

Detecting attacks on an industrial water system from its operational status - a naive IADS

Laurent Pipitone - RAMP^(*) Python Project (2024-11)

() Rapid Analysis and Model Prediction*

Introduction.....	2
Dataset description.....	3
Overview of available files and datasets.....	3
Operational data Structure and Contents.....	3
Attacks and data reliability.....	4
Data simplifications.....	4
Attacks visualization.....	4
Attack data completion.....	4
Exhaustivity for attack periods.....	4
Construction of water tank specific attack flags.....	5
Visuals and analysis.....	5
Intuitive conclusion.....	6
Classification approach - supervised learning.....	7
Objective.....	7
Analysis.....	7
With and without class balancing.....	7
Feature Importance.....	8
Conclusion on the predictive power and visible limits.....	8
Statistical approach - unsupervised learning.....	8
Objective.....	8
Feature Transformation: accounting for statefulness.....	9
Results and analysis.....	9
Performance Summary.....	9
Analysis of Limitations.....	10
Conclusion.....	10
Conclusion.....	10

Introduction


In recent years, the critical need to safeguard Industrial Control Systems (ICS) against cyber threats has become evident, particularly as these systems play a crucial role in sectors such as energy, water distribution, and manufacturing. Attacks targeting these infrastructures can lead to severe disruptions and risks, impacting public safety, economic stability, and environmental security. The rapid digitization of industrial systems has made them increasingly vulnerable to cyber intrusions, which exploit both IT and operational technology (OT) layers.

I am particularly interested in this area due to my emerging responsibilities with Cyberium, where I focus on advancing industrial cybersecurity. Cyberium operates in the realm of cybersecurity for critical industrial sectors, including governmental, defense, and high-stakes industrial entities. Our current mission is to ensure that these essential infrastructures remain protected from potential cyber threats whilst allowing certain communications to flow in and out the critical networks (depending on their criticality level). By exploring cyber-physical attack detection, I mostly aim to deepen my expertise in this area.

For this project, I have chosen to work with data from the [BATADAL](#) (BATtle of the Attack Detection ALgorithms) dataset¹, a well-regarded source in the field of ICS cybersecurity. BATADAL was created to benchmark algorithms that detect cyber attacks on water distribution systems, simulating scenarios relevant to critical infrastructure security. The dataset consists of operational and attack scenarios for a simulated water treatment plant, including normal operations data and periods when various cyber attacks were introduced. With data collected over extended periods, BATADAL includes sensor readings from the plant's network, allowing a detailed view of both the physical and cyber states of the system.

This dataset provides a realistic foundation for analyzing whether attacks on such a system can be effectively detected based solely on operational data. The scope of this project will be to test different algorithms, evaluating their ability to distinguish between normal operational states and potential attacks. Through this approach, I aim to uncover insights into the most relevant factors for reliable attack detection in industrial contexts, laying the groundwork for practical applications in industrial cybersecurity.

The BATADAL dataset served in 2016 as the foundation for a competitive challenge designed to advance research in cyber attack detection for industrial systems. Participants in this challenge were tasked with developing algorithms capable of distinguishing between normal and attack states within a simulated water distribution network. Detailed rules and criteria for the challenge are outlined in the official document, available here: [Challenge rules](#). I can only recommend to watch this short video to have a better and synthetic understanding of the BATADAL context and instakes:

 [Characterizing Cyber-Physical Attacks on Water Distribution Systems](#) .

To go further: [competition high-level results by approach and more information](#).

¹ Riccardo Taormina and Stefano Galelli and Nils Ole Tippenhauer and Elad Salomons and Avi Ostfeld and Demetrios G. Eliades and Mohsen Aghashahi and Raanju Sundararajan and Mohsen Pourahmadi and M. Katherine Banks and B. M. Brentan and Enrique Campbell and G. Lima and D. Manzi and D. Ayala-Cabrera and M. Herrera and I. Montalvo and J. Izquierdo and E. Luvizotto and Sarin E. Chandy and Amin Rasekh and Zachary A. Barker and Bruce Campbell and M. Ehsan Shafiee and Marcio Giacomoni and Nikolaos Gatsis and Ahmad Taha and Ahmed A. Abokifa and Kelsey Haddad and Cynthia S. Lo and Pratim Biswas and M. Fayzul K. Pasha and Bijay Kc and Saravanakumar Lakshmanan Somasundaram and Mashor Housh and Ziv Ohar; "The Battle Of The Attack Detection Algorithms: Disclosing Cyber Attacks On Water Distribution Networks." Journal of Water Resources Planning and Management, 144 (8), August 2018. ([doi link](#), [bib](#))

Dataset description

This section provides a detailed description of the BATADAL dataset, including the files and datasets available, the nature and structure of the data, and the specific variables contained within each dataset.

Overview of available files and datasets

The BATADAL dataset comprises several key files that provide a comprehensive basis for analyzing both normal operations and cyber attack scenarios within the water distribution system. These files include:

- **Training Dataset 1** (file BATADAL_dataset03.csv): a full year of data reflecting normal operations, with no attacks, which serves as a baseline for analyzing typical system behavior.
- **Training Dataset 2** (file BATADAL_dataset04.csv): approximately six months of data that includes labeled attack events. This dataset contains multiple instances of cyber attacks introduced to the water distribution network, allowing for a comparative analysis between normal and abnormal states.
- **Attack Scenarios List** (Attacks_TrainingDataset2.csv): a supplementary file detailing each cyber attack event, including the exact duration of each attack within the dataset and the context for each attack, such as whether it involves tampering with pump flow, changing valve statuses, or manipulating sensor readings, as well as SCADA Concealment Information, indicating whether the attack was visible or concealed within the system's SCADA (Supervisory Control and Data Acquisition) interface.
- **INP File** (CTOWN.INP): a file describing the layout and physical properties of the water distribution system. It contains structural information about nodes, junctions, pipes, pumps, valves, tanks, and reservoirs, defining how components are interconnected and regulated.

Dataset 1 and 2 are considered as **Operational data**.

Operational data Structure and Contents

The datasets 1 and 2 are organized in tabular format, with columns that represent system parameters, and each row capturing a specific timestamp. Key elements include:

- **Timestamp:** A unique identifier for each entry, recording the date and time of each measurement. This column is essential for chronological tracking and time-series analysis.
- **Sensor Data:**
 - **Water Levels (m):** Water levels are recorded for each of the 7 **water tanks** in the system. These measurements reflect the state of water storage at various points in the distribution system.
 - **Pump and Valve Status:** binary status indicators (0 for OFF/CLOSED, 1 for ON/OPEN) for pumps and valves are included, allowing analysis of component states and their changes over time.
 - **Flow Rates (L/s):** flow measurements through pumps and valves, crucial for tracking water movement and identifying unusual flow patterns during attack scenarios.
 - **Pressure (in m, as in meters of water column):** both suction and discharge pressures are captured for pumping stations, helping to monitor system load and detect anomalies related to pressure fluctuations.
- **Attack Flag ATT_FLAG:** (partial and optional) a binary column (0 for normal, 1 for attack) indicating the presence of an attack. For unlabeled periods, the value -999 is used.

Each variable in the dataset is named according to a standardized convention by a **Suffix** (T1, PU1, V2, etc.), which is the component / node identifier and a **Prefix**:

- **L_**: Water levels (e.g., L_T1 for Tank 1)
- **S_**: Status of pumps and valves (e.g., S_PU1 for Pump 1)
- **F_**: Flow rates (e.g., F_V2 for Valve 2)
- **P_**: Pressure readings, often associated with junctions or stations

Attacks and data reliability

The attacks were originally provided as an image, which I transformed to tabular data, so that it can be later used for operational data enrichment. Here are the known attacks scenario included in the Dataset 2:

ID	Starting time [dd/mm/YY HH]	Ending time [dd/mm/YY HH]	Duration [hours]	Attack description	SCADA concealment	Labeled [hours]
1	13/09/2016 23	16/09/2016 00	50	Attacker changes L_T7 thresholds (which controls PU10/PU11) by altering SCADA transmission to PLC9. Low levels in T7.	Replay attack on L_T7.	42
2	26/09/2016 11	27/09/2016 10	24	Like Attack #1 but replay attack extended to PU10/PU11 flow and status.	Replay attack on PU10/PU11 flow and status.	0
3	09/10/2016 09	11/10/2016 20	60	Attack alters L_T1 readings sent by PLC2 to PLC1, which reads a constant low level and keeps pumps PU1/PU2 ON. Overflow in T1.	Polyline to offset L_T1 increase.	60
4	29/10/2016 19	02/11/2016 16	94	Like Attack #3.	Replay attack on L_T1, PU1/PU2 flow and status, as well as pressure at pumps outlet.	37
5	26/11/2016 17	29/11/2016 04	60	Working speed of PU7 reduced to 0.9 of nominal speed causes lower water levels in T4.	None.	7
6	06/12/2016 07	10/12/2016 04	94	Like Attack #5, but speed reduced to 0.7. L_T4 drop concealed with replay attack.	L_T4 drop concealed with replay attack.	73
7	14/12/2016 15	19/12/2016 04	110	Like Attack #6.	Replay attack on L_T1, as well as PU1/PU2 flow and status.	0

Looking at the attack impacts, one should understand SCADA data (such as the Water tank level) is not to be trusted, as they can be hacked by the attacker.

Data simplifications

As this is only a student's (and short) project, I have decided to (over) simplify the case by:

- not using the model (CTWON.INP) in the analysis. The intuition is that a ML system modelization would naturally tend to recreate the water system model numerically, and enforcing the model by the network diagram (such as calculating a distance between nodes in the system graph for instance, and weighting the features by these losses)
- not using the dataset 1 although this would have probably allowed a better modelization of the water system.
- not trying to enrich the provided operational data about the current water system by typical / standardized water system model's data

Attacks visualization

Attack data completion

Exhaustivity for attack periods

In the initial dataset, the **ATT_FLAG** column is inconsistently populated, with certain attack periods either missing labels or marked ambiguously (e.g., with the value -999). To ensure accuracy in

identifying attack periods, I chose to enrich this column by cross-referencing it with detailed attack timing information provided in the attack scenario list. This completion step allows for a more comprehensive and precise labeling of attack intervals, which is crucial for reliable analysis and visualization. By standardizing the attack labels, we ensure that subsequent visualizations and analyses accurately reflect periods of normal operation versus cyber intrusion, enabling the development of more robust detection algorithms.

After this operation, as shown in the code, the percentage of data containing an `ATT_FLAG = 1` increases from 5.24% to 5.43%.

Construction of water tank specific attack flags

To gain more granular insights into the relationship between specific tank levels and potential cyber attacks, we created additional features: **T1_ATT_FLAG** through **T7_ATT_FLAG**. Each of these new features corresponds to one of the tanks (T1 through T7) and indicates whether an attack specifically targets that tank during a given time period.

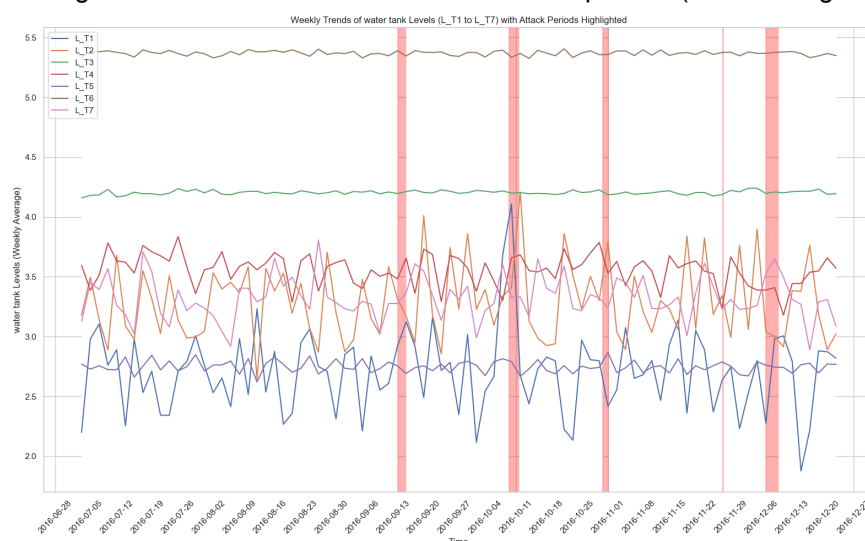
The primary dataset's `ATT_FLAG` provides a general indication of whether the system is under attack, but it lacks specificity regarding which component is impacted. By creating tank-specific flags, we can now distinguish attacks that directly influence each water tank individually. This approach allows us to explore potential correlations between a tank's water level and attack incidents specifically directed at that tank.

To achieve this, we cross-referenced the general attack periods with information on which tanks were affected in each incident. For each attack, if it involved a particular tank (e.g., T1), the **T1_ATT_FLAG** was set to 1 for the duration of that attack. This process was repeated for each tank, ensuring that only relevant attacks were marked in the corresponding tank-specific flag.

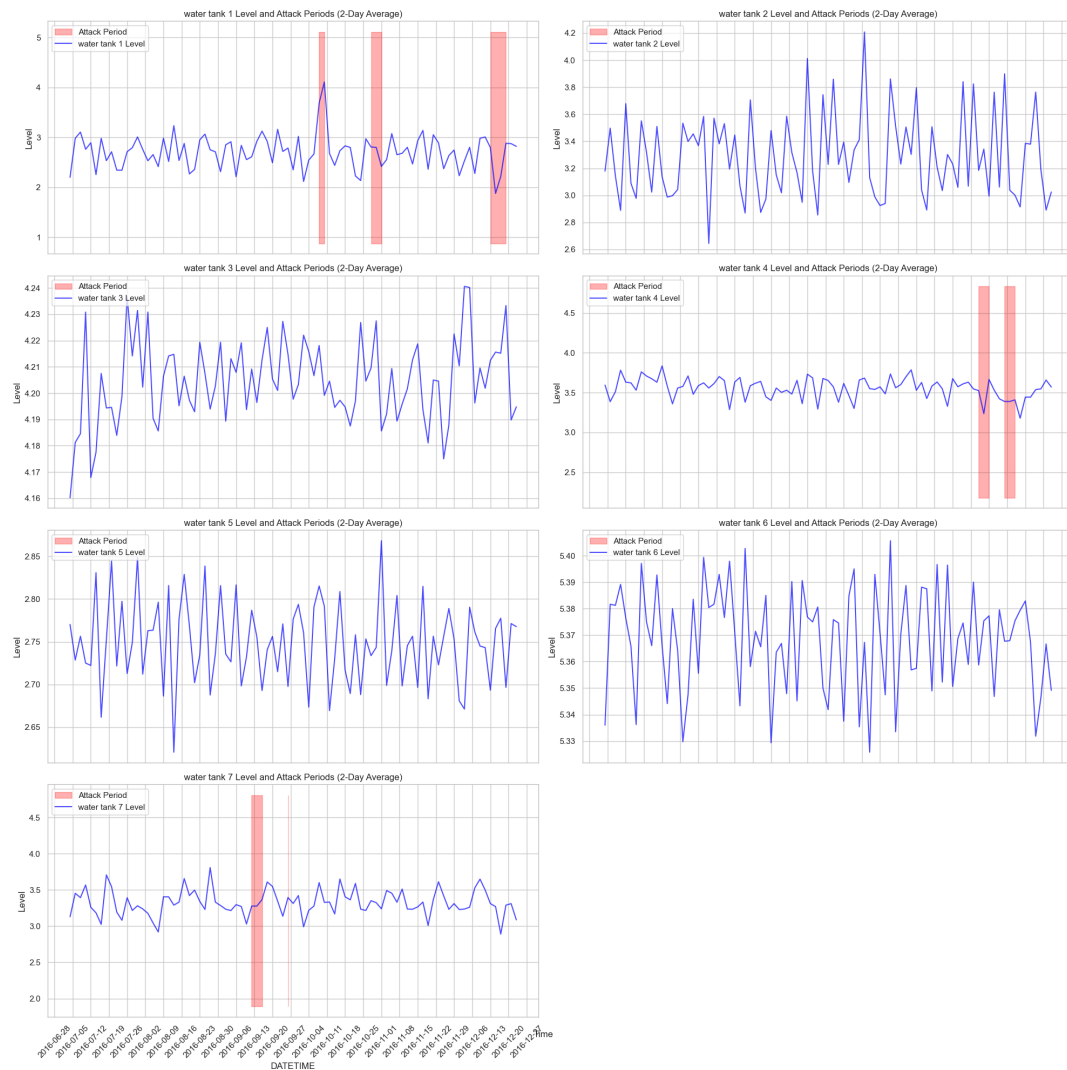
The expected outcome is that these tank-specific attack flags enhance our ability to visualize relationships between tank levels and targeted cyber intrusions. **This setup will allow us to evaluate the hypothesis that variations in water levels might signal specific attacks on a tank, thereby supporting the development of more refined, component-level detection algorithms.**

Visuals and analysis

This figure shows all water tanks levels vs attack periods (in red histograms)



This figure shows each water tank level (from 1 to 7) vs targeted attack periods (in red histograms)



Intuitive conclusion

The visual analysis shows that there is no clear or direct correlation between the periods of attack and the observed water levels in the tanks (which I have confirmed by fitting a linear regression with a very high RMSE compared to the values at hand). Despite detailed examination, the attack periods do not consistently align with significant or distinct variations in tank levels, suggesting that visual inspection alone is insufficient for detecting such events.

This outcome is not entirely unexpected. Given that the data recorded in the SCADA system is known to be compromised during attacks, it is understandable that attack indicators may be obscured or distorted. Cyber attackers often target SCADA data precisely to conceal or manipulate operational anomalies, making it challenging to rely on direct measurements for reliable detection. **This insight highlights the necessity for advanced analytical and algorithmic approaches to discern subtle patterns that may be indicative of attacks within such tampered datasets.**

Classification approach - supervised learning

Objective

Through this approach, a primary objective is to develop a classifier capable of reliably identifying instances of cyber-attacks based on operational data from the water distribution system. Given the binary nature of our target variable, ATT_FLAG (where 1 indicates an attack and 0 indicates normal operation), the classification task can theoretically provide an automated mechanism for early attack detection.

However, one of the challenges encountered in this classification is the significant imbalance in class representation. In the dataset, instances labeled as attacks (ATT_FLAG = 1) account for approximately 5% of the total data. This imbalance presents a classic challenge in machine learning, where minority classes are at risk of being underrepresented in training. In practical terms, a naive classifier might achieve high accuracy simply by predicting "no attack" for every instance, thereby failing to recognize actual attack patterns. This would be particularly problematic in industrial cybersecurity contexts, where failing to detect an attack could lead to significant disruptions or safety risks.

To address this, it seems essential to apply techniques that adjust for class imbalance. Several approaches are commonly employed, such as resampling (either oversampling the minority class or undersampling the majority class), assigning class weights to emphasize the importance of the attack class, or using algorithms that inherently handle imbalance well, such as certain ensemble methods. Additionally, evaluating model performance through metrics like precision, recall, and F1-score rather than relying solely on accuracy ensures that the model's effectiveness in detecting attacks is rigorously assessed.

Ultimately, addressing this class imbalance not only improves the classifier's performance on the minority class but also aligns the model with the project's real-world goal: to prioritize the detection of rare but critical cyber-attack events within industrial environments.

Analysis

The goal of this classification approach was to evaluate the ability of a machine learning model to predict attack periods within an industrial control system based on system data. Using a Random Forest Classifier with and without balancing for the underrepresented attack class, we achieved a relatively high overall accuracy (around 97%) in both cases. However, a deeper look into precision, recall, and f1-score reveals limitations in detecting the minority class (attack periods), despite this overall accuracy.

With and without class balancing

Without balancing, the classifier shows strong overall accuracy and a high precision for the minority class (attack) at 93%. However, the recall for attacks is low, at 56%, indicating that the model misses many true attack instances. This is a significant drawback, as low recall implies an inability to identify all attack events, which is crucial in a security-focused application. When class weights are adjusted to account for the imbalance, the recall for attack detection unexpectedly decreases further, to 50%, and precision for the attack class

remains similar. The model sacrifices some overall accuracy to focus on identifying attacks, yet it still struggles to capture all attack periods effectively. This imbalance highlights a limitation in the model's sensitivity to the minority class, despite our efforts to counteract it. Given these results, we will not pursue further optimization of this model or a deeper exploration of its performance metrics, as our focus will now shift to alternative methods. The limitations observed here emphasize the need to explore more sophisticated techniques, potentially better suited to handle the inherent class imbalance and to improve the model's ability to detect all relevant attack events.

Feature Importance

In both balanced and unbalanced cases, the most important features remain consistent, centering on key pump and pressure variables. Components such as S_PU6, P_J302, P_J307, and L_T1 hold high importance, which aligns with the expectation that specific pump and pressure-related variables would be strong indicators of operational anomalies. However, the small variation in feature importance across balancing scenarios implies that the features most predictive of attacks are relatively stable. This suggests a limited impact of class balancing on feature interpretability but also emphasizes that the current features may not provide sufficient discrimination for attack prediction.

Conclusion on the predictive power and visible limits

The model's overall accuracy is high, yet its predictive power for attack detection remains limited. The class imbalance, where attacks constitute only a small portion of the dataset, affects the classifier's ability to capture all attack periods. The model's recall for attack events is low, highlighting a key limitation: it does not consistently detect attack periods, even with balanced class weights.

Moreover, the SCADA data itself may be compromised, especially during attacks, which could distort the very indicators the model relies on. This factor complicates the detection process, as traditional machine learning methods assume data integrity—a condition not guaranteed here.

While the model can achieve high accuracy in distinguishing between normal and attack states in the overall dataset, it faces limitations in reliably predicting attack periods due to the class imbalance and possible compromised data. Improving predictive power may require additional features that are specifically indicative of attack events, or alternate methods like anomaly detection or ensemble models tuned for imbalanced datasets. **Furthermore, a more tailored approach, such as using time-series analysis or sequential models, may offer improved performance in capturing temporal attack patterns within industrial control systems.**

Statistical approach - unsupervised learning

Objective

In this section, we aim to implement a statistical anomaly detection approach using an Isolation Forest algorithm. Unlike supervised methods, the Isolation Forest offers an unsupervised strategy that models the normal operational patterns of the water distribution system without relying on labeled attack data. By learning typical behavior and relationships within the dataset, this approach enables the detection of deviations that could indicate potential cyber-physical attacks.

The trained Isolation Forest model will establish a baseline of "normal" system behavior by creating splits that isolate data points. Anomalies are identified when data points are isolated early in the splitting process, suggesting significant deviations from typical system behavior. These anomalies can potentially indicate abnormal system states or attack scenarios.

Feature Transformation: accounting for statefulness

The goal of feature transformation in this approach is to capture the system's dynamic behavior by incorporating sequential dependencies. To effectively identify statistical anomalies at a given time step N , it is essential to consider the system's previous state at $N-1$ and its recent evolution.

To achieve this, we include the previous value of each water tank (e.g., Tank 1's value at $N-1$) as a feature. This allows the model to assess both the current state and the rate of change, giving it insight into the system's short-term trajectory.

For a more comprehensive temporal understanding, we could extend this approach to include values from multiple past time steps (e.g., $T-1$, $T-2$, ..., $T-n$), with n acting as a hyperparameter. However, for this initial model, we restrict the scope to $n = 1$, focusing solely on the immediate previous state to predict the subsequent water tank value. This serves as a foundational approach for evaluating the effectiveness of stateful information in anomaly detection.

Results and analysis

The Isolation Forest approach applied to our dataset yielded promising results for detecting statistical anomalies that could indicate potential attack scenarios. By training the model to recognize normal operational patterns, we were able to observe how well it identified deviations from these patterns, which we hypothesize may correlate with cyber-physical attacks.

Performance Summary

Anomaly Detection Effectiveness

The model demonstrated a strong ability to isolate unusual data points, achieving a recall rate of approximately 68% for detecting data points labeled with $ATT_FLAG = 1$. This recall rate suggests that a substantial portion of the attack scenarios were correctly flagged as anomalous by the model, highlighting its potential to detect unusual activity that aligns with known attacks.

Best Parameter Configuration

After testing multiple hyperparameter configurations, the optimal setup was found to include:

- $n_estimators = 100$
- $max_samples = 1.0$
- $max_features = 0.7$
- $contamination = 0.2$

These parameters allowed the model to balance between capturing normal operational patterns and isolating anomalies effectively.

Analysis of Limitations

Despite the model's promising performance, some challenges remain:

- **Class Imbalance:** The dataset's under-representation of attack periods limited the model's sensitivity, as evidenced by the recall rate of 68%. Improving this sensitivity would be crucial in a real-world deployment where detecting every attack is essential.
- **False Positives:** Some normal operational data points were occasionally flagged as anomalies, potentially due to natural system variations that resemble attack patterns. Further fine-tuning, such as dynamic thresholding based on context or adding more complex temporal dependencies, could improve accuracy.

Conclusion

The Isolation Forest model shows potential as a statistical anomaly detection tool within an industrial control system setting, capable of identifying attack periods with reasonable effectiveness. However, future work may benefit from exploring more sophisticated temporal models, such as sequence-based models, or combining multiple detection algorithms to improve recall without increasing false positives. This approach establishes a strong foundation for anomaly detection, with room for enhancements to improve precision and real-time application capabilities.

Conclusion

This project has focused on detecting cyber-physical attacks within an Industrial Control System (ICS), a task essential for maintaining the integrity of critical infrastructure. Our approach primarily falls under the category of **Industrial Anomaly Detection Systems (IADS)**, which aims to identify deviations from typical operational behavior to signal **potential** attacks _ "potential" being the keyword here as those untypical behaviors could come from other factors, which the algorithm could not necessarily distinguish. While the Random Forest Classifier has provided valuable insights into the most influential features, the results highlight both the promise and the limitations of using traditional supervised classification in this context.

The classification approach in this work has shown an overall high accuracy but is limited in reliably detecting attack periods due to inherent class imbalances and the possible compromise of SCADA data. This limitation emphasizes the need to enhance detection mechanisms in the face of real-world complexities. Although IADS can serve as an initial layer of security, they inherently react to abnormalities post-factum, which can leave gaps in proactive threat detection and response.

The statistical approach, using an unsupervised learning method, provides a complementary perspective to the classification approach by focusing on deviations from normal operational behavior rather than direct attack signatures. By implementing an Isolation Forest algorithm, the statistical approach allows us to model "normal" behavior based on patterns in the data and subsequently flag deviations as potential anomalies. This approach is particularly useful when labeled attack data is limited or unavailable, as it does not rely on predefined classes. Instead, it identifies anomalies based on statistical rarity, which may correspond to attack scenarios. While effective in detecting unusual system states, this method has limitations, especially with regard to distinguishing between benign anomalies (due to regular operational variability) and true attack events. However, it highlights the

critical role of unsupervised learning techniques in capturing system anomalies in a manner that could be refined further to enhance ICS security.

Moving forward, an **Industrial Intrusion Detection System (IIDS)** would offer a more robust solution by focusing not only on anomalies but also on patterns and behaviors indicative of specific cyber-attack tactics, techniques, and procedures (TTPs). IIDS are designed to identify known attack signatures or potential threat behaviors more proactively. Implementing an IIDS would involve additional components:

1. **Integration of Known Threat Intelligence:** IIDS can leverage threat intelligence feeds and signature-based detections, enhancing accuracy for known attacks and potentially improving recall.
2. **Behavioral and Sequence-Based Models:** IIDS could apply time-series or sequence-based models (such as LSTMs or HMMs) to detect patterns over time, capturing attack sequences that might go undetected in single-instance anomaly detection.
3. **Real-Time and Multi-Layered Analysis:** Unlike IADS, which often operate retrospectively, IIDS could perform real-time threat analysis, offering a proactive approach that combines both network and operational data.

In summary, while this work has provided a foundation for a naive IADS, focusing primarily on abnormality detection within operational data, transitioning toward an IIDS would enhance resilience by adding layers of preemptive threat identification and response capabilities. This progression from IADS to IIDS represents a necessary evolution to address the sophisticated, evolving cyber threats targeting ICS environments today.