PROJECT REPORT ON

“TELEMETRY DATA ANALYSIS AND MODELLING”

Submitted By

Kumar Aditya
Mohith Krishna V

Under The Guidance Of:

Mr. Mallikarjuna G M
Division Head of Power Stimuli & TTCD Division
Spacecraft Checkout Group,
URSC,ISRO

July 2024

Spacecraft Checkout Group
U R Rao Satellite Center,ISRO
HAL Old Airport Rd, PO, Vimanapura, Bengaluru, Karnataka -560017

CERTIFICATE

This is to certify that Kumar Aditya from GIET University, Gunupur, Odisha and Mohith Krishna V from Amrita Vishwa Vidyapeetham, Bengaluru have successfully completed the project titled "Telemetry Data Analysis and Modelling" at U R Rao Satellite Centre, ISRO under my supervision and guidance. This certificate is awarded in recognition of their exemplary efforts and successful completion of the project.

Mr. Mallikarjuna G M
Division Head of Power Stimuli & TTCD,
Spacecraft Checkout Group,
U R Rao Satellite Center,ISRO

ACKNOWLEDGEMENT

We extend our heartfelt gratitude to Mr. Mallikarjuna G M (Division Head of Power Stimuli & TTCD, Spacecraft Checkout Group ,URSC, ISRO) for his invaluable guidance, unwavering support, and insightful feedback that greatly contributed to the successful completion of this project.

We also express our sincere thanks to Mr. Krishnan V (Group Director of Spacecraft Checkout Group,URSC,ISRO), whose expertise and encouragement were instrumental in shaping our efforts and enhancing the quality of our work.

Furthermore, we acknowledge U R Rao Satellite Center,ISRO for providing the necessary resources, facilities, and conducive environment that facilitated our research and made this project achievable.

Kind Regards,

Kumar Aditya
GIET University,Gunupur,Odisha

Mohith Krishna V
Amrita Vishwa Vidyapeetham, Bengaluru, Karnataka

CONTENTS

INTRODUCTION

Telemetry data analysis and modelling is crucial for spacecraft operations because they require understanding predicting and optimizing such mission-critical processes. Telemetry specifically refers to the real-time feedback coming from sensors and devices on-board space vessels. In order to enhance operational efficiency and reliability, this project focuses on comprehensive analysis and modelling of spacecraft telemetry data.

This starts with gathering raw telemetry data from spacecraft sensors that provide vital information concerning various parameters such as temperature, pressure, position etc. Thereafter, exploratory data analysis (EDA) is carried out to unveil patterns and anomalies in the dataset, thus giving a preliminary understanding on how the spacecraft operates.

After EDA, the subsequent step involves rigorous application of data cleansing methods to clean the data. This calls for correcting inconsistencies as well as dealing with missing values so that subsequent analyses are accurate and reliable enough. Furthermore, feature engineering techniques are utilized to extract meaningful features from raw telemetry datasets which will be useful while building predictive models.

This project's main objectives focus on the use of machine learning models in prediction and anomaly detection. In other words, predictive models utilize past telemetry data in an attempt to predict the future states and behaviours of spacecraft systems.

On the other hand, anomaly detection is concerned with the identification of unusual or unexpected behaviours in streaming telemetry in real time. In space craft operations, early anomaly detection buys time to prevent malfunction or failure. Exploring machine learning techniques, more particularly unsupervised learning algorithms, detects automatically deviations from normal operating conditions. This proactive approach shall help mission control teams take pre-emptive measures to ensure the success and reliability of spacecraft missions.

DATA ANALYSIS

2.1 Importing Necessary Libraries:

numpy: Basic for numerical computations and array, matrix operations, and manipulation.
pandas: Applied in data manipulation and analyses by offering robust data structures like DataFrames.
matplotlib: This is one of the broadest Python libraries used for creating static, animated, and interactive visualizations.
datetime: This library provides classes for date and time manipulations from elementary to complex uses.
scipy: It has modules for scientific and technical computing with additional functionality in statistics, optimization, signal processing, special functions, and more.
Sklearn: This is a general library on machine learning, containing various algorithms for classification, regression, clustering, and dimensionality reduction.

All these libraries provide total data preprocessing, EDA, statistical modelling, and development of machine learning models. This will integrate and ensure the robustness and efficiency to handle and analyse the dataset presented in this report.

2.2 Data Reading:

Initially, the project encountered challenges in efficiently reading and processing a large telemetry log file comprising over 9 lakh lines. A straightforward approach involved iteratively reading portions of the log file, using techniques like "strip()" and "split()" to parse each line and extract relevant data into separate lists for timestamps, variables, current values, previous values, and limits. However, this method proved inefficient for the entire dataset due to its time-consuming nature and inability to handle the entire log file at once.

To address these limitations, a more robust solution was implemented using the "read_csv" function from the "pandas" library. Leveraging "delim_whitespace=True" parameter along with "on_bad_lines='skip' ", this approach efficiently extracted the necessary information and organized it into a structured DataFrame. The DataFrame consisted of five columns: "TIMESTAMP", "VARIABLE", "CURRENT_VALUE", "PREVIOUS_VALUE", and "LIMIT", aligning with the project's requirements for comprehensive telemetry data analysis and modelling.

This optimized method not only streamlined the data reading process but also facilitated subsequent data manipulation, exploratory data analysis (EDA), and machine learning model development. By leveraging `pandas` capabilities to handle large datasets effectively, the project achieved improved efficiency and accuracy in analysing spacecraft telemetry data, ultimately contributing to enhanced operational insights and decision-making in spacecraft missions.

2.3 Data Cleansing:

1.**Replacing White spaces with Underscores:** In the dataset, whitespace characters were replaced with underscores ('_'). This standardization ensures consistency in variable naming conventions and facilitates easier data manipulation and analysis.

2. **Handling Variables Beginning with Numbers:** Variables that began with numbers were adjusted by adding an underscore before the variable name. This adjustment is necessary because variable names starting with numbers can lead to syntax errors or ambiguities in programming languages.

3. **Replacing Special Symbols:** Special symbols and inappropriate characters within the data were replaced with suitable alternatives. This process eliminates potential errors during data processing and ensures that the dataset adheres to standard formatting conventions.

4.**Integer Encoding for Text Data:** Textual data within the dataset underwent integer encoding. This transformation converts categorical or textual data into numerical equivalents, which are easier for machine learning algorithms to process and analyse.

5.**Removing Noisy Data:** Noisy or irrelevant data entries were identified and subsequently removed from the dataset. This step helps improve the overall quality and reliability of the dataset for subsequent analysis and modelling tasks.

6.**Changing Columns into Appropriate Data types:** Columns were converted to suitable types: numerical to `int` or `float` for calculations, date/time to "datetime" for chronological analysis, and categorical to "category" for efficiency. This ensures data integrity and facilitates accurate analysis, crucial for optimizing spacecraft operations and decision-making in missions.

By implementing these data cleaning procedures, the project aimed to enhance the integrity and usability of the spacecraft telemetry data. This clean and standardized dataset serves as a solid foundation for conducting exploratory data analysis (EDA), developing predictive models, and detecting anomalies in real-time telemetry streams. These efforts contribute to improved decision-making and operational efficiency in spacecraft missions, ultimately supporting the project's objectives of optimizing mission-critical processes and ensuring the reliability of space operations.

2.4 Exploratory Data Analysis:

The dataset for this project comprises 889,381 instances and consists of 5 columns. The exploration revealed that:

- There are 32,666 unique timestamps, indicating the temporal span covered by the telemetry data.
- The dataset contains 2,831 unique variables, representing the different parameters monitored during the spacecraft operations.
- The current_value column exhibits 48,174 unique integer values, reflecting the diverse range of measurements recorded.
- Similarly, the previous_value column shows 48,224 unique integer values, capturing historical data points.
- The limit column includes 1,972 unique values, specifying operational thresholds or constraints.
- Notably, all columns are complete without any null values. The data type information reveals that:
- Timestamp is stored as Datetime objects, facilitating chronological analysis.
- Variable is categorized as object data type, representing categorical information.
- Current_value and previous_value are stored as integers, enabling numerical computations and statistical analysis.
- Limit is stored as object data type, potentially requiring further parsing or conversion for numeric operations.

This initial EDA provides a comprehensive overview of the dataset's structure and characteristics. The absence of missing values and the diversity in unique values across columns underscore the dataset's richness and complexity. Subsequent analyses, including visualization and statistical summaries, will delve deeper into these insights to uncover patterns, relationships, and anomalies critical for optimizing spacecraft operations and decision-making in mission scenarios.

2.5 Organizing Telemetry Data by Variable:

1. **Dynamic Segmentation with Dictionary Structure:** Each unique variable from the 'Variable' column was isolated into separate DataFrames using a dictionary approach.

2. **Iterative Filtering**: Data filtering involved iterative extraction of each unique variable's data from the original dataset into distinct DataFrames stored within the dictionary.

3. **Ensuring Data Integrity and Usability**:
   - 'Timestamp' columns were converted to datetime objects.
   - 'Current_value' and 'Previous_value' columns were cast to integers for numerical operations.
   - The 'Limit' column underwent transformation to float data type, aligning with specific data requirements.

4. **Global Variable Assignment**: Using the globals keyword, segregated DataFrames were assigned as global variables, facilitating efficient access and tailored analysis for each telemetry variable.

5. **Enhanced Data Processing Efficiency**: This systematic approach not only improves the organization and management of spacecraft telemetry data but also optimizes data processing efficiency for comprehensive exploratory analysis and modelling tasks.

MODEL BUILDING

3.1 Models Used:

1. **Linear Regression:**
Linear Regression is a statistical method used for modelling the relationship between a dependent variable and one or more independent variables. It assumes a linear relationship between the variables and is widely used for predicting continuous outcomes.

2. **Isolation Forest:**
Isolation Forest is an anomaly detection algorithm that isolates anomalies by randomly selecting features and splitting data points until anomalies are isolated in few steps. It is effective for identifying outliers or anomalies in high-dimensional datasets.

3. **One Class SVM (Support Vector Machine):**
One Class SVM is a machine learning algorithm used for anomaly detection in which it learns the distribution of normal data and detects outliers as data points that lie far from the learned distribution.

4. **Local Outlier Factor (LOF):**
LOF is a method for detecting outliers by comparing the local density deviation of a data point with that of its neighbours. It identifies anomalies based on the density of neighbouring data points, making it effective for local outlier detection.

5. **DBSCAN (Density-Based Spatial Clustering of Applications with Noise):**
DBSCAN is a clustering algorithm that can also be used for outlier detection. It groups together points that are closely packed together and identifies points that lie alone in low-density regions (outliers).

6. **SVM Classifier (Support Vector Machine):**
SVM Classifier is a supervised learning model used for classification tasks. It finds the optimal hyperplane that best separates classes in a high-dimensional space, making it effective for both linear and non-linear classification problems.

7. **Ridge Model (Ridge Regression):**
Ridge Regression is a regularization technique used to mitigate multicollinearity in linear regression models. It adds a penalty term to the least squares objective, improving the model's stability and performance.

8. **Decision Tree Regressor:**
Decision Tree Regressor is a non-parametric supervised learning method used for regression tasks. It splits the dataset into subsets based on features and predicts the average target value of each subset, making it interpretable and versatile for regression problems.

3.2 Model Training:

1. **Linear Regression:**
The Linear Regression model was trained with a test size of 20% to evaluate its performance. It utilized 'previous_value' and 'current_value' as input features to predict 'current_value'. Linear Regression assumes a linear relationship between the input features and the target variable, making it suitable for predicting continuous outcomes in a straightforward manner.

2. **Isolation Forest:**

Implemented with a contamination rate of 0.04, the Isolation Forest algorithm was employed for anomaly detection. It was trained on both 'current_value' and 'previous_value'. Isolation Forest works by isolating anomalies in the dataset as instances that are few in number and different from the majority of data points, making it effective for detecting outliers in high-dimensional datasets.

3. **One Class SVM:**
Utilizing the One Class SVM model with nu=0.1, the dataset's 'current_value' was scaled using MinMaxScaler to normalize the feature values. This SVM variant is designed to identify outliers by learning the distribution of normal data points and isolating deviations as anomalies based on their anomaly scores. An anomaly threshold was applied to distinguish between normal and anomalous data points effectively.

4. **Local Outlier Factor (LOF):**
Trained with n_neighbours=20 and a contamination rate of 0.1, LOF examined anomalies by assessing the local density deviation of each data point with respect to its neighbours. Features included the difference between 'current_value' and 'previous_value', as well as 'slope', 'current_value', and 'previous_value'. LOF is particularly useful for detecting outliers in datasets with varying densities and complex distributions.

5. **DBSCAN (Density-Based Spatial Clustering of Applications with Noise):**
Employing parameters eps=0.5 and min_samples=5, DBSCAN was utilized for clustering and outlier detection based on the density of data points. It was trained on the difference between 'current_value' and 'previous_value'. DBSCAN identifies outliers as data points lying in low-density regions and groups together densely packed data points into clusters, making it robust for detecting outliers in spatial data.

6. **SVM Classifier:**
Utilizing an 'rbf' kernel, the SVM Classifier was trained with a test size of 0.2 for classification tasks. SVMs are effective for both linear and non-linear classification tasks by finding the optimal hyperplane that best separates different classes in the feature space.

7. **Ridge Model:**
Trained with a regularization parameter alpha=0.1 and a test size of 0.2, the Ridge Regression model aimed to mitigate multicollinearity issues in the dataset. It utilized 'previous_value' and 'current_value' as features to predict 'current_value', enhancing model stability and performance in regression tasks.

8. **Decision Tree Regressor:**
Employed for predicting 'current_value' using 'previous_value' and 'current_value' as features, the Decision Tree Regressor partitions the dataset into subsets based on feature thresholds. This model is advantageous for its ability to capture non-linear relationships and interactions between features, making it suitable for complex regression problems in telemetry data analysis.

For the SVM Classifier, a synthetic dataset was generated containing labelled sawtooth and not_sawtooth curves. Each curve was characterized by features such as peak-to-peak amplitude, mean amplitude, kurtosis, and frequency. These features were extracted to differentiate between the two classes of curves. The SVM classifier, utilizing an 'rbf' kernel and trained with a test size of 0.2, was employed to learn the optimal hyperplane that separates the sawtooth and not_sawtooth curves in the feature space. By iterating through different feature combinations, the model aimed to accurately classify unlabelled data into either the sawtooth or not_sawtooth category based on their extracted

features. SVMs are renowned for their ability to handle both linear and non-linear classification tasks effectively, making them suitable for discerning complex patterns and structures in synthetic datasets like the one described.

RESULT

4.1 Best Models:

In our analysis of spacecraft telemetry data using various machine learning models, distinct performances were observed across different algorithms. The **Decision Tree Regressor** algorithm excelled in predictive tasks, demonstrating robust performance in accurately forecasting future states based on historical data. Its ability to capture complex relationships and interactions between variables made it particularly effective in this regard.

Conversely, for anomaly detection tasks, the **Local Outlier Factor (LOF)** algorithm exhibited superior performance. By assessing the local density deviation of each data point with respect to its neighbours, LOF successfully identified anomalies in the telemetry data. This capability is crucial in space missions where early anomaly detection can prevent potential malfunctions or failures, thereby ensuring the reliability and safety of spacecraft operations.

These findings underscore the importance of selecting the appropriate machine learning algorithm based on the specific task at hand. While **Decision Tree Regressor** excel in predictive modelling by leveraging the dataset's intrinsic patterns, **LOF** shines in anomaly detection by identifying outliers indicative of irregular spacecraft behaviour. By leveraging the strengths of each algorithm, our analysis contributes to optimizing mission-critical processes and enhancing operational efficiency in space missions.

4.1.1 Decision Tree Regressor:

In the analysis of spacecraft telemetry data, the Decision Tree Regression model was employed to predict the "CURRENT_VALUE" based on historical observations stored in the 'PREVIOUS_VALUE' column of the dataset ("FRAME_ID_df").

Model Training and Methodology:

The dataset was partitioned into predictor ('X') and target ('Y') variables, where 'PREVIOUS_VALUE' served as 'X' and 'CURRENT_VALUE' as 'Y'. The DecisionTreeRegressor from scikit-learn was then trained using these variables to learn the relationship between past and current telemetry readings. The decision tree algorithm recursively splits the dataset into subsets based on the values of 'PREVIOUS_VALUE', aiming to minimize the variance of 'CURRENT_VALUE' within each subset.

Evaluation Metrics:

After training, the model's performance was evaluated using three key metrics:

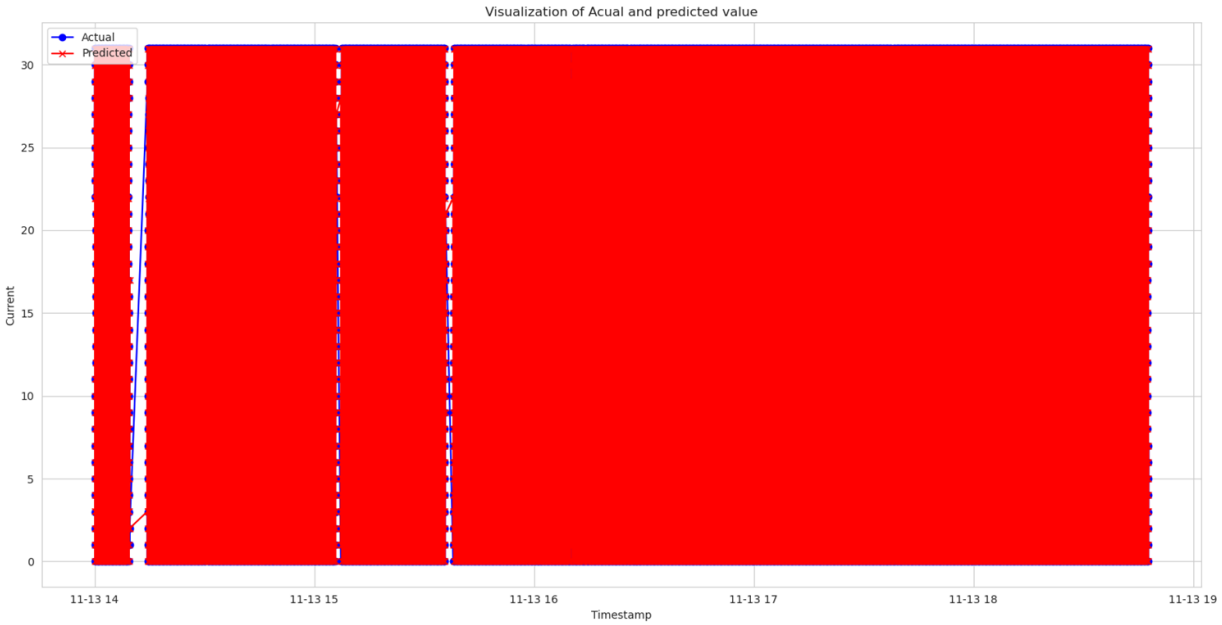
Mean Squared Error (MSE): Measures the average squared difference between the predicted and actual values of 'CURRENT_VALUE'.

Mean Absolute Error (MAE): Provides the average absolute difference between predicted and actual values, offering a straightforward measure of prediction accuracy.

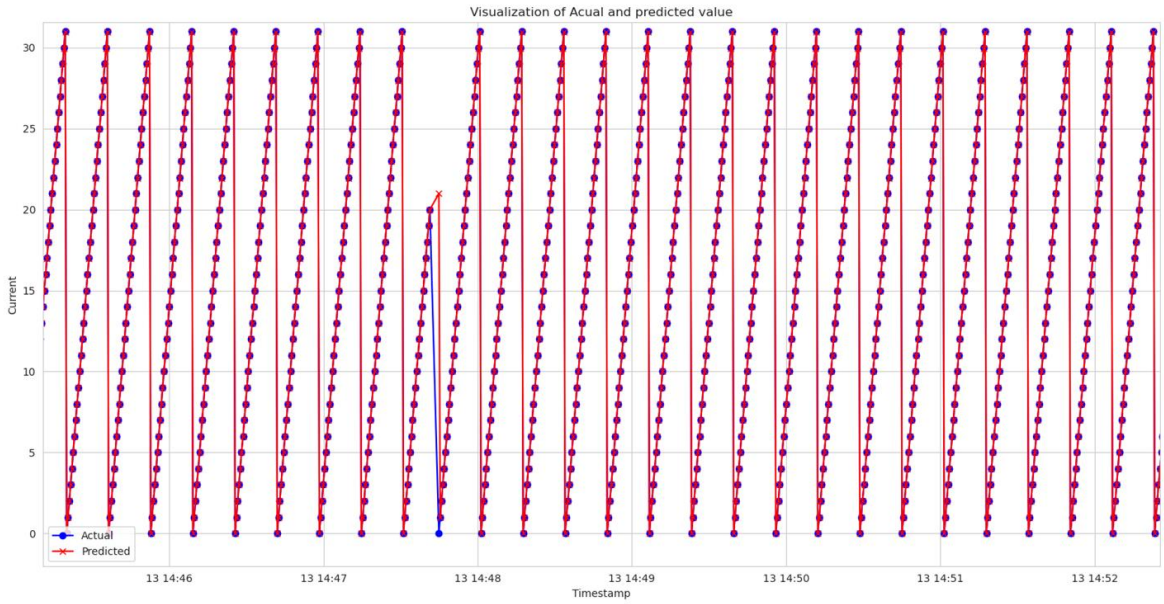
R-squared Score (R2): Indicates the proportion of variance in 'CURRENT_VALUE' explained by 'PREVIOUS_VALUE'. A score closer to 1.0 suggests a better fit of the model to the data.

Performance Assessment:

The obtained MSE of 0.19977193858 and MAE of 0.0179681327 indicate that the model's predictions are consistently close to the actual values of 'CURRENT_VALUE'. Additionally, the high R-squared score of 0.997656108686 signifies that the model captures a substantial portion of the variance in 'CURRENT_VALUE' based on 'PREVIOUS_VALUE', demonstrating strong predictive capability.



Visualization of Actual VS Predicted Values using Decision Tree Regressor



Sample Visualization of Actual VS Predicted Values

Conclusion:

The Decision Tree Regression model proves effective in predicting 'CURRENT_VALUE' from 'PREVIOUS_VALUE' in spacecraft telemetry data. Its ability to provide accurate forecasts

enhances decision-making processes, aids in anomaly detection, and supports operational efficiencies in space missions. By leveraging such predictive models, organizations can optimize resource allocation, ensure mission reliability, and enhance overall mission success.

4.1.2 Local Outlier Factor:

In our analysis of spacecraft telemetry data, we utilized the Local Outlier Factor (LOF) algorithm to detect anomalies in the 'CURRENT_VALUE' parameter based on historical observations from the 'PREVIOUS_VALUE' and engineered features.

Data Preparation and Feature Engineering:

Firstly, a subset of telemetry data (`sample_frame_df`) containing 'TIMESTAMP', 'CURRENT_VALUE', and 'PREVIOUS_VALUE' was extracted. To enhance anomaly detection, additional features were engineered:
DIFF: Calculated as the absolute difference between 'CURRENT_VALUE' and 'PREVIOUS_VALUE', capturing sudden changes.
TIMEDIFF: Represents the time difference between consecutive timestamps in seconds, providing temporal context.
SLOPE: Reflects the rate of change in 'CURRENT_VALUE' over time, computed as the differential quotient.

Missing values in 'SLOPE' and 'TIMEDIFF' were handled by imputing them with their respective mean values to ensure continuity in data processing.

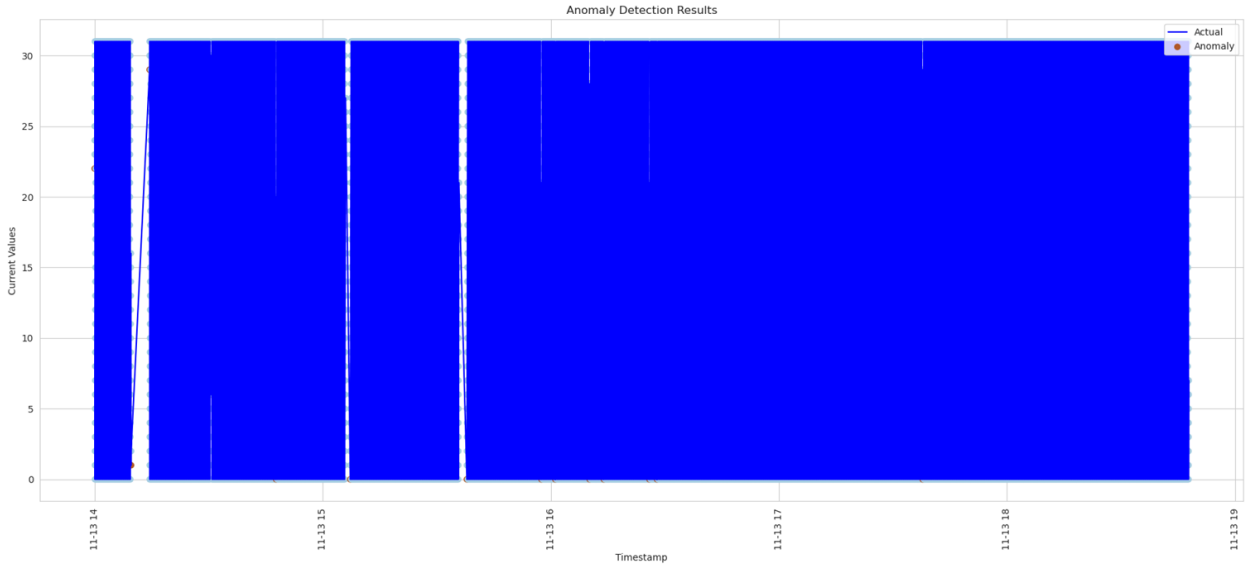
Anomaly Detection with LOF:

The LOF algorithm was applied to the dataset with specific parameters:
n_neighbours: Defined as the minimum between a predefined threshold (20) and the dataset's length, determining the number of neighbours considered for anomaly scoring.
contamination: Set to 0.01, indicating the expected proportion of anomalies in the dataset.

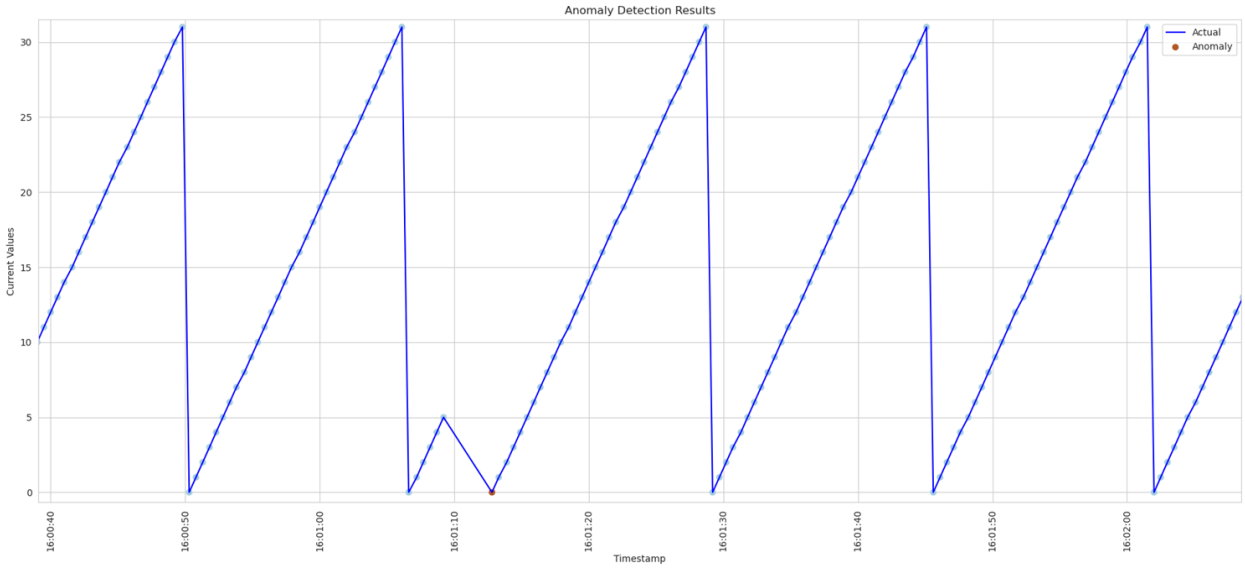
Algorithm Operation: LOF computes an anomaly score for each data point based on its deviation from its local neighbourhood. Points with scores below a threshold are labelled as anomalies (`Y_PRED == -1`), indicating potential irregularities in spacecraft telemetry readings.

Visualization of Anomalies:

To visualize detected anomalies, a plot was created using matplotlib:
- The plot juxtaposes the actual 'CURRENT_VALUE' (represented by a blue line with markers) against detected anomalies (depicted by red crosses), offering a visual overview of potential irregularities identified by the LOF algorithm.



Visualization of Anomaly Detection using LOF



Sample Visualization of Anomaly Detection using LOF

Conclusion:

Utilizing the Local Outlier Factor (LOF) algorithm for anomaly detection in spacecraft telemetry data proves effective in identifying deviations that may indicate critical operational issues. By leveraging engineered features and advanced anomaly detection techniques, such as LOF, organizations can pro actively monitor spacecraft health, mitigate risks, and ensure mission reliability. Continuous refinement and validation of these methods with real-time telemetry data will further enhance their utility in supporting decision-making and operational efficiency in space missions.

CONCLUSION AND FUTURE WORKS

5.1 Conclusion and Future Works:

In this project we looked at spacecraft telemetry data to improve operational efficiency and decision making in space missions. We did data preprocessing, exploratory data analysis (EDA) and built machine learning models to predict future telemetry values and detect anomalies that indicate spacecraft issues.

Data Analysis and Preprocessing:

First we tackled the problem of dealing with a large telemetry log file by using pandas for data manipulation and organization. We standardized variable names, handled anomalies and optimized data types to ensure data integrity and usability. This was key to getting a solid dataset for further analysis.

Exploratory Data Analysis (EDA):

Our EDA gave us insights into the dataset structure, time spans, variable diversity and distributions. This phase helped us choose the right machine learning algorithms and feature engineering techniques to extract meaningful patterns and relationships from the data.

Model Building and Evaluation:

We tried out several machine learning algorithms, Decision Tree Regressor for predictive modelling and Local Outlier Factor (LOF) for anomaly detection. Each model was evaluated using performance metrics to see how well it would work in real world scenarios. Decision Tree Regressor did great in forecasting telemetry values and LOF did well in detecting anomalies that are important for spacecraft health monitoring.

5.2 Future Works:

Next steps:

1. Real-Time Monitoring and Anomaly Detection: Streaming data processing to enable real-time anomaly detection and response during operations.
2. Advanced Feature Engineering: More feature engineering and advanced techniques to improve model performance and robustness.
3. Domain Expertise: Work closely with domain experts to get their knowledge and insights into the modelling process to make the model more relevant and reliable.
4. Ensemble and Hybrid Models: Explore ensemble methods or hybrid models to combine multiple algorithms to leverage their strengths for telemetry analysis and decision support.
5. Scalability and Performance Optimization: Test model scalability and optimize for deployment in resource constrained environments.
6. Validation and Deployment: Run extensive validation tests across multiple datasets and operational scenarios to validate model performance and deploy in mission critical environments.

By doing these future works we hope to push the boundaries of data driven insights in spacecraft telemetry analysis. This will ultimately lead to better operational efficiency, reduced risks and success of future space missions.

So there you have it. Data analytics and machine learning in the aerospace industry. We've opened the door to better decision making and more robust space exploration. Science and technology in space missions.