# Machine-Learning Techniques to Protect Critical Infrastructure From Cybersecurity Incidents or Equipment Incidents

**Dr. R. Scott Starsman**
sstarsman@avineon.com
757-232-7043

Defense Systems Information Analysis Center

AVINEON®

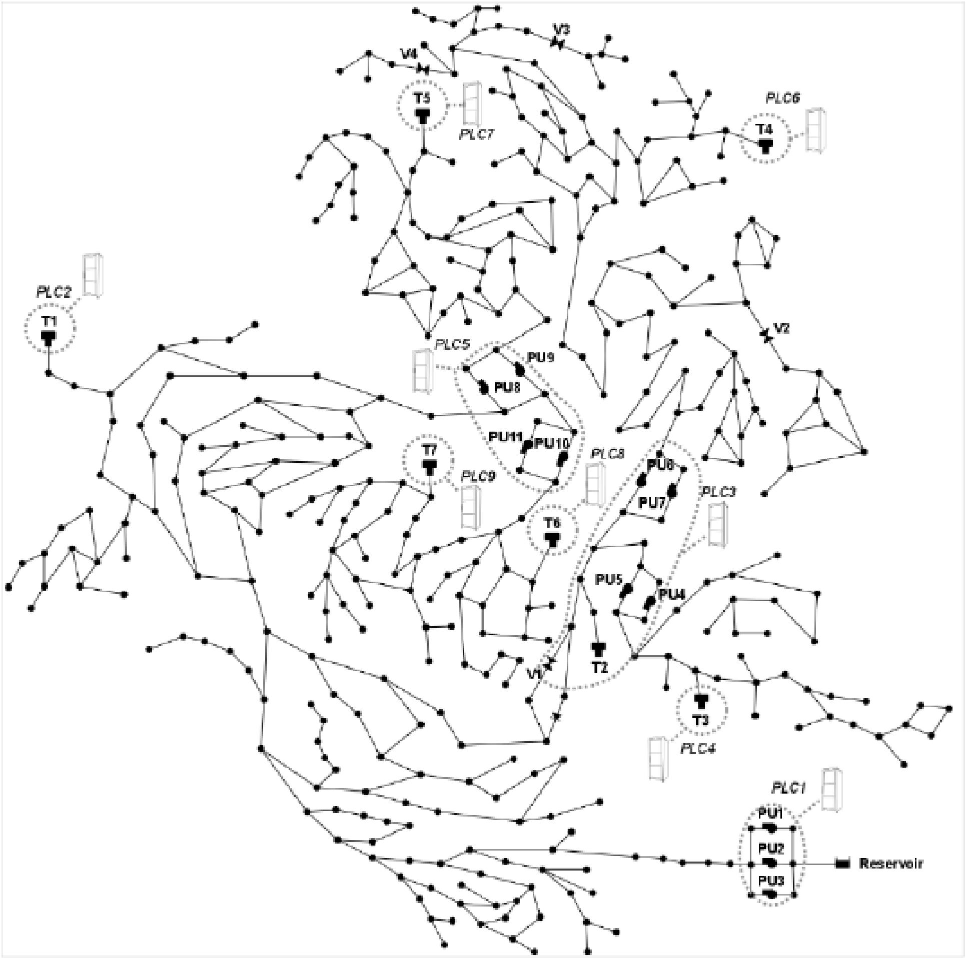
DSIAC is a DoD Information Analysis Center (IAC) sponsored by the Defense Technical Information Center (DTIC), with policy oversight provided by the Office of the Under Secretary of Defense (OUSD) for Research and Engineering (R&E). DSIAC is operated by the SURVICE Engineering Company.

# Agenda

- Challenge Introduction and Goal
- Data Source
- Exploratory Data Analysis and Feature Engineering
- Solution Design
- Unsupervised Learning Approaches
- Implementation
- Anomaly Detection Results
- Conclusion

# Problem Introduction

- Industrial control system cybersecurity remains a critical challenge
- Goal: detect cyber attacks on the industrial control system supporting water distribution
- Illustrate the machine learning (ML) design processes involved in solving this challenge



# Data Source

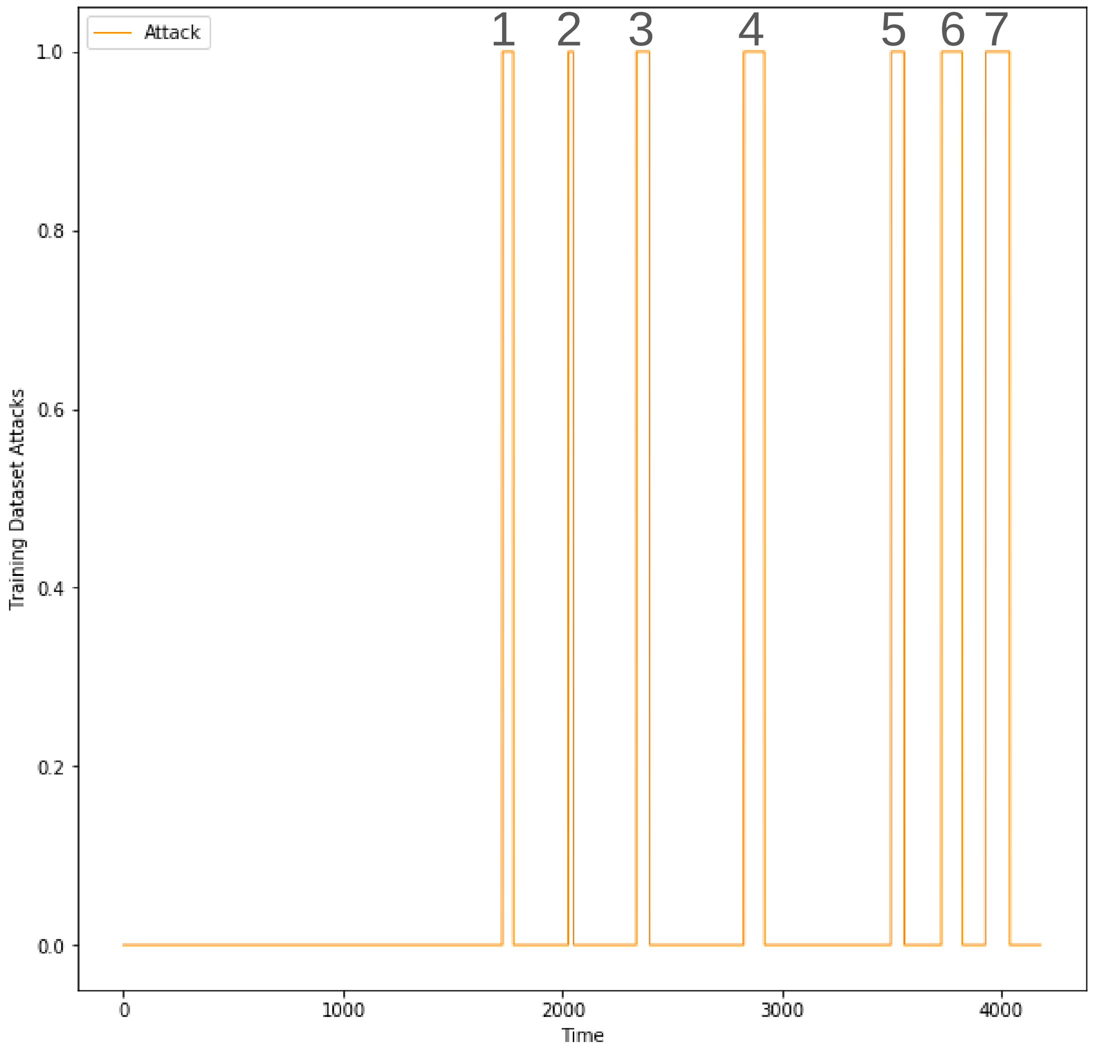
- **Data from the Battle of the Attack Detection Algorithms (BATADAL) - https://batadal.net/**
- **Scenario based upon a cyber attack on a water distribution system**
- **Normal system performance data provided**
- **3 datasets provided**
  - Normal operation (8,761 rows)
  - Under attack, available for training (4,177 rows)
  - Under attack, not available for training (2,089 rows)

# Data Set Sample

| DATETIME | L_T1 | L_T2 | L_T3 | L_T4 | L_T5 | L_T6 | L_T7 | F_PU1 | S_PU1 | F_PU2 | S_PU2 | F_PU3 | S_PU3 | F_PU4 | S_PU4 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 06/01/14 00 | 0.50973 | 2.049003 | 3.191145 | 2.792634 | 2.656091 | 5.316831 | 1.562321 | 98.99844 | 1 | 99.01815 | 1 | 0 | 0 | 35.53669 | 1 |
| 06/01/14 01 | 0.41258 | 2.009072 | 3.642565 | 2.831673 | 3.126387 | 5.494855 | 1.852043 | 99.0959 | 1 | 99.11564 | 1 | 0 | 0 | 34.45491 | 1 |
| 06/01/14 02 | 0.320112 | 1.986093 | 4.140192 | 3.256733 | 3.574601 | 5.5 | 2.246126 | 98.42096 | 1 | 98.4405 | 1 | 0 | 0 | 33.48709 | 1 |
| 06/01/14 03 | 0.332879 | 2.009203 | 4.673478 | 3.744497 | 3.952379 | 5.5 | 3.203573 | 97.57517 | 1 | 97.59446 | 1 | 0 | 0 | 32.58554 | 1 |
| 06/01/14 04 | 0.483496 | 2.089049 | 5.237937 | 4.409456 | 3.504676 | 5.5 | 4.439714 | 97.35106 | 1 | 97.37028 | 1 | 0 | 0 | 31.46968 | 1 |
| 06/01/14 05 | 0.791114 | 2.773177 | 5.155802 | 3.937262 | 3.191528 | 5.322743 | 3.988906 | 94.13547 | 1 | 94.15375 | 1 | 0 | 0 | 0 | 0 |
| 06/01/14 06 | 1.186589 | 3.536068 | 4.983953 | 3.018011 | 2.859591 | 5.066728 | 2.977463 | 95.258 | 1 | 95.27661 | 1 | 0 | 0 | 0 | 0 |
| 06/01/14 07 | 1.420449 | 3.872926 | 4.747458 | 3.581882 | 2.359944 | 5.152646 | 2.953742 | 96.94746 | 1 | 96.96656 | 1 | 0 | 0 | 0 | 0 |
| 06/01/14 08 | 1.534827 | 4.138434 | 4.417932 | 3.959265 | 1.748313 | 5.395835 | 3.228596 | 96.97029 | 1 | 96.9894 | 1 | 0 | 0 | 0 | 0 |
| 06/01/14 09 | 1.576541 | 4.50004 | 4.130157 | 4.232002 | 1.666737 | 5.5 | 3.628678 | 97.15647 | 1 | 97.17564 | 1 | 0 | 0 | 0 | 0 |
| 06/01/14 10 | 1.55855 | 4.96201 | 3.665213 | 2.962582 | 2.107416 | 5.5 | 3.445807 | 97.81398 | 1 | 97.83334 | 1 | 0 | 0 | 0 | 0 |

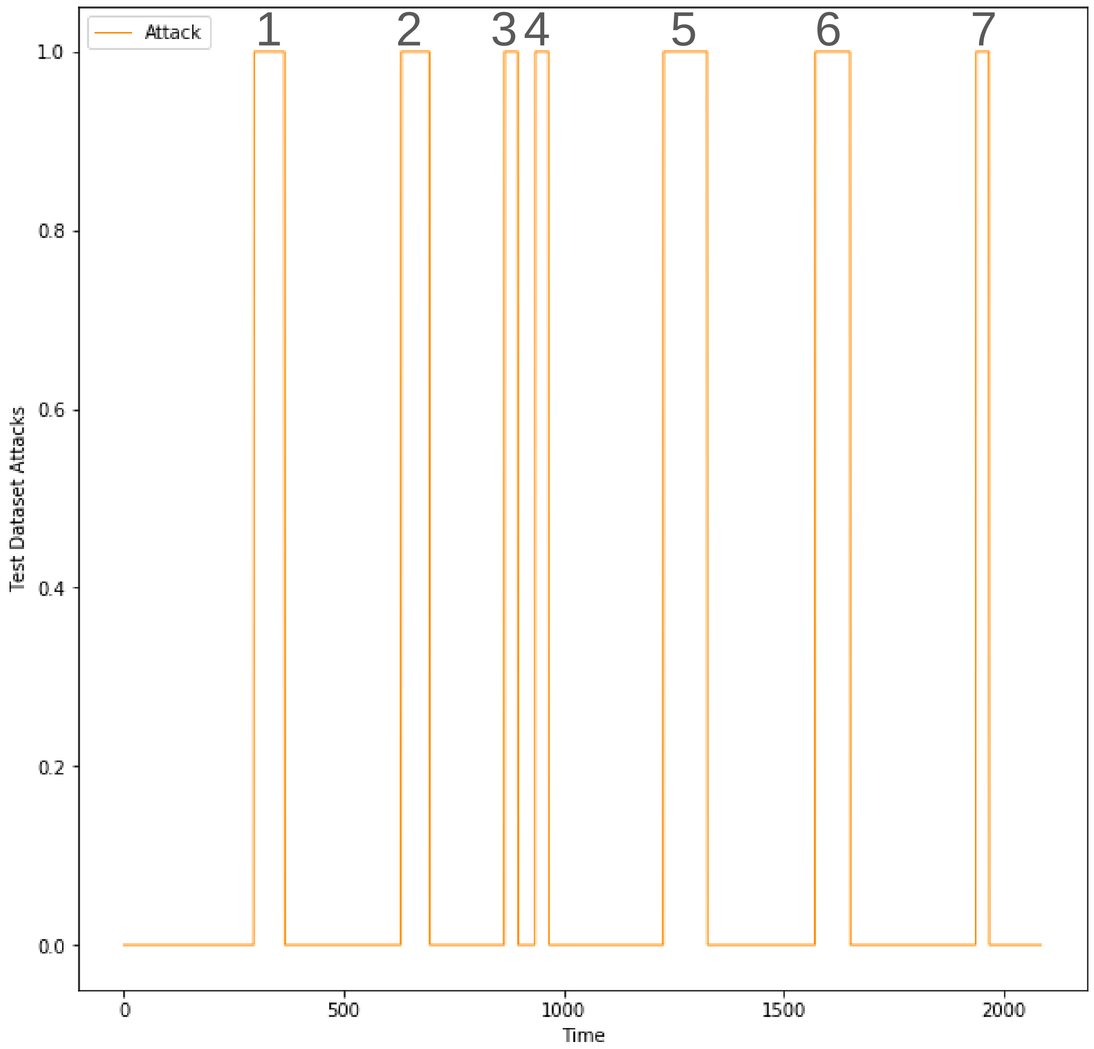
- **Each data set includes 43 input features representing:**
  - Tank levels
  - Pump switches
  - Pump flow rates
  - Valve positions
  - Valve flow rates
  - Pressures at various sensors

# Attacks Conducted –Training Dataset




1. Replay attack on tank 7 level
2. Replay attack on tank 7 level and pumps 10 and 11 flow and status
3. Alter tank 1 level readings causing pumps 1 and 2 to remain on and tank 1 overflow
4. Same as attack 3
5. Speed of pump 7 reduced causing low water levels in tank 4
6. Similar to attack 5 but increased speed reduction
7. Same as attack 6

# Attacks Conducted – Test Dataset



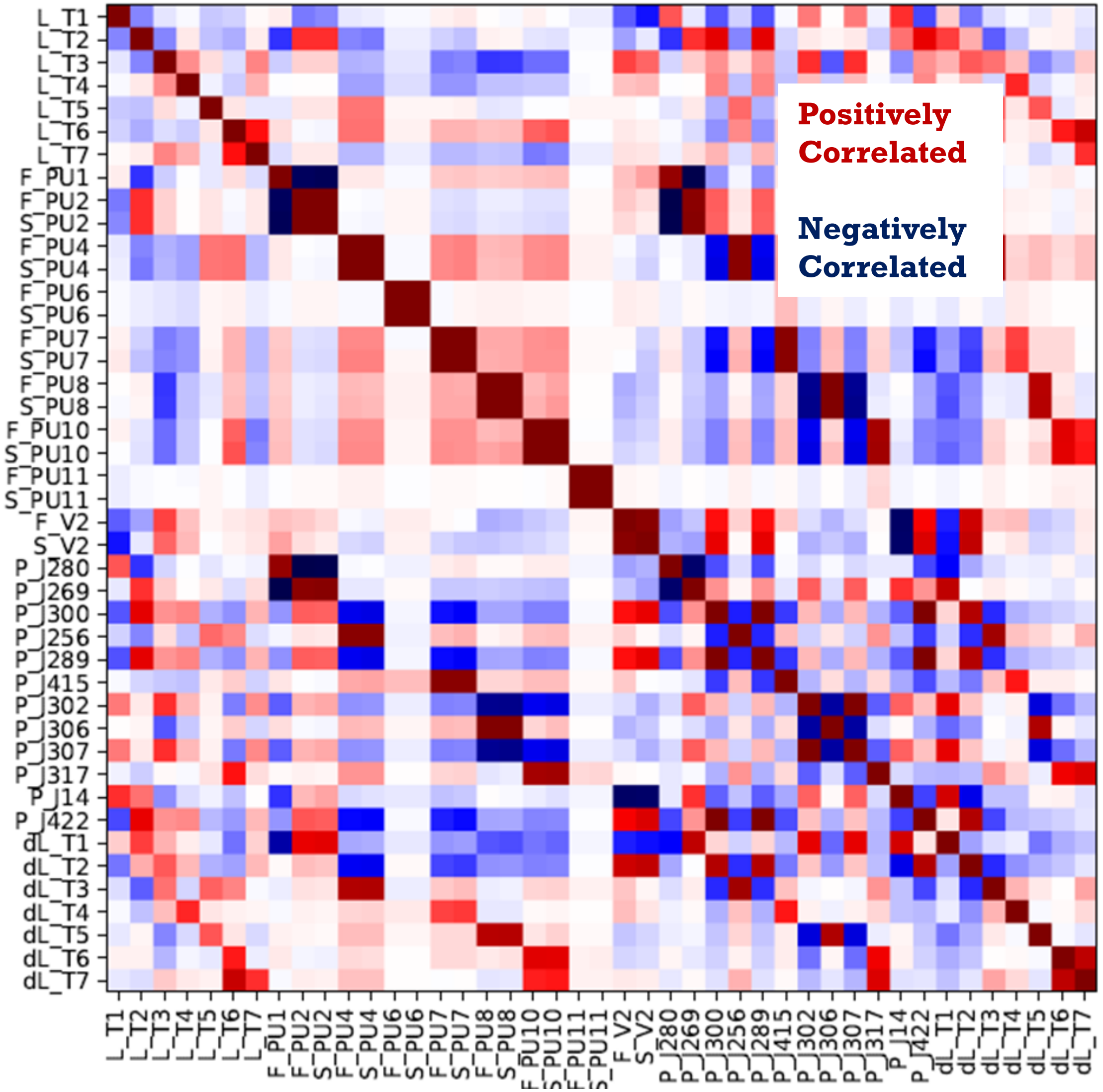

1. **Replay attack on tank 3 level and pump 4 and 5 flow and status**
2. **Attack alters tank 2 levels causing tank 2 to overflow**
3. **Activates pump 3**
4. **Similar to attack 3**
5. **Similar to attack 2**
6. **Replay attack on tank 7 level and pumps 10 and 11 flow and status**
7. **Manipulation of tank level signal leading to overflow of tank 6**

# Exploratory Data Analysis

AVINEON

- Notable correlations
  - Correlation between pump switch and flow
  - Negative correlation between pump 1 and 2 flows
  - Correlation between tanks 6 and 7
  - Correlation between tank 3 and pump 4

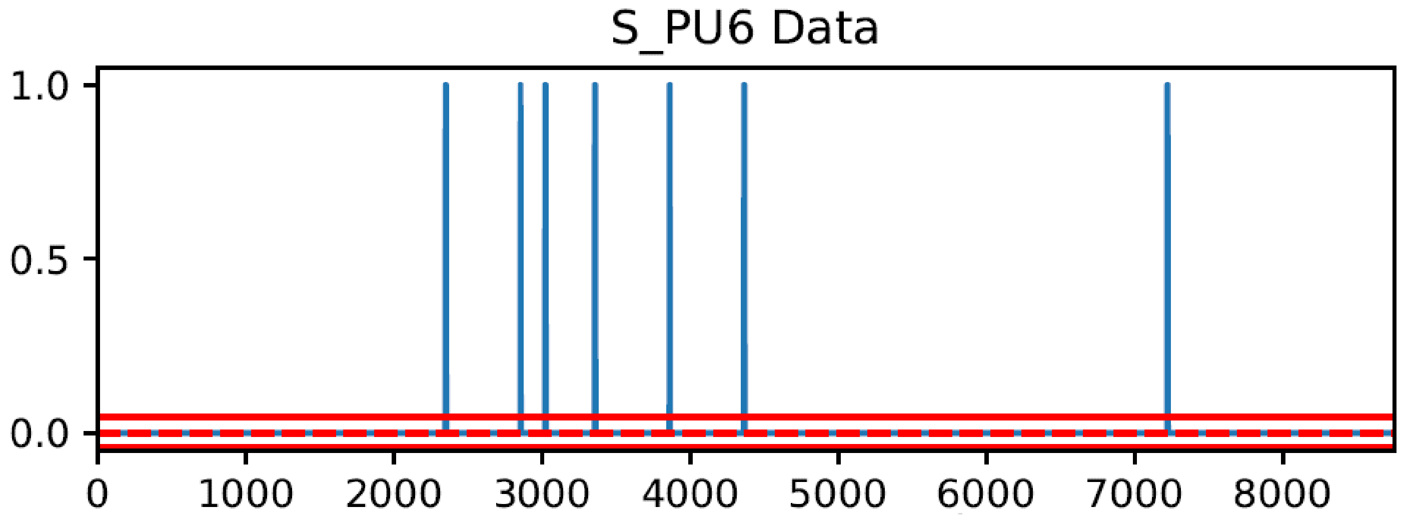
- In some situations, it is useful to remove highly-correlated data
  - Breaking of a correlation might indicate an attack so they are left in
- Pump 1 is in constant use
- Pumps 3, 5, and 9 are never used
- Pumps 6 and 11 are rarely used




DSIAC

# Feature Engineering – Sparse Features

- **Some of the data elements have significant outliers**
  - Due to only occasional equipment use
- **These can cause very high values in the normalized data and negatively impact training**
- **Solutions include limiting the magnitude of the normalized values or not normalizing these type of features**

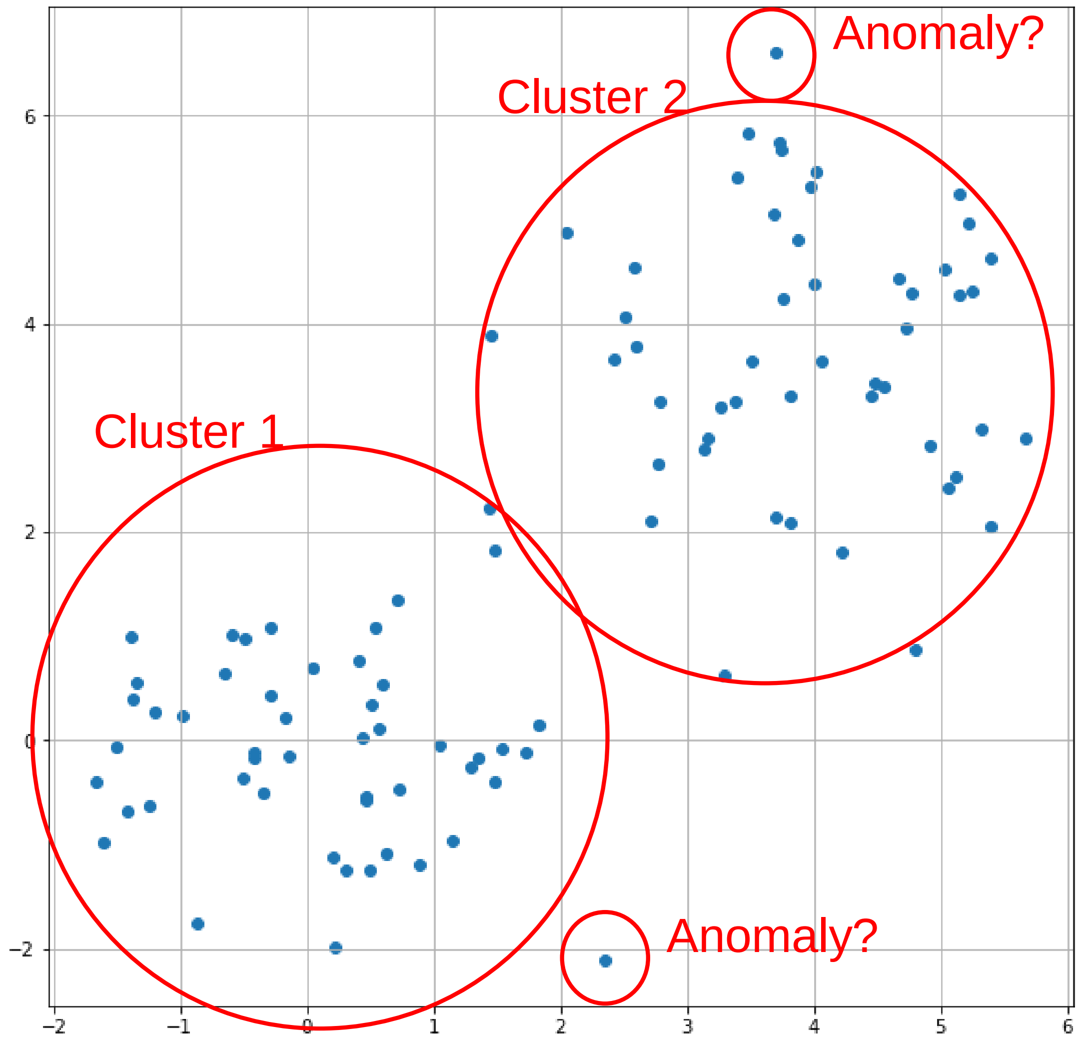


# Solution Design

- **Solution architecture**
  - Data set with no attacks provided
  - Limited data with attacks provided
  - Because very little attack information was provided:
    - Use an unsupervised training method to detect data anomalies that indicate a cyber attack
- **Two unsupervised approaches investigated**
  - Clustering
  - Neural network  autoencoder

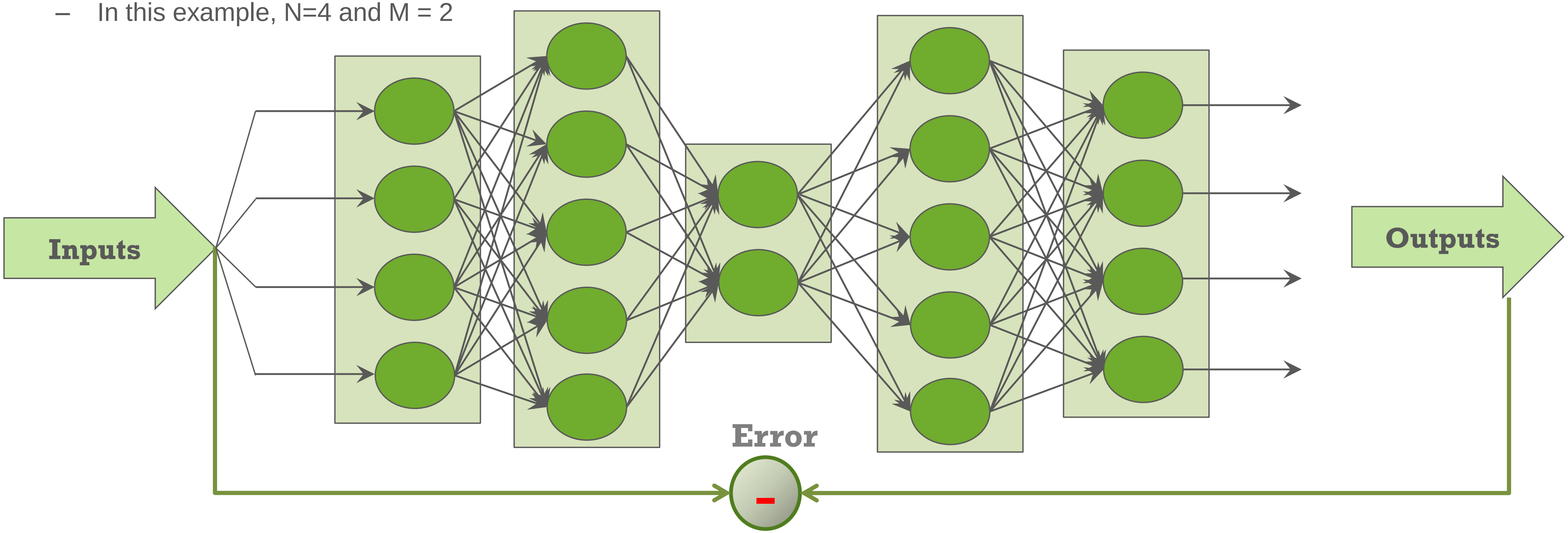
# Unsupervised Learning Approaches

- Useful for unlabeled data sets
- Common approaches include:
  - Clustering
  - Anomaly detection
  - Neural network autoencoders
- Capable of detecting anomalies



# Neural Network Autoencoder

- The desired output is the same as the actual input
- The network is trained to produce this output
- The compression layer in the center reduces the dimensionality from N input nodes to M center nodes
  - In this example, N=4 and M = 2



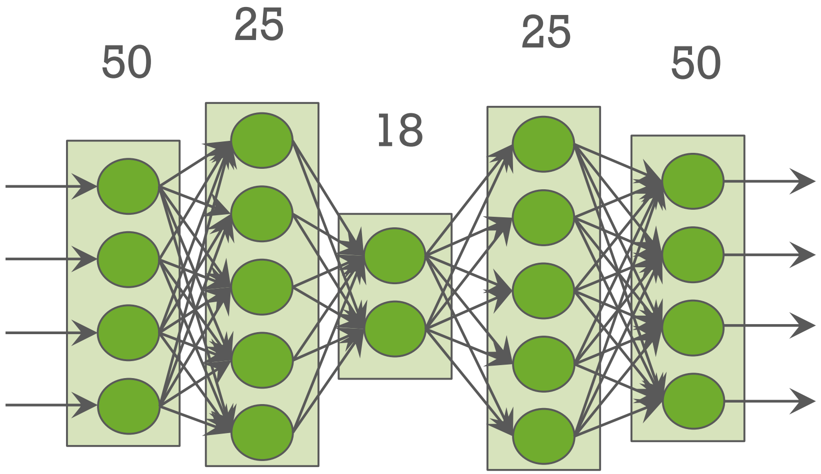

Anomalous data will produce high errors so the autoencoder can be used as an outlier detector

# Solution Design (cont'd)

- **Conducted some experiments with clustering**
  - Was not on track to provide a good solution
- **Experimented with an autoencoder and this promised significant improvement**
- **Continued refinement of architecture throughout development**
  - Number of layers
  - Minimum number of nodes in a layer

# Implementation

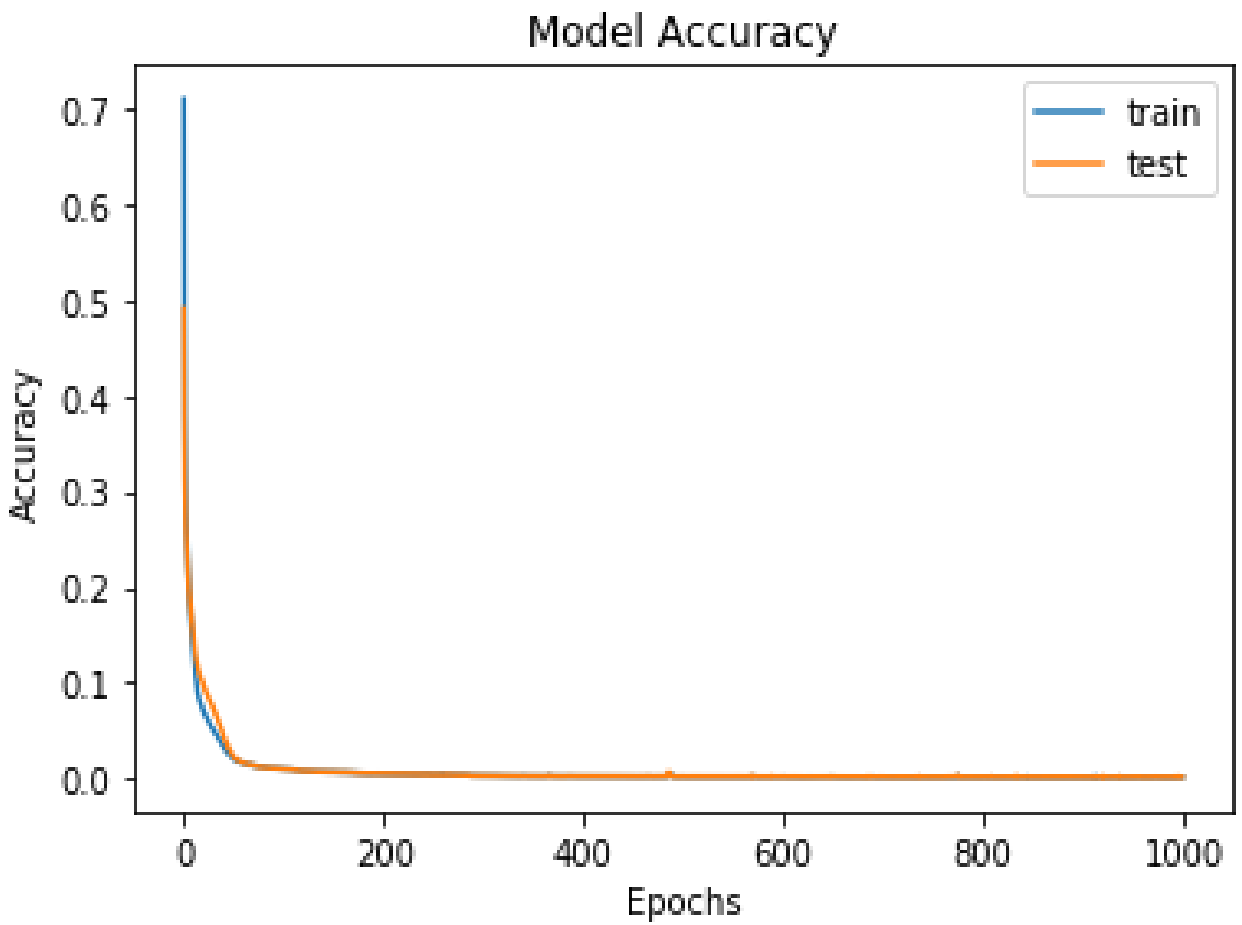
- **Used a neural network autoencoder with:**
  - 50 input nodes
  - 1 for each feature
  - An intermediate layer with 25 nodes
  - An encoding layer with 18 nodes
  - This is the compression factor
  - An intermediate layer with 25 nodes
  - A result layer with 50 nodes
  - The activation function only outputs positive values
  - An output layer with 50 nodes
  - This allows the output to handle both positive and negative values




- **Split the data into a training set with 5,869 samples and a testing set with 2,891 samples**
- **Trained the network for 1,000 epochs (complete passes through the data)**

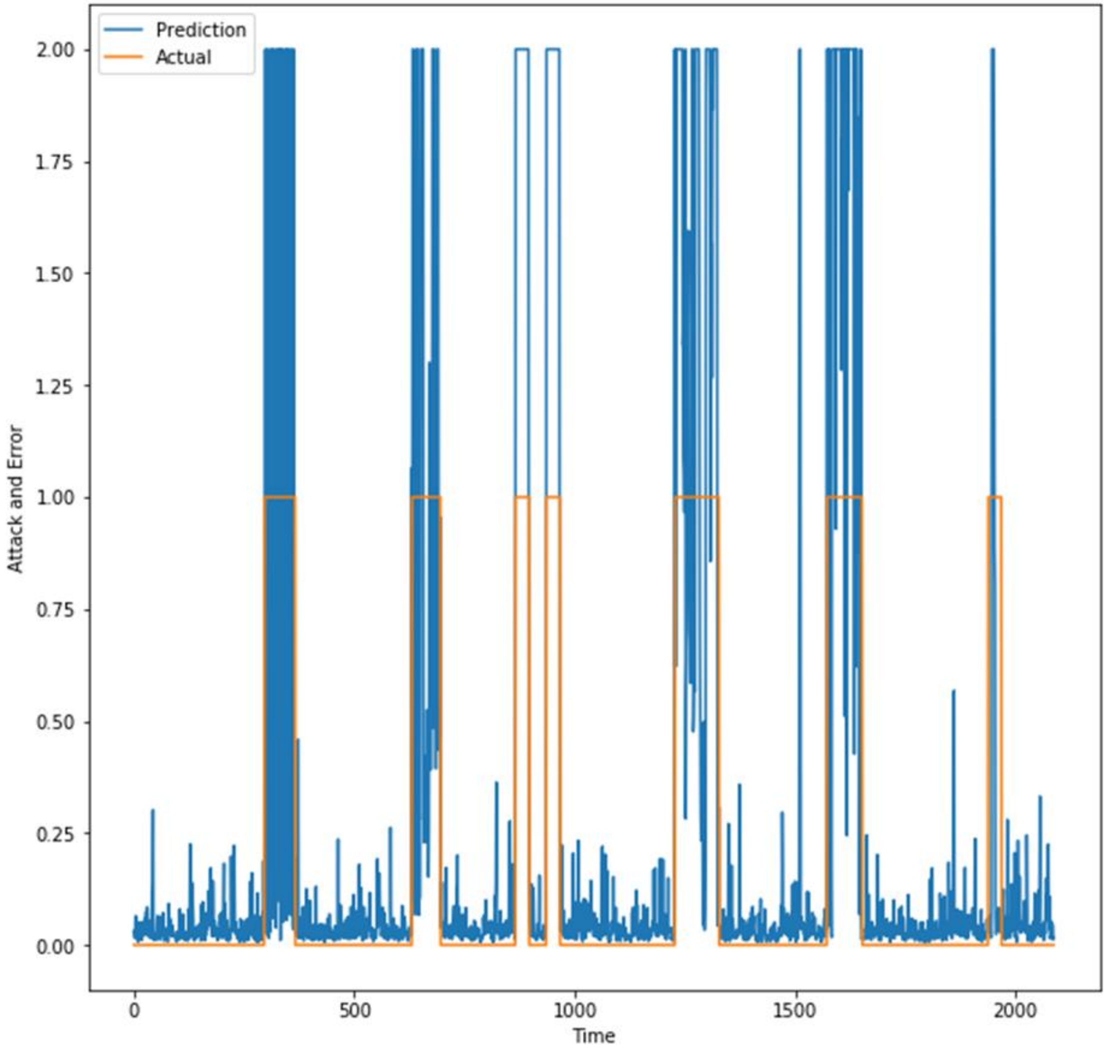
# Implementation

- **Trained model**
- **Reduced mean squared error to more than 99.8%**
- **Completed on a business-class laptop in less than 2 minutes**



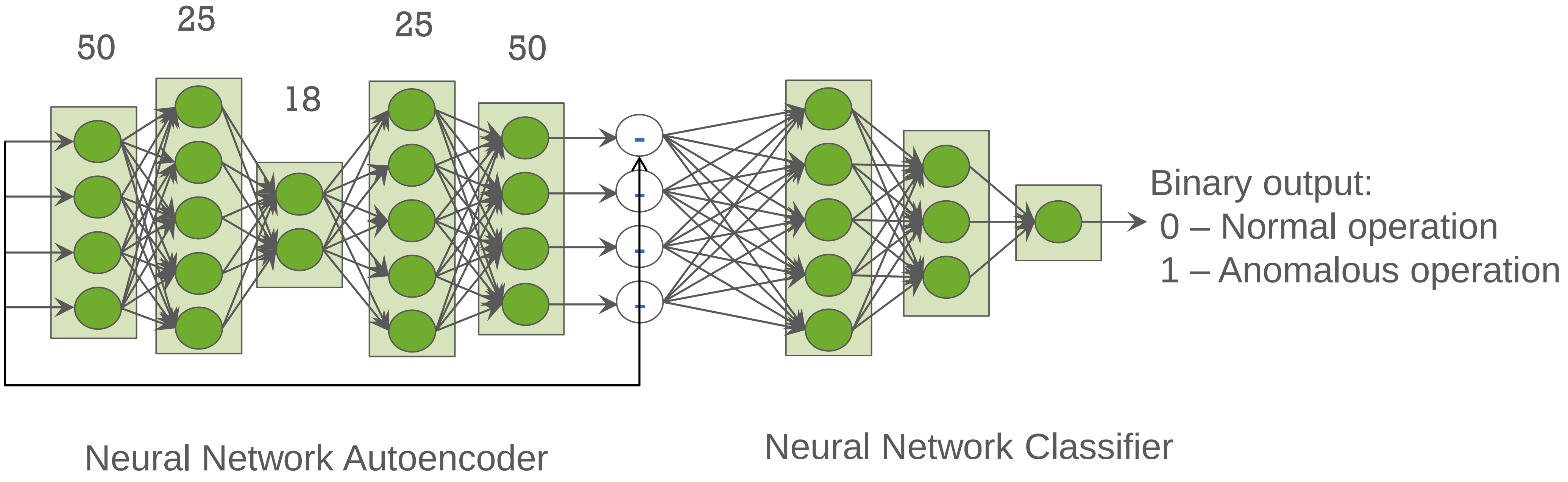
# Initial Results

- **Blue line indicates the magnitude of the error between the input and output of the autoencoder**
- **Orange line indicates the actual attacks**
- **Generally good detection of the hacks**
  - 1 false positive
- **Areas for improvement**
  - Data preprocessing
  - Addition of neural network post-processing layer



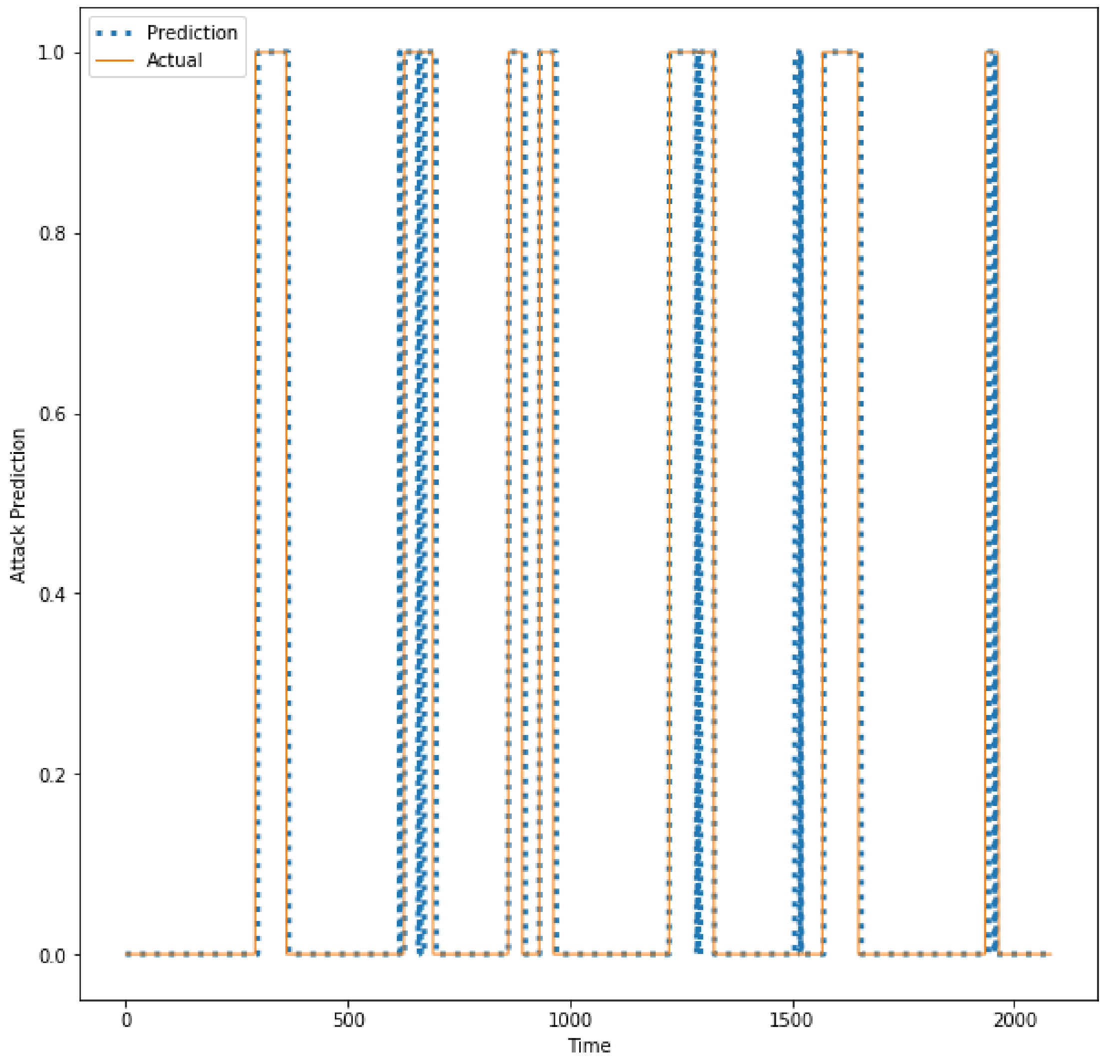
# Neural Network Post Processing

- **Implemented a two layer neural network that post-processed the autoencoder error**
  - Output - Input
- **Both a binary classified and a regression model were tested**
  - Binary classifier performed better



# Results (cont'd)

- **One false positive remains**
- **System able to reliably detect cyberattacks**
- **Areas for improvement remain:**
  - Localize the components impacted by the intrusion
  - Improve handling of infrequent events



# Conclusion

- **Autoencoder approach able to successfully detect performance anomalies in an industrial control system**

- **Results achieved despite relatively small data set**

- **Improvements to approach planned:**
  - Better rejection of false positives
  - Identification of specific equipment being targeted

# Questions

AVINEON
MACHINE LEARNING

**Dr. R. Scott Starsman**
sstarsman@avineon.com
757-232-7043