Robocoast Challenge: Measuring with AI

Report

Jesse Myrberg, Jarno Sallila, Juha Kurkela

Noux Node Oy

FI28124646

Euroopan unioni
Euroopan aluekehitysrahasto

Vipuvoimaa EU:lta 2014–2020

SATAKUNTALIITTO

Prizztech

Introduction

The challenge provider’s production equipment provides measurement data that is analyzed in this report using various analysis tools. The challenge provider’s production equipment measures a parameter of the production process that affects the quality of the end product. The end products are made in batches, where the production parameter may differ batch by batch in terms of average and standard deviation.

The challenge is to identify anomalies from each production batch, especially in terms of unexpected changes in the average and standard deviation of the process parameter. By identifying anomalies in the current batch, the challenge provider could set automatic alarms and stop the production process to ensure the quality of the end product. An example of four batches of measurement data from the parameter of interest is presented in Figure 1.

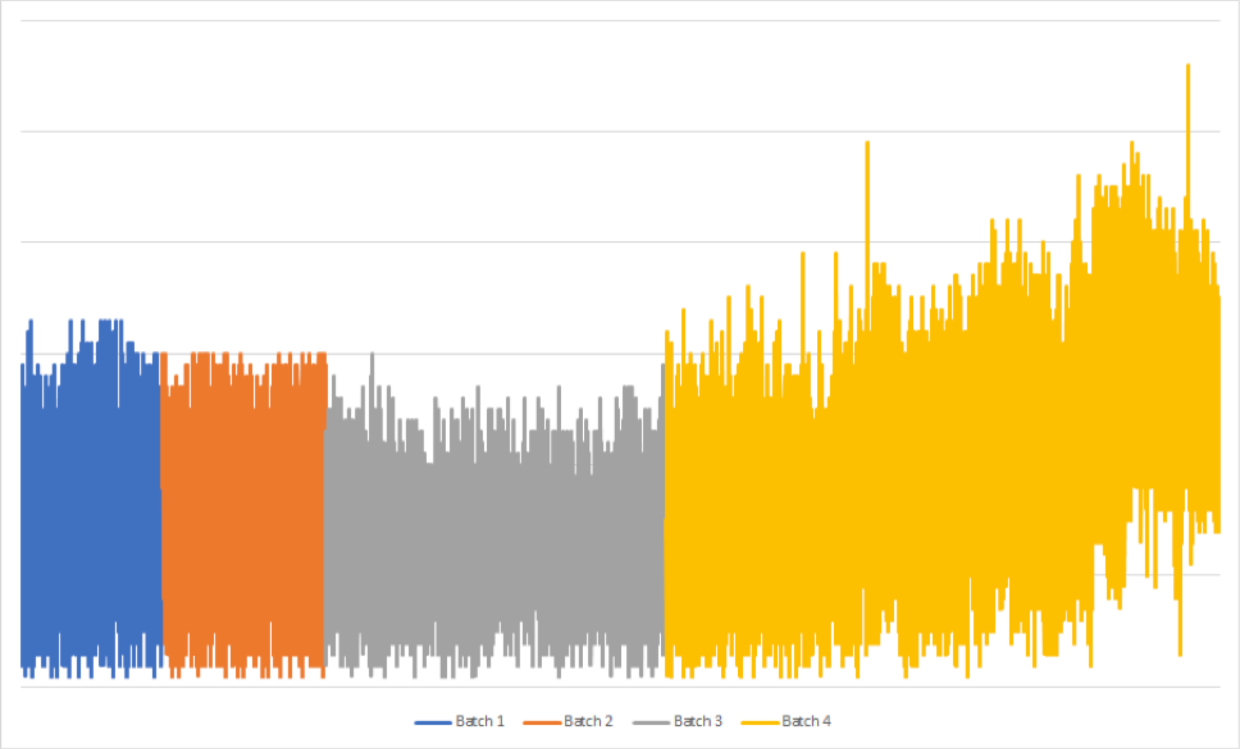


Figure 1. Example data from four production batches, where Batch 4 contains anomalies in terms of increasing average and standard deviation.

NouxCloud is a cloud based platform for industrial data collecting, warehousing and analysis. Through edge computing methods and communication protocols, data can be collected from new or already installed automation systems regardless of system components or manufacturers. NouxCloud has modules for analyzing the collected data through reports, dashboards, alerts and machine learning. The system also allows for data imports in Excel or .csv formats when a direct connection to automation system is not available.

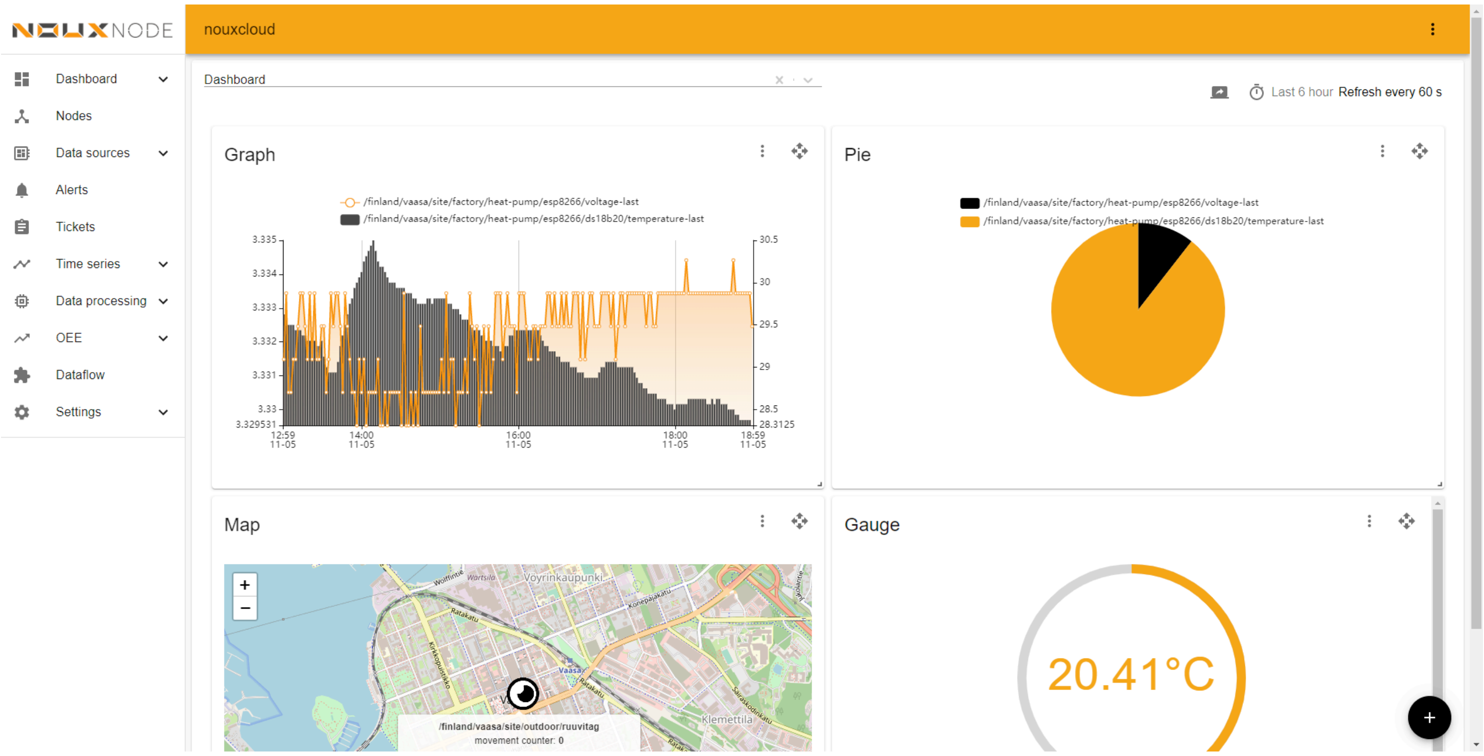

NOUXNODE
nouxcloud
Dashboard
Nodes
Data sources
Alerts
Tickets
Time series
Data processing
OEE
Dataflow
Settings
Dashboard
Last 6 hour Refresh every 60 s
Graph
/finland/vaasa/site/factory/heat-pump/esp8266/voltage-last
/finland/vaasa/site/factory/heat-pump/esp8266/ds18b20/temperature-last
3.335
3.334
3.333
3.332
3.331
3.33
3.329531
30.5
30
29.5
29
28.5
28.3125
12:59 11-05
14:00 11-05
16:00 11-05
18:00 11-05
18:59 11-05
Pie
/finland/vaasa/site/factory/heat-pump/esp8266/voltage-last
/finland/vaasa/site/factory/heat-pump/esp8266/ds18b20/temperature-last
Map
+
−
/finland/vaasa/site/outdoor/ruuvitag
movement counter: 0
Gauge
20.41°C
+

Figure 2. An overview of NouxCloud and especially its Dashboard module.

More technical details regarding NouxCloud function are presented in **Figure 3**. Automation system variables, i.e. process data is transferred via communication bus and protocols and temporarily saved within the system. Most automation systems use common communication protocols (e.g. OPC UA, S7, Modbus etc.) regardless of system components or manufacturers. The saved data can be read by using an edge app that utilizes the communication bus and protocols. The edge app can send the data securely (Transport Layer Security) to NouxCloud unfiltered or alternatively filtering or more complicated analysis can be made already before sending.

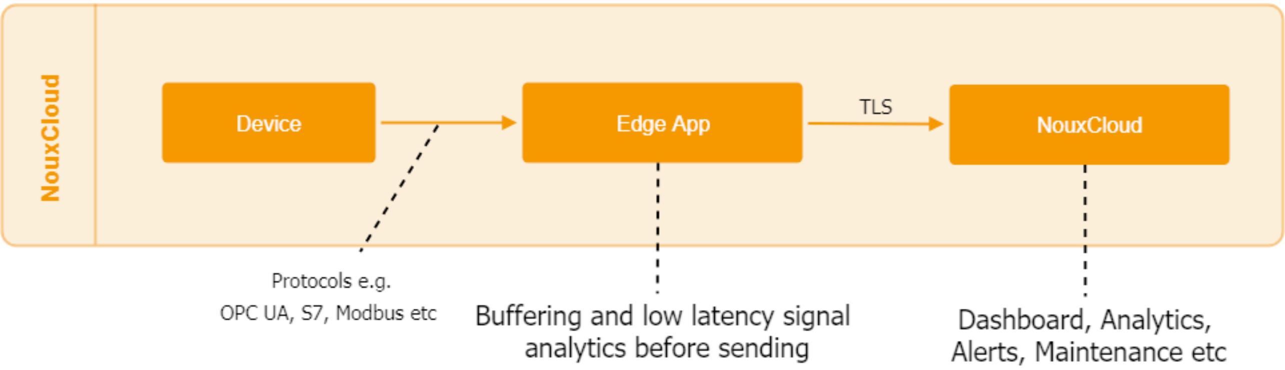

NouxCloud
Device
Edge App
TLS
NouxCloud
Protocols e.g.
OPC UA, S7, Modbus etc
Buffering and low latency signal analytics before sending
Dashboard, Analytics, Alerts, Maintenance etc

Figure 3. NouxCloud operating principle.

Methods

Problem definition

The challenge provider monitors a production parameter that is essential to the batch production quality. There are two cases which are considered as quality failure of a production batch:

- Parameter values are consistently above the batch average
- Parameter values are consistently far from the batch average

In this analysis, these cases are considered as anomalies, and they are analyzed by monitoring changes in the following three cases for each production batch:

A) Average
B) Standard deviation
C) Average and standard deviation

The production parameter data given by the challenge provider consists of ten batches, where seven of them contain data under normal operating conditions, and three of them contain anomalies (one for change in average, one change in standard deviation, and one for both anomalies at the same time). Each batch contains 1000-6000 samples with no timestamp information, adding some additional noise to the data. In the following, the production parameter of interest is also referred to as variable.

Euroopan unioni
Euroopan aluekehitysrahasto
Vipuvoimaa EU:lta 2014–2020
SATAKUNTALIITTO
Prizztech

Approach

NouxCloud is designed and optimized to handle data gathered directly from automation systems, allowing manually imported data as well. When importing data manually, the system generates timestamps automatically, allowing simulation of real time data streaming.

The analytics modules of NouxCloud are used for solving the problem in question. Because the data is in batches instead of being a continuous stream of data, some customizations are required. The problem is approached by utilizing the dataflow functionality of the software, and the ability to connect to external algorithms via an Application Programming Interface (API). The system is used for solving the problem with two different approaches:

1) Detecting changes in rolling average and standard deviation by utilizing dataflow
2) Training an isolation forest[1] anomaly detection algorithm by utilizing an API

In 1), the idea is to monitor a batch from the beginning in order to define typical values and the average level of a batch. Based on these values, alerts can be set to trigger when the new values of a batch exceed the normal values by a certain percentage. To control the number of false positives, users can set the number of data points from the beginning that are considered as normal, and also limit the value that triggers an alert (e.g. 10 % of over the normal values). The same approach is used for testing both the average and standard deviation cases.

In 2), an isolation forest algorithm is trained on data batches that have no anomalies. The given data is normalized, and four additional features are created from the normalized values with a window size of 100: rolling average, rolling standard deviation, 25 % rolling average quantile, and 75 % rolling average quantile. The model is trained on non-anomaly batches, and then evaluated against batches that are known to have anomalies. The predicted values by the model are scaled to be 0.5 or below when the data should contain no anomalies, and data samples with predicted anomaly above 0.5 are considered as an anomaly.

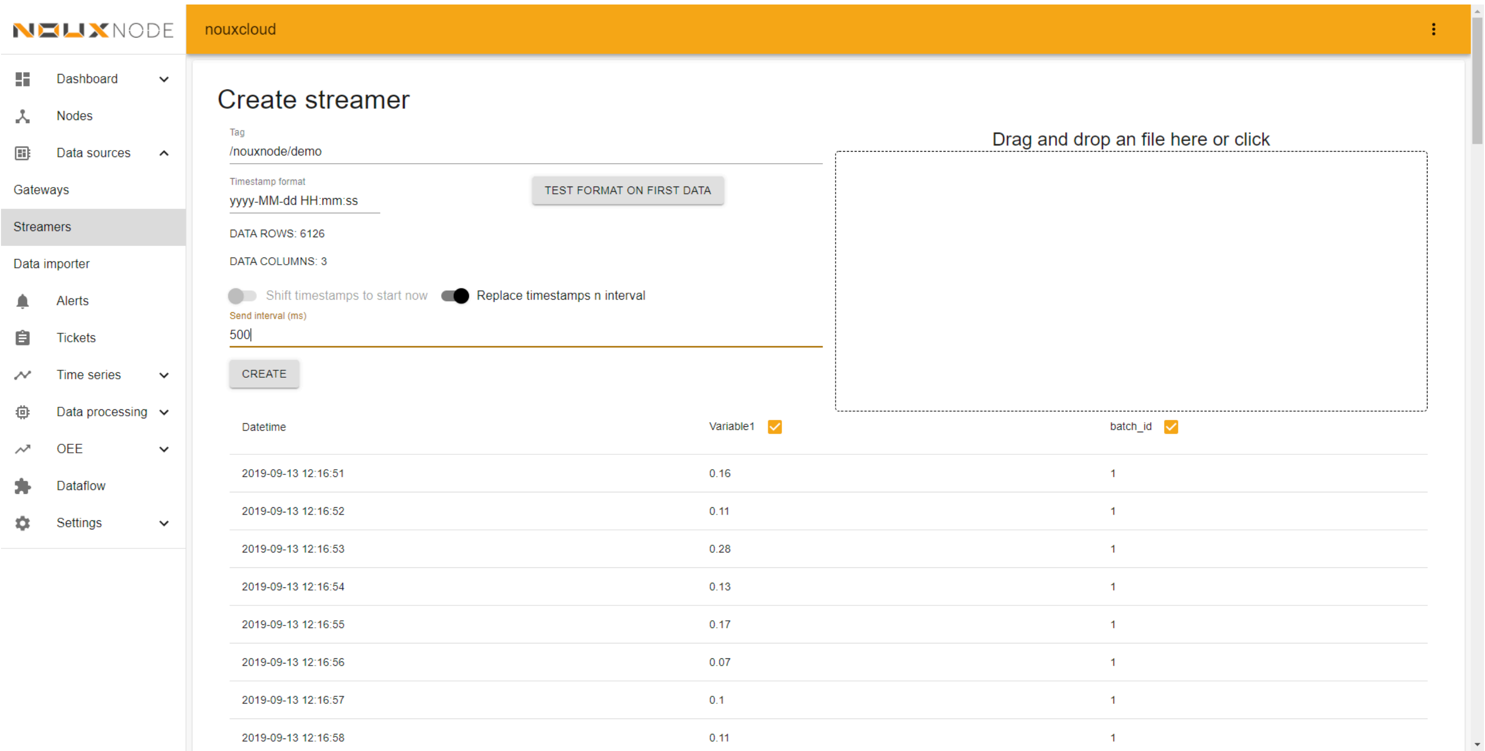
[1] Liu, F. T. (2009). Isolation Forest. ICDM '08. Eighth IEEE International Conference. DOI: 10.1109/ICDM.2008.

Results

The required steps in NouxCloud to analyze the cases are:

1. Import data
2. Configure dataflow
3. Apply the dataflow and present results

Data batch importing by NouxCloud streamer function is presented in Figure 4 and Figure 5. For approach 1), only the data batches with anomalies are imported. For approach 2), both the normal condition and anomaly batches are imported. An example of data batch with anomalies is presented as time series in Figure 6.



Create streamer

Tag: /nouxnode/demo

Timestamp format: yyyy-MM-dd HH:mm:ss

DATA ROWS: 6126

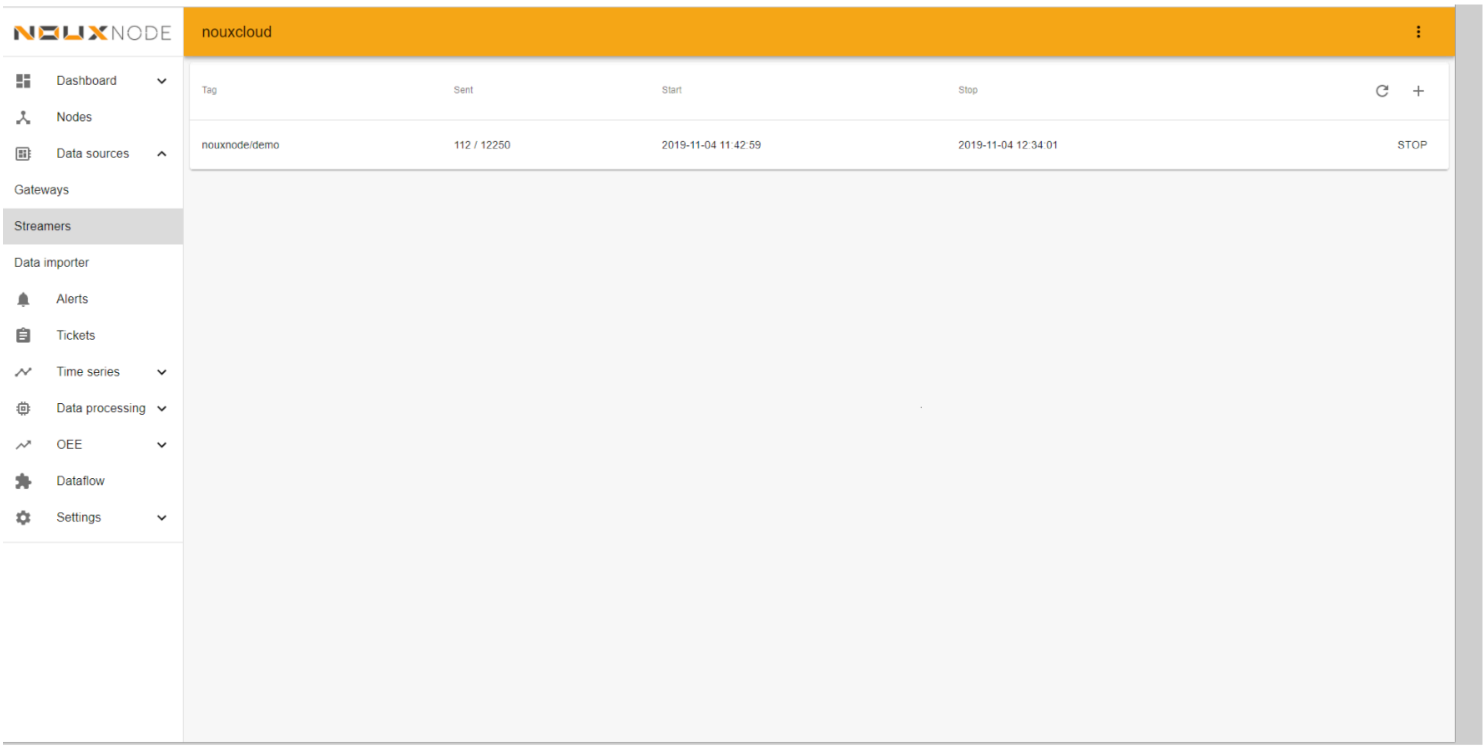
DATA COLUMNS: 3

○ Shift timestamps to start now ● Replace timestamps n interval

Send interval (ms): 500

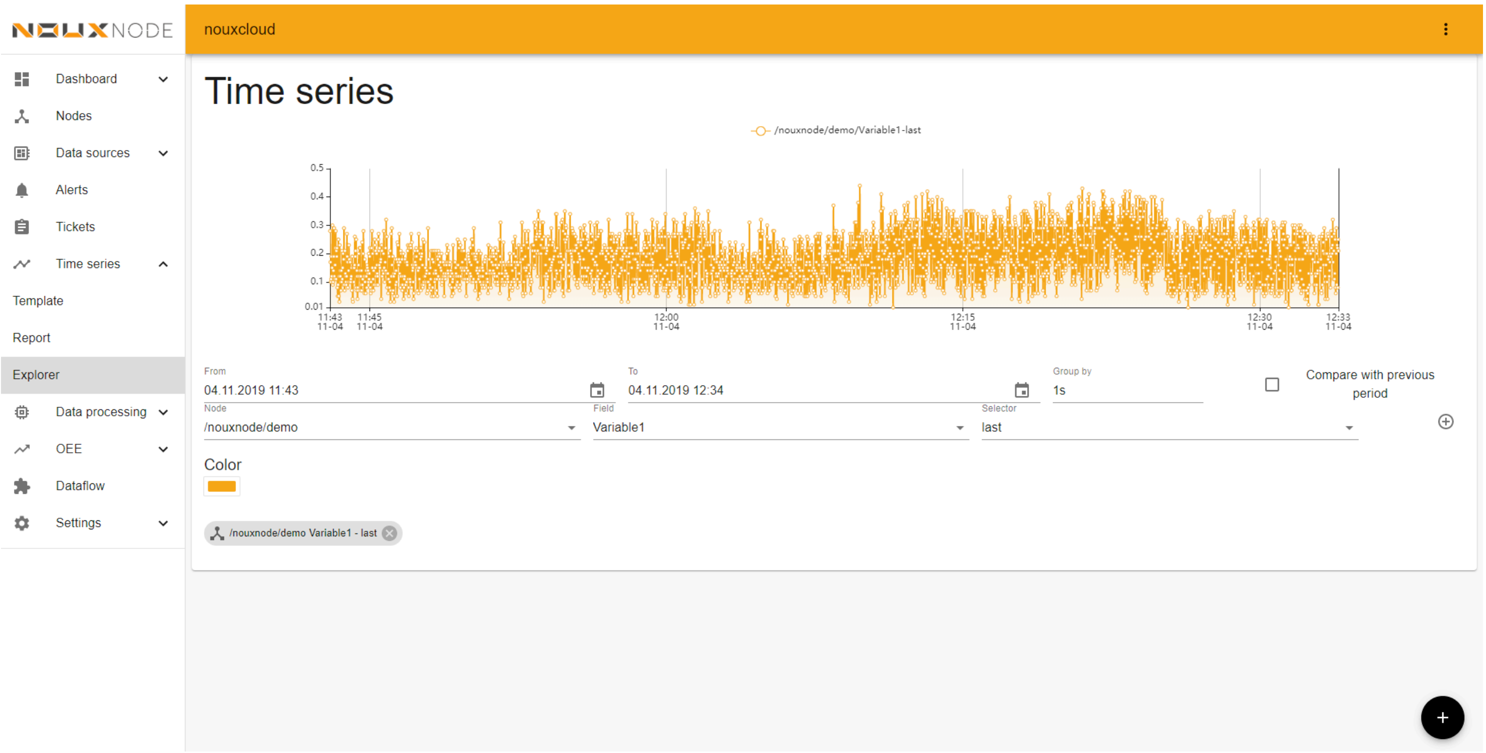
Datetime	Variable1 ☑	batch_id ☑
2019-09-13 12:16:51	0.16	1
2019-09-13 12:16:52	0.11	1
2019-09-13 12:16:53	0.28	1
2019-09-13 12:16:54	0.13	1
2019-09-13 12:16:55	0.17	1
2019-09-13 12:16:56	0.07	1
2019-09-13 12:16:57	0.1	1
2019-09-13 12:16:58	0.11	1

Figure 4. Streamer setup.



Tag	Sent	Start	Stop	
nouxnode/demo	112 / 12250	2019-11-04 11:42:59	2019-11-04 12:34:01	STOP

Figure 5. Variable values imported with the streamer tool.



Time series

From 04.11.2019 11:43

To 04.11.2019 12:34

Group by 1s

☐ Compare with previous period

Node /nouxnode/demo

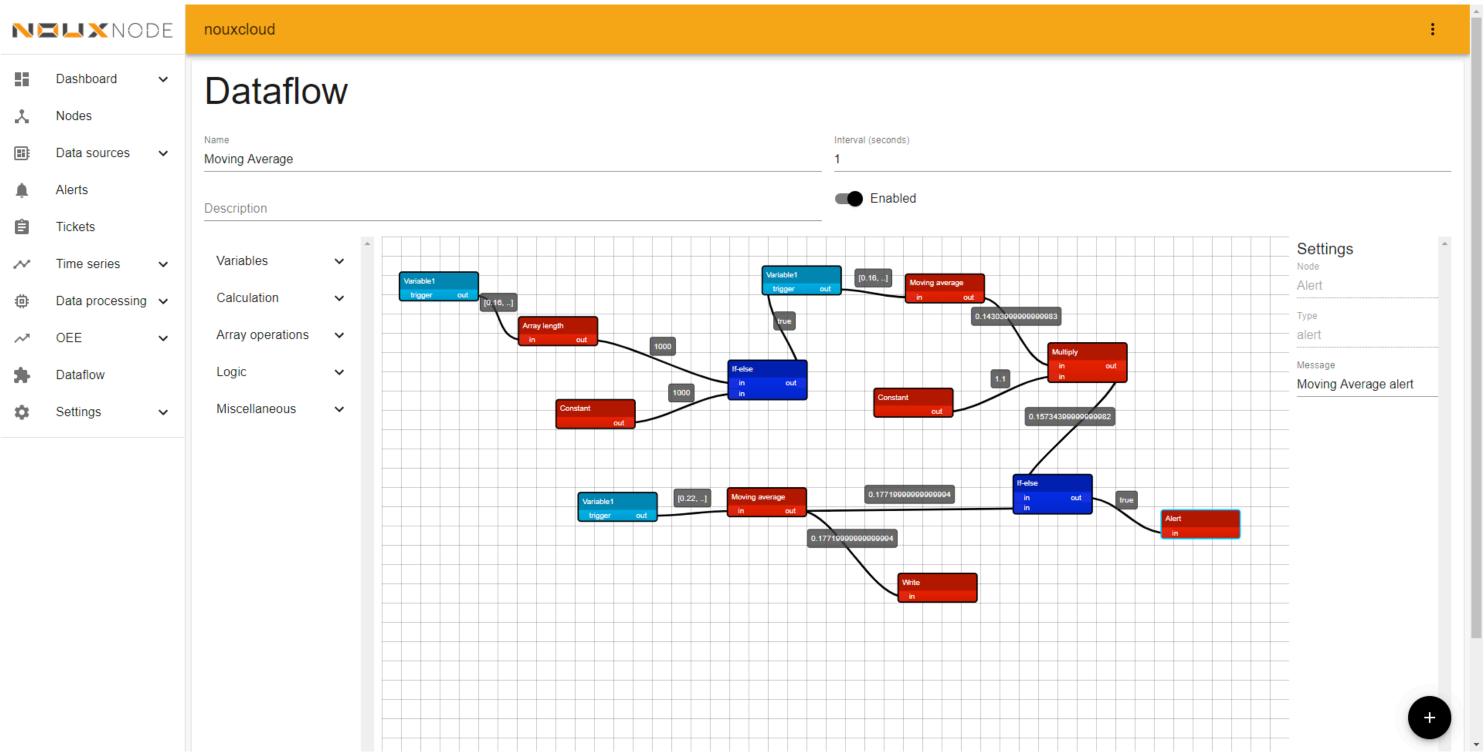
Field Variable1

Selector last

Color 🟧

Figure 6. A time series representation of the batch with anomalies imported with the streamer tool.

The dataflow module allows for applying functions to variables. This creates new derived variables which can be reused, and functions can be further on applied to them. In this analysis, both cases 1) and 2) can be configured through the dataflow module. For the sake of clarity, the average and standard deviation cases are divided into separate dataflow configurations, as demonstrated in Figure 7 and Figure 8.



Dataflow

Name: Moving Average

Interval (seconds): 1

Enabled

Description

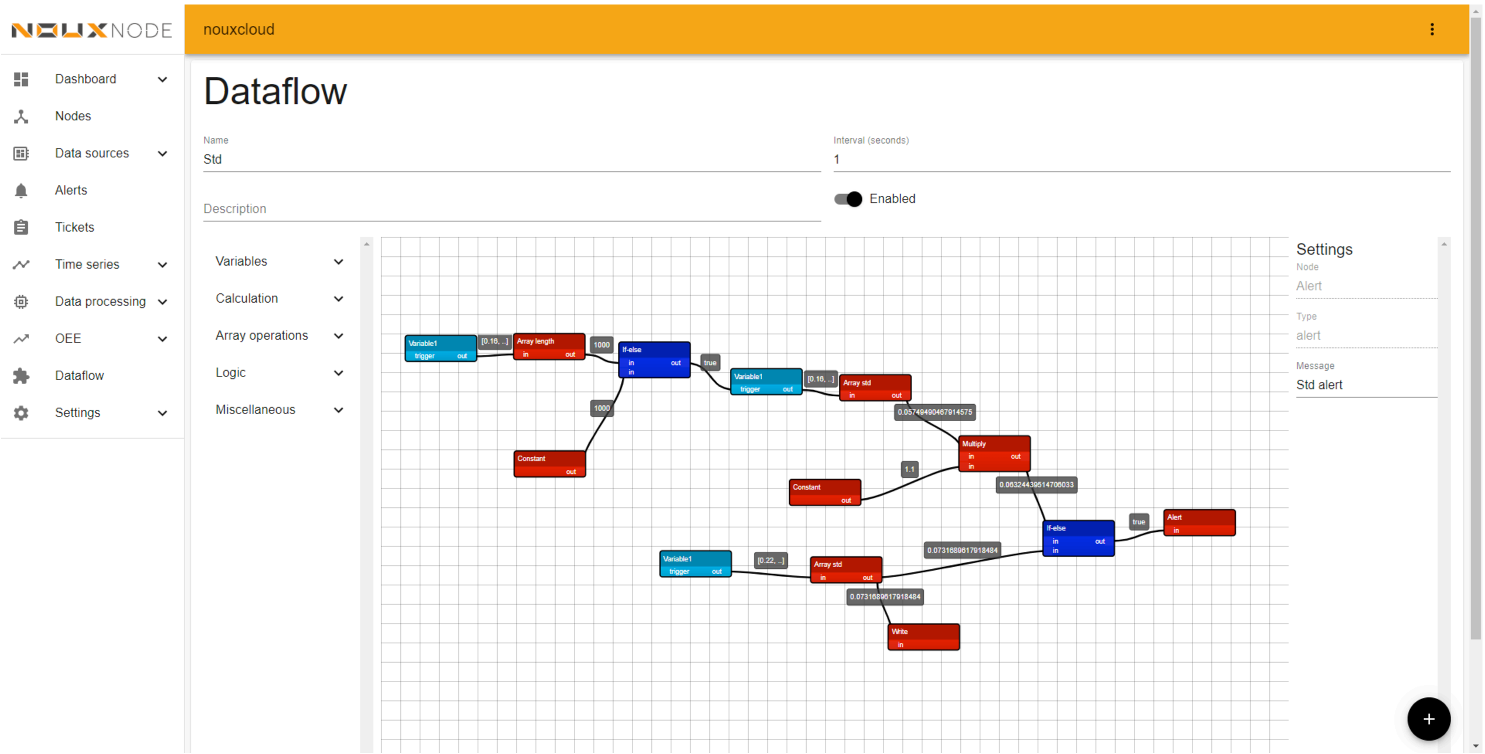
Settings

Node: Alert

Type: alert

Message: Moving Average alert

Figure 7. Moving average method dataflow configuration.



Dataflow

Name
Std

Interval (seconds)
1

Description

Enabled

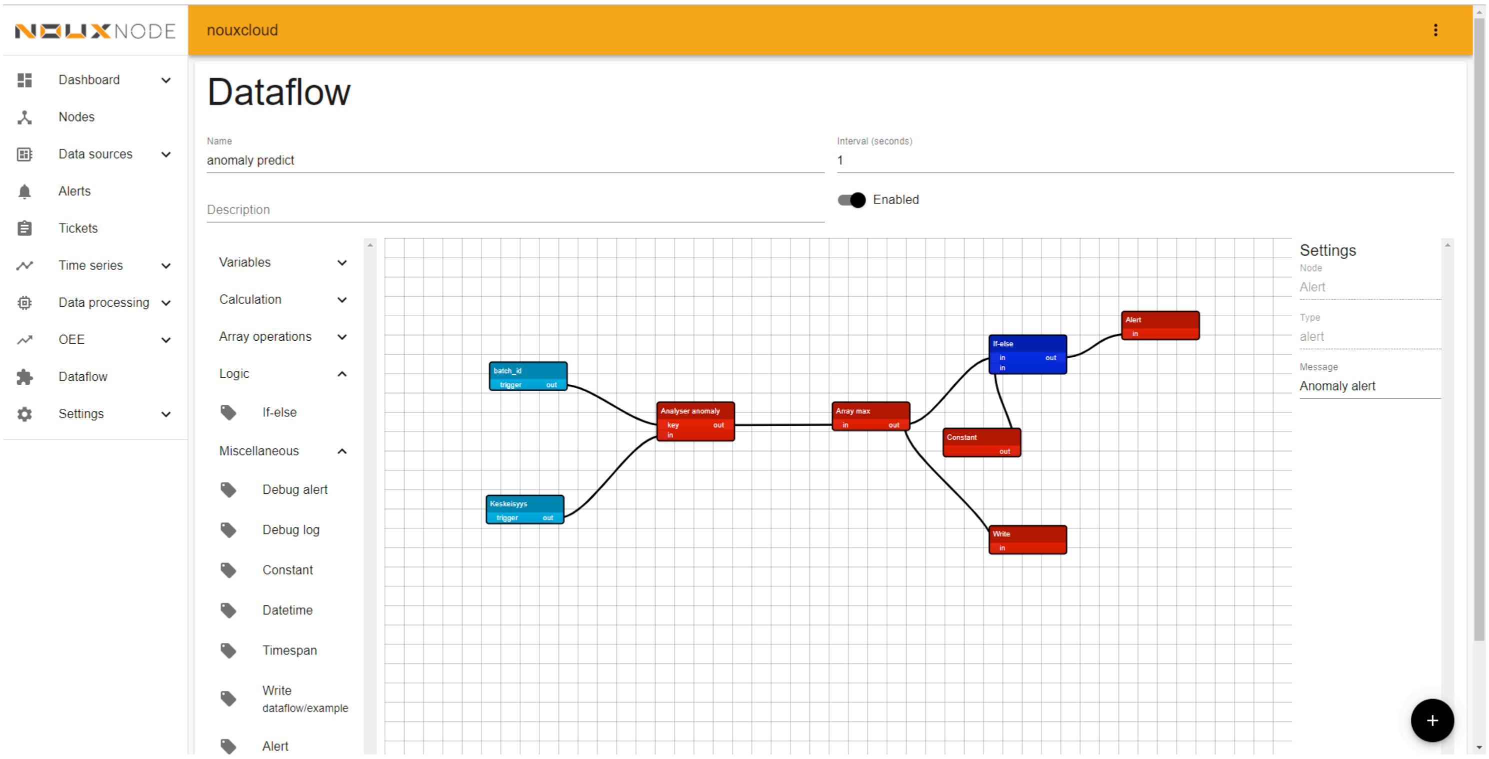
Settings

Node
Alert

Type
alert

Message
Std alert

Figure 8. Standard deviation method dataflow configuration.



Dataflow

Name
anomaly predict

Interval (seconds)
1

Description

Enabled

Settings

Node
Alert

Type
alert

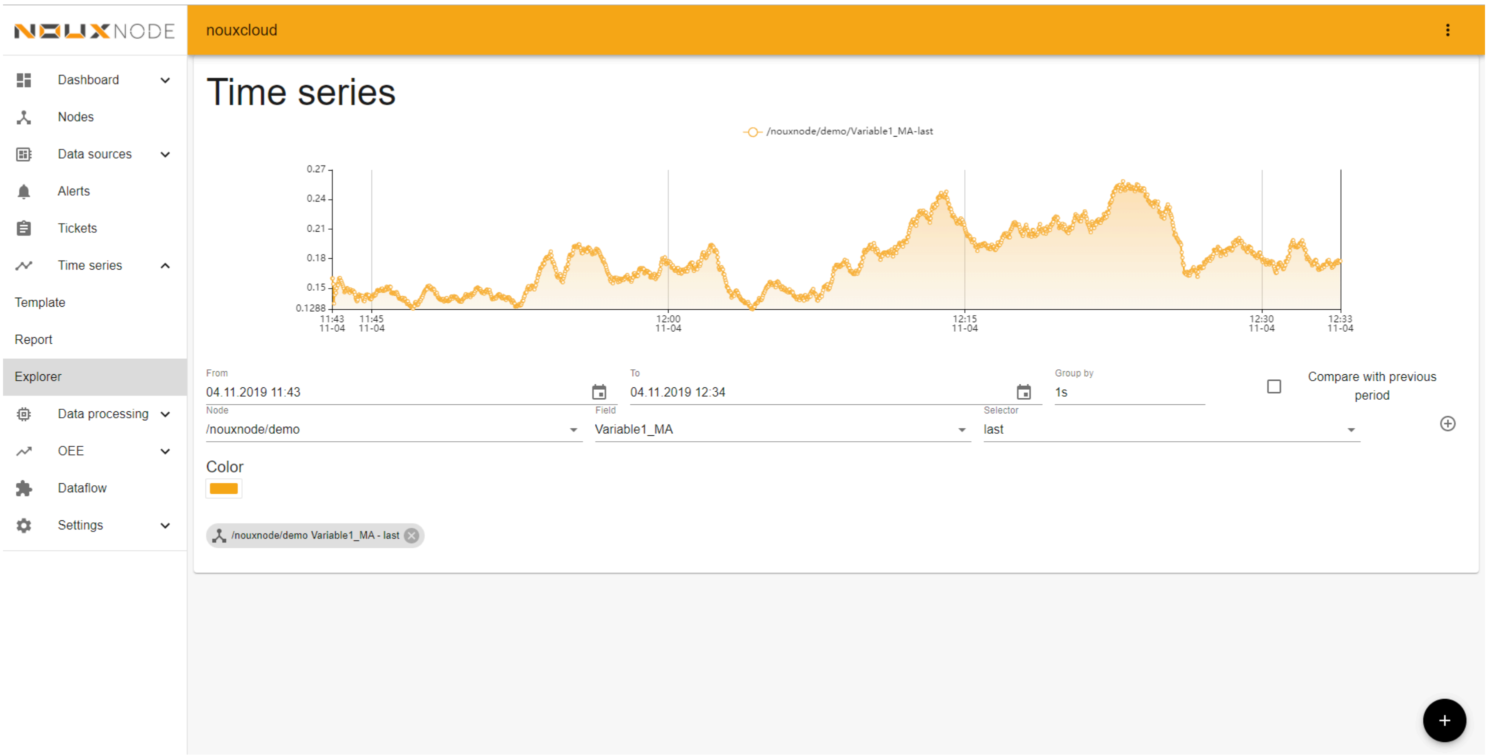
Message
Anomaly alert

Figure 9. Isolation forest model dataflow configuration.

Based on the described dataflow configurations, the following rules are used for triggering an alert in approach 1):

- First 1000 samples define the average level of the batch
- If the batch average is exceeded by over 10% an alert is triggered

For approach 2), new variables are created in dataflow, but alerts are not set. These new variables are moving average, standard deviation and anomaly predictions from an isolation forest algorithm trained with normal condition batches. Their graphs are presented in Figure 10, Figure 11 and Figure 12.



From: 04.11.2019 11:43

To: 04.11.2019 12:34

Group by: 1s

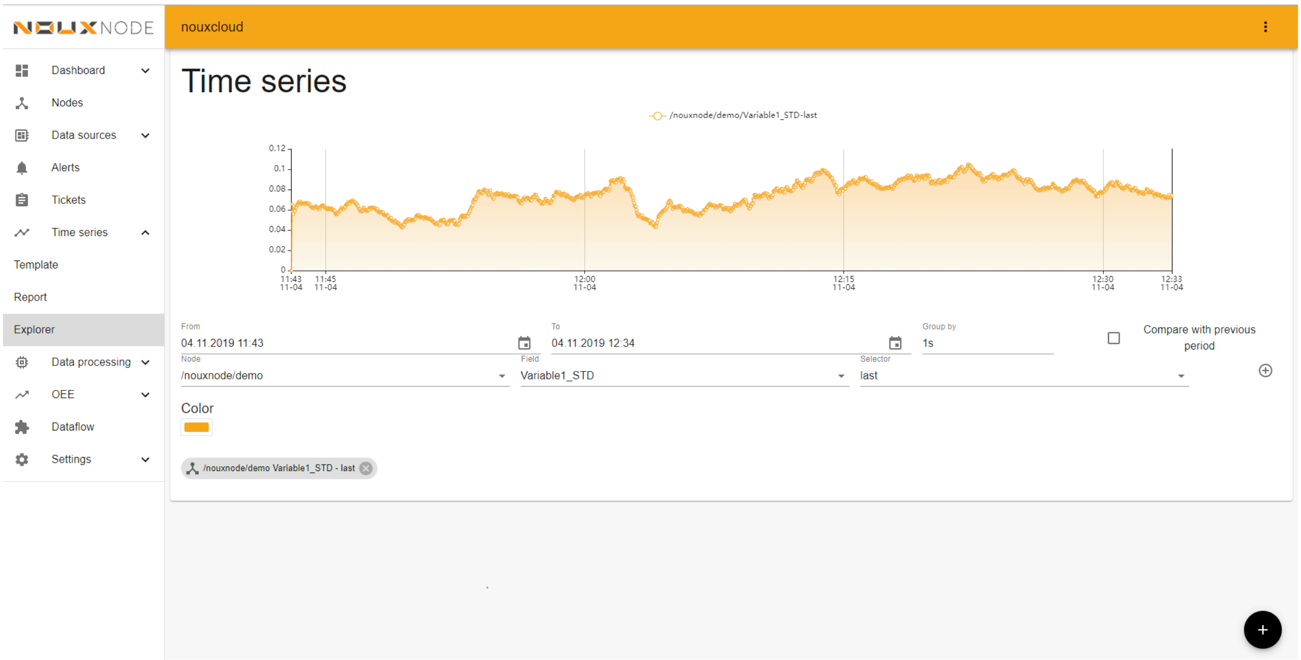
□ Compare with previous period

Node: /nouxnode/demo

Field: Variable1_MA

Selector: last

Figure 10. A time series representation of the moving average method applied to batch with anomalies.



Time series

From 04.11.2019 11:43

To 04.11.2019 12:34

Group by 1s

□ Compare with previous period

Node /nouxnode/demo

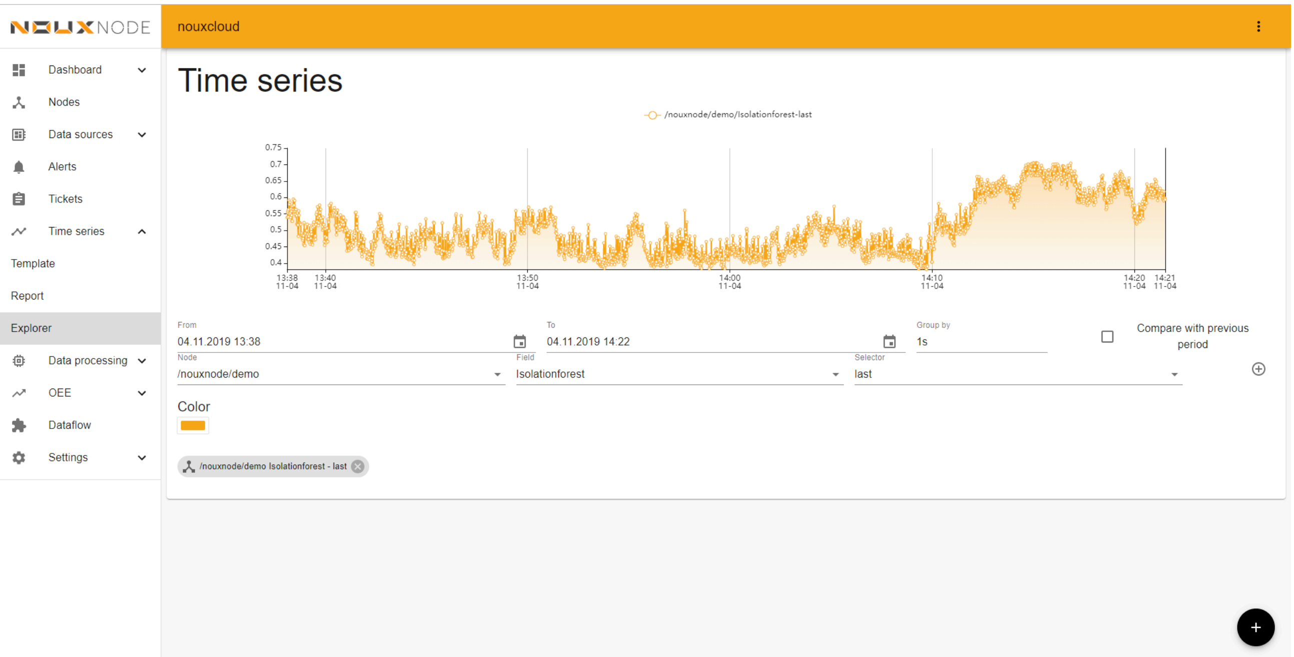
Field Variable1_STD

Selector last

Color

/nouxnode/demo Variable1_STD - last

Figure 11. A time series representation of the standard deviation method applied to batch with anomalies.



Time series

From 04.11.2019 13:38

To 04.11.2019 14:22

Group by 1s

□ Compare with previous period

Node /nouxnode/demo

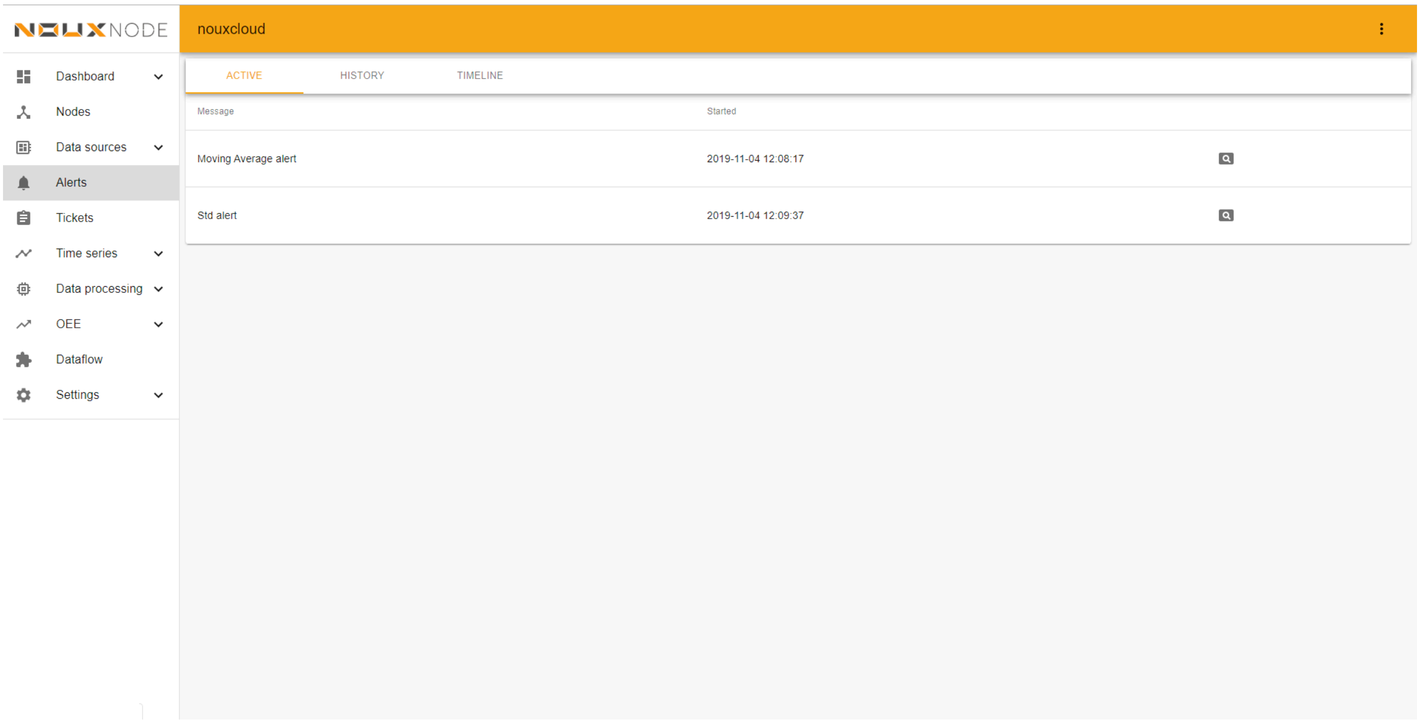
Field Isolationforest

Selector last

Color

/nouxnode/demo Isolationforest - last

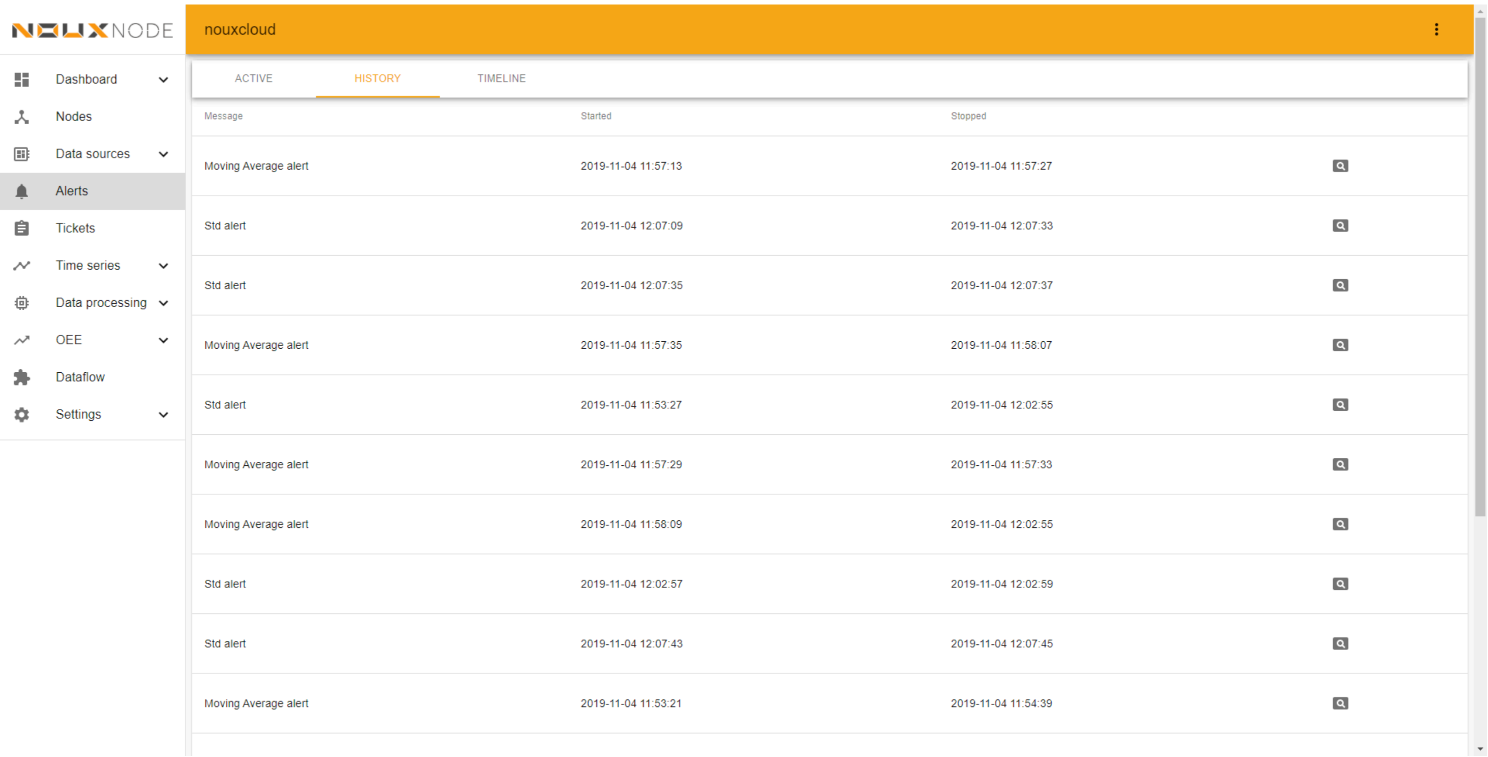
Figure 12. A time series representation of the isolation forest predictions on batch with anomalies. The predictions show high probability for anomality near the end of the batch.



nouxcloud

Message	Started
Moving Average alert	2019-11-04 12:08:17
Std alert	2019-11-04 12:09:37

Figure 13. Alerts created by moving average and standard deviation dataflows.

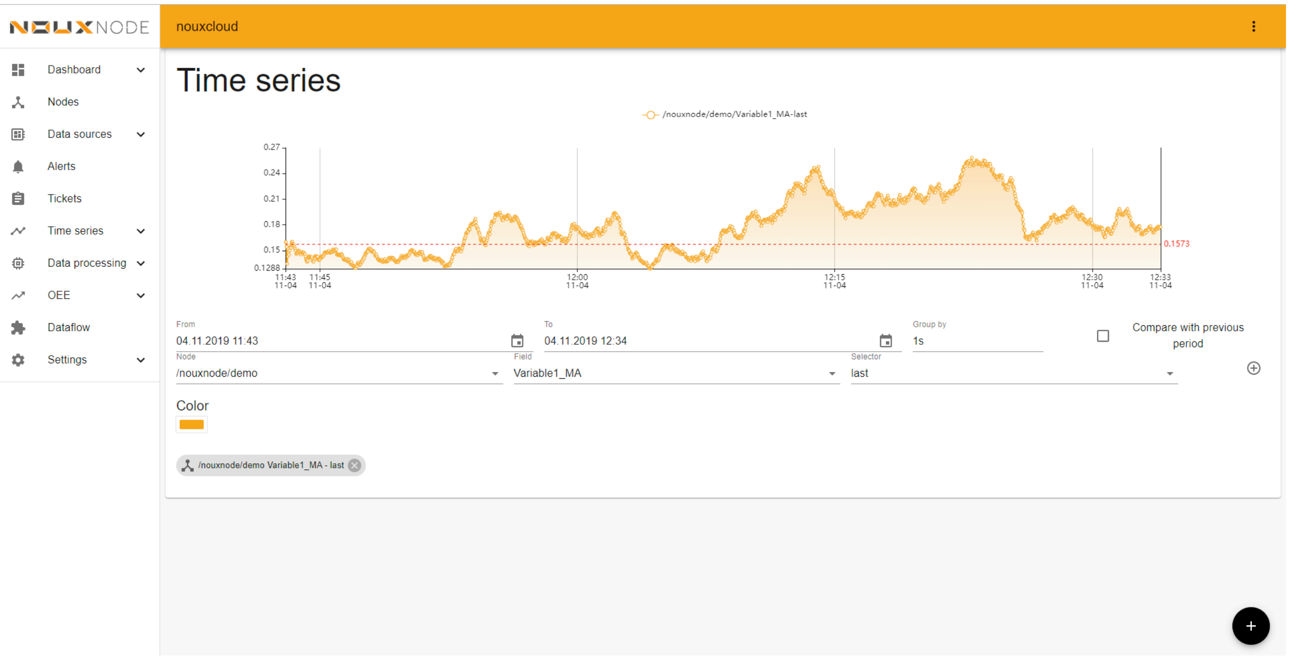


nouxcloud

Message	Started	Stopped
Moving Average alert	2019-11-04 11:57:13	2019-11-04 11:57:27
Std alert	2019-11-04 12:07:09	2019-11-04 12:07:33
Std alert	2019-11-04 12:07:35	2019-11-04 12:07:37
Moving Average alert	2019-11-04 11:57:35	2019-11-04 11:58:07
Std alert	2019-11-04 11:53:27	2019-11-04 12:02:55
Moving Average alert	2019-11-04 11:57:29	2019-11-04 11:57:33
Moving Average alert	2019-11-04 11:58:09	2019-11-04 12:02:55
Std alert	2019-11-04 12:02:57	2019-11-04 12:02:59
Std alert	2019-11-04 12:07:43	2019-11-04 12:07:45
Moving Average alert	2019-11-04 11:53:21	2019-11-04 11:54:39

Figure 14. Alerts created by moving average and standard deviation dataflows. Please note that some alerts have both start and end time and some only start time. Alerts without end time means that value exceeded the alert limit and stayed above it and never returned to the average level of the beginning of the batch.

Alerts created for approach 1) in cases A) and B) are presented in Figure 13 and Figure 14. Alert limits that create the actual alerts are presented in Figure 15 and Figure 16. An alert limit is also set for approach 2), as presented in Figure 17. The actual alerts are not set due to high number of false positives that would occur with a limit of 0.



Time series

From 04.11.2019 11:43

To 04.11.2019 12:34

Group by 1s

□ Compare with previous period

Node /nouxnode/demo

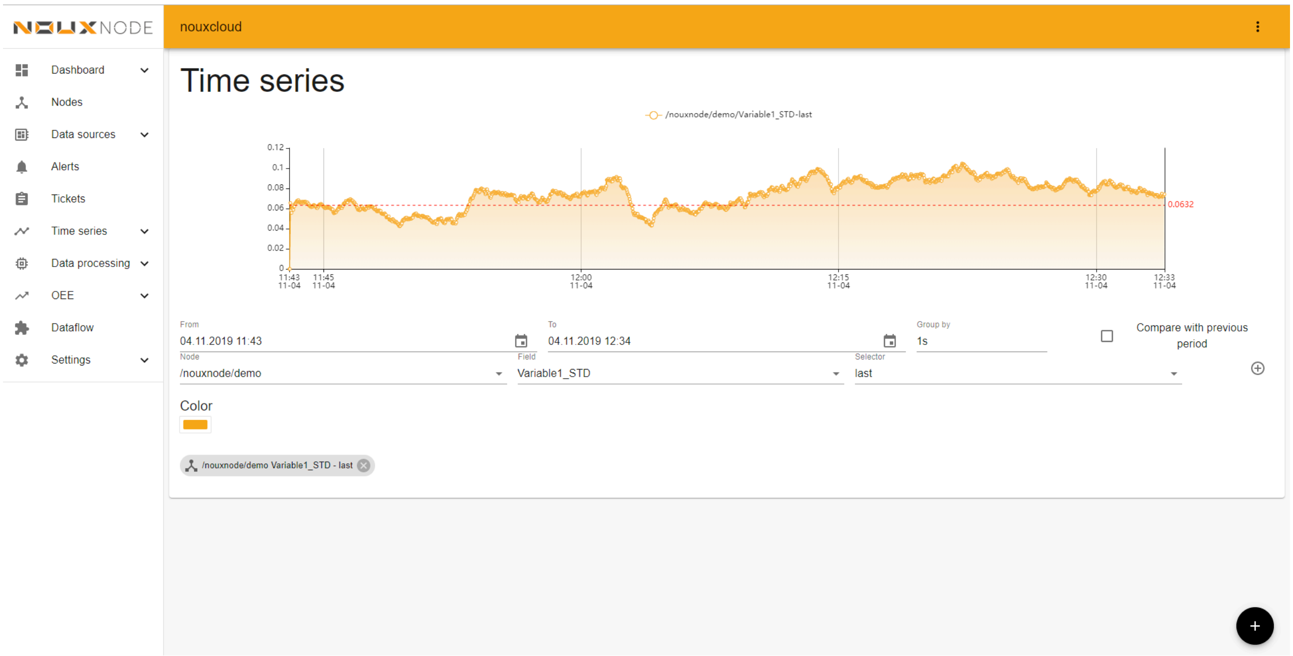
Field Variable1_MA

Selector last

Color

/nouxnode/demo Variable1_MA - last

Figure 15. The moving average case alert limit (red dotted line) presented in time series format.



Time series

From 04.11.2019 11:43

To 04.11.2019 12:34

Group by 1s

□ Compare with previous period

Node /nouxnode/demo

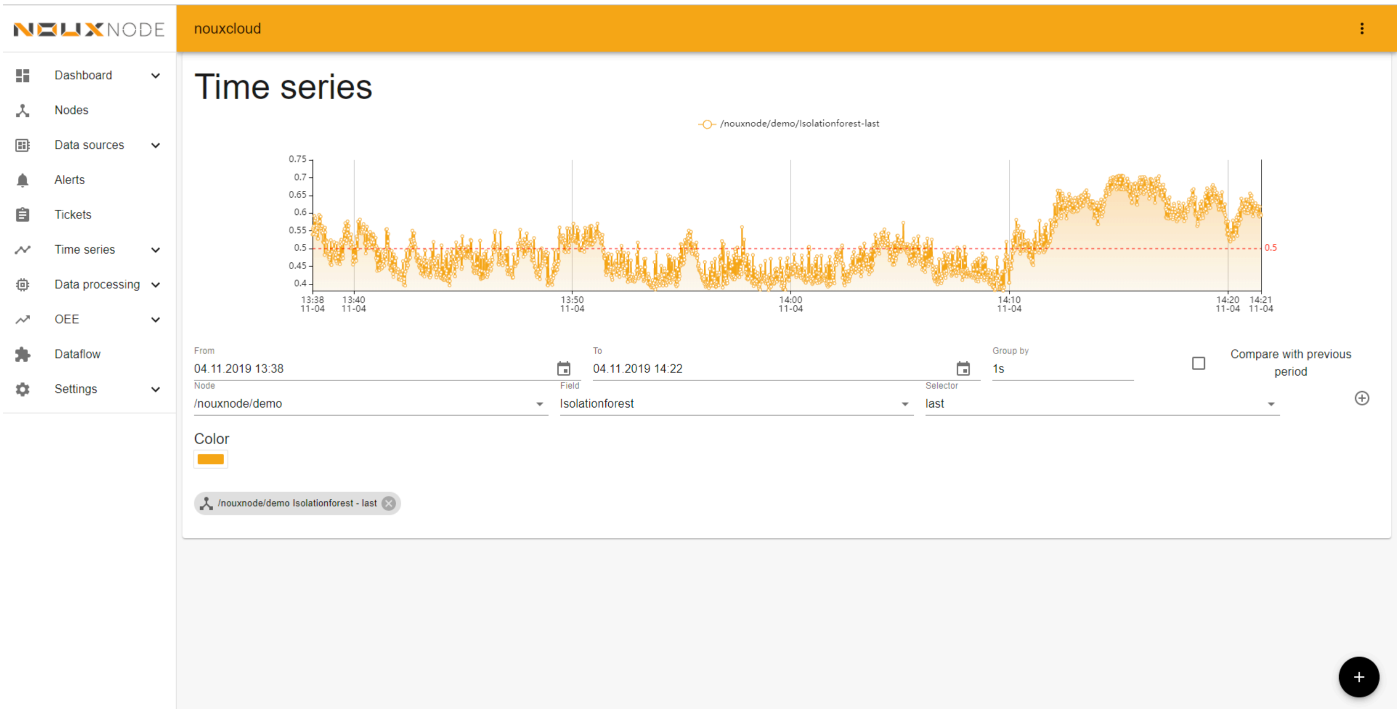
Field Variable1_STD

Selector last

Color

/nouxnode/demo Variable1_STD - last

Figure 16. The standard deviation case alert limit (red dotted line) presented in time series format.



Time series

From 04.11.2019 13:38

To 04.11.2019 14:22

Group by 1s

□ Compare with previous period

Node /nouxnode/demo

Field Isolationforest

Selector last

Figure 17. The isolation forest alert limit (red dotted line) presented in time series format. Individual prediction values exceed the limit from the beginning already, but a rolling average over the predictions would indicate anomalies near the end of the batch, which is where the anomalies are.

Discussion

The results show that it is possible to achieve satisfactory results for all cases with simple methods like calculating the percentage change in the rolling mean or standard deviation of a batch. Moreover, the results indicate that all types of anomalies can be captured by machine learning models as well, but model robustness remains questionable with the limited number of batches available. In general, the more example batches from different kinds of normal conditions, the more accurate the model would be in detecting anomalies for future samples.

To further analyze the potential of automating the detection of deviations in average and standard deviation, more data would be required. This would be needed not only for improving the accuracy of the model, but also for ensuring robustness in various different circumstances through reliable model validation setup. Moreover, having real timestamps for the measurements would improve the signal to noise ratio and help in preprocessing the data properly for the model. To fully unleash the potential of machine learning, other production parameters from the same process could be utilized as well, e.g. size or material of the batch.

The solution presented in this report could be applied to any similar production data. The dataflow models used in these solutions are available in NouxCoud, and other machine learning models could be easily implemented using APIs. Data can be manually imported or read directly from machinery by connecting NouxCloud to it. The former is suitable for piloting but the latter is preferred to allow for utilizing the full potential of machine learning. However, simple and straightforward methods can already provide insightful monitoring and help in improving manufacturing quality.

How much more data would be necessary to be more confident about the predictions of a machine learning model? The challenges arise from differences between the production batches - even though there are thousands of measurements per batch, there are only 10 data samples in terms of batches, being a very small dataset for machine learning. A careful estimate for the number of batches would tens of more batches with and without anomalies.

In the end, the best modeling approach for the also depends heavily on the availability of data. If a production parameter is such that only one sample per minute is available, it would take weeks or months to get a trustworthy amount of data for complex machine learning models. In these cases, simpler modeling approaches might work better. Then again if sample rate is e.g. one per second the required amount can be gathered relatively fast but in a batch-based production differences between batches could still be problematic.

In conclusion, the most useful method to be used for monitoring the production quality depends strongly on the data quality, amount and availability. If the data is not directly from machinery (data quality cannot be confirmed and samples are difficult to obtain), then the most reliable and meaningful methods might be the more simple ones, such as moving average or standard deviation. On the other hand, having more data in combination with machine learning could prove to be even more reliable and accurate. Quality improvements could be achieved with both approaches. Since the machine learning would benefit from all available data, it is advisable to start gathering data from production machinery even when the end-use of the data is uncertain. This allows for implementing data-intensive machine learning algorithms and other state-of-the-art technologies in the future.