Effects of Data Cleansing on Load Prediction Algorithms

Boye A. Høverstad, Axel Tidemann and Helge Langseth

Norwegian University of Science and Technology

2013 IEEE Symposium on Computational Intelligence Applications in Smart Grid
Singapore 2013-04-16
Motivation

● Advanced metering infrastructure is being rolled out in many countries worldwide
● Huge influx of data
● Errors can be introduced from faulty sensors or during transmission from the reader to the database
● The paper focuses on data cleansing as a pre-processing step for short-term (24 hour) load prediction
Application: load prediction

- Short-term load prediction is important for market operators, as well as for efficient grid operation.
- Load prediction on transmission level has been studied before, we want to look at distribution substation and single households as well.
- Data cleansing is performed in order to enhance load prediction.
Parameterization of models

- Both data cleansing and model training needs to be parameterized
- This requires experience and intuition
- Model parameters are often not valid across datasets
An evolutionary approach

● We use evolution to find the best parameters for both data cleansing and model training
● The genome codes parameters for both, so the evolution takes into account the interdependency between the two

![Diagram of evolutionary approach]
Dataset granularity

- In order to validate our approach, we perform load prediction on three levels of data granularity: transmission, distribution substation (150 users), single household
- Load signal characteristics vary greatly between these three levels
- The variation in signal characteristics is another reason to automate the parameter setting, since one set of parameters is unlikely to be optimal for a different dataset
Dataset

- Single household readings are from an earlier research project conducted at SINTEF Energy Research
- We select 150 users from this dataset to represent an urban distribution substation
- Transmission level is from British Columbia, available online
- For the SINTEF data, we also have local temperature readings which are used by the models
Examples of load profiles: single household

- Very irregular and noisy signal
Examples of load profiles: distribution substation (150 users)

- Daily peaks are evident, some irregularities
Examples of load profiles: transmission level

- Smooth signal with regular peaks
Data cleansing

- We use the B-spline approach of Chen et al. (2010) to clean temperature and load data
- Observed data points are approximated by a smooth curve (i.e. spline)
- Data points outside a defined confidence interval are outliers
- The outliers are replaced with the upper or lower bounds of the confidence interval
Data cleansing example
Data cleansing

- Load cleansing frameworks tend to assume normally and independently distributed noise.
- We see that there is a strong diurnal pattern in power load signals.
- The gradient leading up to the peaks are sharper than the rest of the day, leading to higher error by non-parametric regression.
- As a result, peak hour loads are often falsely identified as outliers.
- Identifying and predicting peak load is very important for safe and efficient grid operation.
Subtracting the daily average

- To overcome this problem, we transform the signal prior to spline regression
- First, the historical average load of each calendar day is subtracted for a given day, yielding a day profile with zero mean
- Second, a "prototypical" day is estimated by calculating a 24 hour average (daily) and a 168 hour average (weekly) for the training data, which is subtracted from the shifted signal in the previous step
- Reversed before model training
Prediction models

- Autoregressive model
- Indexed autoregressive model (uses load data only for the previous hour/day/week)
- Echo State Network
- Wavelet prediction, using the "à trous" Haar transform and linear regression
Genetic algorithm

- Searches the parameter space for each model
- 100 individuals for 100 generations
- 30 evolutionary runs for each dataset for each predictor
- Fitness is $1/(1 + \text{RMSE})$
Validation

- After evolution, the evolved genomes are validated on a 6 month dataset to eliminate over-trained models
- The best genomes were examined on a test set
Results: prediction

- 10-day prediction period on transmission level load, best model based on validation period
Results: prediction error

- $R =$ Raw (no cleansing), $O =$ Original, $D =$ Daily, $W =$ Weekly
- Notice the spread of the ESN
Results: best performers (RMSE)

- In 9 out of 12 cases, our proposed method of cleansing improves upon predictions with cleansing as proposed by Chen et al. (2010)

<table>
<thead>
<tr>
<th></th>
<th>Single user</th>
<th></th>
<th>Distribution substation</th>
<th></th>
<th>Transmission</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Raw</td>
<td>Original</td>
<td>Day</td>
<td>Week</td>
<td>Raw</td>
</tr>
<tr>
<td>ESN</td>
<td>0.308</td>
<td>0.111</td>
<td>0.110</td>
<td>0.113</td>
<td>20.72</td>
</tr>
<tr>
<td>Wavelet</td>
<td>0.083</td>
<td>0.082</td>
<td>0.084</td>
<td>0.084</td>
<td>22.34</td>
</tr>
<tr>
<td>AR</td>
<td>0.087</td>
<td>0.087</td>
<td>0.084</td>
<td>0.085</td>
<td>21.43</td>
</tr>
<tr>
<td>AR Index</td>
<td>0.083</td>
<td>0.083</td>
<td>0.083</td>
<td>0.078</td>
<td>21.68</td>
</tr>
</tbody>
</table>
Results: subtracting daily average

- Top figure shows the original algorithm, bottom figure shows our method
- Notice how the top figure tends to fill troughs and shave peaks, which is undesirable
Results: subtracting daily average

- Cleansing frameworks assume noise is normally and independently distributed
- Top figure shows error distribution (raw signal - spline) for original method, bottom figure shows our method
Conclusion

- Data cleansing and prediction should be considered as a combined endeavour, since the two influence each other.
- We use an evolutionary approach to parameterize load cleansing and prediction.
- Typical load patterns violate the assumption of normally and independently distributed noise; we propose a simple remedy.
- This method improves the performance of load prediction algorithms.
Questions?
<table>
<thead>
<tr>
<th>Model</th>
<th>Gene</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data cleansing</td>
<td>Load smoothing</td>
<td>0.001 - 1000 (log)</td>
</tr>
<tr>
<td>Data cleansing</td>
<td>Load Z-score</td>
<td>0.1 - 3</td>
</tr>
<tr>
<td>Data cleansing</td>
<td>Temperature smoothing</td>
<td>0.001 - 1000 (log)</td>
</tr>
<tr>
<td>Data cleansing</td>
<td>Temperature Z-score</td>
<td>0.1 - 3</td>
</tr>
<tr>
<td>ESN</td>
<td>Network size</td>
<td>10 - 500</td>
</tr>
<tr>
<td>ESN</td>
<td>Leak rate</td>
<td>0 - 2</td>
</tr>
<tr>
<td>ESN</td>
<td>Input scaling</td>
<td>0.1 - 0.75</td>
</tr>
<tr>
<td>ESN</td>
<td>Bias scaling</td>
<td>0 - 1</td>
</tr>
<tr>
<td>ESN</td>
<td>Spectral radius</td>
<td>0.5 - 2</td>
</tr>
<tr>
<td>AR</td>
<td>Order</td>
<td>100-200</td>
</tr>
<tr>
<td>AR Index</td>
<td>Indices</td>
<td>1, 24, 168</td>
</tr>
<tr>
<td>Wavelet</td>
<td>Scale</td>
<td>1-10</td>
</tr>
<tr>
<td>Dataset</td>
<td>Hindsight (hours)</td>
<td>24, 72, 168, 336, 672</td>
</tr>
<tr>
<td>System</td>
<td>Seed</td>
<td>5 random numbers</td>
</tr>
</tbody>
</table>