Applications of Deep Learning for Anomalous Energy Consumption Detection

GROUP 13: EE-163, EE-164, EE-177, EE-194

INTERNAL ADVISOR: DR. M. M. ALI BAIG – DEPT. OF ELECTRICAL ENGINEERING, NEDUET

EXTERNAL ADVISOR: MR. SHAHZEB ANWAR – ENI PAKISTAN LIMITED

The Problem: Electricity Theft

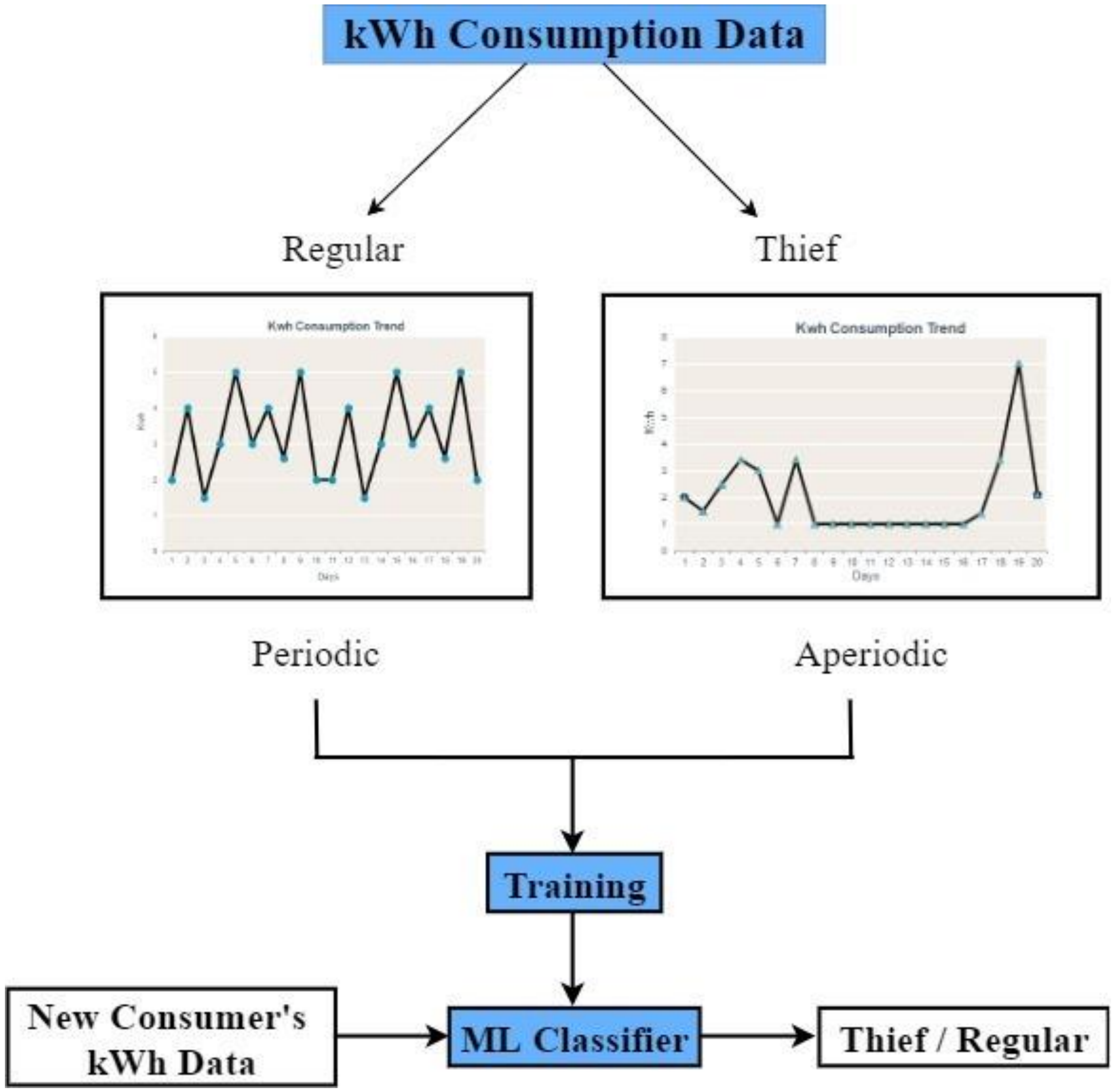
- Illegal consumption of electrical energy from DX grid.
- Bypassing energy meters, unmetered connections.
- Desynchronize energy meter communications
- Approx. annual cost: PKR 45 billion (ARY News - 2019)
- Global problem – CAD 100M annual NTL losses.
- Technical: Overloading, $\downarrow \eta$
- Safety hazard

Possible Solutions: Industry and Academia

- Industry: K-Electric – based on interviews with DX AM
  - PMT AMRs: $(P_{Supplied} - \Sigma P_{Consumed}) > \epsilon? \rightarrow$ Manual inspection
  - Manual inspection in batches every month.
  - Aerial Bundled Cables (ABCs) – deterring unmetered connections
  - Digital smart energy meters – replacing analog – easy to tamper.
  - Time consuming, costly, low granularity, poor to scale.
- Academia – Based on Literature Review
  - Advanced Metering Infrastructure (AMI)
  - Non-Intrusive Load Monitoring (NILM)
  - Host-based Intrusion Detection
  - Artificial Intelligence (AI) Techniques

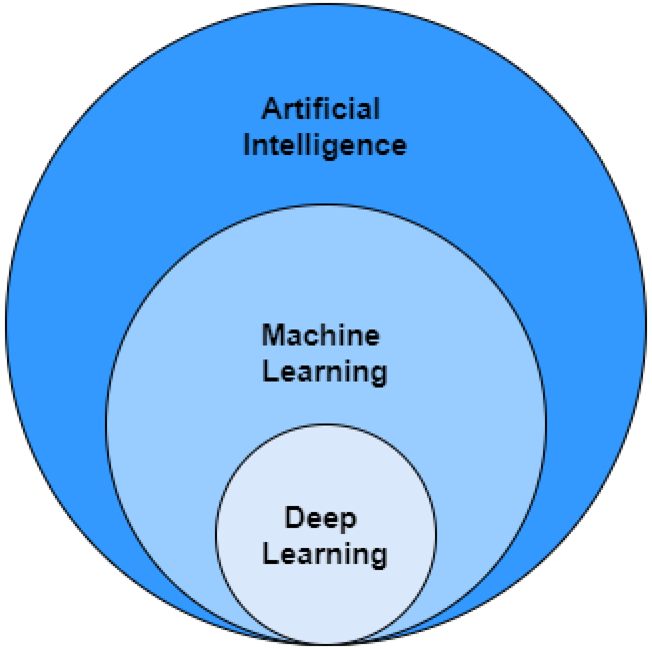
Our Proposal

- DX grids: pivoting to AMI.
- More AMI → More data.
- Data: consumption patterns.
- kWh consumption patterns: thieves ≠ regular consumers
- Thief consumption: aperiodic, anomalous
- ML models: learn from data.
- Distinguish thief/non-thief.
- Anomalous Energy Consumption Detection: **supervised binary classification problem**



Machine Learning

- Teaching computers to learn from data
  - Without being explicitly programmed.
- Supervised vs Unsupervised Learning
  - Supervised – labels provided, find $f: X \rightarrow Y$
  - Unsupervised – no labels, find structure
- Within supervised
  - Regression: continuous-valued output
  - Classification: predict discrete-valued output
- Shallow learning: Few non-linear transformations → small hypothesis space
  - LogReg, SVM, Decision Trees, RF, KNN
- Deep Learning: Multiple successive non-linear transformations → **larger hypothesis space**
  - Machine learning with neural networks



Anomalous Energy Consumption Detection: A Supervised Binary Classification Problem

HOW?

- Features X: daily kWh values
  - Of m consumers over n days
  - x_{ij} = consumption of i^{th} consumer on j^{th} day
  - $i \in [0, m]$ and $j \in [0, n]$, $x_i \in \mathbb{R}^n$
- Labels Y: Thief (1) or Regular Consumer (0)
  - 2 classes – binary classification
- ML model: learn transformation $X \rightarrow Y$
  - Using $(x_0, y_0), (x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)$
  - Iteratively update weights θ for $f(\theta, X) \rightarrow Y$

WHY?

- Operational Efficiency
  - Minimizes need for manual meter inspections.
  - Actionable identification of theft from kWh consumption data → fewer false flags.
  - Scalable – training/prediction can be automated.
  - Online Learning – performance ↑ w/experience
- Financial Benefits
  - Eliminates cost of manual meter inspections.
  - Help minimize NTL losses.
  - Adds business value to unmonetized AMI data.

Project Objectives

FEASIBILITY ASSESSMENT

- How viable is ML-based detection of anomalous energy consumption?
- How useful is the data?
  - kWh timeseries enough for model to ID thieves?
- ML classifiers vs Non-ML benchmark
  - Non-ML classifier: always predicts modal class
- Can models be built with ML toolkits such as `scikit-learn`, `keras`, or `tensorflow`?

IDENTIFYING THE RIGHT ML ALGORITHM

- Identifying best classifier
  - Based on evaluation score on unseen data.
- Shallow Learning vs Deep Learning
  - DL > ML? Why or why not?
  - If comparable, what incentive for DL models?
- Complexity – does the model scale well?
  - m – Number of samples (consumers)
  - n – Number of features (kWhs)

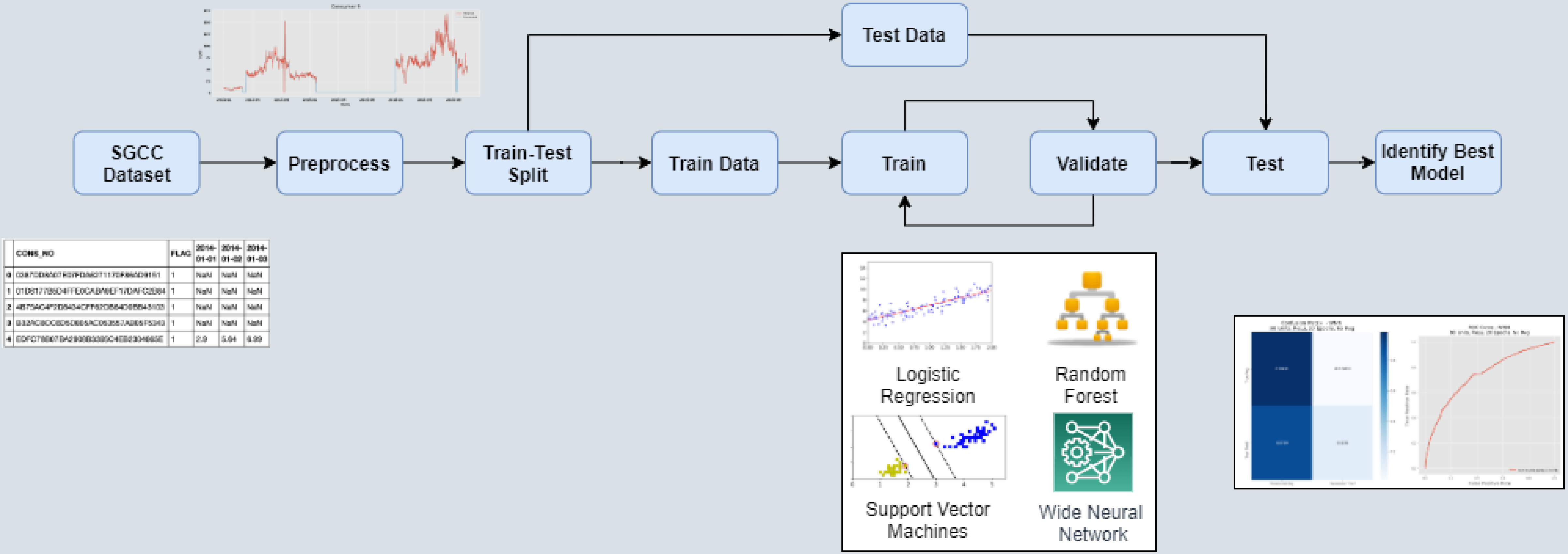
Project Benchmark: Zheng et. al [1]

- Zheng et. al's publication: popular work in AI-based electricity theft detection – 25 IEEE citations.
- Premise: periodicity in kWh consumption of regular consumers – aperiodic in thieves.
- Compared and contrasted performance of shallow and deep learning models.
- Has explored feature scaling, hyperparameter tuning.
- Our goal: verify and/or improve these results.

Learning Approach	Learning Algorithm	Classification Score (AUC)
Shallow Learning	Logistic Regression	0.7060
	Support Vector Machine	0.7413
	Random Forest	0.7385
Deep Learning	Wide Neural Network	0.6965
	Convolutional Neural Network	0.7797

Summary of results achieved by Zheng et. al on the SGCC dataset. Classification score is the area under the receiver operating characteristic curve (ROC curve AUC) for a training ratio of 80%.

[1] Z. Zheng, Y. Yang, X. Niu, H.-N. Dai, and Y. Zhou, "Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids," *IEEE Transactions on Industrial Informatics,* vol. 14, no. 4, pp. 1606-1615, 2017.



Methodology

Original Data → Preprocessing → Train-Test Split → Training & Validation → Test Set Evaluation → Identify Best Model

Tools and Libraries Used

DATA PREPROCESSING AND VISUALIZATION

pandas

matplotlib

NumPy

MACHINE LEARNING

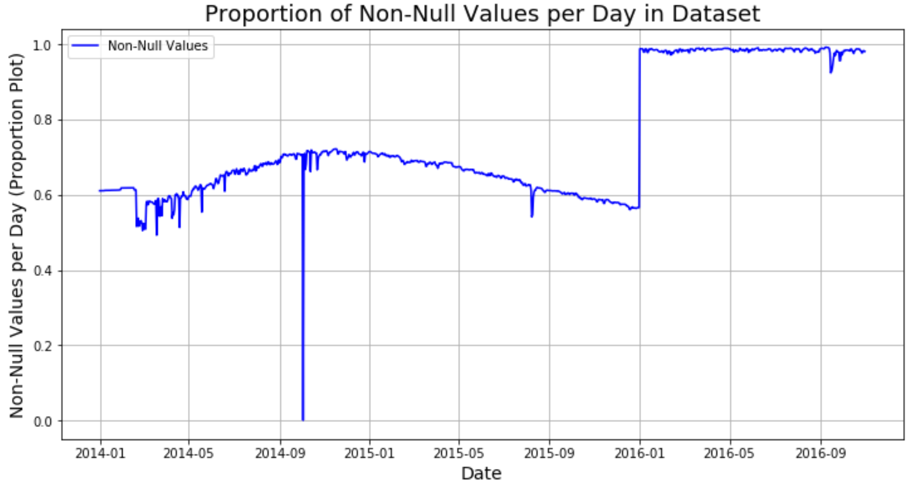
colab

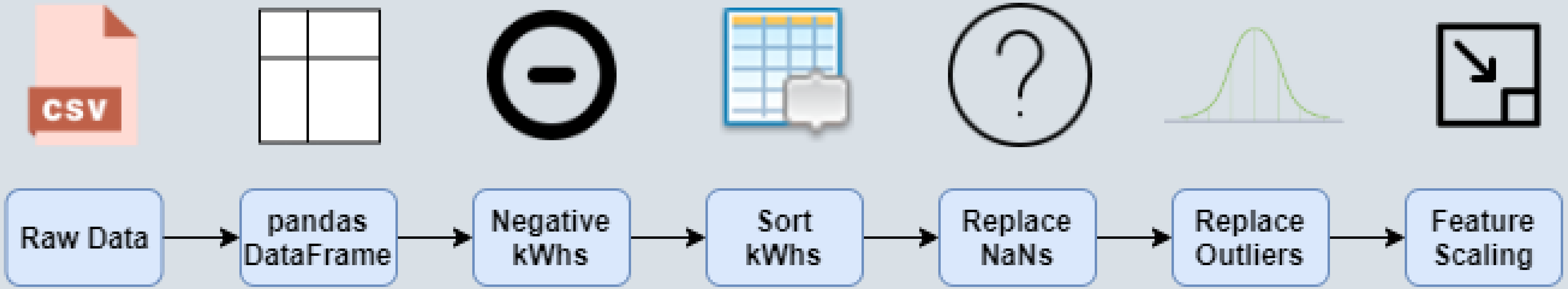
scikit learn

Keras

SGCC Dataset

- SGCC – State Grid Corporation of China
- Publicly available, labeled dataset of daily kWhs of residential consumers from 1/2014 – 12/2016
- From an ML perspective
  - m – Number of consumers: 42,372
  - n – Number of features: 1,034
  - $X = [x_0, x_1, x_2, \dots, x_i],\ i \in [0, m-1], x^i \in \mathbb{R}^n$
  - $y \in [0, 1]$ – [Regular consumer, thief]
- Problems in the dataset
  - Imbalanced – 91.5% - 8.5% split between [0, 1]
  - Unsorted – kWhs not sorted chronologically
  - Outliers – instances of $x_{ij} \sim 10^4$ kWh!
  - Missing values – 30 – 40% kWh missing before 2016



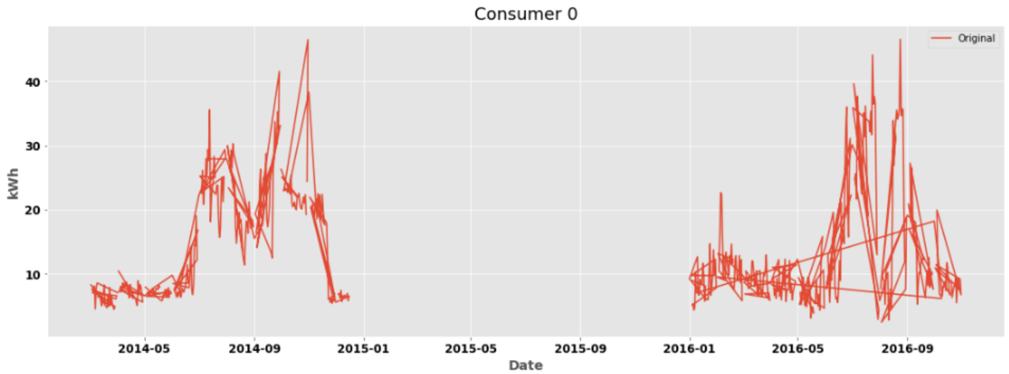


Preprocessing

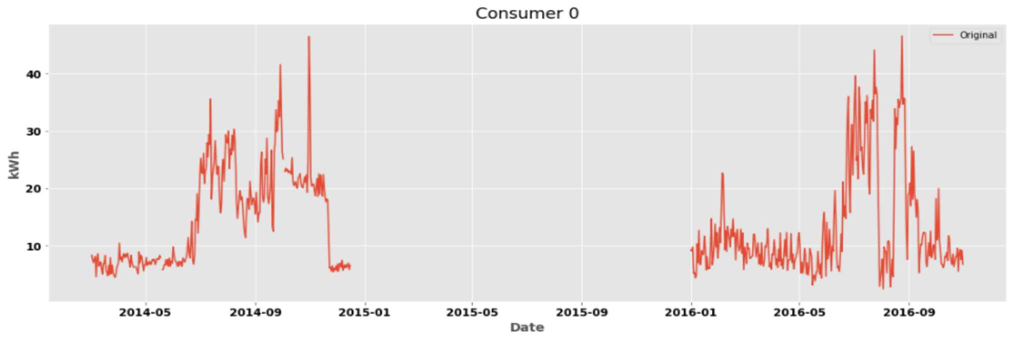
Raw data is read from a CSV file into a `pandas DataFrame`.

Preprocessing: Negative kWhs, Sorting

- Negative kWhs
  - kWhs always ≥ 0. If $< 0 \rightarrow$ erroneous.
  - Using `numpy` and `pandas` API, confirmed no negative kWhs in the dataset.
- Sorting kWhs Chronologically
  - Column headers sorted lexicographically.
  - Dates as strings: 1/1/14, 10/1/14, 11/1/14
  - Converted headers to `DatetimeIndex`.
  - Sorted with call to `sort_values`.
  - Now chronological



kWhs for Consumer 0 – Before Sorting

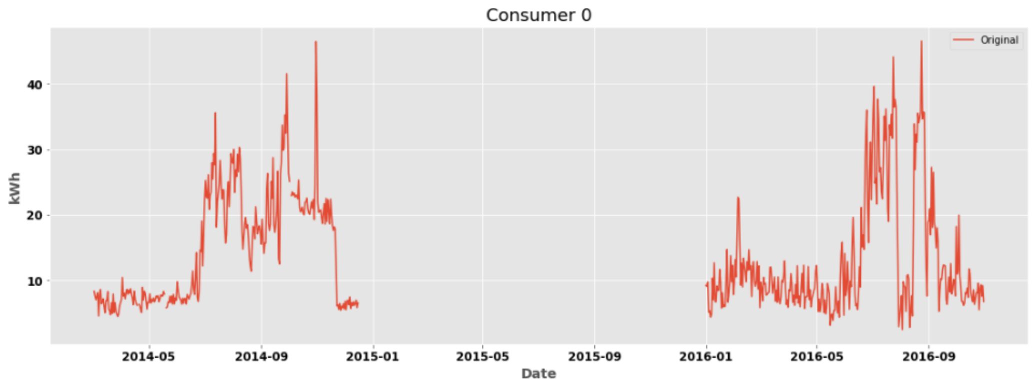


kWhs for Consumer 0 – After Sorting

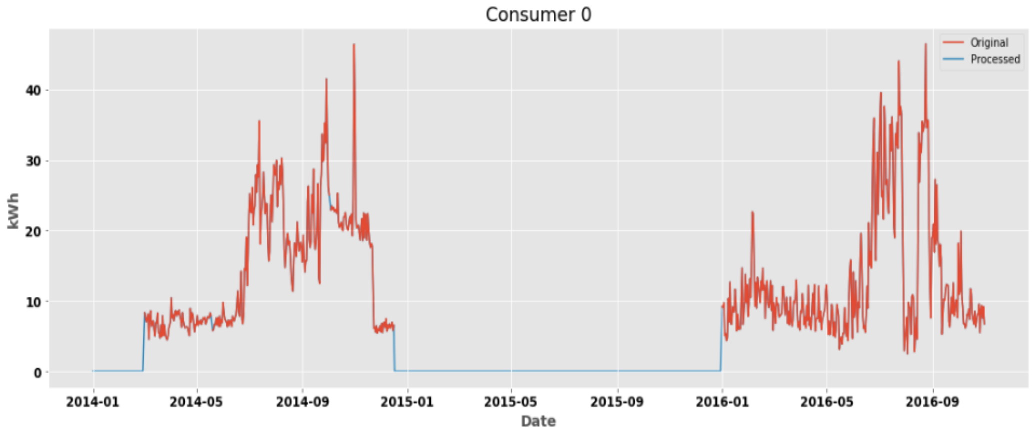
Preprocessing: Missing Values

- Assume x_{ij} is the consumption of the i^{th} consumer on the j^{th} day.
- If x_{ij} is undefined, represented by NaN, and replaced using the following rules.
  - x_{ij} is mean of previous and next days' kWhs <u>if both values are known</u>
  - x_{ij} is 0.0 if either previous or next value is NaN
- Concretely, if $x_{ij} \in NaN, x_{ij}$

$$= \begin{cases} \dfrac{x_{ij-1} + x_{ij+1}}{2}, & x_{ij-1}\ AND\ x_{ij+1} \in \mathbb{R} \\ 0.0\,, & x_{ij-1}\ \ OR\ x_{ij+1} \in NaN \end{cases}$$



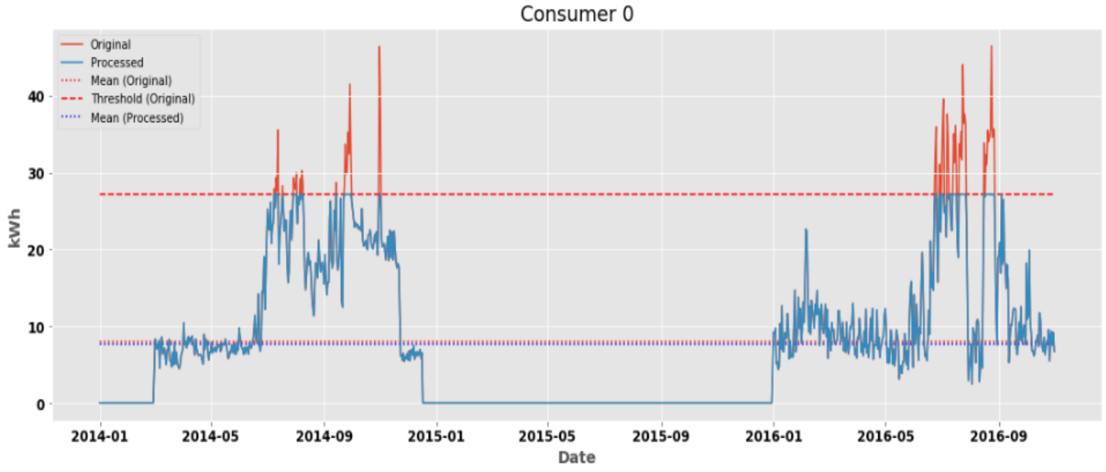
kWh for Consumer 0 – Before Replacing Missing Values



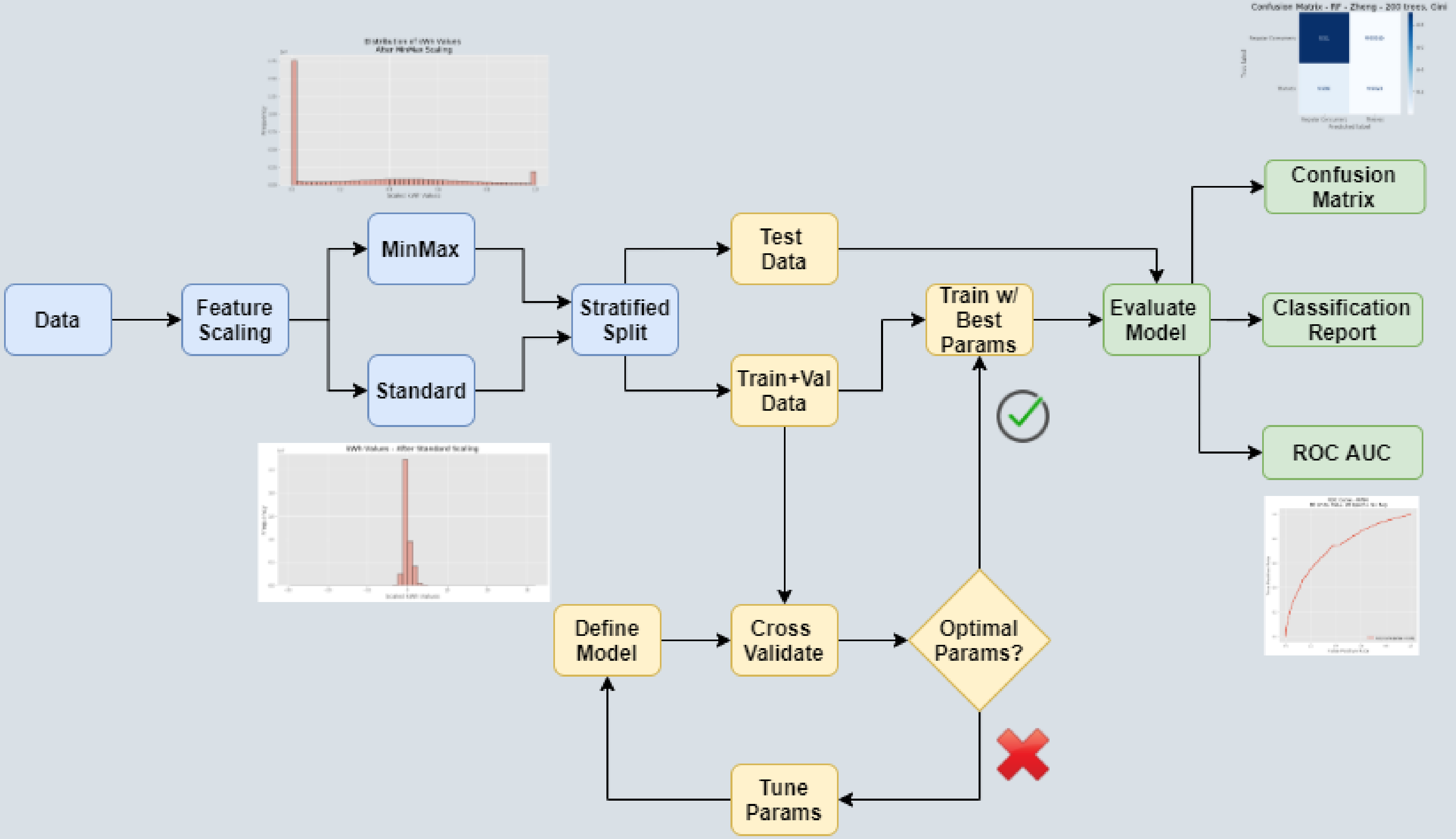
kWh for Consumer 0 – After Replacing Missing Values

Preprocessing: Replacing Outliers

- Outliers – kWh values which differ significantly from all other observations.
- Why replace? Models sensitive to outliers
  - SVMs, WNN - will not generalize well.
- Single consumer kWhs – normal distribution.
- Based on variant of three-sigma rule
  x_{ij} is an outlier if $x_{ij} > \mu_i + 2 \times \sigma_i$
  - x_{ij} – kWh i^{th} consumer's kWh on j^{th} day
  - μ_i– mean of all kWh values for consumer i.
  - σ_i – standard deviation of x_i's kWhs
- All outliers capped at $\mu_i + 2 \times \sigma_i$



kWhs for Consumer 0 – After Replacing Outliers



Training, Testing, and Validation

Each model is trained and tested using data scaled with both Min-Max normalization and Standardization. Where possible, models have been cross validated to find optimal hyperparameters, and then evaluated on a test set in terms of ROC AUC.

Training, Testing, and Validation

MACHINE LEARNING MODELS

- Logistic Regression
- Support Vector Machine
  - Large margin classifier.
  - Aims to fit the 'widest street' b/w classes.
- Random Forest
  - Ensemble classifier – 200 decision trees.
- Wide Neural Network
  - Single layer of 1034 ReLu activation units.
  - Single sigmoid output unit.
  - Memorization of features correlated with target.

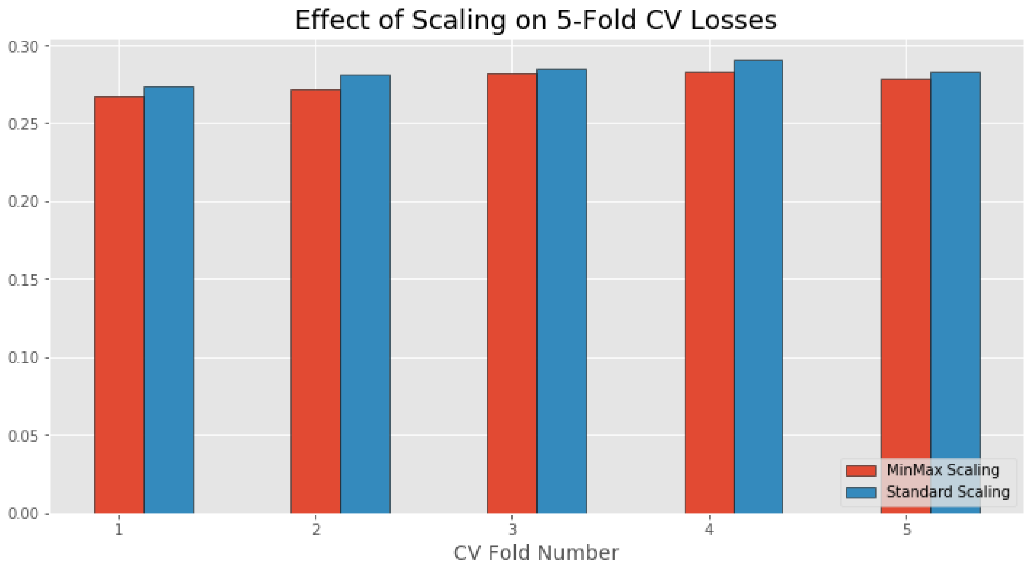
EVALUATION METRICS

- Loss Function – Binary Crossentropy
  - Loss: $f(\widehat{y_i} - y_i)$ - measures of difference between predicted and actual outcomes.
  - Goal of any ML model: minimise loss.
  - Optimal binary crossentropy → 0
- Scoring Function: ROC AUC
  - Area under receiver operating characteristic curve.
  - All models: $P(y = 1|x_i) > \epsilon \, ? \, \widehat{y_i} = 1 : \widehat{y_i} = 0$
  - ROC curve – TPR against FPR for different ϵ.
  - Goal – $\forall\ \epsilon$, TPR ↑, FPR ↓ ⇒ AUC → 1

Validation Methods Visualized

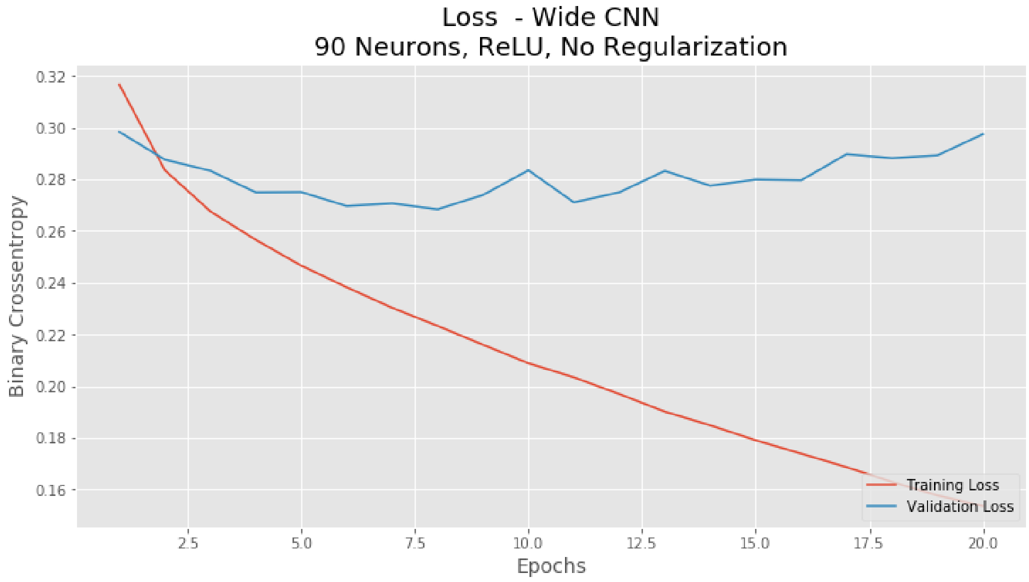
SHALLOW LEARNING MODELS:

CROSS VALIDATION SCORES



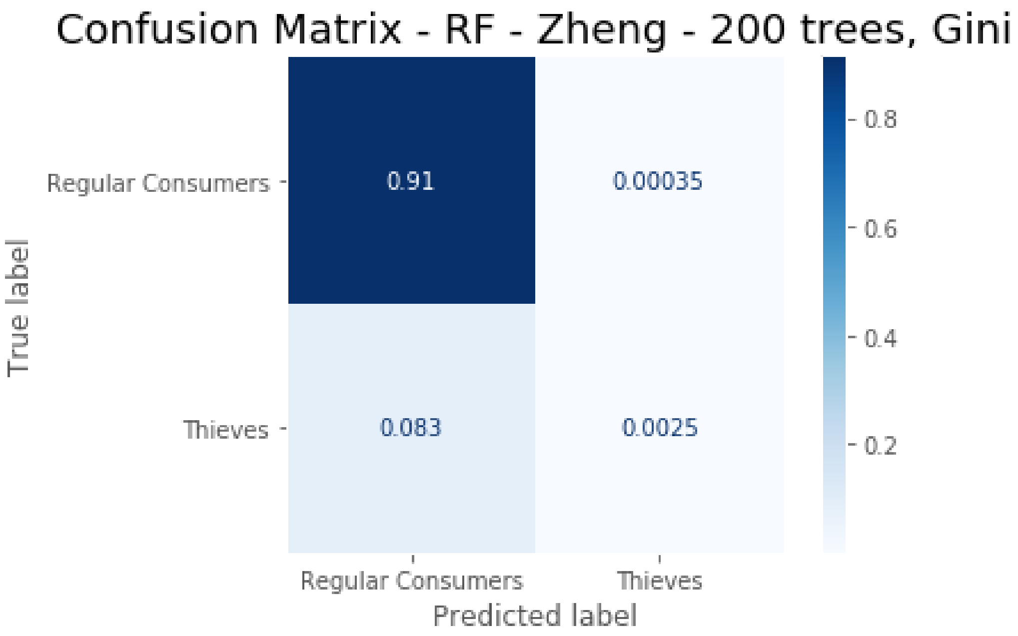
WIDE NEURAL NETWORK

TRAINING HISTORY

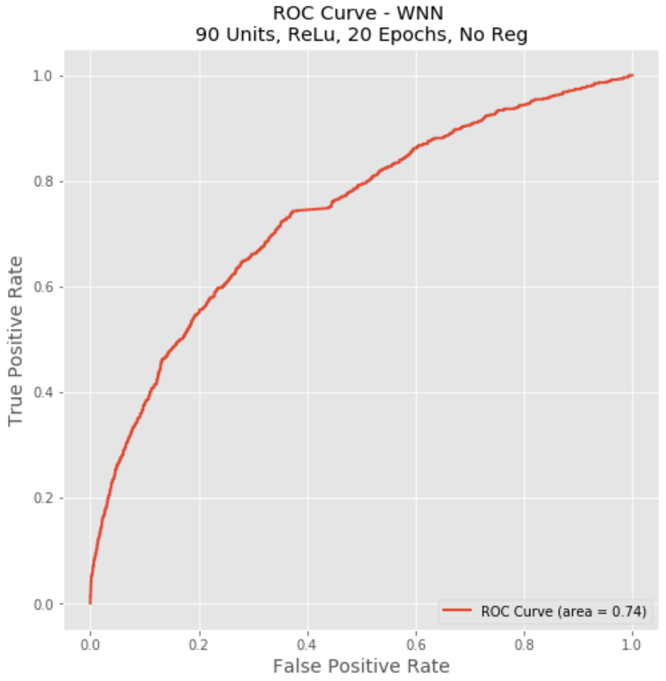


Evaluation Methods Visualized

CONFUSION MATRIX

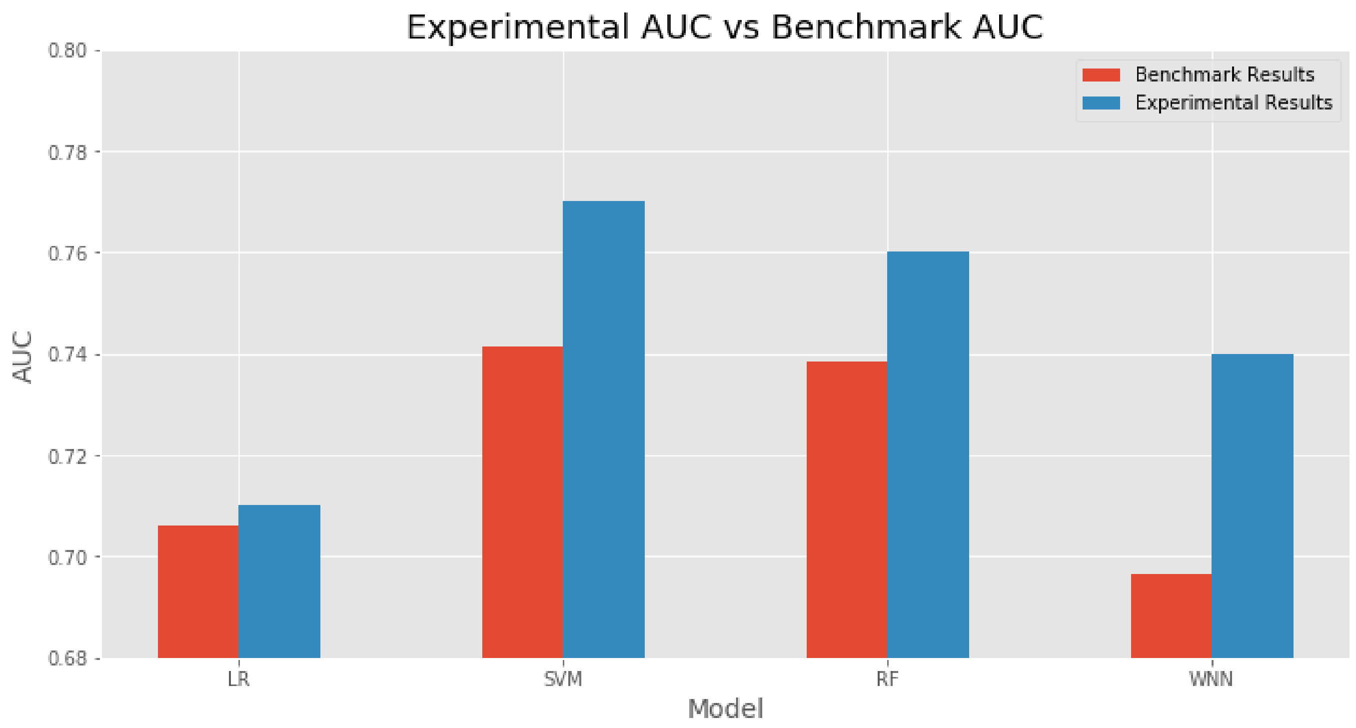


ROC CURVE



Preliminary Results

- Experimental AUC > benchmark AUC for all models.
- Max %age improvement over benchmark in WNN.
- Highest AUC achieved with SVM trained on standardized data – but poor scalability.
- RF & WNN next best alternatives – comparable AUC, complexity ↓
- Standardisation > normalization (exception: RF)
- ML models > non-ML classifier



Summary of preliminary results. All models have been trained with a training ratio of 80% using a stratified train-test split.

Preliminary Results

- Experimental AUC > benchmark AUC for all models.
- %age improvement over benchmark max in WNN.
- Highest AUC achieved with SVM trained on standardized data – but poor scalability.
- RF & WNN next best alternatives – comparable AUC, complexity ↓
- Standardisation > normalization (exception: RF)
- ML models > non-ML classifier

	Benchmark		Experimentation Results		Analysis
Model	**Feature Scaling**	**AUC**	**Feature Scaling**	**AUC**	**Improvement**
Non-ML Classifier	NA	NA	NA	0.50	NA
Logistic Regression	Min-Max	0.706	Standard	0.71	0.57 %
SVM	Min-Max	0.7413	Standard	0.77	3.87 %
Random Forest	Min-Max	0.7385	Min-Max	0.76	2.91 %
Wide Neural Network	Min-Max	0.6965	Standard	0.74	6.24 %

Summary of preliminary results. All models have been trained with a training ratio of 80% using a stratified train-test split.

Conclusions

- ML-based approaches for anomalous energy consumption detection are viable.
  - ML models worthwhile – Drastic improvements over non-ML classifier.
  - Data is relevant – models have been able to identify thieves based solely on kWh patterns.
  - Development Feasible – all models can and have successfully been built using open-source toolkits.
- Shallow and Deep Learning models offer comparable performance.
  - Best model: Random Forest Classifier, with WNN as next best choice.
  - Disregard SVM ∵ good performance, but poor complexity → Will not scale IRL
  - RF vs WNN – unfair comparison: – ensemble learning: 200 decision trees > 1 (barely) deep learning classifier.
  - WNN can be optimized further – signs of overfitting, not mitigated by Dropout, L1, or L2 regularization.
  - Feature Engineering – DNN will not require manual feature engineering – bottleneck for RF.
- Deep Learning models worth investigating further.

Expected Outcomes & Future Work

EXPECTED OUTCOMES

- Convolutional Neural Network
  - Operates on 2D (weekly) kWh data.
  - Feature extraction → periodicity and more.
  - Generalization ↑, but memorization ↓.
- Wide and Deep Convolutional Neural Network
  - Multimodal input neural network
  - WNN and CNN trained in parallel as single model
  - Generalization ↑, Memorization ↑
- Hyperparameter Tuning
  - Of both shallow and deep learning models

FUTURE WORK

- Periodicity & Manual Feature Engineering
  - Use consumer autocorrelation vector as feature.
  - Confirm Zheng's assumption about periodicity.
- Ensemble Learning
  - Mix and match shallow learning models to make predictions on same data together.
  - Voting classifiers → possible improvement
- Outlier and Novelty Detection
  - Semi-supervised or unsupervised learning.

Thank You