A Statistical Analysis Framework for ICS Process Datasets

Federico Turrin
University of Padua
Padua, Italy
turrin@math.unipd.it

Alessandro Erba
CISPA Helmholtz Center for Information Security
Saarbrücken Graduate School of Computer Science,
Saarland University
Saarbrücken, Germany
alessandro.erba@cispa.saarland

Nils Ole Tippenhauer
CISPA Helmholtz Center for Information Security
Saarbrücken, Germany
tippenhauer@cispa.saarland

Mauro Conti
University of Padua
Padua, Italy
conti@math.unipd.it

ABSTRACT

In recent years, several schemes have been proposed to detect anomalies and attacks on Cyber-Physical Systems (CPSs) such as Industrial Control Systems (ICSs). Based on the analysis of sensor data, unexpected or malicious behavior is detected. Those schemes often rely on (implicit) assumptions on temporally stable sensor data distributions and invariants between process values. Unfortunately, the proposed schemes often do not perform optimally, with Recall scores lower than 70% (e.g., missing 3 alarms every 10 anomalies) for some ICS datasets, with unclear root issues.

In this work, we propose a general framework to analyze whether a given ICS dataset has specific properties (stable sensor distributions in normal operations, potentially state-dependent), which then allows to determine whether certain Anomaly Detection approaches can be expected to perform well. We apply our framework to three datasets showing that the behavior of actuators and sensors are very different between Training set and Test set. In addition, we present high-level guides to consider when designing an Anomaly Detection System.

CCS CONCEPTS

• **Security and privacy** → Network security; **Intrusion detection systems**.

KEYWORDS

Industrial Control Systems; Anomaly Detection; Dataset

ACM Reference Format:
Federico Turrin, Alessandro Erba, Nils Ole Tippenhauer, and Mauro Conti. 2020. A Statistical Analysis Framework for ICS Process Datasets. In *2020 Joint Workshop on CPS&IoT Security and Privacy (CPSIOTSEC'20), November 9, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3411498.3419961

CPSIOTSEC'20, November 9, 2020, Virtual Event, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8087-4/20/11…$15.00
https://doi.org/10.1145/3411498.3419961

1 INTRODUCTION

Industrial Control System (ICS) security is fundamental to guarantee the safety and reliability of industrial plants. Devices deployed in ICS are often relatively old (e.g., > 10 years) and do not feature even basic security schemes. A promising solution to secure ICS are Process-Based Anomaly Detectors [4] that analyze the system sensor readings and can be easily integrated with ICSs.

A number of datasets to design and study Anomaly Detectors have been published (e.g., BATADAL [10], SWaT [8], and WADI [3]). Those datasets consist of multivariate time series of sensor readings that occurred in an ICS (real plant, testbed, or simulation). Datasets for ICS anomaly detection are often provided in different data captures. Usually, there are at least two data captures. The first (generally used as a Training set) contains data collected during normal operating conditions. The second (generally used as the Test set) contains data collected while attacks are occurring.

Sensor data contained in ICS datasets depends on the control logic configuration applied to the system. According to this system configuration, data can be differently distributed even for normal operating conditions between the two datasets. This can causes false alarms in anomaly and attack detection. To the best of our knowledge, no systematic analysis of the ICS dataset was done before although it might have a big (overlooked) impact on the performance of the detection schemes. Indeed, if we analyze results obtained by *state-of-the-art* detectors for ICS in terms of Recall score (i.e., True Positive Rate), rarely Recall surpasses 70% meaning that 3 alarms are missing every 10 anomalies. This is a non-negligible miss-rate, especially if we compare this result to what the application of Machine Learning techniques achieves in other classification tasks, e.g., image classification over CIFAR-10 dataset [7].

In this work, we propose a general framework to check whether a given ICS dataset fulfills the properties of the signal that are assumed to hold to detect anomalies (stable sensor distributions in normal operations, potentially state-dependent). Our framework compares different versions of ICS datasets captured from the same industrial plant (testbed or simulation). This helps to understand their degree of similarity and guides the design of an anomaly detector. If two datasets from the same testbed and recorded at different times have the same data distribution, it means that we can feed the Anomaly Detection System with more Training data, increasing the training phase performances. On the other hand, if two datasets do not belong to the same distribution it may imply that the initial condition of the recording phase were different, or maybe some sensors ruined over time. These aspects must be taken into account while designing an anomaly detector.

Our main contributions are:

- We present a framework to evaluate the suitability of a dataset for Anomaly Detection tasks.
- The results of our framework applied to BATADAL, WADI, and SWaT, three water distribution systems dataset by taking into consideration the state of the actuators and the statistical distribution of the sensors measures.
- A high-level guideline of best practice to take into consideration when designing an Anomaly Detection System with the presented dataset.

2 BACKGROUND

2.1 Industrial Control Systems

ICS security is a crucial topic to guarantee the proper functioning of an ICS. Historically attacks to ICS have occurred [11], harming the safety and reliability of the industrial plant. Security of ICS is challenging since devices are often outdated and communication among devices needs to be retro-compatible with insecure protocols that do not support authentication or encryption.

For this reason, in recent years, process-based anomaly detectors were proposed to overcome this technology and security gap. Process-based anomaly detectors monitor process sensor readings and detect anomalous deviations from system expected behavior.

2.2 Kolmogorov–Smirnov Test

The Kolmogorov–Smirnov (K-S) test is a non-parametric statistical test. Given two distribution A and B, the K-S test measures whether the observations in A and B belong to the same probability distribution. For the implementation of the used K-S test [1], the *Null Hypothesis* H_0 states that the two independent samples considered, A and B, are drawn from the same continuous distribution.

Given two cumulative distribution function $F_A(x)$ and $F_B(x)$ the K-S statistic for the two function $F_A(x)$ and $F_B(x)$, indicated as K-S score in the reminder of the paper, is the maximal difference between their Empirical Cumulative Distribution Functions (ECDF):

$$\text{K-S score} = \sup_x \mid F_A(x) - F_B(x) \mid . \tag{1}$$

As reported in [1], if the K-S score is small or the p-value is high, then we cannot reject the hypothesis, and therefore the two samples A and B belong to the same distribution.

3 FRAMEWORK

In this section, we present the framework used to analyze sensor data distributions between ICS datasets. Our Framework, depicted in Figure 1, is composed of an analysis of the sensors distribution on the entire dataset (Section 3.1), and a state-based analysis (Section 3.2 and Section 3.3). In particular, the upper side of the Figure shows the analysis performed on the whole dataset, which output the K-S score for each couple of sensors considered. Instead, the bottom side shows the state-based analysis which output the percentage of common states between two datasets considered and the average K-S score of each sensor after the state division.

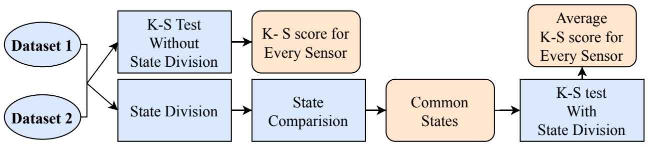
Dataset 1
Dataset 2
K-S Test Without State Division
K- S score for Every Sensor
Average K-S score for Every Sensor
State Division
State Comparision
Common States
K-S test With State Division

Figure 1: Framework showing the steps applied for the analysis. Orange boxes represent the obtained output.

3.1 K-S Test Without State Division

Given two sensor readings from two different datasets (e.g., Training set and Test set), we used the K-S test to compare the sensor readings data distribution in the two datasets to verify if the data for a given sensor belongs to the same distribution. We used the *MinMaxScaler* to scale all the sensors' values with respect to the first of the two datasets considered. Moreover, if one of the dataset considered contains attacks (i.e., is a Test set), we removed the rows with attacks and we kept only the *Normal* samples. The goal of this analysis is to compare if the distribution of the sensors under normal conditions is equally distributed.

According to Equation 1, two distributions collected from the same sensor in different dataset pass the K-S test if its K-S score is lower than 0.20 and a p-value greater than 0.05. If this condition is true, then the *Null Hypothesis* H_0 cannot be rejected and the two sensors considered record the same process distribution.

3.2 State Comparison

The second element that we considered in our analysis is *System State*. We define *System State*, or state, as the combination of the actuators values at a given time step. For example, if the systems is composed of three pumps (P1, P2, P3) connected to two water tanks (T1, T2), a *System State* $S1$ where *P1: Open*, *P2: Close*, and *P3: Open* will be represented by the vector $S1 = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$.

If the dataset is composed of n actuators with two discrete possible values (i.e., 0 = OFF and 1 = ON), then the *System State* space can be composed at most by 2^n states. Every time a certain *System State* occurs, the physical process is supposed to have the same statistical behavior. Given the state S1 from the previous example, it is reasonable to suppose that tank T1 connected to inlet pump P1 and outlet pump P2, would raise its level while tank T3 connected to inlet pump P2 and outlet pump P3 are emptying. It is reasonable to assume this hypothesis if the internal logic of the controllers and the conditions that change the state of the actuators are not updated while the system is operating. In our analysis, given two data datasets, we first extract all the different states in the dataset, then we count how many states are common among the two versions of the dataset and therefore how similar are the operational conditions of the two datasets. For example, given two datasets *df1* and *df2*, we will calculate the percentage of states of *df2* that are also present if *df1* and vice versa.

3.3 K-S Test With State Comparison

For each common *System State* between the two datasets (as reported in Section 3.2) we extracted the sensor readings that occured during a given state. Then we applied the K-S test to each couple of sensors in the different datasets to identify which of them maintains the same behavior in the given state.

If the two datasets considered have n common states, for each sensor we would compute n K-S scores. For a given sensor, K-S scores are then averaged according to Equation 2, to obtain the Average K-S score per sensor (across all the states).

$$\text{AVG K-S score per sensor} = \frac{1}{n}\sum_{i=0}^{n} \text{K-S score}_i. \qquad (2)$$

A sensor passes the test if its Average K-S score per sensor is less then 0.20 (same as in Section 3.1). We must note that not all states appear with the same frequency, some of them are very short, while others are very long, and this could affect the results.

4 BATADAL ANALYSIS

4.1 Dataset Description

BATADAL dataset was released with 'The Battle Of The Attack Detection Algorithms' [10], a competition to detect cyber attacks on water distribution networks. BATADAL competition dataset was generated with [9] which allows to model the hydraulic response of a water distribution network under attack. The dataset is divided into three data chunks: the first contains the sensor readings collected during 365 days of normal operations, the second and the third contain the sensor data collected during 14 cyber attacks.

BATADAL dataset features a collection of 43 sensors and actuators, i.e., water levels, the pressure at pumping stations, flow, and actuator status. There are two versions of the dataset available, the original one[1] (i.e., *TrainV1* and *TestV1*) contains replay attacks to conceal the true system state. A second version[2] (*i.e., TrainV2*) contains sensor readings without concealment. This second version is composed of two different Test sets (i.e., *TestV2.1* and *TestV2.2*).

4.2 K-S Test Without State Division

We performed the K-S test over the various versions of the BATADAL dataset. All the different versions of the dataset have the same sensors and actuators. The results are reported in Table 1. Among the three different datasets considered in this work, BATADAL is the one with a lower K-S score between the sensors (i.e., it passes the K-S test with a lower score than the other datasets). This is probably because BATADAL is generated synthetically, therefore it is subject to less noise. In Figure 2, the Violinplot is presented as example of sensors not passing the K-S test. The Violinplot confirms the effective different distribution of the sensors that did not pass the K-S test and sensors which pass the K-S test.

4.3 State comparison

We reported in Table 2 the percentage of common state states between a couple of datasets considered. The percentage of common states between the different versions of BATADAL is higher than the other dataset considered in this work. Again, this can be due to the simulated environment on which BATADAL is generated.

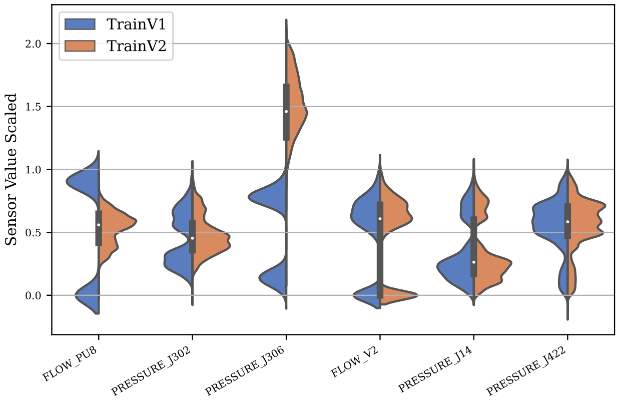

Figure 2: Violinplot which compare the distribution between the sensors of BATADAL *TrainV1* and *TrainV2*. In particular *FLOW_PU8*, *PRESSURE_J302*, and *PRESSURE_J306* failed the test, while *FLOW_PU2*, *PRESSURE_J14*, and *PRESSURE_J422* passed the test.

		Train		Test		
		V1	V2	V1	V2.1	V2.2
Train	V1	-	-	-	-	-
	V2	26/31	-	-	-	-
Test	V1	30/31	25/31	-	-	-
	V2.1	26/31	31/31	25/31	-	-
	V2.2	31/31	31/31	27/31	31/31	-

Table 1: Number of BATADAL sensors which pass the K-S test between the different dataset considered.

		Train		Test		
		V1	V2	V1	V2.1	V2.2
Train	V1	-	39.77%	96.61%	76.00%	87.80%
	V2	50.00%	-	50.84%	76.00%	82.92%
Test	V1	81.42%	34.09%	-	66.00%	80.49%
	V2.1	54.28%	43.18%	55.93%	-	87.80%
	V2.2	51.42%	38.63%	55.93%	72.00%	-

Table 2: Percentage of BATADAL state in common between the two datasets considered (i.e., how many states of the column dataset are contained in the row dataset).

4.4 K-S Test With State Comparison

We reported the results of the K-S test on sensors divided into states in Table 3. In this case, the analysis of the system state performs worse than the analysis of the whole dataset. However, results show that at least eight sensors pass the test among all the common states in the datasets.

[1] http://www.batadal.net/data.html

[2] https://github.com/scy-phy/ICS-Evasion-Attacks

		Train		Test		
		V1	V2	V1	V2.1	V2.2
Train	V1	-	-	-	-	-
	V2	10/31	-	-	-	-
Test	V1	10/31	9/31	-	-	-
	V2.1	8/31	13/31	8/31	-	-
	V2.2	9/31	13/31	8/31	11/31	-

Table 3: Number of BATADAL sensors which pass the K-S test after the state division.

4.5 Discussion of Results

The overall result on BATADAL confirms the similarity of the distribution of the sensors without state division and the common states in the different versions of the dataset. The K-S test with the state division shows that at least eight sensors preserve a common behavior in a given state. We can conclude that BATADAL is a suitable dataset to use in Anomaly Detection tasks in a state-less analysis even if a state-based analysis can also be performed by deriving invariants on the eight sensors previously identified.

5 SWAT ANALYSIS

5.1 Dataset Description

The Secure Water Treatment (SWaT) plant [8] is a scaled-down water treatment plant able to produce five US gallons/hr of filtered water. It consists of six steps process in which the water is gradually filtrated and purified. Each step is equipped with a precise number of sensors and actuators.

There are many versions of the SWaT dataset, opening also a fragmentation problem when using it as a benchmark for the detection algorithms. In the following, we reported the complete list. We refer to the network traffic as the *pcap* dump of the network communications and as physical data the value of the sensors and actuators recorded:

- *SWaT.A1 & A2_Dec 2015* contains 11 days of continuous operation, seven under normal operation (i.e., *TrainV1*) and four days normal operations with attacks (i.e., *TestV1*). Among the two sets of physical data under normal conditions provided we considered Version 1 where the authors removing the first 30 minutes of data corresponding to water drainage.
- *SWaT.A3_Jun 2017* contains 136 hours of traffic and physical data of normal operation without attacks (i.e., *TrainV2*).
- *SWaT.A4 & A5_Jul 2019* includes three hours of normal operating condition (i.e., *TrainV3*) and 1 hour during which six attacks were carried out (i.e., *TestV3*). Among the two versions of this dataset provided, we considered only the second one which contains a correction on a column name.
- *SWaT.A6_Dec 2019* contains a series of malware infection attacks on the SWaT Engineering Workstation. The malware attacks include Historian Data Exfiltration attack and Process Disruption attacks. This version includes three hours of SWaT running under the normal operating condition and one hour in which six attacks were carried out (i.e., *TestV4*).
- *SWaT.A7_May 2020* contains five recording of both network traffic and physical data over 4 days under normal operation (i.e., *TrainV5*). Each run lasted four hours. Between each run, there is a break of 30 minutes and a “reset run” of one hour, for which no data was collected. We removed the first eight hours because they are composed of empty data.

		Train			Test		
		V1	V2	V5	V1	V3	V4
Train	V1	-	-	-	-	-	-
	V2	2/25	-	-	-	-	-
	V5	0/25	1/25	-	-	-	-
Test	V1	6/25	2/25	0/25	-	-	-
	V3	1/25	2/25	2/25	1/25	-	-
	V4	2/25	3/25	3/25	2/25	3/25	-

Table 4: Number of SWaT sensors which pass the K-S test between the different dataset considered.

		Train			Test		
		V1	V2	V5	V1	V3	V4
Train	V1	-	<0.01%	<0.01%	53.16%	0.0%	0.0%
	V2	<0.01%	-	28.00%	0.0%	50.00%	0.0%
	V5	<0.01%	25.52%	-	0.0%	0.08%	0.0%
Test	V1	57.53%	0.0%	0.0%	-	0.0%	0.0%
	V3	0.0%	11.46%	31.81%	0.0%	-	0.0%
	V4	0.0%	0.0%	0.0%	0.0%	0.0%	-

Table 5: Percentage of SWaT state in common between the two datasets considered (i.e., how many states of the column dataset are contained in the row dataset).

5.2 K-S Test Without State Division

Among the three datasets considered SWaT is the one with a highest K-S score between the sensors. This result confirms the observation of other works [6, 12]. Therefore SWaT has a very variable operational behavior. In particular, the version with the most different behavior is the last one, i.e., *TrainV5*. Version five of SWaT implements 18 new sensors and six new actuators which cannot be compared with the other versions, therefore we decided to remove them. Furthermore, we removed five rows from this last SWaT version due to “NaN” values and “Bad Input” values. The different behavior of this version may be due to different usage of the testbed with respect to the modality of the prior years, but it could also be due to sensors degradation.

5.3 State comparison

Table 5 reports the results of the state comparison between the different versions of SWaT. Among all the datasets considered this is the worst result. This may be due to a change of logic in the actuators code. After the first version of the dataset, the authors introduced a new state in the actuators (i.e., actuators now also have the state value “2”). Therefore both the Version 1 of the Training set and Test set have no states in common with the other versions. The *TestV4* version does not have any states in common with the other versions. This could be probably due to a reconfiguration of the internal logic, or a different operating condition of the testbed.

		Train			Test		
		V1	V2	V5	V1	V3	V4
Train	V1	-	-	-	-	-	-
	V2	6/25	-	-	-	-	-
	V5	3/25	1/25	-	-	-	-
Test	V1	0/25	N/A	N/A	-	-	-
	V3	N/A	1/25	2/25	N/A	-	-
	V4	N/A	N/A	N/A	N/A	N/A	-

Table 6: Number of SWaT sensors which pass the K-S test after the state division.

5.4 K-S Test With State Comparison

Table 6 reports the results of the analysis based on the K-S test with the state division. Due to the absence of common states in many combinations (as reported is Section 5.3) there are many results set to N/A, since it was impossible to perform the analysis in those cases. The higher number of sensors that pass the K-S test in the comparison between *TrainV1* and *TrainV2*, *TrainV5* is because these versions have only a few common states, and these few states have effectively a very common pattern. However, in this case, the other comparisons highlight a very similar distribution between the sensors in the same state, implying that in these cases the state division performs better than the analysis on the whole dataset.

5.5 Discussion of Results

Among all the dataset considered, SwaT is the one with the most different behavior between the various version considered, even in the Training set and Test set related to the same dataset release. These result confirms also the observation in [6, 12]. Therefore, we can conclude that due to its instability, SWaT represents a very challenging dataset to perform a Anomaly Detection.

6 WADI ANALYSIS

6.1 Dataset Description

WAter DIstribution (WADI) [3] is a realistic ICS testbed that reproduces a water distribution network. It comprises two elevated reservoir tanks, six consumer tanks, and a return tank (for water recycling purposes). It is controlled by 103 sensors and actuators connected to three PLCs. Each PLC controls one of the following stages: P1 (Primary supply and analysis), P2 (Elevated reservoir with Domestic grid and leak detection), and P3 (Return process).

The dataset is divided into two data chunks, the first contains 14 days of normal operations, the second contains 15 attacks on the physical process that occurred over two days of operations [5].

There are two versions of the WADI dataset available on request. As reported by the authors of the dataset, the newer version (i.e., V2) resolves some problems of the data contained in the first version (i.e., V1). The new version refers to the same testbed run (i.e., V1) but with about 35% fewer lines.

6.2 K-S Test Without State Division

The second versions of WADI have new sensors with respect to the first version. In particular, both the Training set and Test set of the second version of WADI contains 4 columns with entire "NaN" values. We removed them, together with the rows with "NaN" values. In Table 7 we reported the results of the K-S test between the different versions of WADI. As we can see WADI has a high number of sensors passing the test, therefore the distributions of the sensors are similar. When comparing the Training sets (since they are the same dataset with some data cropped in Version 2), we can observe that the test is passed by almost all the sensors.

		Train		Test	
		V1	V2	V1	V2
Train	V1	-	-	-	-
	V2	80/84	-	-	-
Test	V1	69/84	69/84	-	-
	V2	69/84	69/84	84/84	-

Table 7: Number of WADI sensors which pass the K-S test between the different dataset considered.

		Train		Test	
		V1	V2	V1	V2
Train	V1	-	100%	82.91%	83.33%
	V2	77.72%	-	79.19%	79.59%
Test	V1	25.44%	31.26%	-	100%
	V2	25.44%	31.26%	99.49%	-

Table 8: Percentage of WADI state in common between the two datasets considered (i.e., how many states of the column dataset are contained in the row dataset).

When comparing the Train sets with the Test sets, our framework reveals that there are 15 sensors that do not keep the same statistical behavior between the Train set and Test set.

6.3 State Comparison

In Table 8 we reported the results obtained from the intersection of states between the different versions of the WADI dataset. As we can see the intersections above the diagonal of the matrix have the highest score, it means that *TrainV2*, *TestV1*, and *TestV2* have a high number of states contained in *TrainV1* (partially due to the higher dimension of *TrainV1* in term of size) and contains about 1900 different states. These results confirm that the *TrainV2* is a subset of *TrainV1* while *TestV1* and *TestV2* are basically the same dataset with some minor changes (one state of *TestV1* not present in *TestV2*).

6.4 K-S Test With State Comparison

The results are reported in Table 9. As we can see, WADI obtains good results in this analysis. When we compare the two Training set, we can identify that there is only one sensor that fails the stateful K-S test. This is due to the data prepossessing that deleted the occurrences of a value for that sensor. When we compare Training with the Test set, our framework identifies 18 sensors that maintain the same statistical behavior in a given state.

		Train		Test	
		V1	V2	V1	V2
Train	V1	-	-	-	-
	V2	83/84	-	-	-
Test	V1	18/84	18/84	-	-
	V2	18/84	18/84	83/84	-

Table 9: Number of WADI sensors which pass the K-S test after the state division.

6.5 Discussion of Results

The results on WADI highlight a similar behavior among the different versions of the dataset. From the state-less analysis, but also from the state-based point of view, there are several consistent sensors. We note that both the versions of the Training set (i.e., *TrainV1* and *TrainV2*) and the Test set (i.e., *TestV1* and *TestV2*) are mostly equivalent with minor differences. We conclude that WADI represents a good (but challenging) dataset to perform anomaly detection, from both the consistent sensor behavior and the training dataset size point of view.

7 RELATED WORK

Anomaly detection in ICS is a popular topic in the related work. There are many solutions proposed by different authors ranging from Invariant-Based approaches [4], Machine-Learning Based approaches [6], or Fingerprint-based approaches [2]. Generally, to validate the detection approach the authors use well-known ICS datasets. In particular, the three most used dataset are BATADAL [10], SWaT [8], and WADI [3]. However, in the majority of anomaly detectors proposed in the literature, the authors do not consider the analysis of the statistical distribution of the datasets. In [6, 12] the authors observed that the distribution of some sensors in these datasets changes not only among different versions, but also in the same Training set and Test set release. In particular, in [6] the authors performed a modified version K-S test on one version of each dataset, highlighting the different distribution between the Training set and the Test set. However, the work proposed did not investigate deeply the distribution of the different versions over time. Moreover, in [6] the authors considered in the K-S test also the actuators, which are composed of binary values and do not follow any type of distribution.

8 CONCLUSIONS

We presented a framework to evaluate the process data contained in datasets for Anomaly Detection. The framework applies the K-S test to evaluate how the data are distributed among two data captures. It also investigates which states of actuators often occur over the system, and checks the data distribution in a given state.

We applied our framework to three ICS datasets publicly available, and we found that SWaT dataset contains certain instability in the sensor readings. The first step of our framework applies K-S test without state division. By using this test, we found that for BATADAL and WADI sensors readings are stable with at minimum 25/31 and 69/84 sensors passing the test respectively. Conversely, in SWaT the K-S test reveals different distributions for the sensors, with at maximum 6/25 sensors passing the test. The second step of our framework checks the share of states among two data captures. If we compare the share between Train and Test sets, the analysis reveals a good share of states in BATADAL dataset (with a maximum 96.61% of share), followed by WADI (maximum 83.33% of share) and SWaT (with a maximum 53.16%). As a last step, our framework performs the K-S test to compare the behavior during system states. This analysis reveals that in BATADAL there are at least 8/31 sensors that on average pass the K-S test in all the states, while in WADI they are 18/84 and in SWaT at most 6/25.

As a result of the application of our framework, we identified a list of high-level guidelines that could help the development of an Anomaly Detection System.

Check data distributions. Use a preliminary statistical analysis to verify if the sensors of the Training Set and the Test set belong to the same distribution. If the distribution of some sensors is different, consider removing them, as they could raise false alarms.

Dataset instability. We observed that after an attack, the testbed remains unstable for a long time. If the dataset does not recover after the end of an attack, his behavior will be *Anomalous* even if flagged as *Normal*. When designing a dataset the authors should take this in mind, and for example, reset the system to restore the normal conditions. Another solution could be to add another label to classify the dataset, e.g., *System unstable*.

System configuration. By the designer of the dataset side it would be useful to specify the system settings used for every data capture, e.g., the logic of the controllers, the alarm values, and triggers.

REFERENCES

[1] [n.d.]. Kolmogorov-Smirnov test on 2 samples. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ks_2samp.html, Last Accessed on : 2020-07-20.

[2] Chuadhry Mujeeb Ahmed, Martin Ochoa, Jianying Zhou, Aditya P Mathur, Rizwan Qadeer, Carlos Murguia, and Justin Ruths. 2018. Noiseprint: Attack detection using sensor and process noise fingerprint in cyber physical systems. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*. 483–497.

[3] Chuadhry Mujeeb Ahmed, Venkata Reddy Palleti, and Aditya Mathur. 2017. WADI: A Water Distribution Testbed for Research in the Design of Secure Cyber Physical Systems. In *Proceedings of the Workshop on Cyber-Physical Systems for Smart Water Networks)*. ACM.

[4] Cheng Feng, Venkata Reddy Palleti, Aditya Mathur, and Deeph Chana. 2019. A Systematic Framework to Generate Invariants for Anomaly Detection in Industrial Control Systems.. In *Proceedings of the Network and Distributed System Security Symposium(NDSS)*.

[5] iTrust, SUTD. 2017. WADI datatset. https://itrust.sutd.edu.sg/itrust-labs_datasets/dataset_info/, Last accessed on: 2020-06-30.

[6] Moshe Kravchik and Asaf Shabtai. 2019. Efficient cyber attacks detection in industrial control systems using lightweight neural networks. *arXiv preprint arXiv:1907.01216* (2019).

[7] Alex Krizhevsky. 2009. *Learning multiple layers of features from tiny images.* Technical Report.

[8] A. P. Mathur and N. O. Tippenhauer. 2016. SWaT: a water treatment testbed for research and training on ICS security. In *2016 International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater)*. 31–36.

[9] R. Taormina, S. Galelli, H.C. Douglas, N. O. Tippenhauer, E. Salomons, and A. Ostfeld. 2019. A toolbox for assessing the impacts of cyber-physical attacks on water distribution systems. Environmental Modelling Software. *Environmental Modelling Software* 112 (02 2019), 46–51.

[10] Riccardo Taormina, Stefano Galelli, Nils Ole Tippenhauer, Elad Salomons, Avi Ostfeld, Demetrios G Eliades, et al. 2018. Battle of the attack detection algorithms: Disclosing cyber attacks on water distribution networks. *Journal of Water Resources Planning and Management* 144, 8 (2018), 04018048.

[11] Sharon Weinberger. 2011. Computer security: Is this the start of cyberwarfare? *Nature* 174 (June 2011), 142–145.

[12] Giulio Zizzo, Chris Hankin, Sergio Maffeis, and Kevin Jones. 2019. Intrusion detection for industrial control systems: Evaluation analysis and adversarial attacks. *arXiv preprint arXiv:1911.04278* (2019).