Evaluating Electricity Theft Detectors in Smart Grid Networks

Group 5:
Ji, Xinyu Rivera, Asier

Agenda

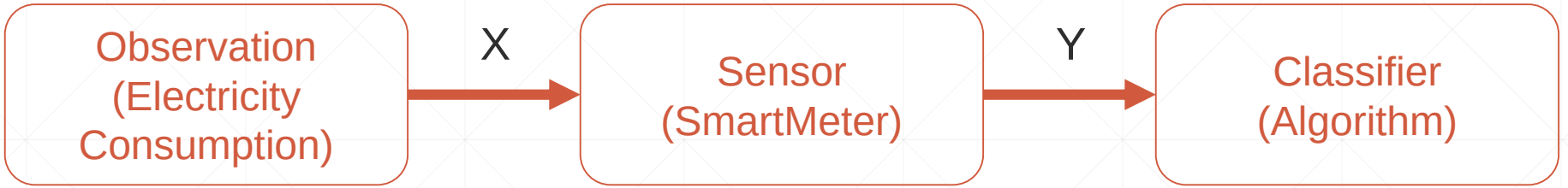
- Introduction
- Background
- Electricity Theft Detectors and Attacks
- Results
- Discussion
- Conclusion

Introduction

Bypass the Meter, so that, it doesn’t count the consumed electricity.

Detected under human inspection.

Background



Average attack: Set **Y** to low value => Really cheap electricity => Easy to detect.

Sophisticated attack: Set **Y** to reasonable value => Cheaper electricity => Difficult to detect.

Adversarial Classification: evaluate the effectiveness of the classifier against undetected attacks.

Adversarial Learning: avoid the attacker to provide false data to the learning algorithm.

Adversarial classification

- Assumptions:
  - A random process generates observations $x \in X$ with probability distribution P_0
  - Use a maximum level of false negatives, α, that can be affordable
  - Each classifier requires a threshold (T) to decide when raise the alert

- Attack vectors:
  1. For each classifier, find the maximum value for the T among the P_0 distribution that fulfils α
  2. Run all the classifiers set with the T selected in step 1, select the results that would produce the worst undetected attacks (more cost for the company)
  3. Select the less costly (for the attacker) among the classifiers selected in step 2

Adversarial learning

- Contamination attacks:
  - The classifiers uses a dataset to detect anomalies
  - The attacker can inject false values to poison the dataset
  - After some time, the data set is modified to fulfil the attackers requirements

Y =

Classifier would find purple as an anomaly

Classifier would find purple as normal

Electricity-Theft Detectors and Attacks

- 1. Average Detector
- 2. ARMA-GLR
- 3. Nonparametric Statics (EWMA and CUSUM)
- 4. Unsupervised Learning (LOF)

Average Detector

- $\bar{Y} = \frac{1}{N} \sum_{i=1}^{N} Y_i$

In the paper they select the threshold in the following way:

1. 1. Given a training dataset, say T days in the most recent past, we can compute T daily averages, D_i (i = 1,...,T).
2. 2. τ = $\min_i(D_i)$

- If $\bar{Y}$ <τ, where τ is a variable threshold.

- sending τ as $\widehat{Y_i}$ all the day

Formulas from “ECI Telecom. Fighting Electricity Theft with Advanced Metering Infrastructure (March 2011) “

ARMA-GLR (Auto-Regressive Moving Average)

- ARMA probability distribution p0:

$$Y_k = \sum_{i=1}^{p} A_i Y_{k-i} + \sum_{j=0}^{q} B_j V_{k-j}$$

- If $Y_k > Y$

- The attacker creates the probability distribution (P_γ) based on:

$$Y_k = \sum_{i=1}^{p} A_i Y_{k-i} + \sum_{j=0}^{q} B_j (V_{k-j} - \gamma)$$

Formulas from "Forecast package for R, http://robjhyndman.com/software/forecast/"

EWMA (Exponentially-weighted Moving Average)

- $EWMA_i = \lambda Y_i + (1 - \lambda)\, EWMA_{i-1}$

λ is a weighting factor and $0 < \lambda \leq 1$
Yi is one of the time series measurements

- If $EWMA_i < \tau$, where τ is a configurable parameter.

- When $EWMA_{i-1} > \tau$, send $\widehat{Y_i} = MAX(0, \frac{\tau - (1-\lambda) EWMA_{i-1}}{\lambda})$
- When $EWMA_{i-1} = \tau$, send $\widehat{Y_i} = \tau$.

Formulas from “EWMA Control Charts, http://itl.nist.gov/div898/handbook/pmc/section3/pmc324.html”

CUSUM (Cumulative Sum Control Chart)

- S_i = MAX(0, S_{i-1} + (μ − Y_i − b))

  (i = 1,...,N)

μ is the expected value of the time-series,

b is a “slack” constant defined so that E[|μ − Y_i| − b] < 0 under normal operation

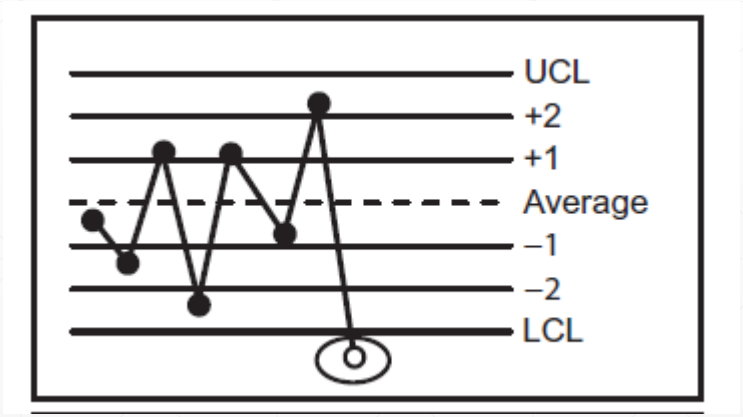
- if S_i > τ

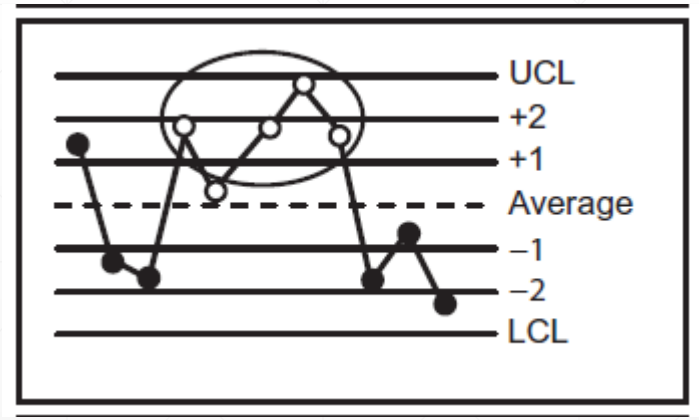
- Calculate M = $\frac{\tau+Nb}{N}$ and send $\widehat{Y_i}$ = μ − M

Formulas from “Non-Parametric Methods in Change-Point Problems. Kluwer Academic Publishers” by Brodsky, B., Darkhovsky, B

EXTRA: XMR chart (individuals and moving range chart)

- EWMA and CUSUM identify minor changes in small regions
- XMR chart identifies large changes over the time, it determines whether the process is stable and predictable or not
- It calculates various thresholds and classifies the anomalies depending on how the process overpasses those thresholds





Images adapted from “Fraud detection in registered electricity time series” by Josif V. Spiric, Miroslav B. Docic and Slobodan S. Stankovic

LOF (Local Outlier Factor)

- 1. Create a vector containing all measurements of a day to be tested in order, $V_{test} = \{Y_1, \ldots, Y_n\}$ where N is the number of measurements per day.
- 2. For all days in a training dataset, create vectors in the same way, $V_i = \{x_{i1}, \ldots x_{iN}\}$ (i = 1,...,T).
- 3. Create a set containing V_{test} and all V_{is}, and apply LOF to this set.
- 4. If $LOF_{test} < \tau$ where LOF_{test} is a score corresponding to V_{test} , conclude V_{test} is normal and exit.

$$\bar{Y}\left(= \frac{1}{N}\sum_{i=1}^{N} Y_i\right) < \frac{1}{NT}\sum_{i=1}^{T}\sum_{j=1}^{N} X_{ij}$$

- Select the value with the minimum consumption among the ones that their LOF score is less than τ.
- Reduce that value without raising the alarm and send it.

Fomulas from “Lof: Identifying density-based local outliers“ by Breunig, M., Kriegel, H.-P., Ng, R.T., Sander, J.

EXTRA: COF (Connectivity-based Outlier Factor)

- It is a variation of LOF, created because of the excessive complexity (N^2)
- LOF creates a neighbourhood based on the main given instance
- COF creates the neighbourhood in an aggregative way from the main given instance

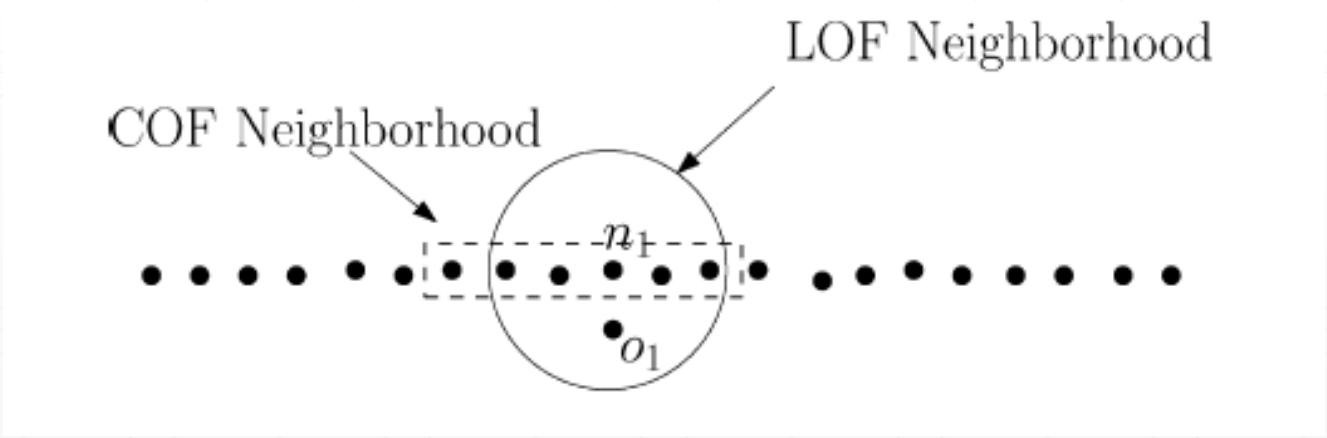


Image adapted from “Anomaly Detection: A Survey” by Varun Chandola, Arindam Banerjee and Vipin Kumar

Results

- In the paper, they use real meter-reading data measured by an electric utility company during 6 months.
- The meter reading consisted of 108 customers with a mix of residential and commercial customers and recorded every 15 minutes.

Adversarial Evaluation: Cost of Undetected Attacks

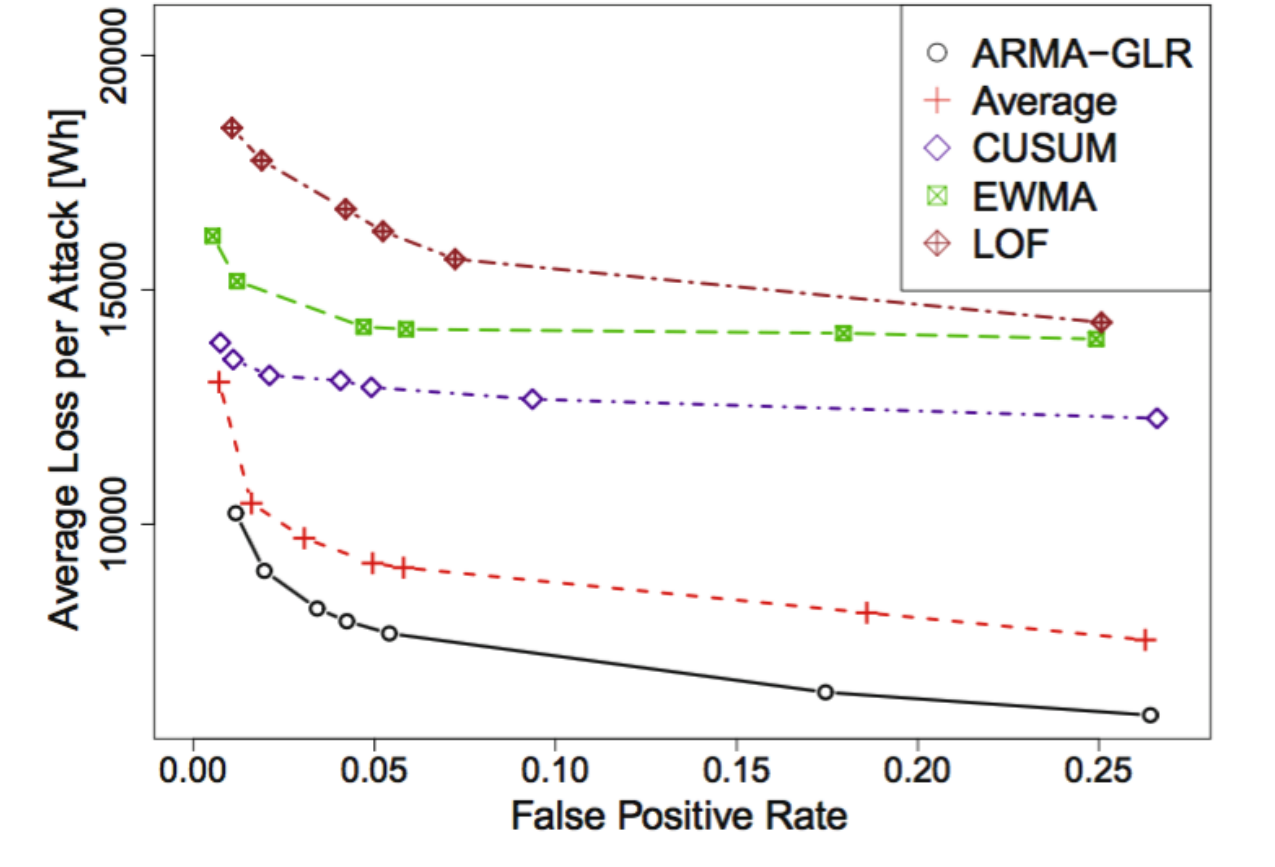
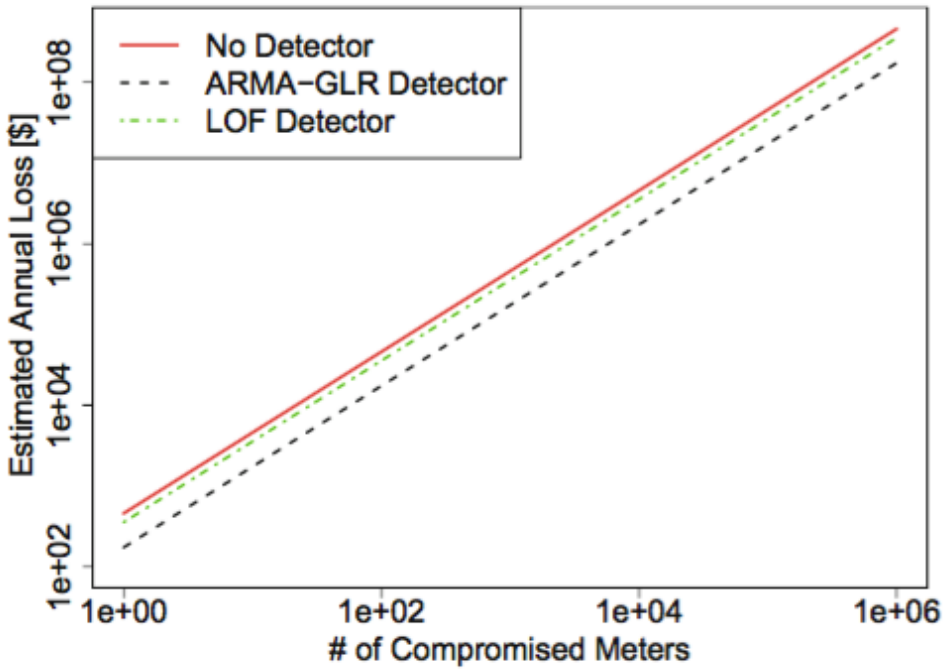
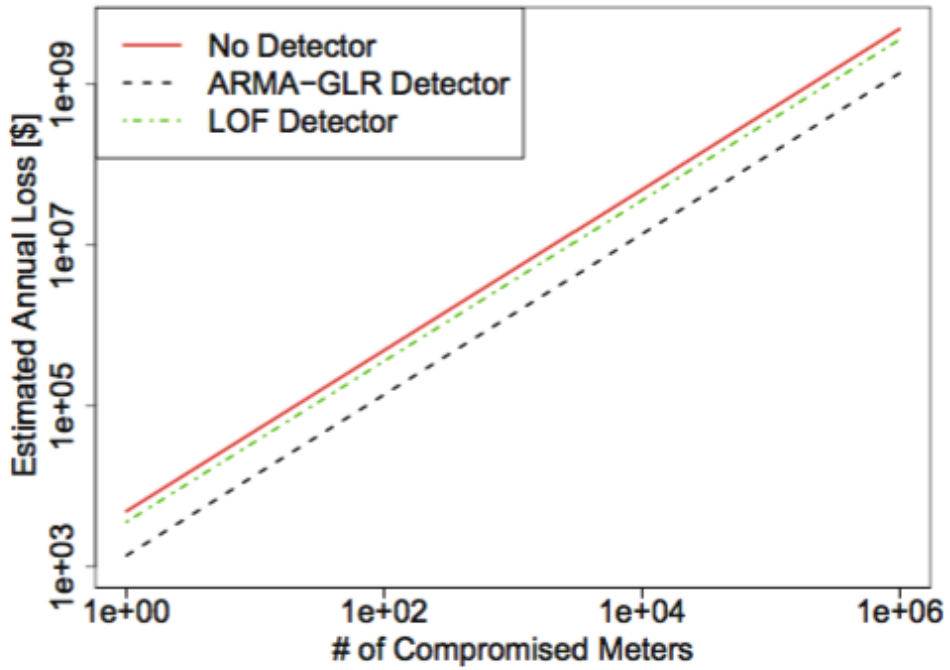


Image adapted from “Evaluating Electricity Theft Detectors in Smart Grid Networks” by Daisuke Mashima and Alvaro A. Cárdenas

Monetary Loss Caused by Different Customers



Average customers



High consumption Customers

Image adapted from “Evaluating Electricity Theft Detectors in Smart Grid Networks” by Daisuke Mashima and Alvaro A. Cárdenas

Adversarial Learning: Detecting Contaminated Datasets

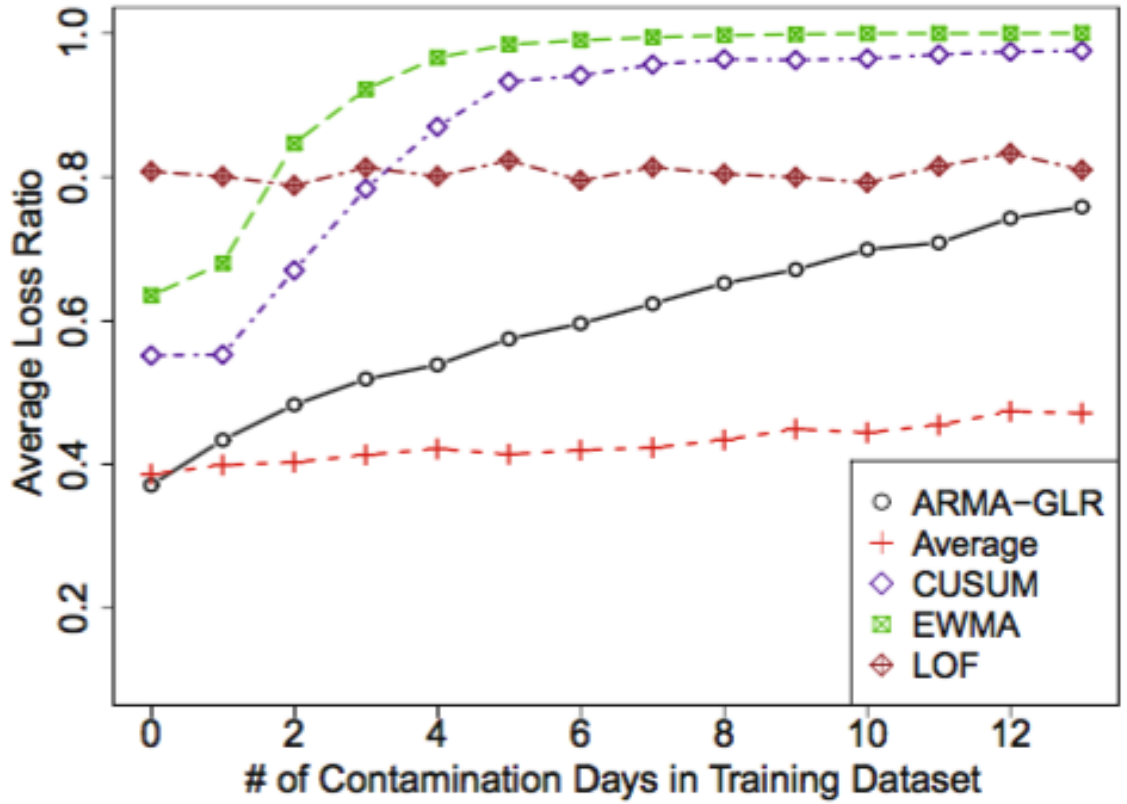
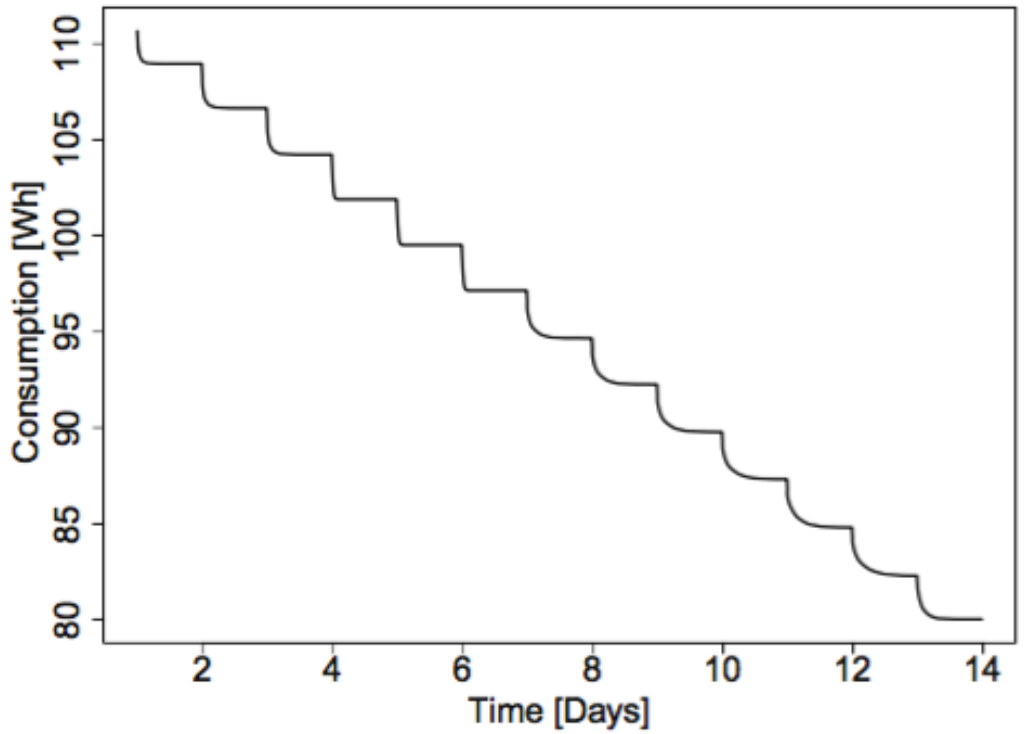


Image adapted from “Evaluating Electricity Theft Detectors in Smart Grid Networks” by Daisuke Mashima and Alvaro A. Cárdenas

Adversarial learning: Types Of Contamination



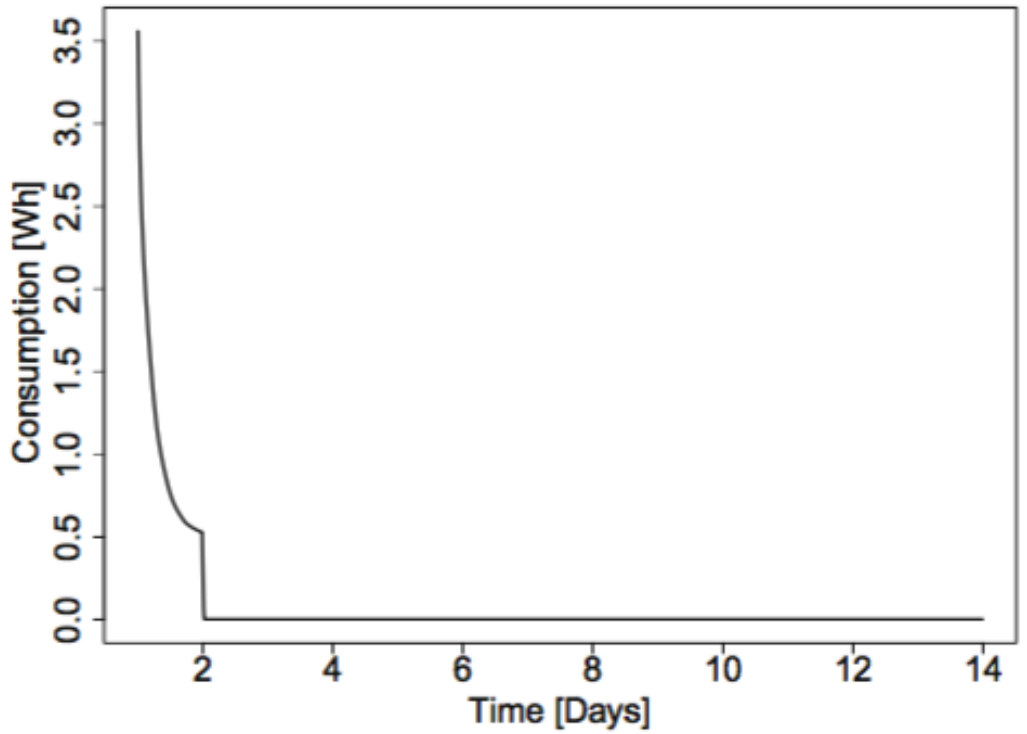


Image adapted from “Evaluating Electricity Theft Detectors in Smart Grid Networks” by Daisuke Mashima and Alvaro A. Cárdenas

Discussion

- Cross-Correlation among customers
  - Attackers should exhibit different trends compared to similar honest customers
  - LOF to identify outliers
- Auto-Correlation in ARMA-GLR
  - The residuals of generated attacks have high auto-correlation
  - Durbin-Watson statistics then can detect attacks against ARMA-GLR
- Energy Efficiency
  - Customers can add green-energy technology to the system, such as, solar panels
  - Company should know about this information to set the classifiers properly
- Not general classifiers
  - Depends on consumers' lifestyle (changes from countries, areas, etc)

Conclusion

- These algorithms will perform much better under average cases where the attacker does not know the algorithm or time intervals we use for anomaly detection
- For companies, they need to combine all the information to consider their network and accurate the electricity-theft reports
- The proposed anomaly detector will only output indicators of an attack