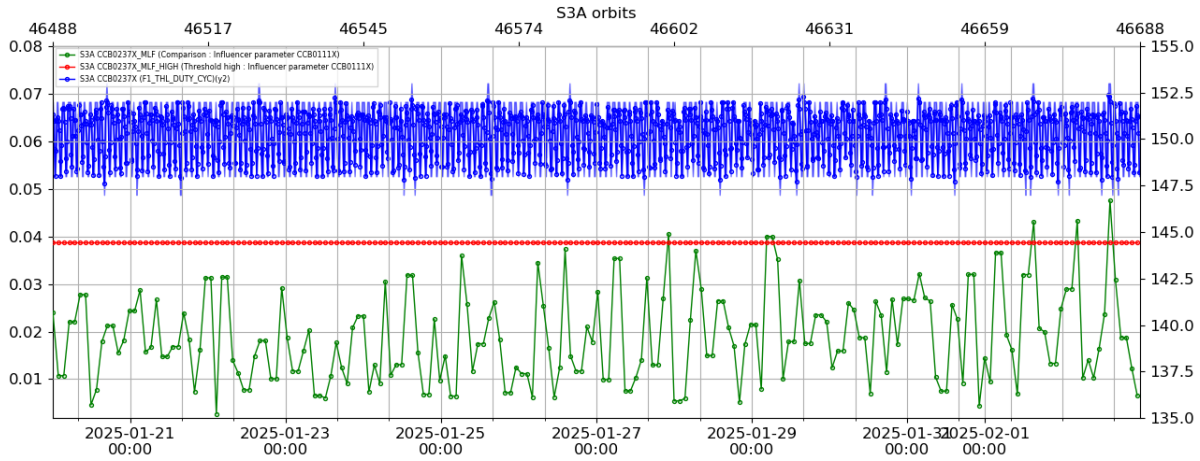


Figure 8 Spacecraft parameter (blue, right axis) showing no obvious deviations in behaviour, compared to the output weighting of the ANN algorithm (green, left axis) for the parameter with a steady increase in value over a period of 2 weeks, and the outlier limit (red, left axis) for the weighting

Figure 8 shows a spacecraft parameter (blue, right axis) over a period of 2 weeks that is not showing any obvious deviations in behaviour, along with the output weighting of the ANN algorithm (green, left axis). This shows a steady increase in the outlier weighted reported by the ANN algorithm, until it ultimately crosses the threshold defined for this parameter (red, left axis). Plotting the temporal evolution of the ANN outputs in this manner brings several advantages. This plot illustrates that the tool's "opinion" of the parameter has a steady trend, which may indicate a subtle aging effect and gradual deviation from the training data. This can be further investigated by the team, and if nominal, may indicate the need to periodically retrain the algorithm. Assessment of the gradient of the ANN output is likely to give a more straightforward indication of the frequency with which the retraining must be done than assessing the number of outliers detected, helping to minimise the workload associated with retraining and enabling the team to plan for the retraining as necessary rather than a retraining being triggered by unwanted paranoid detections and corresponding investigation.

This highlights the benefits of storing the numeric output weighting of the ODA, for plotting alongside the original data.

3.1.1 Persistence of Output Data

The outputs of the kNN tool, which is running in the CHART framework, are accessible to the FCT via reports in the CHART web GUI. Similarly, the outputs of the ANN tool which is running in the MLF were originally only available via reports hosted on the MLF and available via the MLF web GUI. The MLF reports are only available for a limited period – testing the concept of disposable reports, with each new report replacing the previous. This was found to be OK for assessing individual runs of the tool or for determining the latest situation in orbit, but that the ability to review previous reports for comparison was highly advantageous. The outputs of the web GUI were subsequently ingested, together with the value of the outlier thresholds at that time, into the CHART database. This opened new possibilities for plotting and viewing trends of the outlier scores, in addition to the scatter plots, which were also found to be very useful by the team.

In scenarios where the ANN algorithm was tuned or updated, the outlier scores and corresponding thresholds for a given parameter have a step change. This can be clearly seen provided the threshold is stored, in addition to the outlier score itself. However, too many such changes make long-term assessment of the trends impossible, to the team must resist the urge to continually "tweak" the settings. It is possible to re-process the HKTM for the previous time periods, but this is computationally expensive and time consuming so it should be avoided where possible.

3.2 Advantages and lessons learned from using the kNN tool with categorised assessment and tiled 2-D scatter plots

One of the advantages of kNN algorithms when used by spacecraft engineering teams that do not necessarily have expertise in the field of data science is that the tool's methodology is readily understandable to the team, and the assessment can be explored to further the team's understanding of the nature of any reported outlier detections.

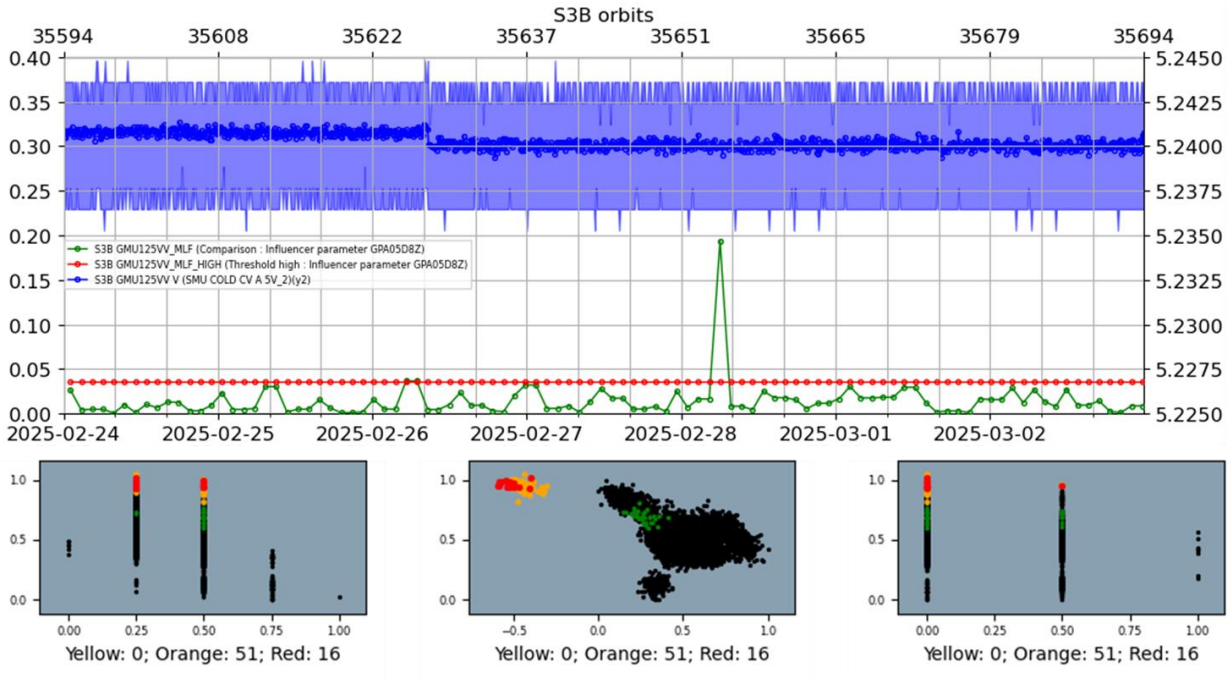


Figure 9 Top: Spacecraft parameter (blue, right axis) compared to the output weighting of ANN tool (green, left axis) for the parameter, and the outlier limit (red, left axis) for the weighting over a period of 2 weeks. Bottom: Output of kNN tool for the same parameter and date range.

Figure 9 illustrates an example where a parameter showed a step change (decrease) in average value, but without any corresponding change in minimum or maximum reported values. This behaviour is notoriously difficult to spot using traditional "limit monitoring" techniques but is easily detected by an ODA.

Below the plot of data and corresponding ANN output, the data is presented as a series of two-dimensional scatter plots with STD on the y-axis of each, and the min/mean/max used for the x-axis sequentially. Other arrangements of plots are also possible (e.g. min Vs avg and std Vs max) but this arrangement has found to be optimal for supporting

the team understanding of the tool outputs. The concept is also extensible, should additional dimensions be incorporated into the tool. The training data is plotted as black dots, with test data plotted as colour-coded points according to the categorisation of the output score (green for nominal, yellow/orange/red for increasingly significant weightings of outliers).

The 2D plots (created using the output of the kNN algorithm) illustrate clearly that the algorithm is flagging all the vectors with lower average values as outliers, and that the std/min/max values are otherwise as expected. This can be visually confirmed by the FCT by looking at the data. If this is considered problematic for the unit, the FCT can immediately communicate clearly and effectively the situation. This demonstrates the value of presenting the tool output as a series of 2D plots.

As the maximum and minimum are both highly discrete with very small number of values reported in the training data, the ANN-based tool will have discarded both dimensions and is testing only on the std and mean. However, only one datapoint is flagged as an outlier from this dataset. As the rationale of the ANN is less apparent on an engineering level than Euclidian distance, it is likely that it would be discounted as a false positive and ignored by the FCT. It is not immediately clear why, given the tool is monitoring only the mean and std and should therefore be giving greater weight to the changes in one dimension of a 2D-set compared to the kNN tool's 4D-set, why the ANN has not flagged the majority of this data as outliers. It is possible that the training dataset used may contain a structure such that the novelty datapoint configuration flagged by kNN was for some reason considered as regular behaviour for the ANN. The definition of the training dataset is of course key, and kNN and ANN have different approaches, so using the same dataset for training may not lead to the same results.

This additionally highlights the benefits of keeping the outputs of the algorithm used as clear as possible, to maximise the translation from automated detection to user understanding.

3.3 Case Study: Thermistor Failure and the Algorithm that Cried Wolf

In November 2020, one of the thermistors onboard Sentinel-3A started to report anomalous values. It is understood that this is because the thermistor, which is physically located on the outside of the spacecraft, has come loose. Every two months the satellite is rotated to perform lunar calibrations, and these operations apply a force to the thermistor that causes it to disconnect further and report increasingly inaccurate values. Figure 10 illustrates the result of monitoring this parameter with a kNN ODA, with colour-coded thresholds for the detection of anomalous values. Data after 1 Jan 2023 is not shown, as the increasing values after that date are not meaningful and detract from the visualisation of the analysis.

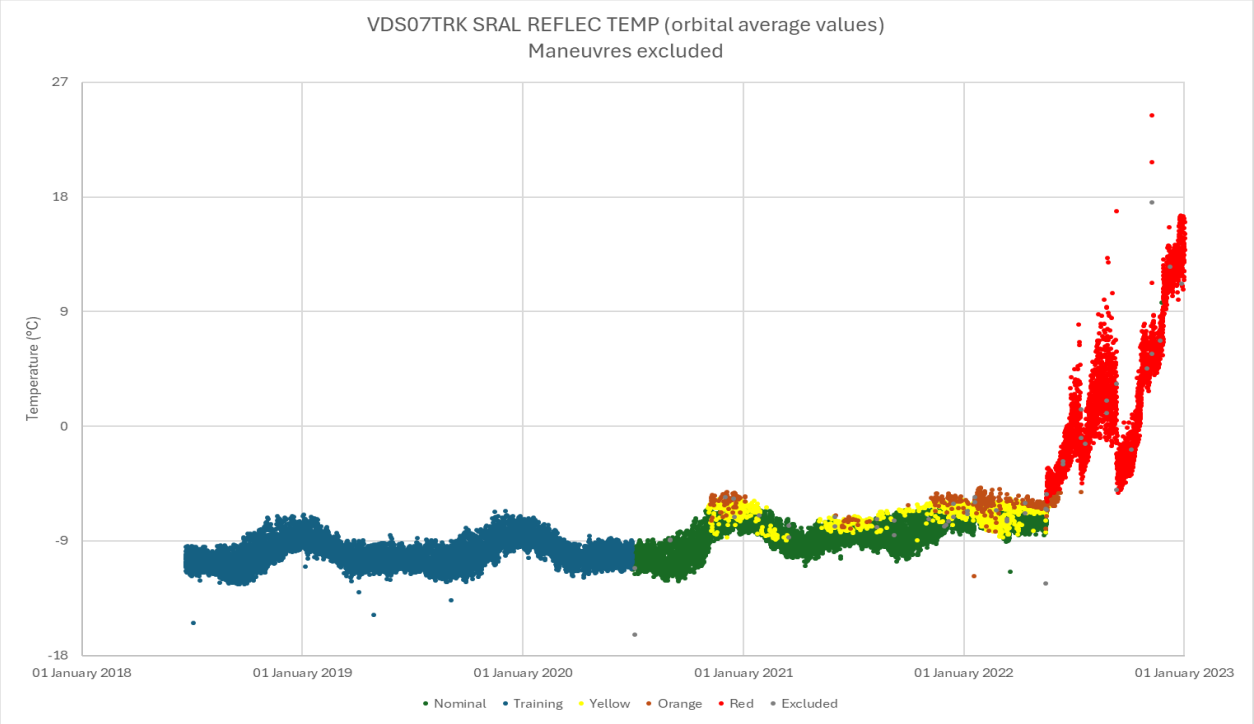


Figure 10 Thermal Data for Thermistor Exhibiting Anomalous Behaviour, Colour Coded By Outlier Classification

Looking at Figure 10, it is very clear that the behaviour of the thermistor has changed significantly on 16 May 2022. However, it can also be seen that the ODA used clearly showed anomalous behaviour more than a year ahead of this and detected correctly the change in behaviour that began in November 2020. This is a very clear example of such a tool detecting behaviour changes faster than traditional monitoring techniques. However, if an engineer was to be presented with the output of this tool in Nov 2020, what would their interpretation of the situation have been?

It would be disingenuous to assert that an FCT engineer would have correctly interpreted the situation in Nov 2020 correctly, based only on the information available at that time. As the values dropped back into the previously seen ranges and followed the established seasonal patterns, it is highly likely that this behaviour would have been considered a result of an "overly paranoid" tool and ignored. Worse, it is likely that the anomalous values seen over the winter would been considered as being caused by domain shift and consequently would have been added to the training data to prevent similar values from being similarly flagged in future. In this context, it is important to recall that it is very common for spacecraft temperatures to increase slightly over time. This would then have resulted in the comparably dimensioned values later in 2021 and again in early 2022 not being detected, until the significant change on 16 May 2022. i.e. although the tool is capable of correctly identifying the existence of a problem, it would likely have been ignored until the same date that it was seen via traditional monitoring methods! This is the spacecraft operations equivalent of the famous "boy that cried wolf".

To unlock the true potential of such a detection, the confidence of the team in the outputs of the tool must be very high. This is only possible if there is a very low number of "paranoid" detections.

3.4 Analysis

The benefits and disadvantages of different ODAs have been explored extensively, and this work does not provide any unexpected results in that context. Although it may be expected that an autoencoder would outperform the kNN algorithm [1], both tools are highly accurate once the training data is well defined which is the most critical factor.

The work done did not apply the ODAs to all 46,000 HKTM parameters routinely reported by Sentinel-3, as doing so would have overloaded the team and was not necessary to assess the tools as desired. Because the selected parameters intentionally included known anomalies and noisy parameters, the calculated values for accuracy, precision, sensitivity and paranoia would not be a fair reflection of the capabilities of the tool. Due to the changes in

ANN settings during the trial period, these values were also not constant. Consequently, the numerical values are omitted here but are discussed in qualitative terms.

The methodology used for performing the semi-automated assessment of the tool outputs is to apply the following logic to an assessment of the input data:

- Short-term (duration < 1 orbit) spikes of the average dimension only are classed as FP.
- Short-term spikes of the std dimension only are classed as FP.
- Short-term spikes of min/max dimensions but within the observed range of the training data are FP.
- Identification of long-term change of behaviour in any number of dimensions are classed as TP, regardless of magnitude.
- TPs due to minor changes (discounted by FCT engineer) are classed as paranoid.

One surprising observation was the presence of an offset in detection timings between the ANN and kNN algorithms on multiple occasions. When a sequential number of orbits were flagged as outliers, it was observed that the kNN ODA would often recognise the change in behaviour one orbit earlier than the ANN ODA. Similarly, as the change in behaviour came to an end, the kNN ODA stopped flagging the parameter as outliers sooner. This was surprising because the *accuracy* of ANN ODA is generally higher than kNN ODA. However, given the time needed for the FCT to investigate and understand the root cause (and given the frequency at which the ODA is performing the analysis, discussed later) the practical impact of this difference is not significant for the context in which the tools are being considered (weekly or monthly assessment).

4. Enhancements Introduced to the EUMETSAT Outlier Detection Toolset

4.1 Introduction of Ground Knowledge

There are times where an understanding exists on ground of expected satellite behaviour, that is not reflected in the status parameters (e.g. unit modes HKTM) at a specific moment in time. For example, the Ocean and Land Colour (OLCI) instrument onboard Sentinel-3 routinely operates in SILENT and NORMAL modes. When the mode is commanded to STANDBY mode, it switches off the fine temperature control algorithm which results in a significant change of thermal environment within the unit. When applying the EUMETSAT ODAs to this temperature, for the duration of time that the mode of the unit (the influencer) does not have the nominal value, analysis of multiple parameters (the followers) can be discarded or autonomously ignored. However, upon returning to the nominal mode, there may still be some time needed before all the follower parameters have returned to their nominal behaviour. This is illustrated in Figure 11, in which the timeframe in which the temperature behaviour is not consistent with routine operations is much longer than the timeframe in which the mode is reported with a non-nominal value.

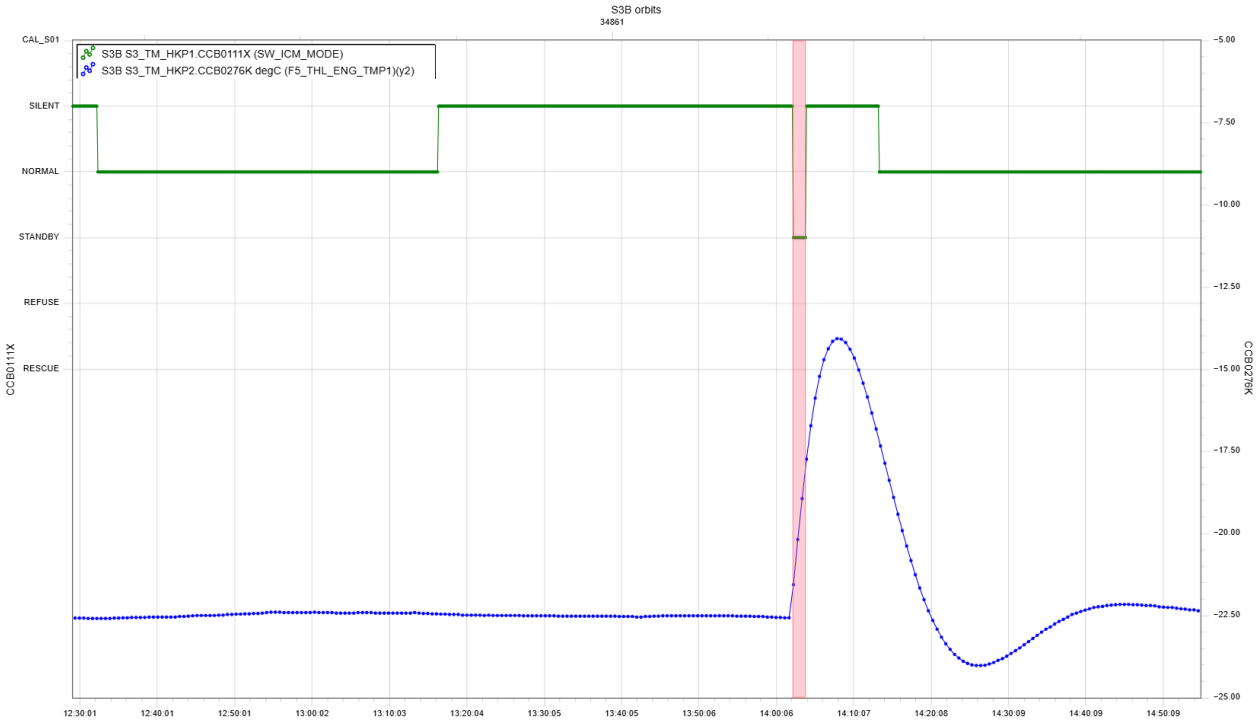


Figure 11 Instrument mode (green, left axis) with change to non-routine mode (red highlighting) with corresponding impact on thermal behaviour (blue, right axis).

To mitigate this limitation of the data available to the tool, we can introduce ground knowledge. On Sentinel-3, the EUMETSAT FCT maintain a scheduling tool known as SATCAT. This tool includes start and end time of operations and anomalies, which can be set by the FCT to include periods such as this for any given operation or anomaly, beyond the timeframe that can be directly read from HKTM values. Through the API provided by SATCAT, the CHART algorithms can identify the presence of such an operation, retrieve the respective start and end times, and avoid flagging the data as outliers for the duration of the operation. This expands the influence-follower concept by enabling the introduction of operations as a class of influencer, in addition to HKTM parameters.

An interesting aspect is that when considering the effect of mode changes on parameter behaviour, many parameters will behave in the same way during a satellite anomaly to a commanded satellite operation. Adding ground knowledge in this way enables the tool to distinguish between intentional operations, known satellite anomalies, and new or potentially as-yet undetected satellite anomalies.

4.2 Introduction of Engineering Expectations

One of the classic problems encountered by flight control teams when attempting to incorporate ODAs into routine operations concepts is *domain shift*. Figure 12 shows a temperature from Sentinel-3 over the course of eight years. As this temperature is not subject to fine temperature control, it exhibits a well-established series of compounding temporal patterns:

- Orbital (101 minutes) variation with eclipse/daylight effects,
- Daily variation from changing albedo effects as the Earth rotates,
- Seasonal variation from the beta-angle of the orbit,
- Annual (subtle) variation from the eccentricity of the Earth’s orbit,
- Long-term logarithmic warming due to aging of the satellite’s reflective surfaces,
- Cooling effect from the 11-year solar cycle,
- Short-lived spikes and dips due to satellite operations and solar eclipses.

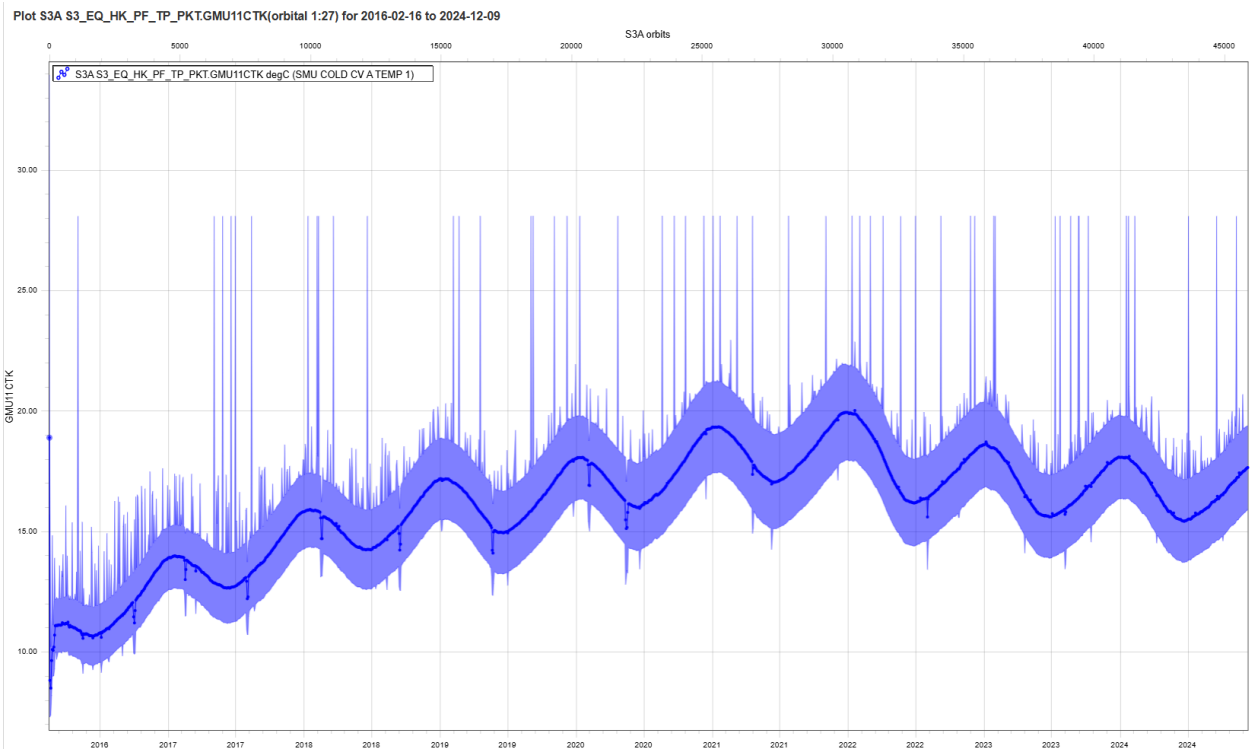


Figure 12 Thermal Data Showing Common Seasonal Behaviour And Long-Term Aging Effects

Figure 12 illustrates the issue of domain shift, where even after four years in flight the data from the subsequent years will be flagged as outliers for a significant proportion of the year. This is a good example of a parameter that can lead to teams misclassifying the detection described in §3.3 previously. In this example, using the first year or two years as training data will result in many (or all) orbits during the warmer months of the subsequent years being flagged as outliers. Two approaches to handle such a parameter are typically adopted: to continually re-train the algorithm, or to declare such parameters as "unsuitable" for inclusion in the set of parameters to be analysed. Neither option is ideal, as automated retraining risks introducing anomalous behaviour into the training data, manual training introduces an unacceptable level of workload and desensitises the team to outlier detections, and excluding all such parameters reduces significantly the capabilities of the toolset. However, these trends and phenomena are well understood by engineers and can be predicted on at least a qualitative level with relatively little effort.

In theory, a deep learning model such as a recurrent neural network may be able to learn and identify these features and significantly outperform the kNN or ANN models used here. However, these models would still be restricted to only learning from the data presented, whereas the "ground knowledge" of an engineer allows for advanced prediction of events such as the cooling effect visible in recent years in advance. To allow for an outlier detection tool to be deployed as early as possible after the launch of a satellite, it is not helpful to require that a significant proportion of the satellite's lifetime as training data.

The satellite working operational database used by the MCS includes information about these parameters being monitored, and that information has been ingested into CHART. This means that an ODA running in CHART has direct access to the engineering unit of the parameter being monitored. The linking between temperatures and heater control loops is also available on ground, and so it is not a significant challenge to identify whether a particular parameter is a) a temperature, and b) tied to a heater control loop. There are other temperatures that are impacted by heater control loops without being linked to the loop, however, which is less straightforward to overcome. Identification of such parameters can be done through feature analysis but has not yet been incorporated into the CHART-S3 ODAs.

As a proof-of-concept, the CHART-S3 kNN ODA was modified to incorporate a simple de-trending method to compensate predictable aging effects of temperatures. When an aging effect is predicted, the tool uses the available data to calculate the magnitude of the asserted effect and applies a compensating factor prior to normalisation according to the difference in dates of the test and training data. The results of this modification are shown in Figure 13: the top row of scatter plots show the output of the kNN tool prior to de-trending, with all data returned as clear outliers as

expected. The bottom row of scatter plots shows the output of the kNN tool after de-trending, with results far more consistent with the test data.

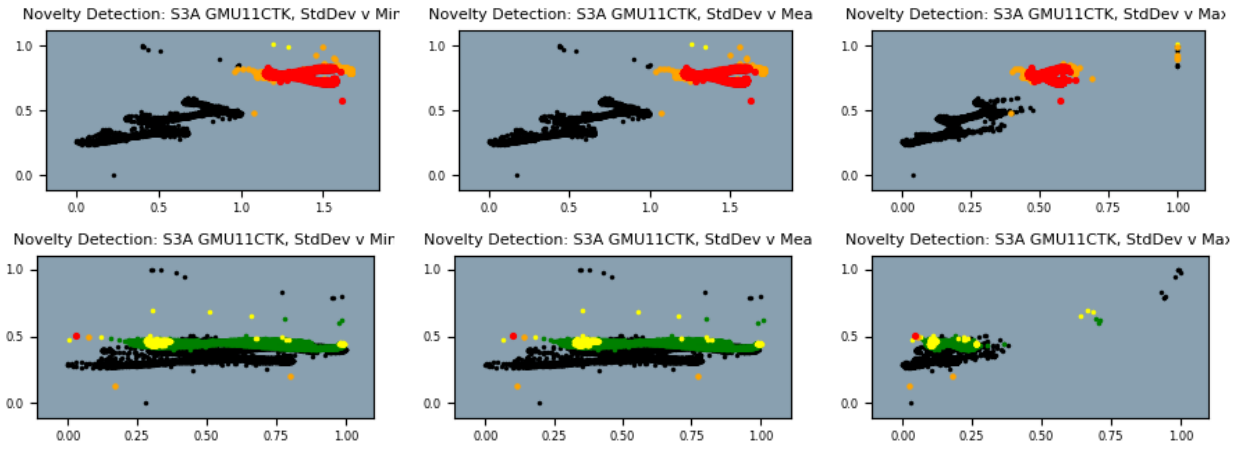


Figure 13 Categorised Outlier Detection Outputs For Thermal Data Before (Top) And After (Bottom) De-Trending

The detailed results are presented in Table 1. One of the mechanisms in place within the tool to prevent noisy reporting of minor deviations from the training data is the definition of thresholds defined for reporting the existing of outliers according to the categorisation level [Losco, 2021]. These thresholds are defined at 30% of the total number of datapoints in the test dataset for yellow, and 5% for orange. Neither of these thresholds are reached after de-trending. However, the presence of the singular red outlier would still trigger an escalation to the FCT for further analysis. This demonstrates that even relatively simple methods that are not computationally intensive can significantly reduce the paranoia score of the algorithm, through the incorporation of established engineering expectations based on the type of parameter being monitored.

Table 1 Comparison of Outlier/Inlier Classification With and Without De-trending

	Before De-trending	With De-trending
Inlier / Nominal	0	4737
Outlier – Yellow	2	457
Outlier – Orange	1229	3
Outlier – Red	3967	1
Total # outliers (%)	5198 (100%)	461 (8.87%)

For the long-term usage on most parameters, the simplified de-trending model demonstrated above would not be sufficient.. For long-term monitoring, a more accurate forecasting model is desired that can be applied in a generic way to multiple parameters. A similar shape as the plot in Figure 12 can be observed for many LEO spacecrafts' temperatures.In this plot, the most obvious and significant properties are:

- A logarithmic or root-shaped aging function.
- An 'annual' sinus-shaped function.
- Another sinus-shaped function over the solar cycle (~ eleven years).
- An offset.

There are many other effects (but with only minor impact) being involved in the overall shape (such as orbital day/eclipse cycle, orbital repeat cycle, etc) which are not considered yet.

An attempt was made to create a coarse forecasting model using these mathematical observations as a baseline and applying a heuristic gradient-descent method over the entire test-dataset to find a suitable fit for a synthetic function to describe the plot in Figure 12. The aging function part (here a root function) has an inclination and an order, the two sinus functions both have an amplitude, a frequency, and a shift. These terms, together with the offset are combined by an addition:

$$f(t) \sim F_{aging} \cdot \left(t^{\frac{1}{a}}\right) + F_{annual} \cdot \sin\left(\frac{2\pi t}{\lambda_{annual}} + \delta_{annual}\right) + F_{solar} \cdot \sin\left(\frac{2\pi t}{\lambda_{solar}} + \delta_{solar}\right) + i \quad (1)$$

Where:

t = time
F = scaling factors (aging, annual, solar)
λ = periods (annual, solar)
δ = tuneable offset
i = intercept at t=0
a = aging factor

The gradient-descent algorithm manipulates each parameter (one at a time) and assesses the fit by determining the Mean Square Error (MSE) between the prediction and the actual data. According to the outcome, the algorithm will either continue to manipulate the parameter, revert the last manipulation and try again with a littler change, or proceed to the next parameter of the equation, when no better fit can be achieved for now. Running this over multiple epochs (=multiple full cycles over all parameters), an MSE of <0.5 converged. Leaving out the launch- and early operations phase, the fit is even better (< 0.25).

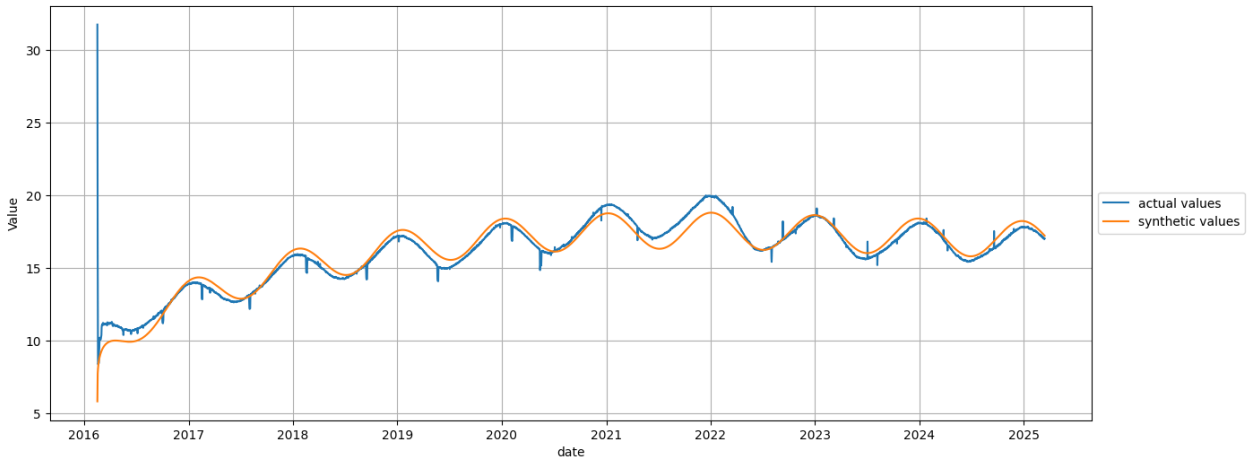


Figure 14 Actual values vs synthetic values

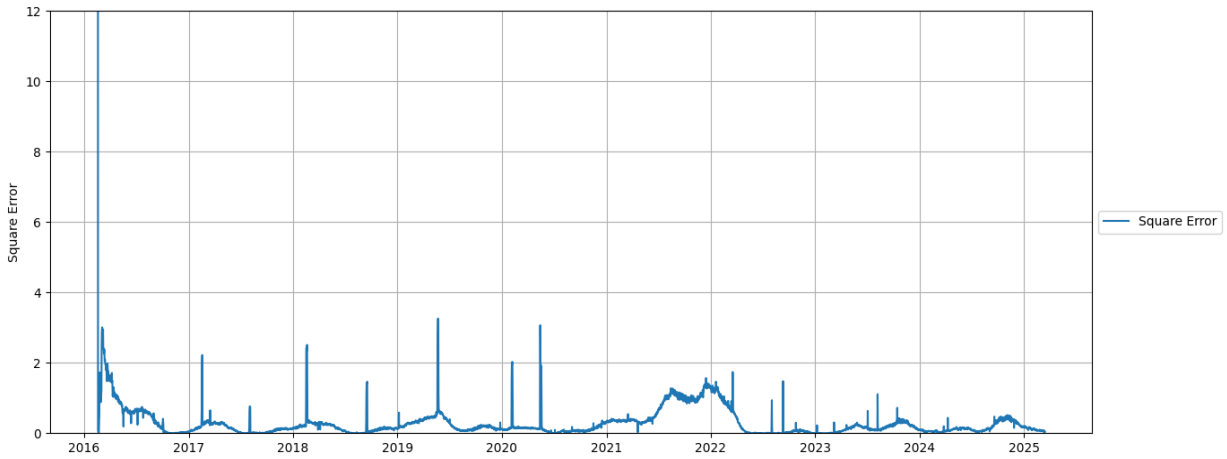


Figure 15 Square Error between actual- and synthetic values

The baseline formula, together with the parameter values of the best fit may be used as a forecasting model. In the context of outlier detection, analogue to the linear de-trending, this more complex model can be used instead.

For the proof-of-concept, the training data had been defined for the period 1.1.2017 to 1.1.2018, while the test dataset was defined for the period 1.1.2024 until 1.4.2024. The value ranges (avg) of these two periods did not overlap.

Without detrending, 1290 red outliers have been detected (what is the entire test dataset), whereas de-trending with the function reduced the number of red outliers to just one which was confirmed to be a true positive detection.

4.3 Introduction of Engineering Judgement

Figure 16 illustrates the output of the kNN tool using one week of measurements (100 data points/orbits) of a regulated voltage, which is expected to be generally stable and relatively discrete. With more than 30% of data points flagged as yellow outliers, the week was identified as sufficiently novel for the tool to escalate it to the attention of the engineering team. Visual examination of the scatter plots allows for a fast understanding: all values are within the nominal range, but many of the average values are lower than in the training data. This is the same parameter Figure 9 discussed previously, but examining a different time range. These outliers are all *paranoid detections*: although they clearly fall outside of the range of the training data, it is not concerning to an experienced engineer with knowledge of the parameter being monitored.

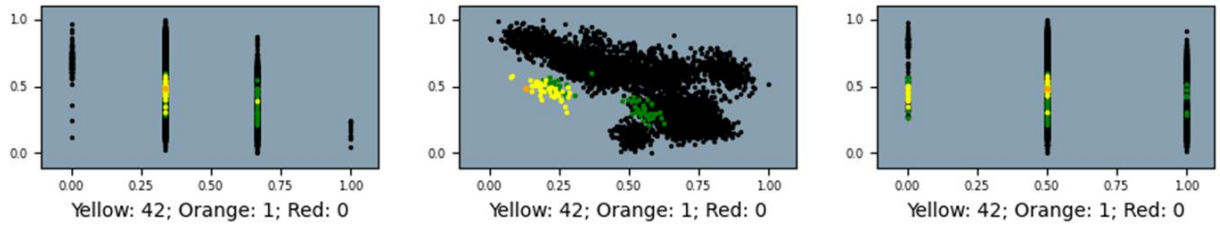


Figure 16 Series of scatter plots (std Vs min, mean, max respectively) for one week of kNN analysis of a regulated voltage on Sentinel-3B. Training data in black, nominal test data in green, outliers in yellow/orange by severity.

An interesting aspect of this particular week is that the minimum, mean and maximum values reported are all within the range of values seen within the training data set, but they are flagged because the std is lower for the reported mean, or the mean is lower for the reported std. Examination of the full training data set (before reduction to orbital stats: 3,952,000 data points) confirms that the data is very discrete, with just ten distinct values being reported in the training dataset (and nine in the this test dataset) and just two values making up ~70% of the training dataset and ~77% of the test dataset. The distribution of the training data is shown as a histogram in Figure 17, which also illustrates the level of variability in this distribution through the year.

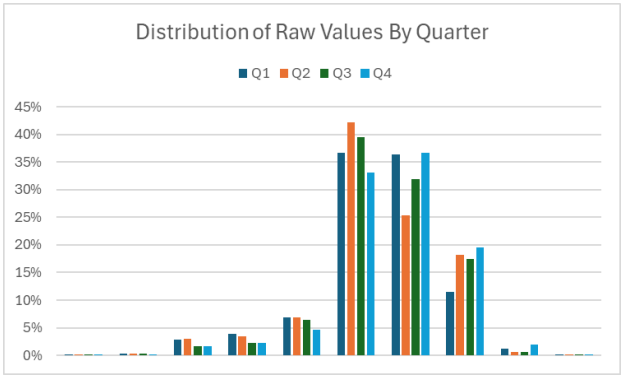


Figure 17 Histogram of raw values in the training data

So, this raises the questions: how many datapoints need to be lower, and by how much, to match the observations within the week highlighted? And can we proof the tool against such minor fluctuations in behaviour? Closer examination of the data shows that the test data flagged as yellow outliers was, on average, reporting values just 1 bit lower for 3.6% of the orbit (~27 data points) compared to orbits identified as inliers. The singular orbit flagged as orange similarly contained mean values that were just 1 bit lower for 4% of the orbit (~30 data points). As the min and max values reported were not at the extremes of the test (or training) data, it can be concluded that the variation came from fewer reported high values i.e. the behaviour was more stable than the training data.

For this parameter, one bit corresponds to just 0.25mV and such variance is not unexpected. Determining the probability distribution of the full 1-D dataset for the period of the training data (1 year) and running a simple Montecarlo simulation to generate the equivalent of one week of data and calculating the corresponding 4-D orbital statistics gives results which match the most central cluster of points in the training data as expected. This is illustrated by the orange dots in Figure 18. It is important to note that because this analysis uses the probability distribution over a full year, any clustering effects within orbits or seasons is absent. This results in the simulated data being far more

tightly clustered around the centre of the true distribution than the training data. This approach could be modified to use smaller subsets of the training data and run sequential simulations, but this was not considered necessary for our purpose.

Modifying the simulation to regress a percentage of the values towards the mean obviously does not change the mean but reduces the std – resulting in a cluster moved directly downwards in the scatter plot (purple) consistent with a subset of both the training and test data sets. Increasing the magnitude of this regression would result the simulation producing values matching the lower-most cluster. Modifying the regression such that it uses a value a 1 bit lower than the average is shown by the green dots, which matches very closely the observed test data. Note that using other values in the training data can be simulated by slightly increasing the level of variation instead of reducing it, by using a value 1 bit higher instead of lower, or a combination of these.

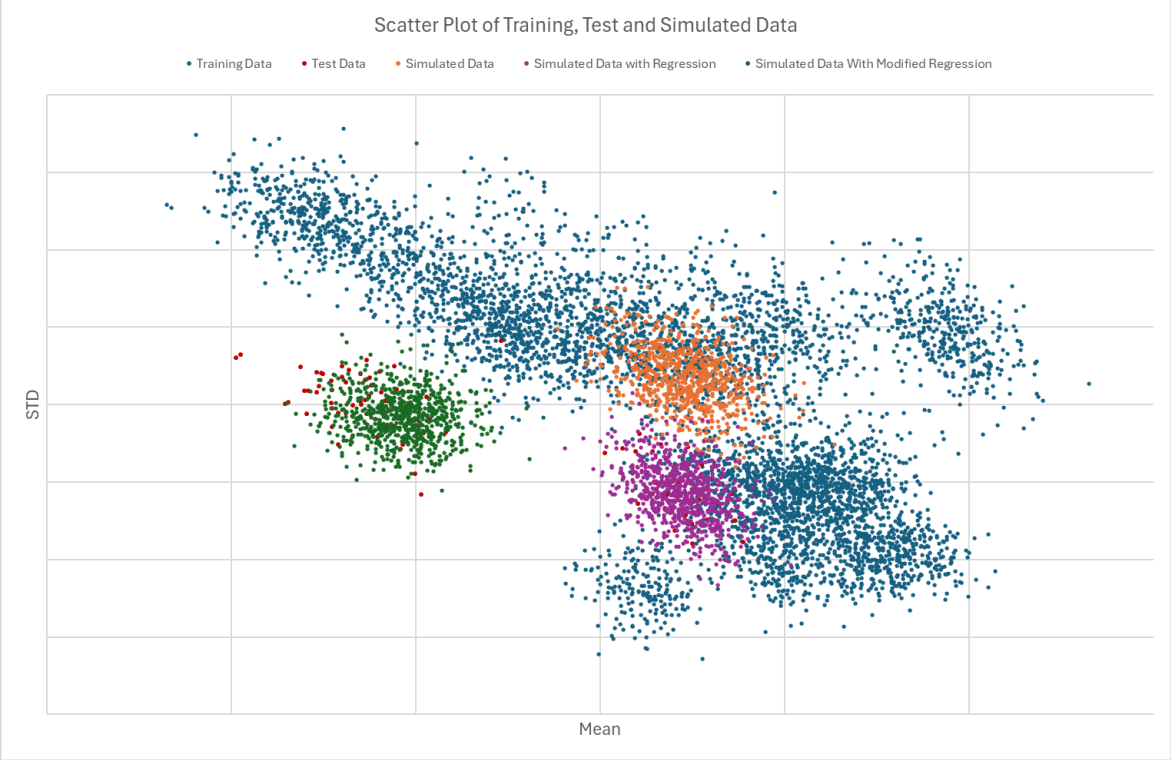


Figure 18 Scatter plot (std Vs mean) of the training (blue) and test (red) datasets, overlaid with one week of simulated data using the probability distribution of the training data (orange), regressed to the mean (purple) and regressed to a truncated (integer) value of the mean offset by one (green).

Note that the "downwards" slope of the training data is an artefact caused by orbits with varying (increased) levels of divergence from this median value in the training data. This can also be simulated through the same methodology, but was not done for aesthetic reasons and in this scenario is not necessary.

Just four of the original 43 outliers would still be flagged as outliers, if this one week of simulated data were added to the training set, which is a significant improvement and well below the defined escalation threshold. This demonstrates that augmenting the training data with simulated data using expecting levels of variation can provide a useful and consistent result with higher accuracy and precision, and fewer unwanted (false positive) detections due to incomplete training data.

The value of a tool for monitoring satellite behaviour that cannot be used until a significant amount of in-orbit time (to collect training data) has passed is reduced in proportion to the amount of time needed. In addition to reducing the FP and paranoid TP rates, the inclusion of simulated data helps to minimise the time needed for an outlier detection tool to be considered operationally ready.

4.4 Additional Dimensionality

All outlier detection methods on satellite HKTM currently investigated at EUMETSAT rely on the same inputs: the 4-dimensional orbital statistics. However, other information is available that could potentially be used as additional dimensions for the analysis. Due to seasonal patterns, the dataset can consist of a broad range of values. One hypothesis that was explored during this work was that the rate of change of orbital values should be slow: values may be expected to raise or lower over a period of months but should not jump quickly from one region of the training data to another. i.e. when the test data falls within the lower phase of a seasonal cycle and has a sudden spike or change in behaviour upwards, the tool might fail to detect this anomaly as the spike remains within the overall range of the training data.

To utilise this hypothesis, a fifth dimension was added to the kNN algorithm using the difference in reported mean value from the previous orbit, or "average difference". Adding this dimension was found to increase the sensitivity of the algorithm significantly, but as this also resulted in a larger number of paranoid detections it was decided not to utilise this concept until further work has been completed in reducing the quantity of such detections. The tool has been updated such that additional dimensions like this can be turned on/off with a simple configuration change.

Additional concepts for different dimensions are still being investigated and evaluated by the team.

4.5 Multivariate Analysis – Detecting Micro-meteorite Or Debris Impacts

All discussion previously in this paper has focussed on univariate analysis (despite the analysis being done using representations of the single variable in 4 or more dimensions), but ODAs are also capable of performing multivariate analysis. This can be more difficult to configure and use, but it is a powerful tool that can help distinguish between the signatures of routine operations, spikes, anomalies and different anomalies and is less susceptible to noise in individual parameters.

The Sentinel-3 reaction wheels occasionally exhibit spikes or short periods of increased friction. This is nominal behaviour. But when multiple AOCS parameters report unusual values at the same time, it could be indicative of something else: such as a micro-meteorite or debris impact.

Small impacts are traditionally detected through monitoring of individual parameters e.g. the Kalman filter. To avoid reliance on significant deviations in a single parameter to flag outliers, an alternative form of multivariate analysis of AOCS parameters was implemented to monitor for possible micro-meteorite or debris impacts by using majority voting of the input parameters. A potential micro-meteorite or debris impact will be shown as an event raised in CHART, as illustrated in Figure 19. The first column shows the orbit timestamp of when the impact was detected (not the exact moment of impact as the outlier detection tool uses the orbital stats). The second column gives a description of the event, and the third shows the "colour" categorisation of the outlier which is given using a majority voting system. The final column indicates which parameters are influenced by the potential impact.

These events can be used to trigger the automatic generation of a CHART report containing detailed plots, statistics tables and background information of the affected AOCS parameters, facilitating the investigations of the FCT. A further evolution of this feature will be to use the outlier detection output to trigger a more precise automated examination of the data, to determine the exact moment of impact by implementing a spike detection system using all the data points over the time range of the impacted orbit.

Orbit Start Time	Description	Colour	Parameters influenced
2019-08-27 12:47	Potential micro-debris impact	red	AAS1033A,AAS3100A,AAS3100R,AAS3200A,AAS3200R,AAS3300A,AAS3300R,AAS3451T
2021-02-03 13:51	Potential micro-debris impact	red	AAS1033A,AAS3100A,AAS3100R,AAS3200A,AAS3200R,AAS3300A,AAS3300R
2021-06-09 19:28	Potential micro-debris impact	orange	AAS1033A,AAS3100A,AAS3100R,AAS3200R
2022-01-19 07:33	Potential micro-debris impact	red	AAS1033A,AAS3100R,AAS3200R,AAS3300A,AAS3300R,AAS3451T

Figure 19 Examples of events raised in CHART for potential micrometeorite or debris impacts.

In this case study, the novelty detection with focus on potential micro debris impacts has been executed over almost the entire mission time (from Jan 2017 until March 2025), leaving out the launch and early operations. The kNN algorithm uses k=20 and analyses the 4-dimensional vector space of [min, max, mean, and std] of statistics of the HKTM parameter set of each orbit throughout the entire period.

An impact by micro-meteorite or debris is expected to cause simultaneous outliers in the behaviour of multiple AOCS parameters together. Outliers during manoeuvring of the spacecraft shall not be considered, thus the spacecrafts' status (AOCS mode) was set as 'Influencer' of the values being monitored. Further, the corresponding values of the

parameter set during known activities on the spacecraft must be discarded, which was achieved using the list of known activities via the SATCAT's API as discussed in §4.1 previously.

For this campaign, annual hypertext-based reports were generated for each year since launch of the respective satellites and analysed by the FCT. Each report contains:

- Plots of Euclidean Distance to the nearest neighbour in the training set found over time for each parameter, including thresholds for the categories yellow, orange, and red, as well as those candidates masked out by SATCAT over reporting period.
- Table of outliers discovered for each parameter over reporting period by category.
- Table of masked events (from SATCAT) for each parameter over reporting period.
- Table of correlations of outliers detected on two or more parameters over reporting period.
- Cumulative plot (voting plot) of outliers over reporting period.

For this analysis, low counts of votes are ignored. To achieve a high reliability and robustness, only higher votes were considered (with a tie-breaker quorum of more than 50 %: here 7 of 13 parameters). The tie-breaker criteria were reached just a few times over the entire mission and validated against the list of known impacts identified during annual satellite performance reviews.

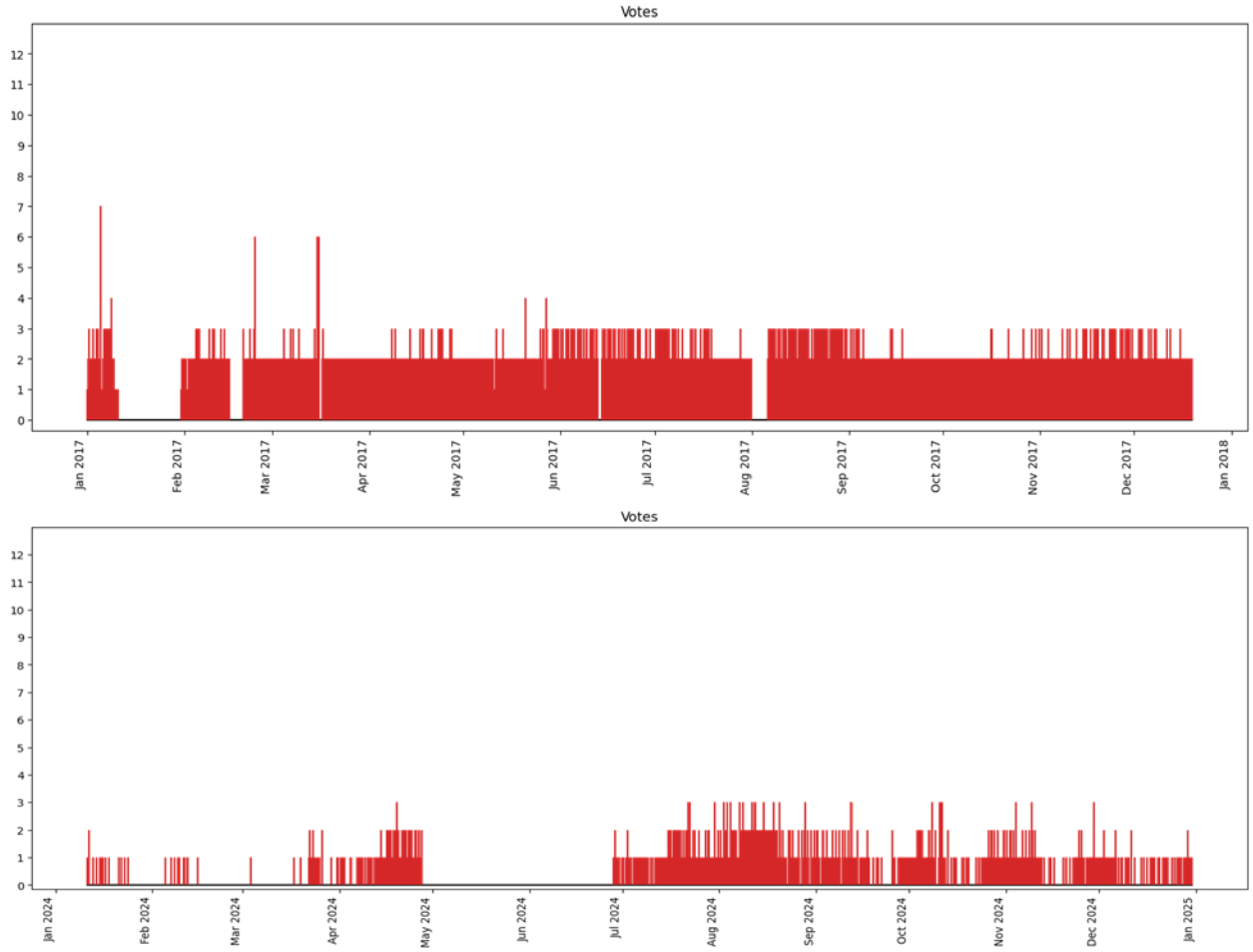


Figure 20 Cumulative (voting) plots of outliers for the years 2017 (top) and 2024 (bottom)

The cumulative (or voting) plots shown in Figure 20 show how the distribution of detected outliers changed over the first and last full year of the Sentinel-3A mission versus the number of parameters which are affected, free of known activities or manoeuvres. During the years 2017, 2018 and partially 2019, two almost constantly persisting outliers were flagged. The root cause of these outlier series is known, but the training dataset does not consider these

behaviours as they are no longer applicable to the flying mission today. Including these behaviours to the training set would mitigate the detection during the first years but would prohibit the detection of potential later reoccurrences which is not desirable given the purpose of the tool. This highlights an important additional factor when maintaining training data sets – behaviour that was nominal in the past is not always to be expected in the future.

The cumulative plots also revealed accumulations of outliers around certain times of the year. These accumulations might be a result from a non-optimal training set and being related to seasonal temperature fluctuation that could not be trained to the same extend as other parameters. Not all training sets covered all seasons equally well, with different time periods defined for each parameter.

In the period April 2019 – June 2024, five suspected minor impacts were recorded as having occurred across the two satellites. The tool implemented successfully detected all five of these events, plus nine additional candidate events. For each of these candidates, the peaks observed in the individual parameters was much lower than for the five previously identified events. However, these detections successfully identified cases in which multiple parameters exhibited a change in behaviour at the same time with a signature matching the known cases but with smaller magnitude that can be masked when looking only at single parameters with significant levels of noise.

5. Integration Into the Routine Operations Concept

Novelty detection methods offer many advantages compared to traditional limit-based monitoring of HKTM, most notably that they can detect anomalous changes in parameter behaviour before the fixed limits are violated (as highlighted by the case study discussed in section 3.3). For this reason, they can be a very useful tool for engineers to monitor spacecraft health alongside traditional methods. In this section, we assess the various options that are available for integrating outlier detection tools into the routine operations concepts of small constellation satellite missions.

The benefits of integrating novelty detection analysis with CHART are manifold, e.g. allowing for seamless integration with existing reports or enabling use of CHART functionality such as ad-hoc plotting, or widgets to present the results in reports.

Integration with CHART can be achieved either by performing the analysis within CHART itself, or by ingesting the results of analysis performed elsewhere. Both approaches have been followed for this paper – the kNN analysis described in section 1.3 is performed within a CHART widget to populate a dedicated report, while the ANN algorithm described in section 1.4 is run as part of the EUMETSAT Machine Learning Framework described in section 1.2. The outputs of the algorithm are automatically ingested into the CHART database.

However, ingesting the data into CHART is only a precursor to integration into the routine ops concept.

5.1 Timeliness of Monitoring Concepts

Typical monitoring concepts for small constellation satellite missions can be divided into 4 categories, illustrated in Figure 21. Used for urgent responses to critical events, Real-Time monitoring is typically performed by the satellite itself, i.e. failure detection, isolation and recovery (FDIR). The need for immediate action is not compatible with the latencies introduced by relying on ground monitoring. Although on-board outlier algorithms are widely discussed and have been used for payload data [12-16], and some advocate for using it to augment FDIR concepts [17,18], these considerations are outside the scope of this paper.

Real-Time Monitoring
Immediate or short-term response needed for satellite safety.
Performed by spacecraft itself.

Near Real-Time Monitoring
Monitoring by ground systems for intervention (telecommanding).
Minimum reaction time determined by orbital constraints, may be several hours

Short-Term Monitoring
Analysis of trends and behaviour changes of potential concern.
Daily / weekly / monthly

Long-Term Monitoring
Analysis of long-term trends and mission performance
Quarterly / annual.

Figure 21 Typical Spacecraft Monitoring Responsibilities By Timeliness

Near real-time (NRT) monitoring includes less critical responses, such as secondary responses to the actions of the FDIR. For example, if a temperature gets too high a payload unit may be switched off by the FDIR, and the NRT monitoring would detect the corresponding mode change events, and any limit violations associated with the unit status. This is typically the monitoring performed in missions operations control rooms and a reaction time in the order of minutes to hours may be expected. It is vital that people or tools working in this domain are not confused by noisy or superfluous alarms and detections, such as those discussed in §2 previously, and so this is also not a suitable domain for the use of outlier detection at this time.

Short-term monitoring is used to analyse and detect trends and behaviour changes of potential concern, with a lower level of criticality. EUMETSAT FCTs use CHART extensively in this context, particularly the automated report generation functionality. To integrate outlier detection in this monitoring category, we can consider three options: namely a daily, weekly or monthly report, each having unique advantages and disadvantages. A higher frequency allows for faster detection of changes in behaviour and helps to identify anomalies or unusual behaviour more quickly. However, this increases the workload for the team if it contains any number of false positives or paranoid detections, and a constant level of false positives at this frequency introduces a significant risk of triggering a drop of focus of the ground team. On top of this, as described above in §3.3, due to the high level of sensitivity of the algorithms it is very possible that significantly more than a day will be needed for the operations team to recognise a change in behaviour properly. A lower frequency lowers the workload for the team and makes it possible to properly detect a change in behaviour. The trade-off is that this increases the response time and makes it slower to identify issues. A balance between these needs to be found and thus a weekly report has been selected for some analysis, with further parameters assessed monthly. Select high-value monitoring, such as checking for debris impacts, may be done on an orbital or daily basis.

Long-term (quarterly, bi-annual or annual) analysis is performed to assess long-term trends with potential impact on mission performance or mission lifetime. No rationale was found so far to consider exclusively using this frequency for outlier detection for any HKTM parameters or other related data. However, this time frame is suitable for assessment of trends in the outlier scores as discussed in §3.1 or for visualising and assessing the frequency of detections, and whether any retraining is needed.

5.2 Importance of Visualisation

Creating a working tool is only a first step in the process of integration into an operations concept - the outputs of the tool must be made available to the engineers in a meaningful and useful way, without introducing significant

additional workload. Ideally, the outputs should be presented with a high-level summary to provide a quick overview of the results, while offering the ability to quickly navigate the data for more detailed analysis.

The first element of the reports is a high-level summary table as shown in Figure 22. This gives a clear overview of all the parameters included in the report and the total number of outliers detected for each one. The parameter names are hyperlinks linking to the relevant plots. When the same data point is marked as an outlier for three or more parameters, the relevant table cells will be shown in bold with a grey background colour - this highlights cases when multiple parameters are flagged as outliers for the same orbit, which may imply a common root cause. The timestamps of these data points, as well as the number of parameters being flagged as an outlier at that time, are shown below the table. Note that this is an evolution of the summary table presented by Losco et al [3], with the highlighting of correlated outliers added as a new feature to support analysis.

Outliers Summary

Parameter	Yellow	Orange	Red
AAS2550Z	0	0	2
AAS01MTT	0	0	0
AAS02MTT	0	0	0
AAS03MTT	1	0	0
AAS1033A	0	0	**1**
AAS1136T	0	0	**1**
AAS1236T	1	0	0
AAS1336T	3	0	0
AAS3100A	0	0	**1**
AAS3100R	0	0	**1**
AAS3151T	0	0	0
AAS3200A	0	0	0
AAS3200R	**1**	0	0
AAS3251T	0	0	0
AAS3300A	0	0	0
AAS3300R	0	0	0
AAS3351T	0	0	0
AAS3451T	0	0	0

The data point **2024-06-09 19:28:12** is flagged for **five** parameters

Figure 22 Example of CHART-S3 Outlier Report Summary Table

To reduce the amount of unnecessary information shown in the report, not all parameters are included in any single report but are instead distributed according to unit and importance (i.e. some parameters evaluated weekly, others monthly). Further, the plots for each parameter are added in subsections which for aesthetics and usability will be automatically collapsed if no outliers were detected for the parameter (see Figure 23). This aims to strike a balance between completeness (by still including all results in the report) and brevity (by ensuring that only plots of interest are displayed by default), minimising visual overload and simplifying navigation of the report.

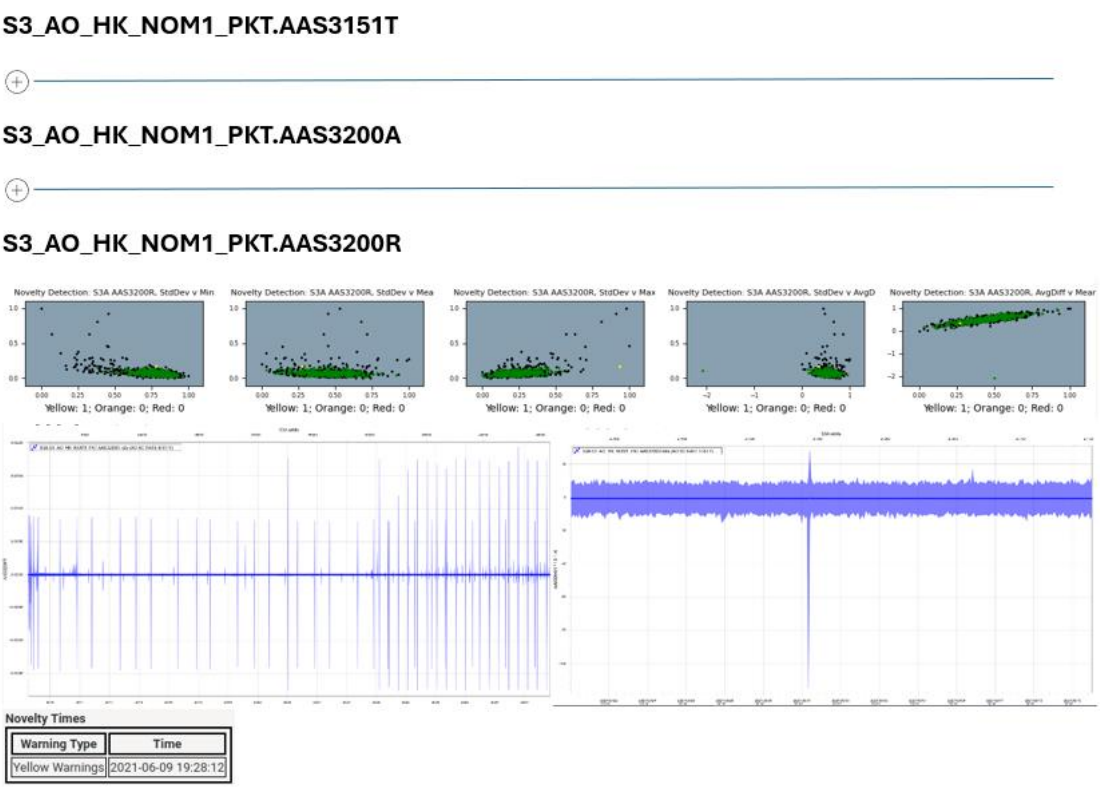


Figure 23 Example of Expanded and Minimised Parameters Within Detailed Outlier Tool Report

Another idea explored is to display in conjunction the corresponding time-series data and scatter plot of outlier distance score as shown in Figure 10. Here the training data is plotted in blue, test data in green, yellow, orange or red (depending on outlier status), data points filtered out due to operations in grey, and data that is not part of the training or test datasets in black. Compared to the scatter plots currently used for this analysis (as in Figure 7), this has the advantage of illustrating trends over time and facilitates the identification of which datapoints within the time-series data have been flagged as outliers.

The micro debris impact case study (§4.5) underlined the importance of visualisation, as it was found that lists and tables of detections grew massively with the number of parameters and the duration of the reporting period. In the scope of the campaign, the visualisations within the generated reports were iteratively fine-tuned to maintain clarity and allow easy identification of correlated outliers.

As an example, below is an excerpt of the table of correlations (Voting List), obtained as output from the micro debris impact analysis for 2017. The number of detected novelties involving at least two parameters over the year was quite immense - the excerpt shown below covers just the first week of the year.

Time	Followers
2017-01-01 05:59	AAS3100A,AAS3351T
2017-01-01 12:43	AAS3300A,AAS3300R,AAS3351T
2017-01-01 16:05	AAS3300A,AAS3351T
2017-01-02 02:11	AAS3200R,AAS3351T
2017-01-02 03:52	AAS3200R,AAS3351T
2017-01-02 10:36	AAS3200R,AAS3351T
2017-01-02 22:23	AAS3300A,AAS3351T
2017-01-03 00:04	AAS3200R,AAS3351T,AAS3451T
2017-01-03 08:29	AAS3200R,AAS3351T
2017-01-03 13:32	AAS3351T,AAS3451T
2017-01-03 15:13	AAS3351T,AAS3451T
2017-01-03 16:54	AAS3351T,AAS3451T
2017-01-03 21:57	AAS3200A,AAS3200R,AAS3351T
2017-01-03 23:38	AAS3100A,AAS3351T
2017-01-04 03:00	AAS3200R,AAS3351T
2017-01-04 11:25	AAS3200R,AAS3300A,AAS3351T
2017-01-04 16:28	AAS3200R,AAS3351T
2017-01-05 00:53	AAS3200R,AAS3351T
2017-01-05 02:34	AAS3351T,AAS3451T
2017-01-05 04:15	AAS3351T,AAS3451T
2017-01-05 05:56	AAS3100A,AAS3100R,AAS3200A,AAS3200R,AAS3300A,AAS3351T,AAS3451T
2017-01-05 07:37	AAS3200R,AAS3351T
2017-01-05 09:18	AAS3300R,AAS3351T
2017-01-05 22:46	AAS3300A,AAS3351T
2017-01-06 00:27	AAS3200A,AAS3351T
2017-01-06 05:30	AAS3300A,AAS3300R,AAS3351T
2017-01-06 08:52	AAS3200R,AAS3351T
2017-01-06 12:14	AAS3300A,AAS3300R,AAS3351T
2017-01-06 17:16	AAS3300A,AAS3351T
2017-01-06 18:57	AAS3200R,AAS3351T
2017-01-06 20:38	AAS3200A,AAS3300R,AAS3351T
2017-01-06 22:19	AAS3200R,AAS3351T
2017-01-07 06:44	AAS3300R,AAS3351T
2017-01-07 11:47	AAS3300A,AAS3300R,AAS3351T
2017-01-07 13:28	AAS3300A,AAS3300R,AAS3351T
2017-01-07 15:09	AAS3200R,AAS3351T

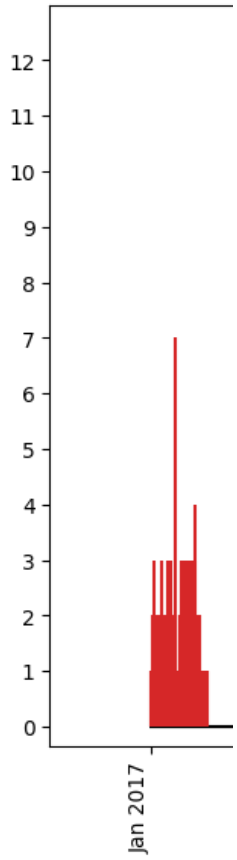


Figure 24 Example of the table of correlations (Voting List) (left) and corresponding cumulative plot (right)

The cumulative plot offers a more intuitive visualisation - clearly visible here is the large spike in the first week of January, which corresponds to a detection involving 7 parameters. Upon noticing this, the user can consult the table to find the exact timestamp, list of parameters involved, and a link to the CHART plotting tool for further analysis.

5.3 Maintenance of Training Data

The maintenance of training data is a critical task for accurate results, but the workload associated with doing so must be minimised for use operationally. Firstly, the training data needs to be well defined accordingly to the expectations of the tool. Behaviour which could be marked as outliers from a mathematical point of view but is considered nominal from an engineering perspective, should be included in the training data. Due to aging and seasonal effect, it might be required to retrain the tool often over the years, which increases the workload. To minimise this, we have demonstrated that a combination of de-trending and simulated data can be very beneficial. Having a GUI to maintain the time range of the training data has also been suggested previously [2] but has not yet been implemented and would also be highly advantageous.

6. Using Sledgehammers To Crack Walnuts

There is no question about the power and potential usefulness of ML applications for spacecraft operations monitoring on ground, and potentially in orbit. However, there may be many opportunities to take the lessons learned from using these tools and apply those, without incurring the computational cost of performing ML itself. For example, the Packet Utilisation Standard (PUS service 4) allows for the creation of orbital statistics onboard. Instead of feeding those statistics into a ML algorithm, application of the basic high/low limits to those statistics would enable greater

flexibility in onboard monitoring onboard PUS satellites. For example, the temperature highlighted previously in Figure 12 contains many spikes above 25°C, an average that remains below 20°C and a typical orbital variation of ~2°C. Applying limits to the orbital statistics for the average and standard deviation would potentially increase the level of flexibility available to FDIR concepts, gaining much of the benefits of ML but with vastly reduced computational cost.

7. Conclusions

Two outlier detection algorithms (ODAs) with different user interfaces and functionality have been assessed for supporting the routine operations monitoring of the Sentinel-3 satellites. This work has highlighted the benefits of storing the numeric output (outlier distance score) of the ODA, for plotting alongside the original data but note that this does not exclude the need for other methods of visualisation such as scatter plots.

The definition of the training dataset is of fundamental importance, and the methodology used for determining suitable training data may vary for different algorithms as using the same dataset may not lead to the same results. Training of the users defining and/or maintaining the training dataset is therefore important, and must be tuned to reflect the algorithm used.

Once deployed for use, tweaking the settings of the ODA disturbs the long-term trend of outlier distance scores. This should be done as infrequently as possible, and the historical HKTM reprocessed by the ODA afterwards.

HKTM being flagged as outliers by highly sensitive algorithms is not sufficient, it is vital that the engineering team be able to understand why a parameter was flagged as an outlier. This may favour more simplistic algorithms whose outputs are directly understandable and reproducible by engineering teams but regardless requires the availability of multiple visualisation methods.

Introducing detrending and simulated data based on ground expectations and understanding of satellite behaviour can be used to avoid the desensitisation of FCT members to noisy (low precision) or "paranoid" detections, which in turn can lead to misclassification of TP detections as FP.

Supplementing HKTM by introducing ground knowledge from other data sources and tools greatly enhances the ability of outlier detection tools to correctly distinguish between intentional satellite operations and anomalies that exhibit similar signatures.

The sensitivity of an ODAs for satellite HKTM can be increased through the inclusion of additional dimensions to the traditional 4-dimensional orbital statistics.

There may be scope for applying lessons learned from use of ML algorithms to spacecraft operations, without incurring the computation cost of the ML algorithms themselves.

Acknowledgements

The authors would like to extend their gratitude to the EUMETSAT Sentinel-3 flight control team, for their patience and support in performing the assessment of the tools used during this work. We would also like to thank Michele Burla, Mike Elson and Paul Raval for their support in developing and deploying the kNN ODA, and to Richard Dyer, Pio Losco and the countless others whose shoulders we stood on to achieve the current level of maturity of the tool. And similarly to Ruth Britton, Luca Garegnani and the rest of the EUMETSAT MLF team for their work in creating and maintaining the ANN ODA. Finally, we would like to thank TU Delft for providing excellent interns to EUMETSAT, without whom our progress on these tools would have been drastically slower.

References

[1] J. Murphy, J. E. Ward, B. Mac Namee., An Overview of Machine Learning Techniques for Onboard Anomaly Detection in Satellite Telemetry, IEEE, 2023 European Data Handling & Data Processing Conference (EDHPC), Juan-Les-Pins, France, 2023, 1-6.

[2] E. Trollope, R. Dyer, T. Francisco, J. Miller, M. P. Griso, A. Argemandy, Automated techniques for routine monitoring and contingency detection of LEO spacecraft operations, in: H. Pasquier, C. A. Cruzen, M. Schmidhuber, Y. H. Lee (Eds.), Space Operations: Inspiring Humankind's Future, Springer, 2019, 413-437.

[3] P. L. Losco, A. De Vincenzis, J. Pergoli, R. Dyer, From Theory to Practice: Operational Implementation of Telemetry Outlier Detection at EUMETSAT, in: C. Cruzen, M. Schmidhuber, Y. H. Lee (Eds.), Space Operations Beyond Boundaries to Human Endeavours, 2022, 235-259.

[4] G. Galet, DATA MINING: Using machine learning for spacecraft housekeeping purpose, SpaceOps Workshop 2017, AIAA, Moscow, Russia, 2017.
[5] J. A. Martínez-Heras, A. Donati, M. G. Kirsch, F. Schmidt, New telemetry monitoring paradigm with novelty detection, 14th International Conference on Space Operations, AIAA, Stockholm, Sweden, 2012, 11-15.
[6] C. O'Meara, L. Schlagy, L. Faltenbacher, M. Wicklerz, ATHMoS: Automated Telemetry Health Monitoring System at GSOC using Outlier Detection and Supervised Machine Learning, 16th International Conference on Space Operations, AIAA, Daejeon, Korea, 2016, p. 2347.
[7] Serpell, E., Miller, J., Collins, P., Theurich, M. METOP–A PLM High Frequency Telemetry Acquisition Operations, AIAA Paper 2008-3421, May 2008.
[8] Schulster, J., Evill, R., Phillips, S., Feldmann, N., Rogissart, J., Dyer, R. and Argemandy, A., CHARTing the Future – An offline data analysis and reporting toolkit to support automated decision-making in flight-operations, 15th International Conference on Space Operations, Marseille, France, 2018.
[9] G. Casonato, R. Britton, L. Garegnani, EUMESTAT Machine Learning Framework System and AI/ML Applications, 18th International Conference on Space Operations, Montreal, Canada, 2025. (submitted for publication).
[10] Y. Dodge, Mahalanobis Distance. In: The Concise Encyclopedia of Statistics. Springer, New York, 2008. pp 325–326.
[11] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey. *ACM computing surveys (CSUR)*, *41*.3 (2009) 1-58.
[12] G. Mateo-Garcia, J. Veitch-Michaelis, C. Purcell, N. Longepe, S. Reid, A. Anlind, F. Bruhn, J. Parr, P.P. Mathieu., In-orbit demonstration of a re-trainable machine learning payload for processing optical imagery, Scientific Reports 13.1 (2023) 10391.
[13] F. Ales, A. Krstova, T. Chabot, M. Ghiglione, M.C. de Lera, F. Gegwein, A. Koch, C. H. Garcia, P. Harikrishnan, M. Mallah, R. Ali, M. Rothe, L. Hili, Edge AI Solutions for Spacecraft Failure Management, In: Y. H. Lee, A. Schmidt, E. Trollope (Eds.), Space Operations Invest in Space to Serve Earth and Beyond, 2025, 461-474.
[14] S. Kacker, A. Meredith, K. Cahoy., Machine Learning Image Processing Algorithms Onboard OPS-SAT, 36th Annual AIAA/USU Conference on Small Satellites, Logan, 2022.
[15] G. Labrèche, D. Evans, D. Marszk, T. Mladenov, V. Shiradhonkar, T. Soto, V. Zelenevskiy, OPS-SAT spacecraft autonomy with TensorFlow lite, unsupervised learning, and online machine learning. In *2022 IEEE Aerospace Conference (AERO)*, IEEE, *2022*, 1-17.
[16] G. Giuffrida, L. Fanucci, G. Meoni, M. Batič, L. Buckley, A. Dunne, J. Aschbacher, The Φ-Sat-1 mission: The first on-board deep neural network demonstrator for satellite earth observation. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 2021, 1-14.
[17] J. Murphy, J.E. Ward, B. MacNamee, Machine Learning in Space: A Review of Machine Learning Algorithms and Hardware for Space Applications, AICS, 2021, 72-83.
[18] J. L. Gonzalo, C. Colombo, On-board collision avoidance applications based on machine learning and analytical methods. In *8th European Conference on Space Debris, ESA/ESOC, Darmstadt, Germany, 2021*, 20-23.