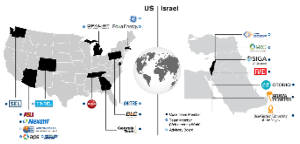
Enhancing Cybersecurity of Grid Operations

BIRD Review Meeting —Tasks 5 and 8

Lalitha Sankar
Associate Professor
Arizona State University

March 2023

ASU Team Members



Obai Bahwal
2nd Year PhD Student

Rajasekhar Anguluri
Postdoctoral Scholar

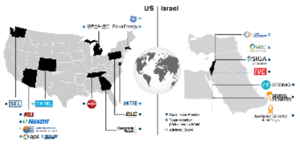
Joel Mathias
Postdoctoral Scholar

Nima T. Bazargani
5th Year PhD Student

Avinash Kodali
MS Student

Also collaborating with John Dirkman, Narsi Vempati, Guanji Hou @ Resource Innovations Inc.

Enhancing Cybersecurity of Grid Operations

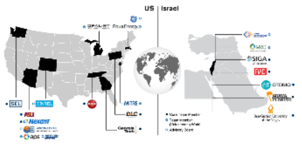


Task 5: Generate event-mimicking attacks

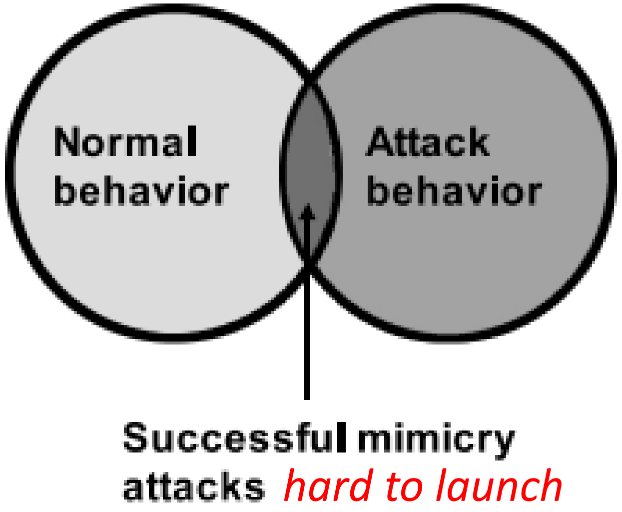
Task 8: Detect event-mimicking attacks

Commercialization: Software development with Resource Innovations

Event-mimicking Attacks and Countermeasures



- Modern grid with renewables is more stochastic in operations and requires real-time monitoring to detect/identify real events (oscillations/outages) and attacks.
- ML-based detectors can be easily evaded by attacks that mimic events, ultimately, causing significant damage on grid operations.

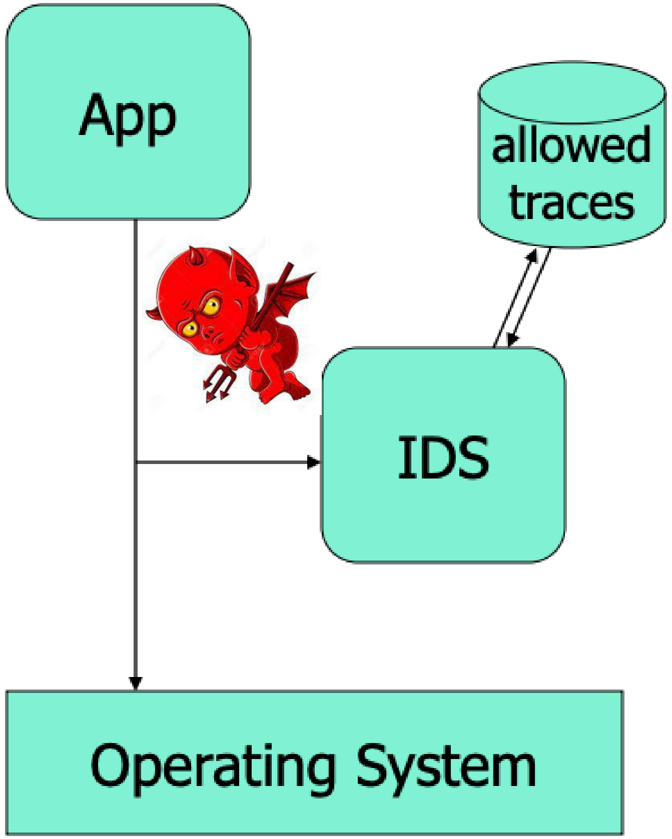


mimicry attack: a careful cyberattack on data that throws off ML detector

Source: https://towardsdatascience.com/evasion-attacks-on-machine-learning-or-adversarial-examples-12f2283e06a1

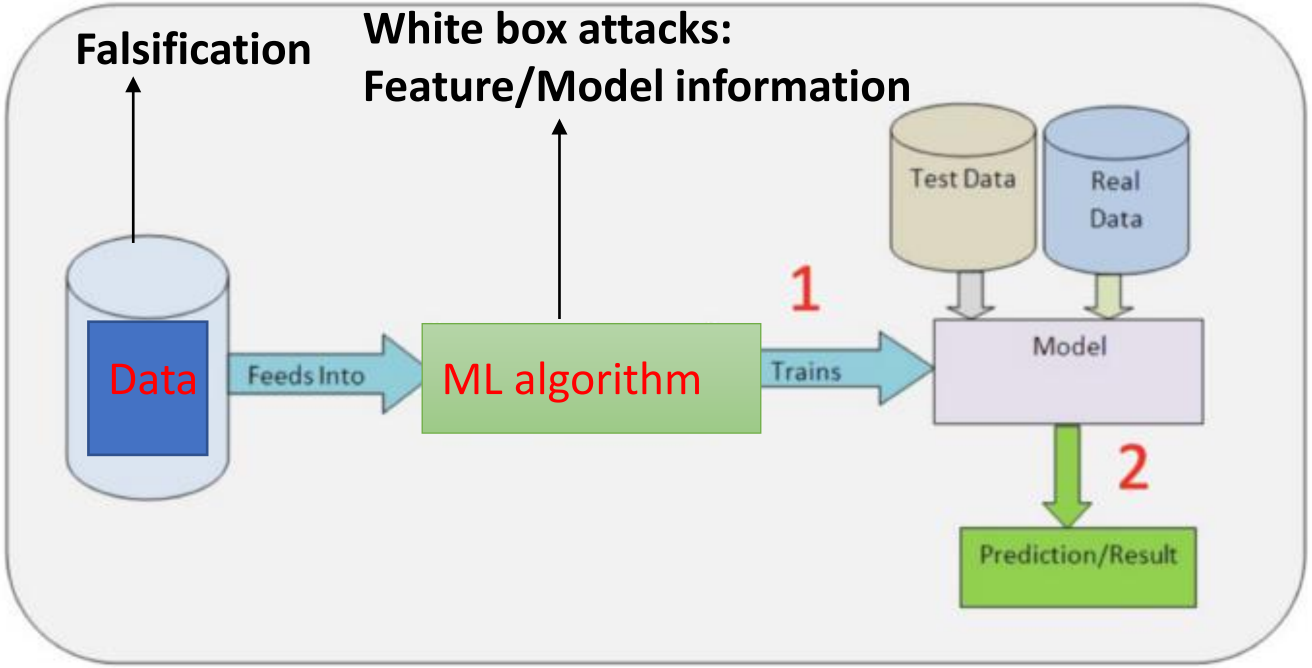
Mimicking Attacks in OT Systems

- Mimicking attacks have historically focused on IT systems
- Operational Technology (OT) systems are also vulnerable to mimicking attacks
- OT systems in power grid consider dynamics, temporal correlations of data, etc.
- Attacker can intrude OT systems at multiple locations



attacks target software internal to a computer

Where Can Attackers Target OT Systems?



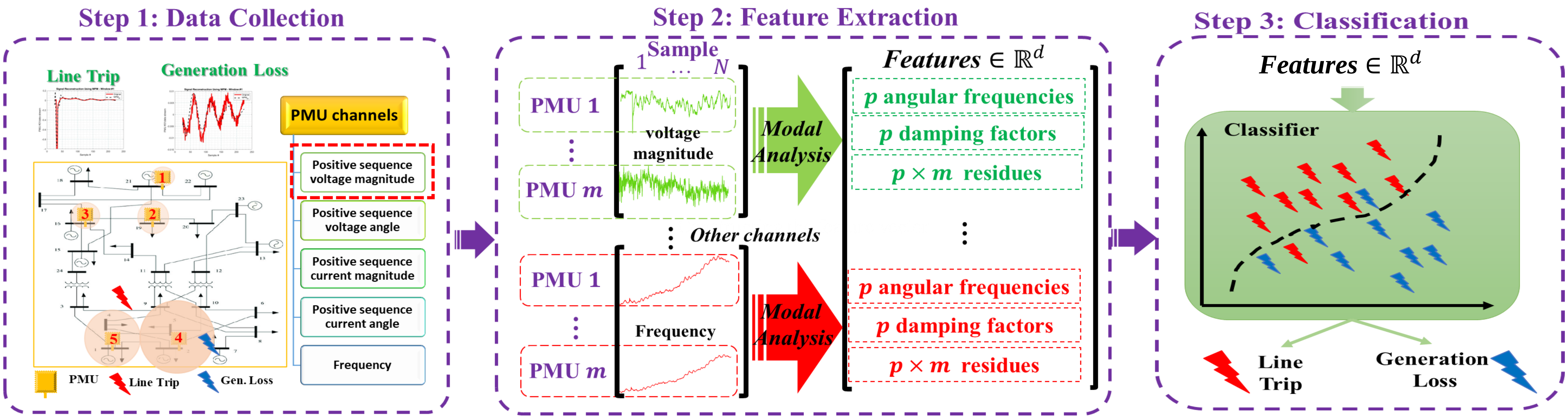
PMU data can be falsified but for mimicking event attacks

- *how to tamper data?*
- *how many PMUs to tamper?*
- *how long to tamper?*

extract and exploit signal physics (modes)

Source: https://towardsdatascience.com/will-my-machine-learning-be-attacked-6295707625d8

Task 5: Learn Event Signatures from Measurements

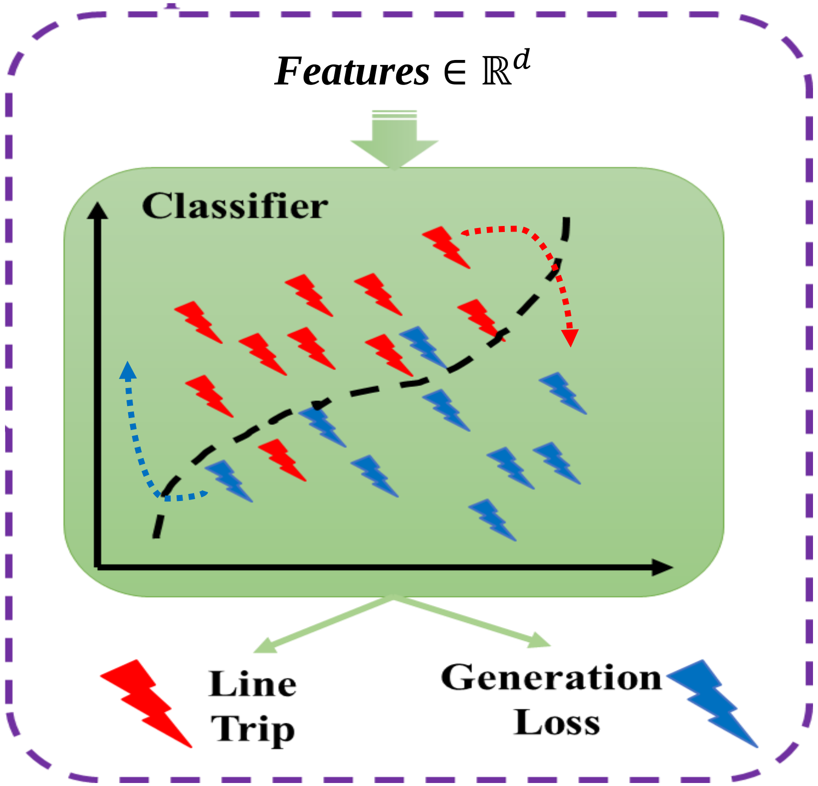


- ✓ Characterizing events based on a set of physically interpretable features
- ✓ Finding the most informative sparse set of features
- ✓ Learning a set of robust classification models to identify the events

[1] N. Taghipourbazargani, G. Dasarathy, L. Sankar and O. Kosut, "A Machine Learning Framework for Event Identification via Modal Analysis of PMU Data," in IEEE Transactions on Power Systems, 2022.

Task 5: Threat Model

- Start with White Box Attack Model: Attacker has full information of the event classifier (LR)
- Untampered Features:
  - Angular Frequency
  - Damping
  - Residual Amplitude
  - Residual Angle
  - Channels: Voltage magnitude, voltage angle, frequency
- Tamper features just enough for the event to be misclassified
  - Move feature sample across decision boundary

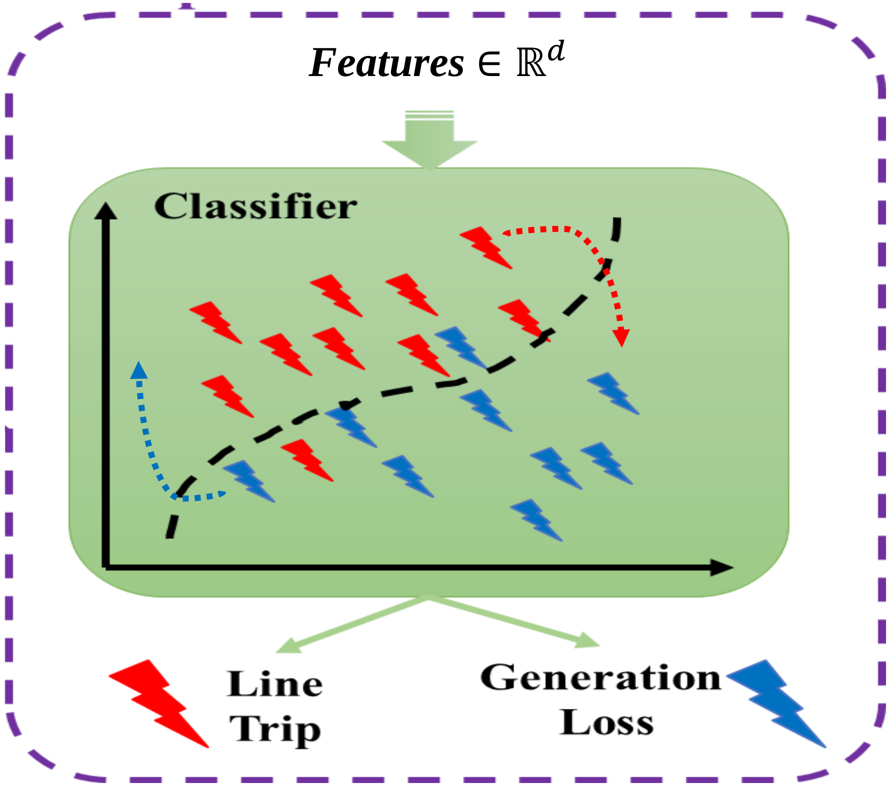


Task 5: Event Mimicking Attack Algorithm

Inputs: LR classifier, attack parameters, PMU data

1. Tamper features until the event is misclassified by employing the knowledge of LR parameters
2. Reconstruct time signals of the tampered data
3. Replace the time domain signals for only the PMUs under the attacker's control
4. Extract features of the new signals set
5. Classify using LR model
6. Repeat 1 through 5 until misclassification

Output: tampered PMU measurements

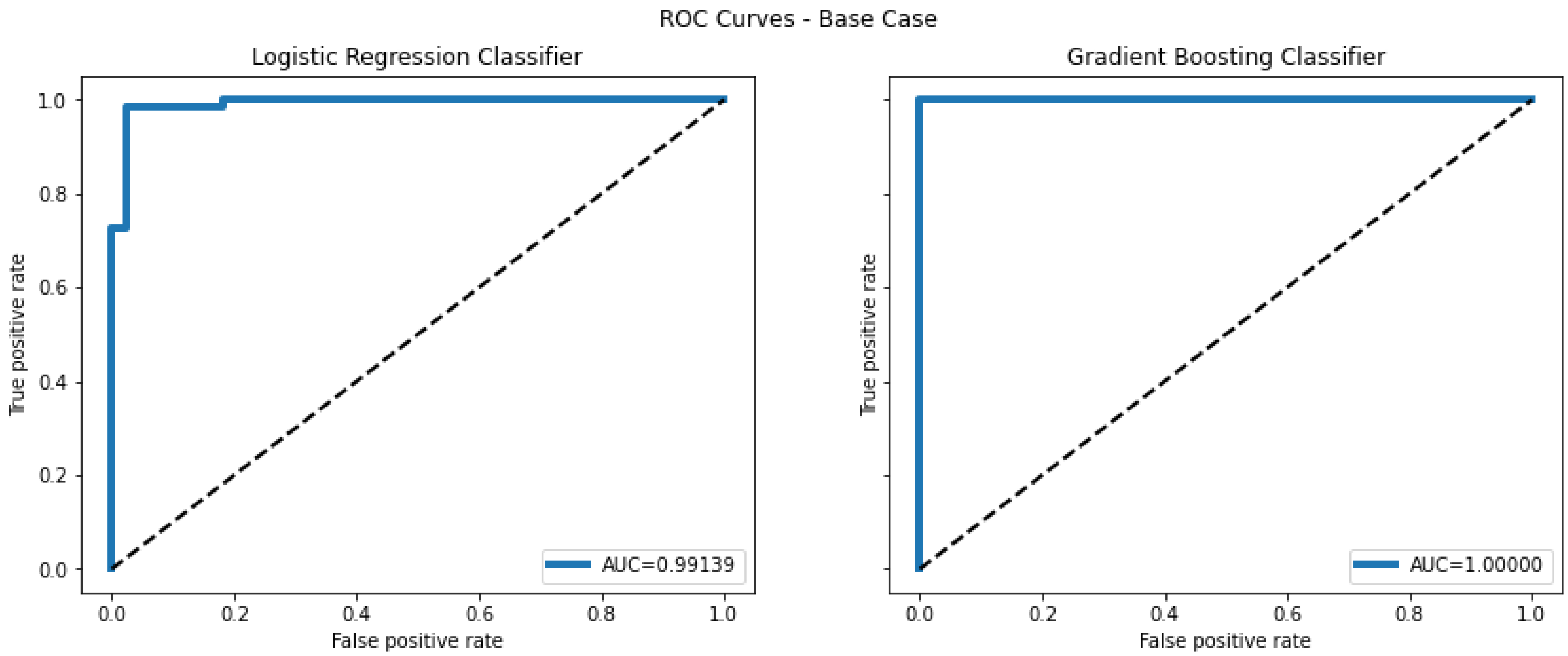


Task 5: Setup and Assumptions for Illustrations

- Network and data: synthetic PMU data generated using PSS\E for Texas 2000-bus system
  - 400 generation loss and 400 line trip events
  - Voltage magnitude, voltage angle, and frequency measurements are collected from 95 PMUs across the system
- Classifiers: Logistic regression (LR) and gradient boosting (GB) algorithms
  - Training data: 317 generation loss and 323 line trip events
  - Test data: 83 generation loss and 77 line trip events
  - Modal analysis is used for feature extraction

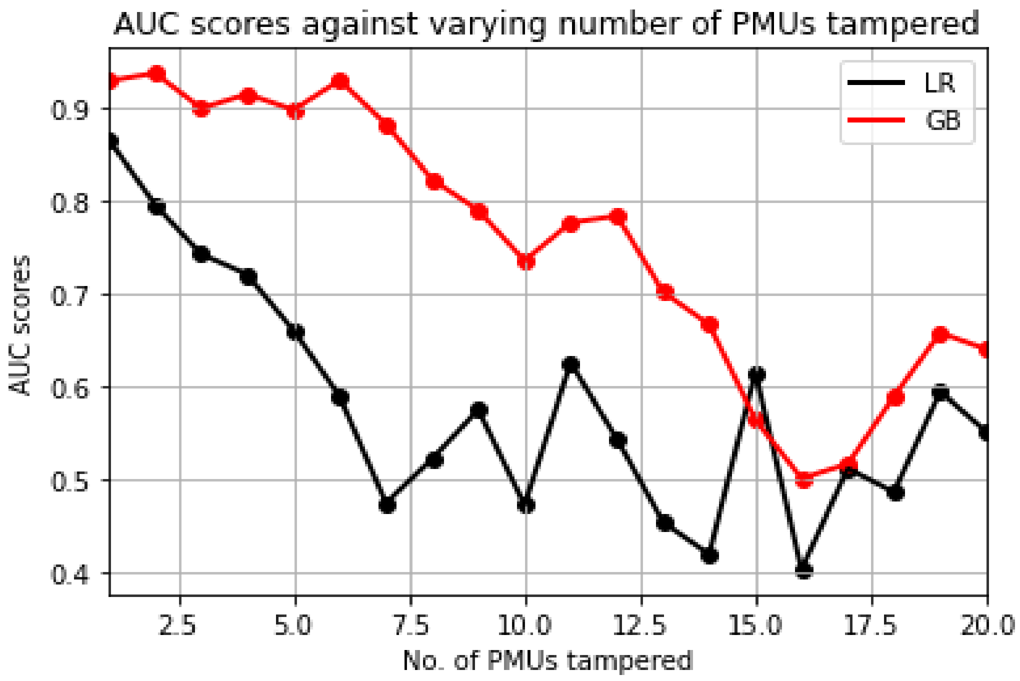
Task 5: Classification of untampered events

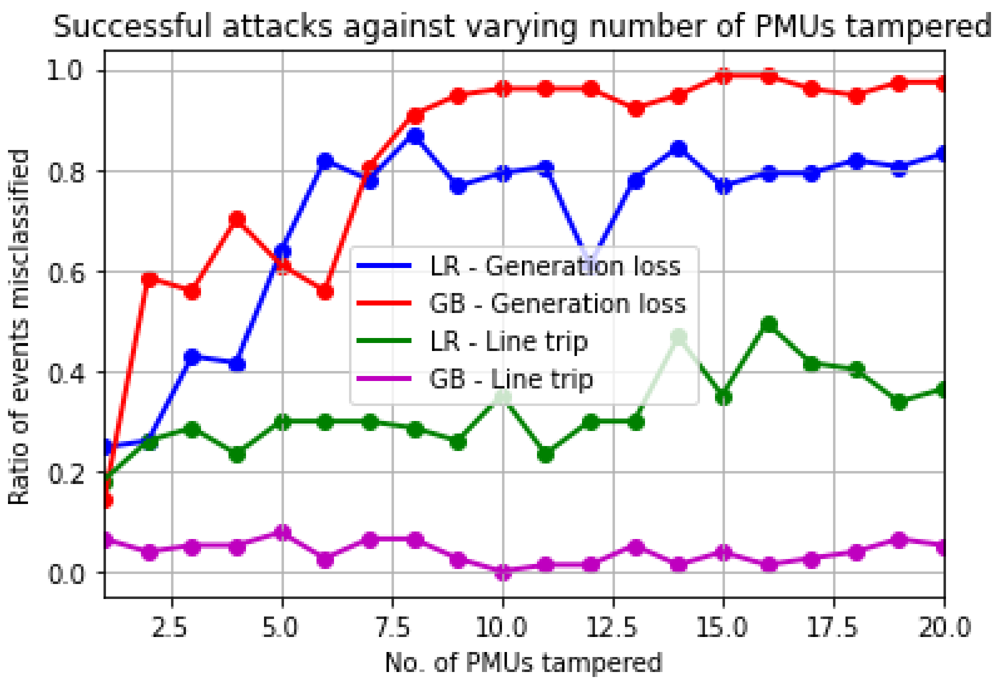
- Event classifier is applied to 160 test data (83 generation loss and 77 line trip events)
- LR and GB classifiers are used to classify untampered test data to establish a base case
  - Both models are trained on the same dataset
- Both models classify the events with very high accuracy

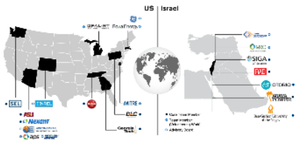


Task 5: Attack Illustration

- Attack Assumptions:
  - Attacker has full knowledge of LR classifier model
  - Attacker has access to a subset of system PMUs (no more than 20)
  - Tampers 160 test data comprised of 83 generation loss and 77 line trip events
- Efficacy of tampered data also evaluated on GB classifier (trained on clean data)
- Results: overall successful attack with higher success rate when applied to generation loss events
  - Line trip events are harder to tamper

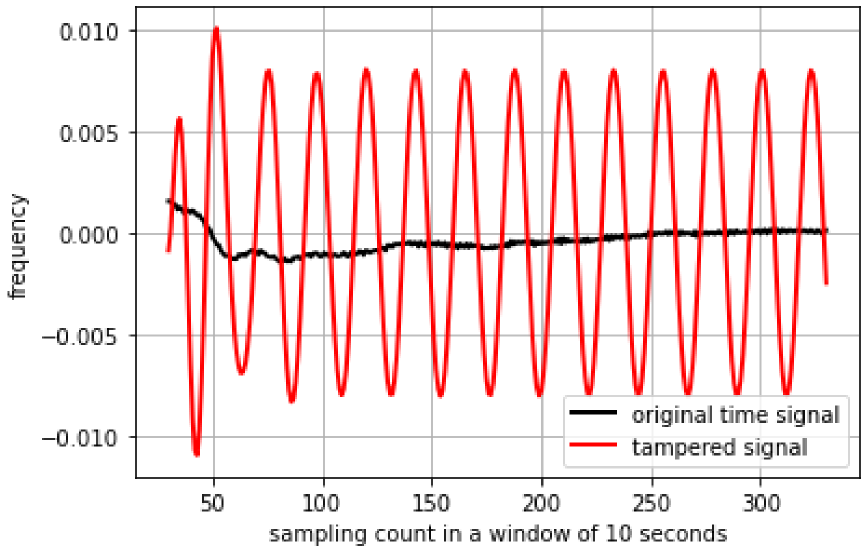


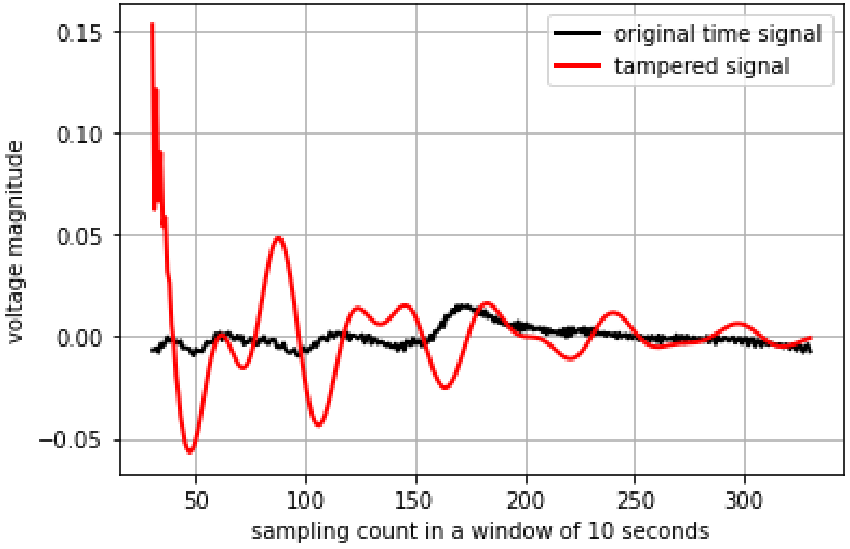


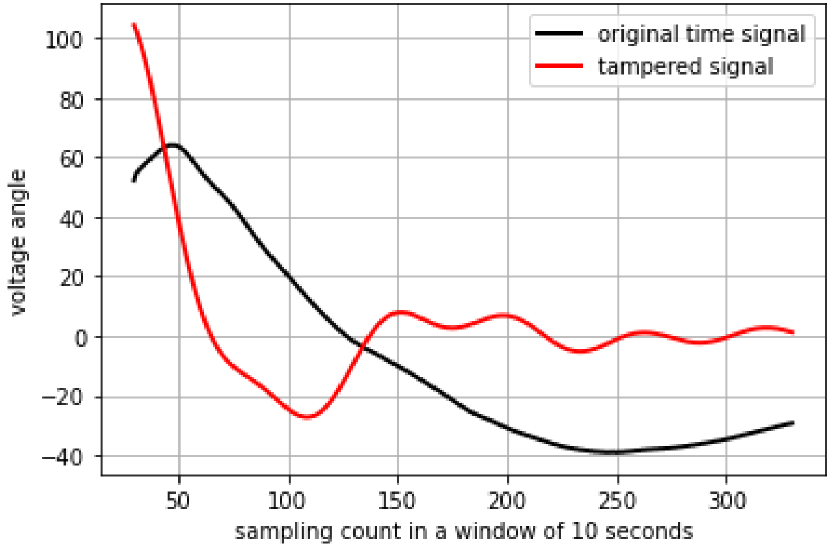


Task 5: Illustration of Event Mimicking Signals

- What is the effect of the attack on the temporal signals?
  - Illustration here for an attack limited to 10 PMUs
  - **Attack: Tamper Generation Loss event**
- Tampered time signal for one such PMU:
  - Frequency, voltage magnitude, and angle plotted
  - All channels are tampered in this attack

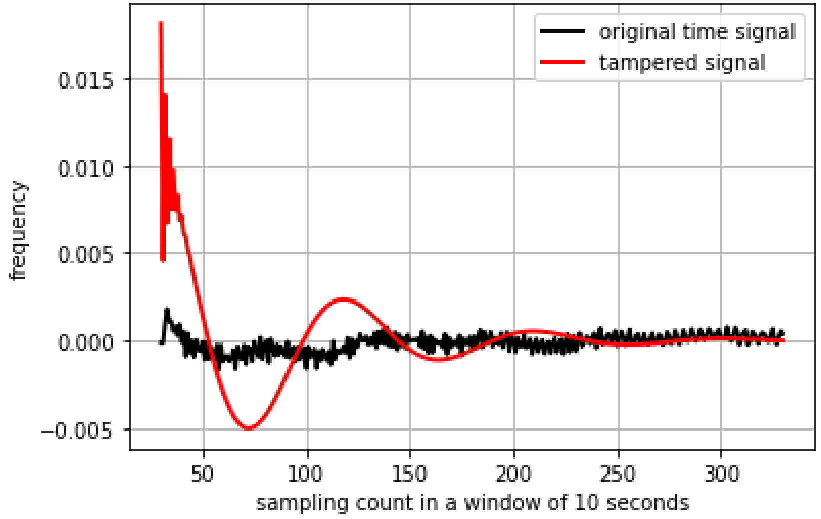


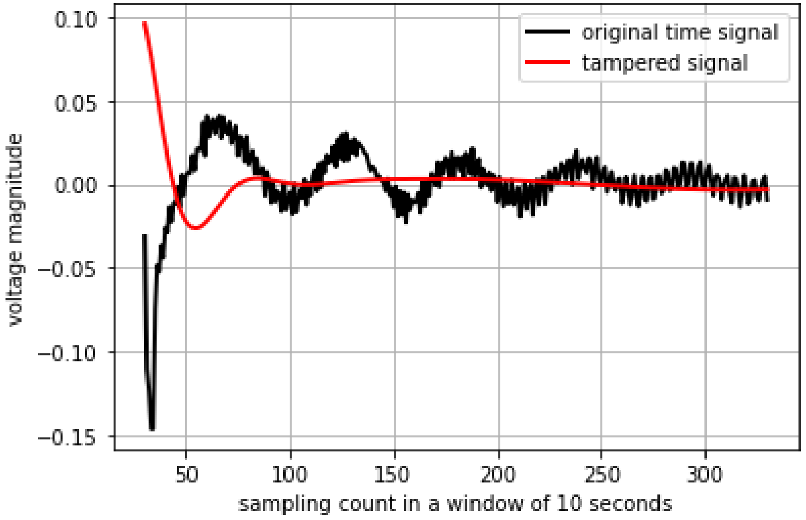


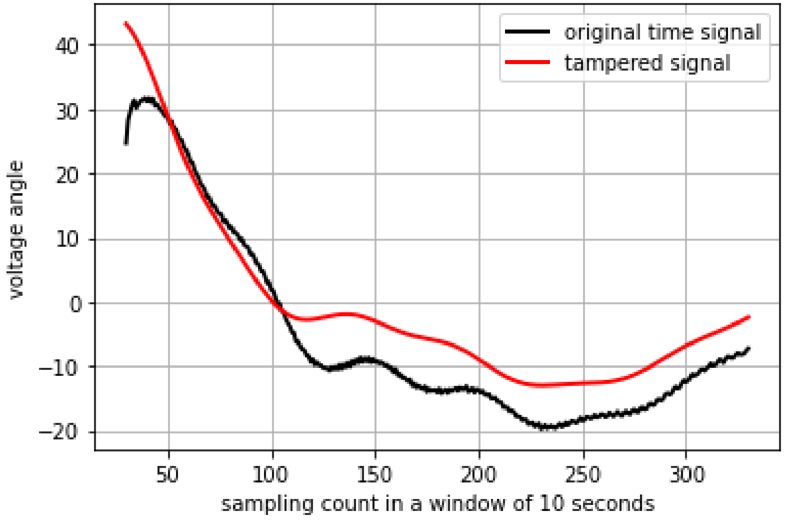


Task 5: Illustration of Event Mimicking Signals

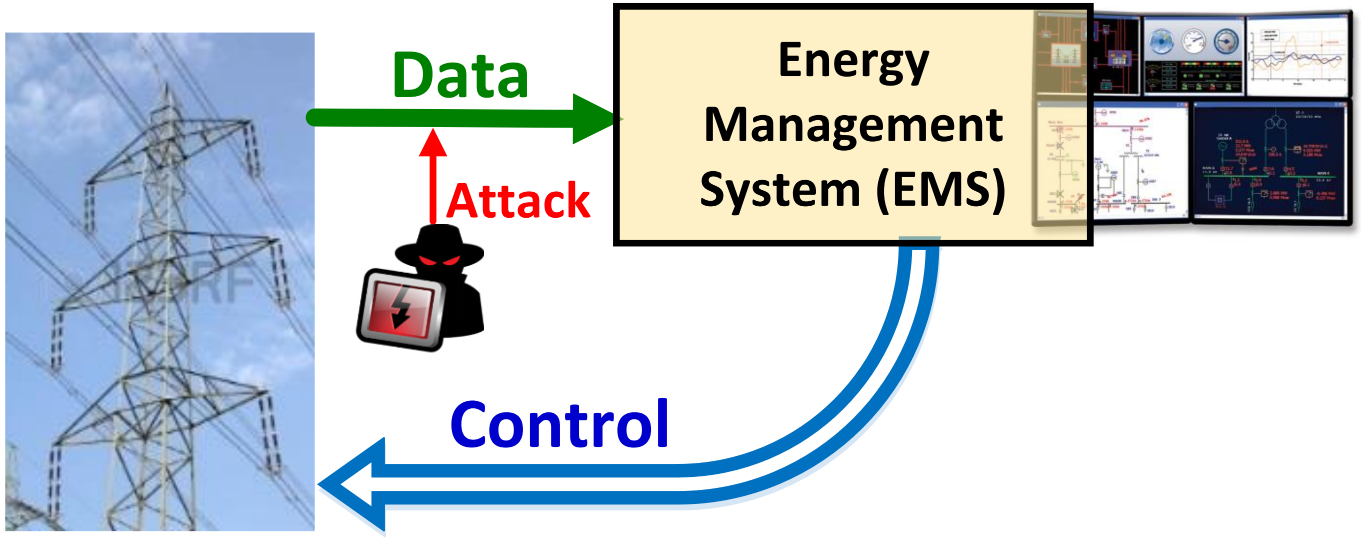
- Illustration shown here for a **tampered line trip** event
  - Illustration here for an attack limited to 10 PMUs
- measurements from an attacked PMU
  - Led to a successful misclassification of line trip as generation loss







Countermeasures for False Data Injection Attacks

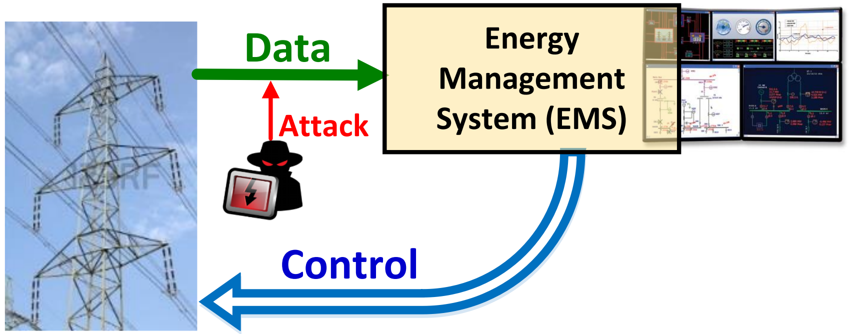


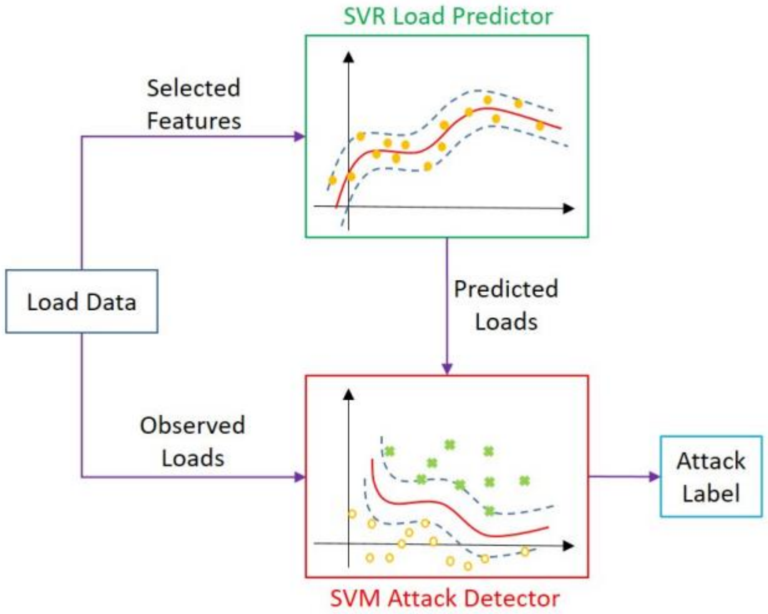
- Knowing network configuration, attackers can maliciously change a subset of measurements with counterfeits before they reach the EMS
- Requires attacker to have access to measurement devices or data concentrators
- Can be unobservable and result in physical [2] / economic [3] consequences

[2] Zhang, J., Sankar, L.: ‘Physical system consequences of unobservable state-and-topology cyber-physical attacks’, IEEE Transactions on Smart Grid, 2016, 7, (4), pp. 2016–2025
[3] Moslemi, R., Mesbahi, A., Velni, J.M.: ‘Design of robust profitable false data injection attacks in multi-settlement electricity markets’, IET Generation, Transmission Distribution, 2018, 12, (6), pp. 1263–1270
[4] Liang, J., Sankar, L., Kosut O.: ‘Vulnerability analysis and consequences of false data injection attack on power system state estimation’, IEEE Transactions on Power Systems, 2015, 31, (5), pp. 3864-72

Detecting Load Redistribution Attacks via Support Vector Models

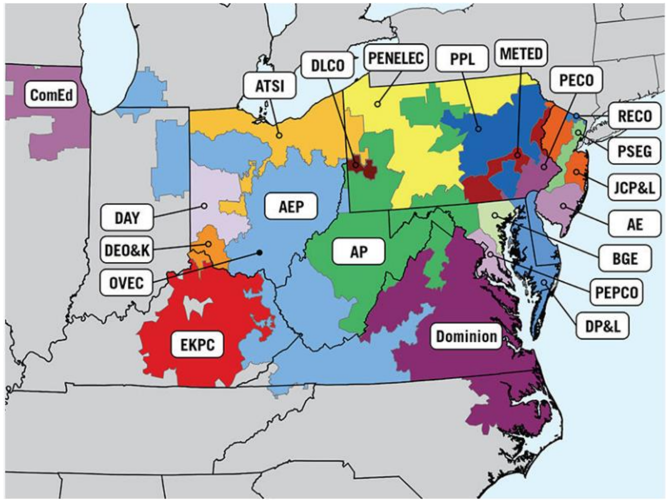
- Load Redistribution (LR) attacks: redistribute loads across buses without any change in net load
  - Current net load prediction approaches can miss this entire class of false data injection attacks (FDIA)
- Our detection methodology:
  - Grid telemetry including loads follow diurnal and seasonal patterns
  - Historical data can be used to predict such patterns
  - ML algorithms trained on such temporally correlated data can be used to predict loads at the bus-level
- Use multi-output support vector regression (SVR) load predictor
  - predicts loads by exploiting both spatial and temporal correlations
- Combine with a support vector machine (SVM) classifier to classify incoming load estimate as either normative or attacked

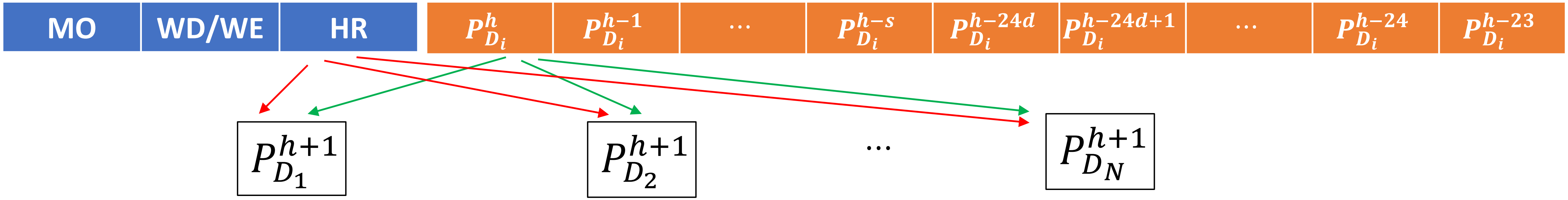




Load Prediction using SVR

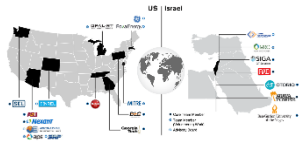
- Dataset
  - PJM hourly zonal load data [5], 20 zones in total
  - Mapped publicly available PJM load data to the 30-bus system
- Feature selection to predict loads at hour $h + 1$
  - Time information
  - Historical load values at past s hours, as well as at hour HR and HR+1 at past d days
  - Combine these values for multiple loads to capture spatial correlations
  - Can be applied to predict bus level loads



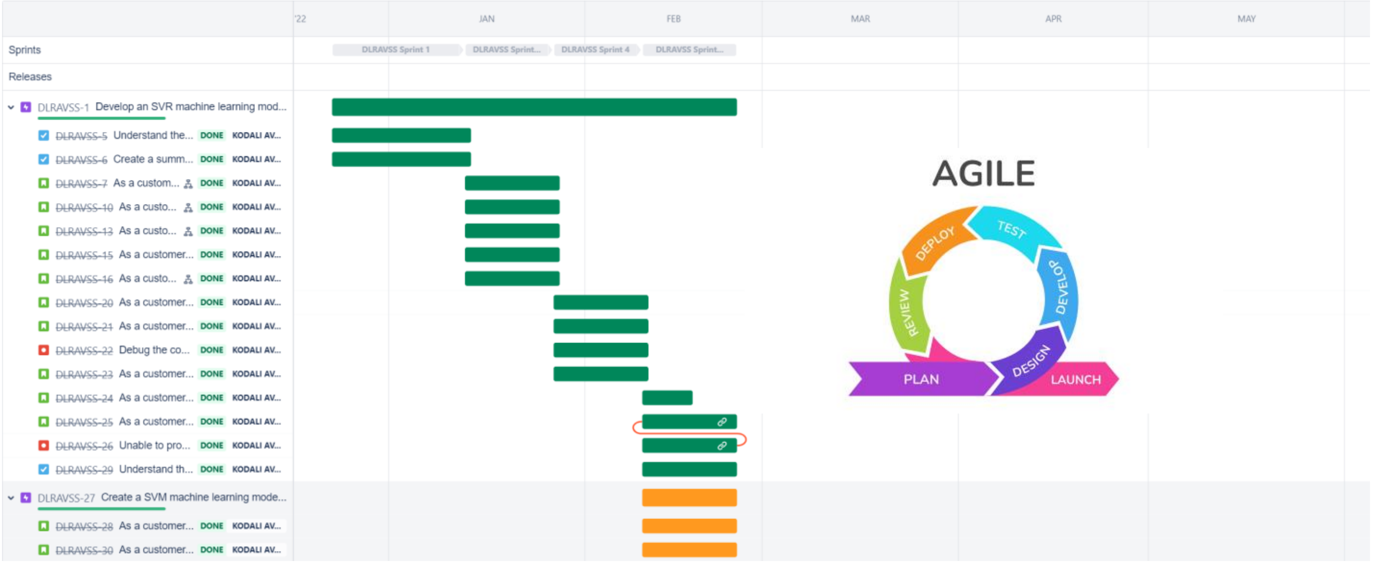


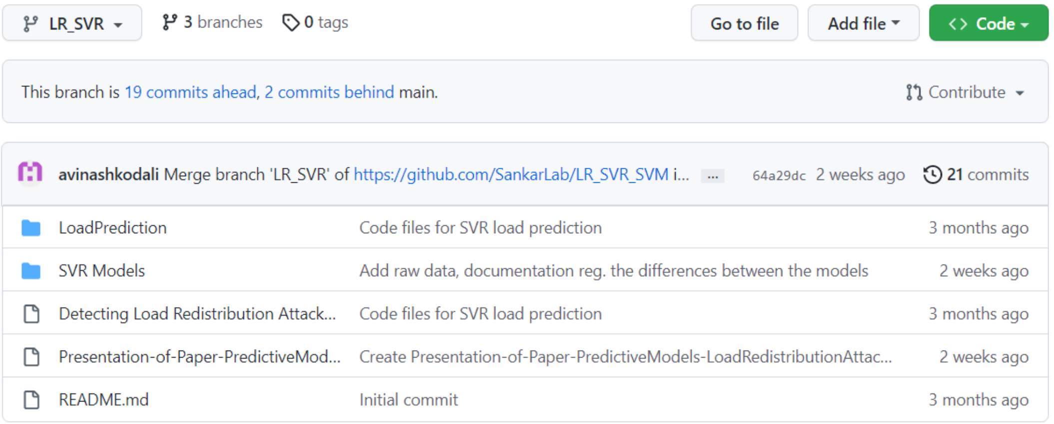
[5] "PJM metered hourly zonal load data," 2019. PJM Data Miner 2, https://dataminer2.pjm.com/feed/hrl_load_metered/definition

Commercialization: Load Prediction using SVR

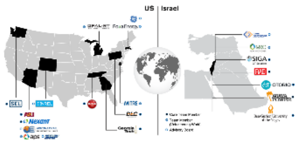


- Modularized and documented the load prediction Python code which makes it easier to understand
- Performed rigorous testing on the load prediction code using the IEEE 30-bus system (map PJM loads to this system)
- Agile methodology using Jira to ensure timely completion of work
- Version control using GitHub throughout the project, enabling efficient tracking and management of code changes



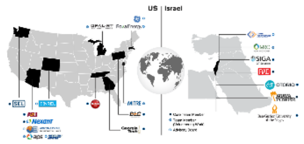


Commercialization Efforts with RI



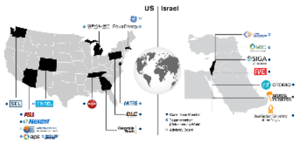
- On-going team meetings with RI to hand-off code
- Corresponding ASU team:
  - Lalitha Sankar (PI)
  - Postdocs: Joel Mathias (commercialization effort liaison), Rajasekhar Anguluri (countermeasure development)
  - Avinash Kodali (load prediction, attack design, and anomaly detection)
  - Nima T. Bazargani (event-ID)
  - Obai Bahwal (event-mimicking attacks and countermeasures)
- Specific questions on data and code changes discussed in these meetings
- Focus is on streamlined commented code (all in Python)
- RI to test algorithms under industry level simulations

Commercialization Meetings with RI



- Corresponding RI team:
  - John Dirkman and Narsi Vempati (leads)
  - Guanji Hou (consultant)
- RI is continuing to engage with industry partners to determine viability and best methods for commercialization of Load Prediction, Redistribution Attack Detection and Mitigation code
  - A new engine to predict, monitor, and mitigate load measurement attacks

Commercialization Process - Load Prediction, Redistribution Attack Detection and Mitigation



Load Prediction

Setup:
- Obtain and install developed code on local machine
- Obtain and install input data on local machine
- Obtain and review user guide/guidance
- Obtain and install third-party applications
- License fee for third-party applications
- License structure for commercialization

Commercialization Plan and Revenue Estimate:
- Lean Canvas
- Discuss product with potential customers
- Revenue Estimate
  - Cost of Commercialization
  - Price for Product
  - Price for Support and Maintenance
  - Number of Installations
  - Revenue from Product
  - Revenue from Support and Maintenance
- Go/No Go Decision

Design:
- User Experience:
  - Data Input
  - Processing
  - Output/Visualization
- Review use of third-party applications and options for mitigating or not using them
- Integration with other applications - APIs
- Testing Plan
- Discuss product design with potential customers

Develop:
- User Experience:
  - Data Input
  - Processing
  - Output/Visualization
- Minimize use of third-party tools
- Integration with other applications - APIs
- Testing and defect resolution
- Installation and User Guides

Deploy:
- Marketing Collateral
- Sales Support
- Installation Support
- Training
- Testing and defect resolution
- Ongoing Support

The Lean Canvas

Designed for: Load Prediction, Redistribution Attack Detection and Mitigation

Designed by: John Dirkman

Date: 9 March 2023

Version: 1.0

Problem

Utilities lack software to predict and detect attacks intended to redistribute load measurement data.

Solution

Develop software to predict and detect attacks intended to redistribute load measurement data that can work with existing SCADA systems.

Unique Value Prop.

There is currently no commercially available software to predict, detect, and prevent attacks on loads.

Unfair Advantage

1. ASU domain knowledge and research.
2. Easier path to commercialization using Grid360 engines framework
3. Established sales and delivery channels.

Customer Segments

Electric Distribution Utility Companies Worldwide

Existing Alternatives

While there have been technical papers published on this topic, no known commercial software currently provides this capability.

Key Metrics

Customer contacts, RFP's received, contracts closed.

High-Level Concept

Use support vector regression (SVR) for enhanced load prediction, then combine with a support vector machine (SVM) classifier to classify incoming load estimate as either normative or attacked.

Channels

1. Direct to utilities
2. Via business partners: GE, Hitachi/ABB
3. Via SI's: Infosys, Accenture, Capgemini, Deloitte, Guidehouse, HCL

Early Adopters

Existing RI and business partner clients

Cost Structure

List your fixed and variable costs:

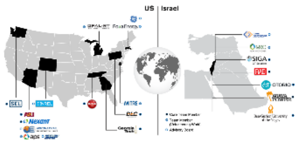
- Business development costs
- Software development and testing costs
- Sales engineering costs
- Project implementation costs

Revenue Streams

List your sources of revenue:

- Software licenses: one-time/perpetual or annual/subscription/SaaS
- Implementation/integration
- Ongoing support and maintenance

Summary



	Details	Status
Task 5 (attack generation)	• Synthesize intelligent attacks that mimic natural events (e.g., line trip, generation loss) by tampering measurements • Develop data poisoning methods using physics-informed machine learning methods to identify subsets of features amenable to perturbation	Completed: • designing data tampering attacks that spoof events • Identified events that are amenable to attacks In progress: • Identify attacks robust to multiclass event classifiers
Task 8 (attack detection)	• Develop ML and data-driven "robust" detectors that detect intelligent false data injection attacks • Algorithms to detect tampering of SCADA telemetry • This effort can also be a relevant countermeasure for the FDIA in task 9	Completed: • **Handed off tested Python code for bus-level load prediction to RI** In progress: • Rigorously testing Python code to generate random and FDI attacks • Developing countermeasures for event mimicking attacks
Industry Collaboration	• Developing commercial grade software for bus level load prediction in collaboration with RI	• Biweekly meetings with RI • RI evaluating business proposition