Anomaly detection in environmental sensor system

I. Introduction

Environmental sensor systems play a crucial role in monitoring air quality, temperature, humidity, and pollution levels. However, these systems face significant challenges, including sensor malfunctions, data loss, and inaccurate readings caused by environmental factors or external interferences. The results of this project may improve environmental policies by providing data -based information to government agencies to combat important threats such as air pollution, water pollution and deforestation. The detection of abnormalities in real time allows the execution of decisions quickly in crises, allowing the government to immediately respond to polluted peaks or fluctuations in water quality. In addition, the project helps to optimize resource allocation by identifying high -risk areas, ensuring effective deployment of monitoring stations and emergency intervention groups. By supporting smart city initiatives and sustainable development.

II. Problem Statement

With the increasing volume of data collected by these systems, big data analytics is essential for identifying abnormal patterns (anomalies) that may indicate real environmental changes or sensor errors. Failure to accurately detect these anomalies can lead to incorrect decisions, negatively impacting environmental policies and delaying responses to environmental crises. Addressing this issue directly supports Sustainable Development Goal (SDG) 13: Climate Action, which emphasizes the need for immediate and accurate monitoring of environmental changes to mitigate climate-related risks

III. Research question

How can machine learning-based anomaly detection models differentiate between sensor errors and real environmental changes in air quality monitoring systems?

In order to improve data accuracy and system performance, this question helps in identifying important technical and environmental elements affecting sensor certainty.

IV. SDG

The sustainable development goals are a set of goals that were developed by the united nations as part of the 2030 agenda for sustainable development. these goals aim to address global challenges such as poverty, environmental sustainability and peace, ensuring a more sustain able future for all.

SDG has a set of 17 goals, for our project we selected SDG 13: Climate Action as it aligns with our focus on improving environmental sensor accuracy. Reliable monitoring helps detect climate changes, supports informed decision-making, and enhances responses to environmental challenges.

V. Understanding the dataset

The dataset comprises 405,184 records and 9 columns, representing IoT telemetry data collected from various devices over time. Each record contains a timestamp (ts), identifying when the data was recorded, and a device identifier (device), which specifies the source of the data. The dataset includes multiple sensor readings, such as carbon monoxide (co), humidity (humidity), light (light), liquefied petroleum gas (lpg), motion detection (motion), smoke level (smoke), and temperature (temp), providing a detailed overview of environmental conditions. The majority of these readings are stored as numerical values (float64), allowing for precise measurement and analysis. However, the light and motion columns are Boolean (True/False), indicating whether a specific event or state was detected. Given the large volume of records, this dataset is well-suited for analyzing trends, detecting patterns, and monitoring environmental changes over time.

VI. Related Work

Our chosen studies demonstrate the significance of AI and machine learning in anomaly detection and monitoring the environment. The article “Artificial Intelligence in Environmental Monitoring: Advancements, Challenges, and Future Directions” explores how AI improves the detection of pollution and disaster prediction, though it faces challenges like data limitations and ethical concerns. The other article ”A Machine Learning Approach to Anomaly Detection” discusses two methods, rule learning (LERAD) and clustering (CLAD), these two techniques enhance intrusion detection by recognizing deviations from normal behavior, despite ongoing issues with false positives and data quality. The paper “Real-time Bayesian Anomaly Detection for Environmental Sensor Data” shows Dynamic Bayesian Networks for real-time detection of anomalies in environmental sensor networks. The last research “machine learning regression techniques for environmental data verification” which showcases how artificial intelligence boosts prediction accuracy in climate observation and resource management. These four studies together highlight artificial

intelligence's potential in enhancing security, sustainability and decision-making.

Limitations:There are several limitations in anomaly detection for environmental sensor systems using AI. One limitation is sample size, as the dataset may not be large enough for generalization. Data quality is also crucial since model accuracy depends on the quality of environmental data. Additionally, SVR requires significant computational resources, posing practical challenges. Rapid fluctuations in environmental conditions can impact anomaly detection accuracy, leading to potential errors. AI models require large amounts of high-quality data for training, and data bias can result in inaccurate predictions. Some models, such as deep neural networks, are difficult to interpret, making decision-making processes less transparent. Training large AI models consumes high energy, which negatively impacts the environment. Limited computational resources in developing regions hinder AI adoption. Furthermore, there are specific model limitations: LEARD assumes that training data is attack-free, which is not always the case, while CLAD struggles with high-dimensional data and may misclassify attack traffic. Both methods also require careful feature engineering tuning for optimal performance.

Contributions:This research introduces a novel methodology for anomaly detection in environmental sensor data using Dynamic Bayesian Networks (DBNs) and multiple sensor data streams to enhance detection accuracy. An SVR-based model was developed for environmental data analysis and anomaly detection, incorporating a new statistical approach based on residual analysis to improve anomaly identification. The model was applied to a real environmental dataset, demonstrating its effectiveness in detecting anomalies. The proposed approach improves the accuracy of predicting environmental disasters such as floods and wildfires, enhances air and water quality monitoring through AI-driven methods, and increases the efficiency of data analysis compared to traditional techniques. Additionally, LEARD was introduced to generate rules for normal behavior, while CLAD was proposed to detect anomalies without requiring labeled attack-free training data. Both methods were evaluated on the DARPA99 dataset, showing improved attack detection performance compared to previous techniques.

conclusion:Understanding the applications of AI and machine learning in anomaly detection, cybersecurity, and environmental monitoring is improved by reviewing these research. For better data analysis and decision-making, they offer insights into methods like rule-based learning and Dynamic Bayesian Networks. Furthermore, by bringing to light issues like data constraints and moral dilemmas, these investigations aid in the creation of creative solutions. We can improve the ability to solve problems, create AI-powered solutions, and support industry or research developments. Whether the goal is to use these methods in practical situations, streamline decision-making, or investigate novel research avenues, this understanding provides useful instruments to enhance security, sustainability, and forecast precision in a variety of domains.
[1] [2] [3] [4] [5]

Paper	Models	Contributions	Limitations	Data and Pre-process	Results
[1]	The study employs Dynamic Bayesian Networks (DBNs) and utilizes Kalman Filtering and Particle Filtering to detect anomalous values in the data.	The paper introduces a new methodology based on DBNs for anomaly detection in environmental sensor data, emphasizing the use of multiple sensor data streams to improve anomaly detection accuracy.	The paper discusses several limitations, including the impact of increasing the number of sensors on detection accuracy and challenges related to handling missing values and noise in the data.	The paper describes the data collection process from two wind sensors in Corpus Christi, Texas, recording measurements every two minutes. The pre-processing steps included injecting synthetic anomalies with a (6/100) probability by altering wind speed values within predefined ranges (R1: 1.03–11.3 m/s, R2: 2.57–12.9 m/s). The dataset was split into training (October 2006) and testing (November 2006) sets for model evaluation.	The paper discusses the performance of the anomaly detection model, comparing false positive and false negative rates across different anomaly detection methods applied to wind data.
[2]	Support Vector Regression (SVR)	• Developed an SVR-based model for environmental data analysis and anomaly detection. • Introduced a new statistical approach based on residual analysis from the SVR model. • Applied the model to a real environmental dataset, demonstrating its effectiveness in detecting anomalies.	• Sample Size: The dataset may not be large enough for broad generalization. • Data Quality: The accuracy of the model depends on the quality of the environmental data used. • Computational Complexity: SVR requires significant computational resources, posing challenges for practical applications. • Changing Environmental Conditions: Rapid fluctuations in environmental factors may affect the model's accuracy in anomaly detection.	The study uses environmental data collected from various sources, such as meteorological stations and environmental sensors, including temperature, humidity, wind speed, and pollution levels. Before analysis, the data undergoes preprocessing, which includes: • Data Cleaning: Removing missing values, errors, and inconsistencies. • Normalization: Adjusting values to a uniform scale. • Data Transformation: Converting data into a suitable format for analysis. • Feature Extraction: Identifying and selecting key features for better model performance.	• Demonstrated the effectiveness of the SVR model in detecting anomalies in environmental data. • Utilized residual analysis from the SVR model to accurately identify anomalies. • Successfully applied the model to a real environmental dataset, proving its accuracy in anomaly detection.
[3]	-CNN -SVM -RNN -Decision Trees -Random Forests -Hybrid Models	-Improved accuracy in predicting environmental disasters such as floods and wildfires. -Enhanced air and water quality monitoring using AI-driven approaches. -Increased efficiency in data analysis compared to traditional methods.	-Requirement for large amounts of high-quality data to train AI models. -Challenges related to data bias, which can lead to inaccurate predictions. -Difficulty in interpreting some models, especially deep neural networks, making decision-making less transparent. -High energy consumption in training large AI models, which can have environmental consequences. -Limited computational resources in developing regions, restricting AI adoption.	-Use of multi-source data, including satellite imagery, sensor data, and environmental reports. -Application of big data processing techniques for handling large datasets. -Implementation of data cleaning and noise reduction methods to improve accuracy. -Integration of data from various sources to create more reliable and accurate models.	-AI models demonstrated significantly higher accuracy compared to traditional methods in environmental data analysis. -Faster response times for environmental emergencies due to improved predictive capabilities. -Scalable AI-driven solutions for large-scale environmental monitoring applications. -Increased precision in detecting pollutants and identifying their sources.
[4]	-LEARD -CLAD	-introduced LEARD to generate rules for normal behavior. -Proposed CLAD to detect anomalies without labeled attack-free training data. -Evaluated both methods on the DARPA99 dataset and showed improved attack detection over previous techniques.	-LEARD assumes that training data is attack free which is not always the case. -CLAD struggles with high dimensional data and can misclassify attack traffic. -Both of these methods need feature engineering tuning for optimal performance.	-Uses the DARPA99 dataset, which contains simulated network traffic with both normal and attack behavior. -Feature extraction includes packet header and payload analysis to improve accuracy.	-CLAD is effective for anomaly detection when the training data is not clean but it detects fewer attacks than LEARD. -Both methods increase coverage of undetected attacks in comparison to existing detection systems.

TABLE I
SUMMARY OF THE ANOMALY DETECTION PAPER

VII. EDA

Exploratory data analysis was conducted using PySpark. The initial step was to remove duplicates and convert timestamp entries into a proper datetime format. Our dataset experienced minimal missing values. To detect outliers, both the standard deviation method and the interquartile range (IQR) approach were used, however, the IQR method was shown to be more successful in detecting extreme values. Data visualization using histograms and boxplots showed that a number of features, especially CO and smoke, were skewed and had noticeable outliers. Interestingly, motion sensor activity seemed to affect the level of gases like smoke and LPG. Clear correlations were shown by using scatter plots to analyze the relationships between variables, especially between variables temperature and smoke, CO and smoke, and humidity and LPG. A heatmap of the correlation matrix further highlighted a strong correlation among gas-related features, which suggests potential environmental interactions. Overall, the dataset was fairly cleaned, and the visual analysis offered insightful information about patterns, relationships and outliers, all of which are crucial to understanding sensor behavior in an internet of Things setting.

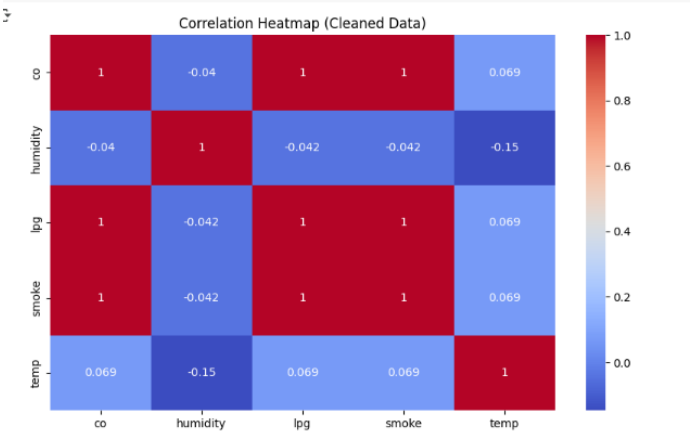

Fig. 1. Correlation Heatmap of Cleaned Data

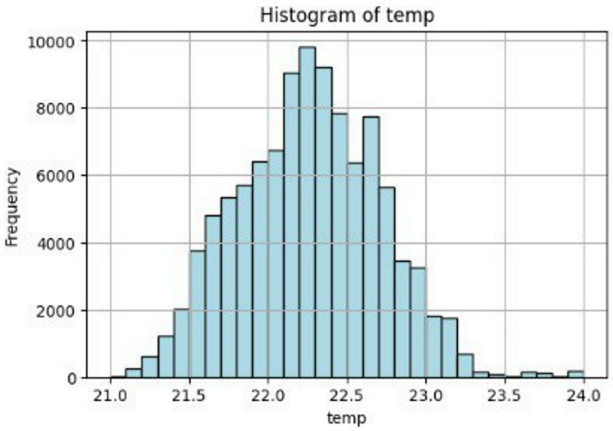

Fig. 2. Histogram of Temp

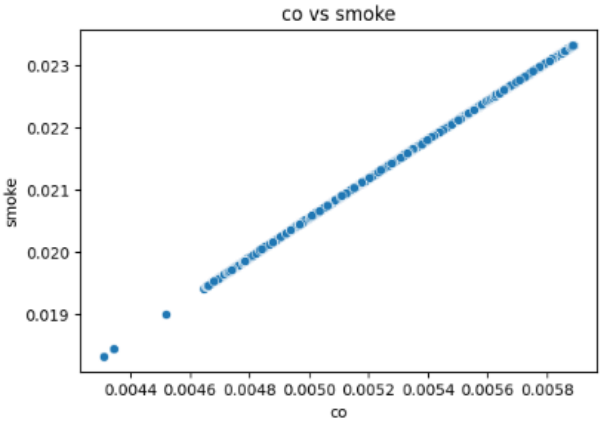

Fig. 3. Scatter Plot of CO vs Smoke

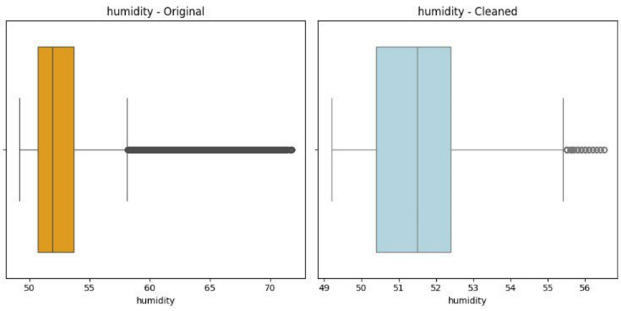

Fig. 4. Humidity BoxPlot

- Fig. 1. The heatmap shows the strength of the relationships between variables. CO, LPG, and Smoke are very strongly correlated, meaning when one increases, the others tend to increase as well. On the other hand, humidity and temperature have weak or almost no correlation with the other variables.

- Fig. 2. The histogram shows that, after removing outliers, the temperature values are approximately normally distributed, with most observations centered around 22°C. This indicates stable and consistent environmental conditions.

- Fig. 3. The chart shows a very strong positive correlation between CO concentration and smoke levels. As the concentration of CO increases, smoke levels increase proportionally, indicating a clear linear relationship between the two variables.

- Fig. 4. The chart shows humidity distribution before and after data cleaning. In the original plot (left), there are many high outlier values, causing the distribution to be right-skewed. After cleaning (right), these outliers were removed, resulting in a more balanced distribution that reflects typical humidity conditions more accurately.

Insights:

- 1. The dataset contains a large number of environmental measurements such as CO, humidity, and temperature, which enables accurate anomaly detection based on changes over time.
- 2. The presence of missing values in some sensors may indicate a malfunction, which should be considered a technical anomaly during analysis.
- 3. Significant variation in the statistical values of certain sensors (e.g., a high maximum CO level compared to the average) suggests the possibility of abnormal readings or environmental leaks during specific time periods.
- 4. The imbalance in the number of detected anomalies across different sensors indicates that some sensors may be more prone to error or degradation, and thus require close monitoring or replacement.
- 5. The distribution of values across sensors is inconsistent: Some sensors record completely different data ranges than others, which may indicate differences in location or sensor sensitivity—an important factor in interpreting anomalies.
- 6. Anomalous values often appear in specific sensors: Repeated anomalies in certain devices may indicate a persistent malfunction or a distinct operating environment (e.g., proximity to a pollution source).
- 7. The presence of outliers in the data highlights the need for flexible models: Values outside the normal range support the use of non-linear algorithms or unsupervised learning to detect them effectively.

VIII. Building Machine Learning Models

Model selection: In this project, we selected three different classification models: Decision Tree, Random Forest, and Logistic Regression. Our choice was based on achieving a balance between model interpretability, robustness, and predictive performance.

Decision Tree was chosen for its simplicity and transparency. It provides a clear visualization of how features influence decision-making, making it highly interpretable and easy to explain to non-technical stakeholders.

Random Forest was selected as an ensemble extension of Decision Trees, offering improved accuracy and reducing the risk of overfitting by combining the outputs of multiple trees.

Logistic Regression was included due to its effectiveness in binary classification problems, particularly when the relationship between the features and the target variable is approximately linear. It also offers fast training time and interpretability through feature coefficients.

By selecting a diverse set of models with different underlying assumptions and strengths, we ensured a comprehensive evaluation of the dataset, allowing us to compare their performance and choose the most suitable model for our anomaly detection task.

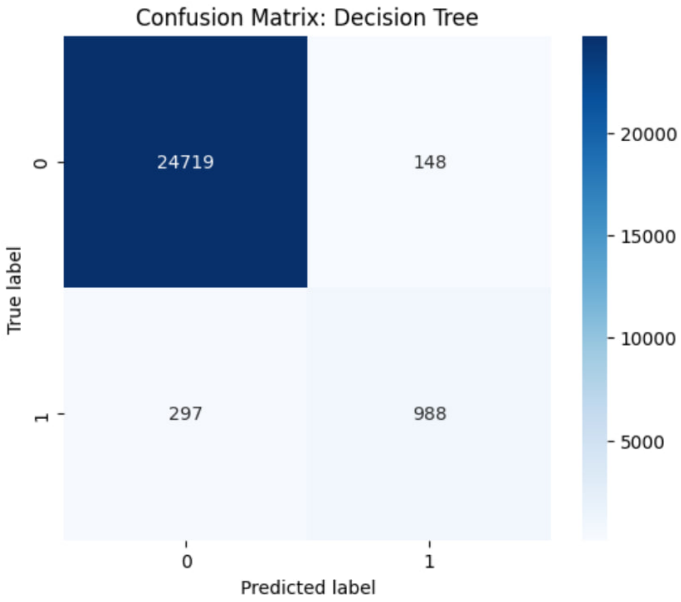

Fig. 5. Confusion Matrix for Decision Tree Model

The Decision Tree model accurately identified 24,719 cases as True Negatives (normal data) and 988 instances as True Positives (anomalies), as we can see in the confusion matrix. However, 148 cases were misclassified as False Positives, where normal cases were mistakenly classified as anomalies, and 297 cases were misclassified as False Negatives, where some anomalies were mistakenly forecasted as normal. Though there is still opportunity to improve the model's dependability by reducing the rates of false positives and false negatives, this indicates that the Decision Tree model is typically successful in differentiating between normal and abnormal cases.

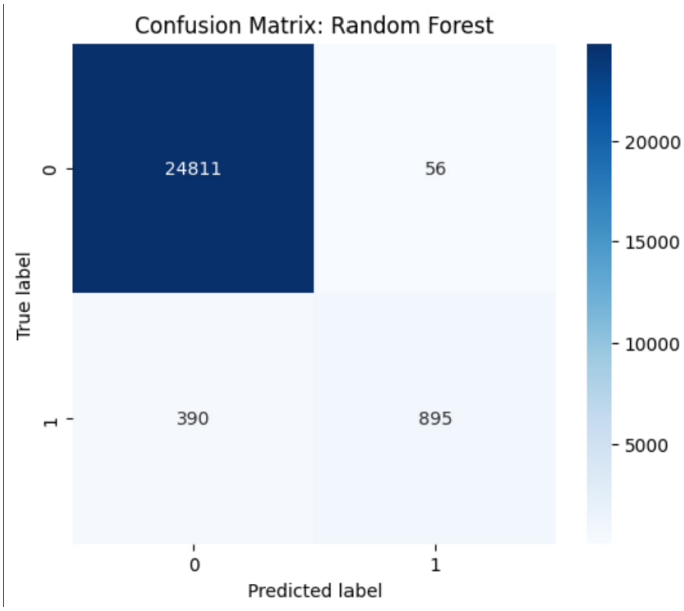

Fig. 6. Confusion Matrix for Random Forest Model

The random forest confusion matrix shows that 24,811 cases properly identified as True Negatives and 895 instances correctly identified as True Positives, which indicates high performance. 56 normal instances were mistakenly identified as anomalies (False Positives), whereas only 390 anomalies were mistakenly classified as normal (False Negatives). Random Forest's capacity to distinguish between normal and aberrant observations is demonstrated by the comparatively low amount of misclassifications. The model is a reliable option for anomaly detection jobs because to its excellent sensitivity (recall) and specificity.

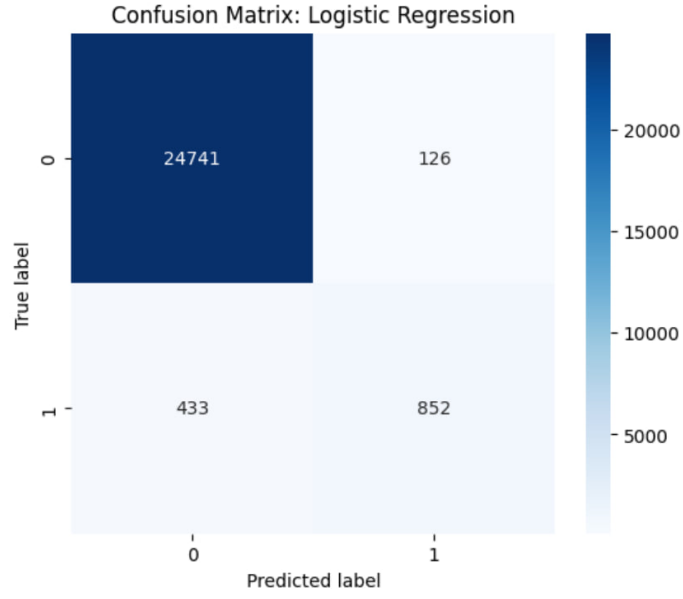

Fig. 7. Confusion Matrix for Logistic Regression Model

The Logistic Regression model correctly classified 24,741 instances as True Negatives and 852 instances as True Positives. However, 126 normal instances were mistakenly categorized as anomalies (False Positives), while 433 anomalies were mistakenly classified as normal cases (False Negatives). While Logistic Regression shows decent classification performance, it has a slightly higher number of misclassifications compared to Random Forest. This suggests that even while the model's predictive capacity is sufficient, it may be enhanced by more fine-tuning or the addition of more intricate characteristics to increase its sensitivity and accuracy.

Model	**Accuracy**	**Effectiveness**
Random Forest Classifier	98%	Achieved high accuracy, indicating strong overall performance and good generalization.
Decision Tree Classifier	98%	Also achieved high accuracy, suggesting it effectively captured patterns in the data.
Logistic Regression	98%	Reached the same high accuracy, showing good separation between classes.

TABLE II
COMPARISON OF CLASSIFICATION MODELS BY ACCURACY AND EFFECTIVENESS

Hyperparameter tuning was conducted to improve the classification models' predictive accuracy. Using PySpark's CrossValidator in combination with ParamGridBuilder, a grid search was applied for three classifiers: Random Forest, Decision Tree, and Logistic Regression. Each model underwent 3-fold cross-validation to evaluate different parameter combinations. For the Random Forest model, the grid included variations in numTrees (50, 100) and maxDepth (5, 10). The Decision Tree model was tuned using multiple maxDepth values (5, 10, 15). Logistic Regression was optimized by testing different combinations of regParam and elasticNetParam. This tuning process led to measurable improvements in model performance, highlighting the importance of parameter optimization in machine learning workflows.

```
# -------- Decision Tree AFTER Tuning --------
dt_paramGrid = (ParamGridBuilder()
                .addGrid(dt.maxDepth, [5, 10, 15])
                .build())

dt_cv = CrossValidator(estimator=dt,
                       estimatorParamMaps=dt_paramGrid,
                       evaluator=evaluator,
                       numFolds=3)

dt_model_tuned = dt_cv.fit(train_data)
dt_preds_after = dt_model_tuned.transform(test_data)
dt_acc_after = evaluator.evaluate(dt_preds_after)
print(f"Decision Tree Accuracy AFTER tuning: {dt_acc_after:.2f}")
```

```
Decision Tree Accuracy AFTER tuning: 0.99
```

Fig. 8. Decision Tree Accuracy After Tuning

```
# -------- Logistic Regression AFTER Tuning --------
lr_paramGrid = (ParamGridBuilder()
                .addGrid(lr.regParam, [0.01, 0.1, 0.5])
                .addGrid(lr.elasticNetParam, [0.0, 0.5, 1.0])
                .build())

lr_cv = CrossValidator(estimator=lr,
                       estimatorParamMaps=lr_paramGrid,
                       evaluator=evaluator,
                       numFolds=3)

lr_model_tuned = lr_cv.fit(train_data)
lr_preds_after = lr_model_tuned.transform(test_data)
lr_acc_after = evaluator.evaluate(lr_preds_after)
print(f"Logistic Regression Accuracy AFTER tuning: {lr_acc_after:.2f}")
```

```
Logistic Regression Accuracy AFTER tuning: 0.95
```

Fig. 9. Logistic Regression Accuracy After Tuning

```
# -------- Random Forest AFTER Tuning --------
rf_paramGrid = (ParamGridBuilder()
                .addGrid(rf.numTrees, [50, 100])
                .addGrid(rf.maxDepth, [5, 10])
                .build())

rf_cv = CrossValidator(estimator=rf,
                       estimatorParamMaps=rf_paramGrid,
                       evaluator=evaluator,
                       numFolds=3)

rf_model_tuned = rf_cv.fit(train_data)
rf_preds_after = rf_model_tuned.transform(test_data)
rf_acc_after = evaluator.evaluate(rf_preds_after)
print(f"Random Forest Accuracy AFTER tuning: {rf_acc_after:.2f}")
```

```
Random Forest Accuracy AFTER tuning: 0.98
```

Fig. 10. Random Forest Accuracy After Tuning

Conclusion:

In this project, titled "Anomaly Detection in Environmental Sensor System", we conducted a comprehensive analysis to develop an effective approach for identifying anomalies in sensor data. We began with an in-depth Exploratory Data Analysis (EDA) to better understand the structure of the data, identify irregular patterns, and select meaningful features for model training. Additionally, we reviewed several scientific papers focused on anomaly detection techniques, analyzing their methodologies, challenges, and the strengths and weaknesses of their models. These insights from the literature review played a crucial role in refining our strategy and avoiding common pitfalls. Based on our findings, we selected and applied three machine learning models: Decision Tree, Random Forest, and Logistic Regression. Each model was carefully tuned to maximize performance, balancing accuracy, interpretability, and computational efficiency. Random Forest demonstrated strong performance due to its ensemble capabilities, Logistic Regression offered simplicity and robustness, and Decision Trees provided clear interpretability. Through the integration of EDA, literature insights, and model evaluation, we successfully built a reliable anomaly detection framework tailored for environmental sensor systems. Our results confirm the effectiveness of combining data-driven analysis with informed model selection for detecting anomalies in complex sensor environments.

References

[1] D. J. Hill, B. S. Minsker, and E. Amir, "Real-time bayesian anomaly detection for environmental sensor data," *Unknown Journal*, 2006.

[2] F. Yuan and J. Lu, "Anomaly detection for environmental data using machine learning regression," in *IOP Conference Series: Materials Science and Engineering*, vol. 472, no. 1. IOP Publishing, 2019, p. 012089.

[3] D. B. Olawade, O. Z. Wada, A. O. Ige, B. I. Egbewole, A. Olojo, and B. I. Oladapo, "Artificial intelligence in environmental monitoring: Advancements, challenges, and future directions," *Hygiene and Environmental Health Advances*, vol. 12, p. 100114, Oct. 2024. [Online]. Available: https://doi.org/10.1016/j.heha.2024.100114

[4] P. K. Chan, M. V. Mahoney, and M. H. Arshad, "A machine learning approach to anomaly detection," Florida Institute of Technology, Technical Report CS-2003-06, Mar. 2003. [Online]. Available: https://repository.fit.edu/ces$_{f}aculty$

[5] United Nations, “Sustainable Development Goals,” 2024, [Online]. Available: https://sdgs.un.org/goals. [Accessed: 24-Mar-2025]. [Online]. Available: https://sdgs.un.org/goals