Semi-supervised monitoring of electric load time series for unusual patterns

2011 International Joint Conference on Neural Networks
San Jose, California    July 31 - August 5, 2011

Nikolaos Kourentzes
Sven F. Crone
Lancaster Centre for Forecasting
Lancaster University Management School – Department of Management Science
I. Electricity Load Outliers
   I. Issues & Algorithms

II. Proposed Semi-Supervised Algorithm
   I. Constructing initial labels
   II. Self-training
   III. Final classification

III. Empirical evaluation: UK E-Load outlier detection
Outlying daily patterns

- Functional outliers
- Outliers in level & shape
- Predictable reasons: Bank Holidays
- Unpredictable: Natural disasters, Industrial actions, etc
Accurate E-Load forecasting required for many applications

A lot of progress in forecasting, BUT not in monitoring and outlier identification

- Forecasting models require data cleaning (outliers)
- Outliers → problem for model estimation, evaluation, etc
- Massive high frequency dataset → Manual identification costly
- Need automatic approach!

Traditional outlier identification does not work

- High frequency data
- Functional outliers
- High cost of manual exploration → Conventional algorithms require knowledge of past data.
Supervised Learning

- Parameterise model based on minimising a cost function on a training sample → Conventional model building
- Requires labelled cases → What is a normal/outlier day?
- Prior labelling of training set → Costly
- E.g. Classification neural networks, discriminant analysis, etc

Unsupervised Learning

- Discover information with no prior knowledge, based on data characteristics.
- Labelling of data is not required → Algorithm discovers labels
- Not use past available information → Loss of accuracy
- E.g. K-means, Self Organising Maps, etc

Semi-supervised Learning

- Use limited labelled cases, self-label remaining unlabelled cases
- Between supervised & unsupervised
- Useful when labelling is costly & unsupervised not accurate
SSL Outlier Detection
Proposed Framework

1. Construct initial (small) set of labelled cases

2. Use self-training neural networks to label remaining dataset

3. Stop self-training once confidence in classification cannot be increased
SSL Outlier Detection

Proposed Framework

1. Construct initial (small) set of labelled cases
   - Identify a small number of outliers \((n<10)\) \(\rightarrow\) Automated by focusing on bank holidays for a short period of time.
   - Use heuristic to identify a large number of normal days.

   i. Low pass filter to remove trend/low frequency seasonalities (annual)
   ii. Separate time series into days:
      
   iii. Kernel density estimation for each hour \(\rightarrow\) connect modes = normal profile

   iv. Correlation between profile and observed days
   v. Rank according to similarity and pick top \(\phi S_L\)
      \[1 < \phi < (S - S_L)/S_L\]
      \(S\) number of days
      \(S_L\) number of labelled (outlier) days
   vi. Initial set of normal and outlying days created
2. Use self-training neural networks to label remaining dataset

SSL Outlier Detection

Self Training

Supervised Learning

Labelled data

Unlabelled data

Neural Network

Labelled output

High confidence labelling

Prediction

Supervised Learning

New Labelled data

Unlabelled data

Neural Network

Labelled output

High confidence labelling

Prediction
2. Use self-training neural networks to label remaining dataset

- Labelled set $X_L$ with labels $C_L$; Unlabelled set $X_U$
- Normal day = [0 1], Outlier = [1 0]
- Train classifier on $X_L$ with targets $C_L$ and predict $X_U$

Define classification confidence as:

$$\psi_i = \min \left\{ |c'_{L_{i1}} - 1| + |c'_{L_{i2}}|, |c'_{L_{i1}}| + |c'_{L_{i2}} - 1| \right\}$$

- Minimum when close to either [0 1] or [1 0]
- Use all labelled cases with $\psi_i < P_{\psi_i}$ (set percentile) as new training set and repeat classification
- Note: At each iteration samples are allowed to leave the training set
3. Stop self-training once confidence in classification cannot be increased
   - At each iteration learn on new training set
   - Converge when no new samples can be labelled with confidence (include in training set)
   - Classify remaining samples
UK E-Load hourly for years: 2001-2008 → 2771 days, 63 true outliers

Provide only 7 outliers from year 2001

Evaluate semi-supervised learning outlier detection ($\phi = 3$) against:

- Supervised: MLP classifier
- Unsupervised: K-means time series clustering

Criteria:

- AUC: Area Under the Curve → Superior to accuracy
- Outlier Rate: $OR = \frac{TP}{TP + FP + FN}$
  - Correctly identified outliers over labelled outliers & missed outliers
  - Metric focused on outliers (due to sample size)
Semi-supervised is more robust to network parameter selection
With $P_\psi$ lower than 100% (=no diffidence evaluation) algorithm is robust
Allowing for re-assessment of labelling confidence has minimal impact
Proposed semi-supervised algorithm identified outliers accurately and more robust than supervised and unsupervised alternatives.

Robustness is crucial → In practice absence of large labelled sets to evaluate performance → Need to be robust to design parameters.

Semi-supervised algorithm robust to MLP size, self-training threshold and scheme.

Proposed heuristic reduces substantially need for costly labelled cases.
Thank you for your attention!

Questions?

Nikolaos Kourentzes
Lancaster University Management School
Centre for Forecasting
Lancaster, LA1 4YX, UK
Tel. +44 (0) 7960271368
email n.kourentzes@lancaster.ac.uk