International Journal of Forecasting 30 (2014) 369–374

Contents lists available at ScienceDirect

International Journal of Forecasting

journal homepage: www.elsevier.com/locate/ijforecast

GEFCom2012 hierarchical load forecasting: Gradient boosting machines and Gaussian processes

James Robert Lloyd

Department of Engineering, University of Cambridge, United Kingdom

ARTICLE INFO

Keywords:
Load forecasting
Gradient boosting machines
Gaussian processes

ABSTRACT

This report discusses methods for forecasting hourly loads of a US utility as part of the load forecasting track of the Global Energy Forecasting Competition 2012 hosted on Kaggle. The methods described (gradient boosting machines and Gaussian processes) are generic machine learning/regression algorithms, and few domain-specific adjustments were made. Despite this, the algorithms were able to produce highly competitive predictions, which can hopefully inspire more refined techniques to compete with state-of-the-art load forecasting methodologies.

1. Introduction

This report details the methods I used when competing in the load forecasting track of GEFCom2012.[1] It is split into sections describing the various techniques used for forecasting temperatures and loads. Within each section, I have given the motivation for the particular choice of algorithm, discussed the way it was used and described how to replicate the results using the spreadsheets and scripts accompanying this report.[2]

I approached the competition in the spirit of a data mining competition. Consequently, some of the choices detailed below were based on intuition, in order to save time and focus on those aspects of the data which were most likely to give the greatest improvements in performance. Many of these choices could have been replaced by appropriate searches and cross validation; where more complex techniques would be required, I have briefly described how one could perform a more objective analysis.

My methodology for load backcasting/forecasting was to try different general purpose regression algorithms and then combine (average) the predictions. The final ensemble forecast comprised predictions from a gradient boosting machine (GBM), a Gaussian process (GP) regression, and the benchmark solution provided by the competition organisers (a linear model).

This paper is organised as follows. Section 2 discusses data cleansing, Section 3 is on temperature forecasting, Sections 4 and 5 introduce the GBM and GP forecasting methodologies respectively, and Section 6 discusses the way in which the final ensemble prediction was formed. A brief introduction to Gaussian processes is provided in the Appendix.

2. Data cleansing

A sensible first step in any prediction task is to look at your data, and in particular, to search for anomalies. This was performed for both the temperature and load data using the spreadsheets `temp/temp.xlsx` and `load/load.xlsx`, by plotting the data as time series and using various types of conditional formatting to do a visual search for irregularities.

This visual analysis revealed a large discontinuity in load series 10 and atypical dynamics in series 9 (see Fig. 1, and others for comparison). No other large anomalies were detected during the initial inspection of the data, and the smaller irregularities were not revisited because the

E-mail address: jrl44@cam.ac.uk.

1 http://www.gefcom.org/.

2 The source code is available at https://github.com/jamesrobertlloyd/GEFCOM2012/.

0169-2070/$ – see front matter © 2013 International Institute of Forecasters. Published by Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ijforecast.2013.07.002

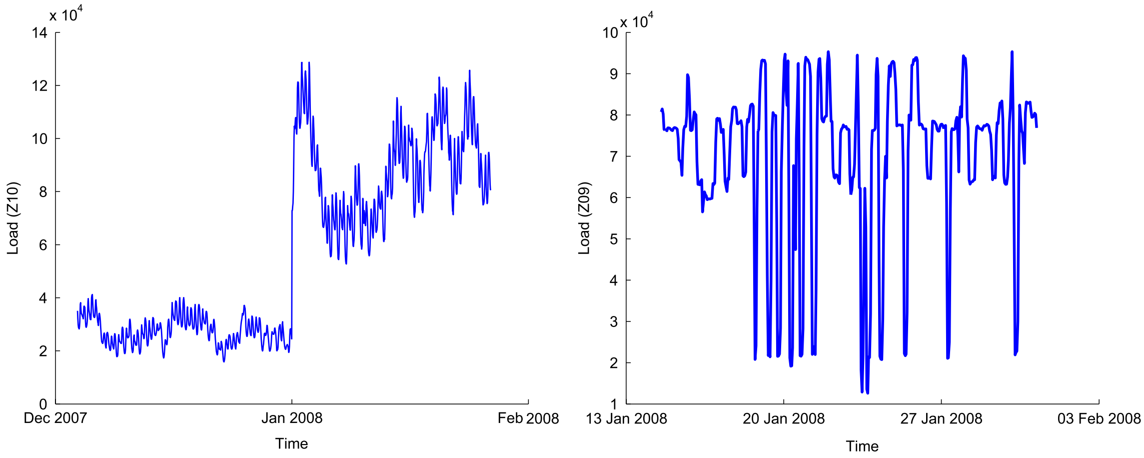

Fig. 1. Left: Raw data at zone 10, showing a large discontinuity. Right: Raw data at zone 9, showing atypical load dynamics.

algorithms detailed below performed acceptably well without requiring further data cleansing.

No adjustments were made for holidays or other irregular events such as black-outs. Ignoring them was a practical response to time constraints, rather than a design choice.

3. Temperature forecasting

3.1. Initial analysis and remarks

The error metric used in GEFCom2012 was heavily weighted towards times at which the temperatures were unknown (i.e., the load forecast rather than backcasts). Consequently, good temperature predictions seemed to be crucial for overall success.

When submitting a solution to Kaggle, the error metric was computed on 25% of the held out data and returned to the user. This allowed the user to optimise their temperature predictions by optimising the score of the resulting load predictions (i.e., computing load predictions based on different temperature predictions and selecting the temperature prediction with the highest corresponding load prediction score). Depending on the way in which the test dataset was split (the 25% test and 75% validation partition), this may have allowed users to come very close to knowing the true future temperatures.

I therefore used a flexible but simple method for forecasting temperatures that could be easily tuned. Fig. 2 shows data from temperature station 1 along with various curves which were used for prediction (described later). The black solid line is the raw data, which shows that temperatures follow a smooth trend, with a daily pattern of rising and falling temperatures. For the sake of simplicity, I modelled the smooth trend and the daily periodicity separately, and modelled each temperature station in isolation.

3.2. Methodology

The smooth trend was estimated within the data using a local linear regression (e.g., Hastie, Tibshirani, & Friedman, 2009, Chapter 6) with a bandwidth of one day. I assumed that the smooth trend would probably return to the historical average smoothly. A suitable modelling technique that would give this type of prediction is Gaussian process regression (e.g., Rasmussen & Williams, 2006, or see the Appendix to this paper for a very brief introduction); the difference between the smoothed temperature and its historical average was regressed against time using a squared exponential kernel and zero mean function.

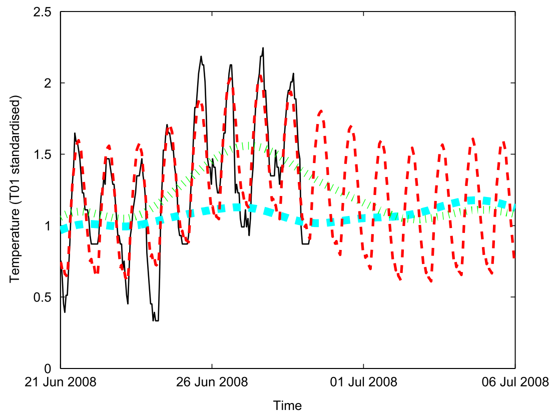

Fig. 2. Raw data at temperature station 1 (after removing the mean and scaling to have a unit standard deviation) (black solid line); the historic average of smoothed temperatures (blue dashed thick line); current and predicted smoothed temperatures (green dotted thick line); and final temperature predictions (red dashed line). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

In Fig. 2, the thick blue dashed curve shows the historical average of smoothed temperatures, while the thick green dotted line shows the current smoothed temperature and prediction. The parameters of the GP model (length scale and scale factor of the squared exponential kernel) were tuned by hand, by initially choosing sensible values based on plots of the data, and then trying to optimise the public test score of the load predictions thus derived.

The difference between the temperature and the smoothed temperature was assumed to be a smooth periodic function with a period of one day. This modelling