U.K.'s National Grid Company - Report on Work for the Eastern Slovakia Demand Forecasting Competition

FOREWORD

This Foreword text will not appear in the published report, it is merely here to orientate the unfamiliar reader, seeing this out of context.

To National Grid:

We have recently come to the end of the formal project on demand forecasting research (MA050), in which some promising results were achieved for normal days. Subsequently, we became aware of an international demand forecasting competition. Following consultation with the client, who offered support in principle but not funding, it was decided to enter this competition. Funding came from "Consultancy". One of the concrete benefits envisaged from this was a demonstration of the flexibility of the new experimental technique, hopefully having applied it to a different country's demand pattern within a relatively short time.

This document is intended as a Report as part of the Results package, as requested by the competition organizers. Details of the competition are at the website http://neuron.tuke.sk/competition/index.php (possibly the "index.php" can be omitted).

To the Competition Representatives:

Any problems over the appearance, including both text and graphics, will be responded to.

WARNING: WORD FILE / EDITING BUG (& WORKAROUND)

There appears to be a slight problem with this particular Word file. Whenever text is modified, for example by selecting text and making it bold, the whole document gets affected. This can immediately be fixed since a single application of the "Undo" (control-Z) command will then leave the document with just the intended text affected (i.e. it will not undo completely). Although this is a nuisance, it appears to be a reliable work-around for the problem. The same problem exists regardless of which computer is used to edit this file. Possibly something is corrupt or out-of-step in the file.

Adaptive Logic Networks for East Slovakian Electrical Load Forecasting

David Esp
Modelling Development Group
Operations & Trading
The National Grid Company plc
St. Catherine's Lodge,
Bearwood Road, Sindlesham,
Berkshire RG41 5BN, United Kingdom
david.esp@uk.ngrid.com, www.nationalgrid.com

Abstract. The author is part of a research and development team in National Grid (UK), independent of the department responsible for routine forecasting. National Grid is owner and operator of the high voltage transmission system in England and Wales. Having recently investigated a novel technique, achieving some promising results on demand modelling for normal days in the UK, it was decided to apply the same technique to the East Slovakian data in the EUNITE competition.

The strategy taken was to generate a model from time and weather information to half-hourly electrical demand, then to execute this based on an estimate of the target period temperatures. The modelling method used was a non-parametric nonlinear modelling method called Adaptive Logic Network (ALN). This is a form of machine learning broadly similar to standard neural networks but with several advantages.

1 Introduction

The author is part of a research and development team in National Grid (UK), independent of the department responsible for routine forecasting. National Grid is owner and operator of the high voltage transmission system in England and Wales.

Having recently investigated a novel technique, achieving some promising results on demand modelling for normal days in the UK, it was decided to apply the same technique to the East Slovakian data in the EUNITE competition. One of our motivations was to find out how flexible and efficient the technique could be in accommodating another country's demand pattern. Another was to use this relatively small problem to experiment with techniques for coping with special days such as holidays.

Our modelling technique was based on the Adaptive Logic Network (ALN), a continuous value generalization [1] of a Boolean modelling technique, originated by W. Armstrong[1] in 1968 [2]. It is a form of machine learning non-parametric nonlinear modelling technique, broadly similar in capability to standard neural networks (e.g. Multiple Layer Perceptron, MLP) but offering several advantages over them.

Prior research on our own company's data had trialled the use of several kinds of neural network related technique. It was this research which identified[2] the Adaptive Logic Network (ALN) as the most effective kind. Their main advantages were found to be accuracy, consistency, controllability of training, speed of training and the analyzability of the resulting model.

Details of further advantages of ALNs and the ALN theory are available from *Dendronic Decisions Limited* of Canada, as is downloadable trial software, from their website: http://www.dendronic.com.

The main idea behind the ALN learning algorithm is least-squares fitting, starting off with plain linear regression, then progressively splitting into a multitude of separate partial linear fits (called linear forms). Combined together, these are able to approximate an arbitrary shape in data-space. One can imagine each linear form as representing a Hyperplane in (multi-dimensional) data space. The amount

[1] Armstrong's original (1968) ALNs were set up as classifiers with Boolean signals passed between elements of the networks, but modern ALNs perform regression using continuous values generated by linear functions and combined by maximum and minimum operations. The new ALNs smooth the combinations by quadratic "fillets". The continuous values support a far more effective learning algorithm. The new ALNs can also be used for classification problems by thresholding the output (0/1).
[2] At least, within the context of our experiments and data.

of splitting and hence the degree of model complexity is limited by a Tolerance parameter. When the error-of-fit for a linear form falls below this tolerance, no further splitting of it will occur. The splitting and fitting algorithms have been designed in a very general computation-efficient manner (related to alpha-beta pruning), and as a result training is much faster than for most standard neural networks. Execution is also faster, not that this was an issue of relevance to the current exercise.

2 Orientation to the Problem

2.1 Problem Description

The set problem was to predict daily peak demands for the month of January 1999.
Both maximum and average error were of concern.

The available data consisted of:

	Average Daily Temperatures	Half-Hourly Loads
1995	Y	no
1996	Y	Y
1997	Y	Y
1998	Y	Y
1999	no	no

In addition, a list of public holidays was provided. For the Jan-Feb period, the only holidays were date-fixed: Jan 1 and Jan 6. The units of the Load data were not specified.

Note that no temperature data were given for 1999. The implication of this was that if we were to develop a temperature-dependent load model, we would have to produce our own estimate of the January 1999 temperatures.

2.2 Establishing a Direction

Established literature (e.g. [3]) points out a variety of techniques for timeseries (e.g. electrical load) forecasting. These include both pure time-series and also model-based methods. For this exercise, we choose the model-based approach, since we have had some success with applying it in experiments involving our own company's historical load data.

Modelling in our case means the production of an algorithmic function to compute the demand at a particular point in time dependent on influential factors such as weather conditions. In general, algorithmic approaches typically range from pure linear model through multiple linear models (e.g. with expert selection) and parameterised models (e.g. polynomials) through to free-form non-parametric models such as neural networks.

The time available to this exercise prohibited the proposal and investigation of more than one of these alternatives. Without prior knowledge of the sorts of relationships to be found in the given data, it was decided to take the path of the free-form non-parametric model. This was seen as a "safe" option - if there was a pattern to be found, that sort of technique would find it. In addition, our prior research into load modelling for our own company's data had established a base of experience to draw upon, such as the type of algorithm and the types of input variable preprocessing. Also, one of our motivations for entering the competition had been a need to “prove” our research techniques on unfamiliar data.

Modelling Method

The chosen modelling technique, the ALN, has already been described in Section 1. As with most such techniques, the use of it implied various options to be explored or at least decided upon. For example there was input selection and transformation as well as training parameters of the ALN software. From an experimental point of view, this constituted our "search space". It is expanded on in Section 2.3.

Model Assessment

In the absence of data for the target period (January 1999) it was decided to judge the relative quality of each experimental model according to its performance against the January 1996 data, both as regards half-hourly and daily-peak load values. Both needed to be considered because not only had the model to get the "right" answer (according to the 1996 data) but, in order for the result to be general, the model had to have obtained it for the right reason.

Initial Prototype - Naive Model

Initially, a simple all-year model, a January and February model and a January-only model were compared, using a "safe but not necessarily best" position in the search-space, as follows:

The input variables were as follows:

> Sin & Cos of part-of-year (with small bug[3]), Sin & Cos of part-of-day, Day of Week, Maximum Illumination (explained in Section 3.2), Average Daily Temperature and Holiday-Flag. The holiday-flag accounted for both Jan 1 and Jan 6 holidays.
>
> Using these inputs, an ALN model was trained using an obsolete but user-friendly demonstration version of the ALN training software, ALNbench. Data records were taken from 1997 and 1998 data sets, then split into blocks of two days. Alternate blocks were assigned to Train or (Cross-) Validation subsets. Training tolerance was set to 30 (an optimum determined experimentally).

The January and February model worked best. For the daily peaks, the Mean Absolute Percentage Error (MAPE) against the January 1996 data was 3%. This gave confidence that the basic idea worked.

2.3 Possible Approaches (Search-Space)

Although the initial prototype model worked, it was not necessarily representative of the best that could be achieved. To get more insight, it was necessary to explore a number of variations. It was desired, in the limited time available for this exercise, to explore the following:

- **Big Decisions:**
  - Predict peaks directly or predict half-hours and take the maximum of these over the day.
  - Training period (1997 & 1998): January or January & February.
  - Model whole training period or use separate model for 1st week of January.
  - Holiday Records: As-is, Replicated, Replicated with diverse pseudo day-of-week values.

- **Input Selection / Preprocessing:**
  - Day-of-Year: Integer or Quadrature (Sin & Cos 2*PI*proportion-of-year)
  - (Day-of-Week: Integer. Would have liked to explore alternatives such as quadrature or separate flags, but insufficient time to do this. Integer is simpler and has worked before on other data).
  - Time-of-Day: Integer or Quadrature
  - Temperature: current day and/or previous day
  - Input *Maximum Illumination*[4]: Include or don't include
  - Holidays: Ignore, Separate flags, Single Flag, 6th Jan as a "pseudo Saturday", Flag for Jan 2nd.

- **ALN Training Parameters**
  - Tolerance variation (mostly around 15 to 30 power units).
  - (would have liked to also try modelling variable tolerance with time-of-day etc., but there was not sufficient time for this).
  - Jitter or no jitter (small random changes of the inputs).
  - Learning rate
  - Number of training epochs.

---

[3] Examining these results retrospectively, an information-preserving but cycle-breaking bug was found in the computation of Sin & Cos of part-of-year. Since it was information-preserving, the bug should only have had a small effect on accuracy.

[4] Explained in Section 3.2

- Use of old or new ALN training software.

No Lagged Demand
An additional kind of input which can be useful in short term demand forecasting, such as day-ahead is lagged demand. However the competition goal is instead one of long-term forecasting. The most recent data to January 1999 would be from December 1998 - a period which is far from typical (thanks to Christmas and New Year). Therefore such inputs were not considered as part of the search space.

3 Data Considerations

Before starting serious experimentation, attention was focussed on the data.

3.1 Given Data Formats

Data File	Nature of Content
Holidays.xls	A descriptive list of dates, some coinciding with weekends.

Load 1997.xls
(& 1998)
	`Year Month Day 00:30 01:00 01:30 ....(etc.)`
	`1997 1 1 797 794 784 ..... (etc.)`
	`1997 1 2 704 697 704 ..... (etc.)`
	`......... (etc.)`

Temperature1997.xls
(& 1998)
	`Date Temperature [°C]`
	
	`01/01/97 -7.6`
	`02/01/97 -6.3`
	`.............(etc.)`

On request, further spreadsheets were obtained for Load and Temperature of 1996. The data format was not precisely identical but the same information was there. There was also further temperature data for 1995, though there was no corresponding demand data.

For the Load spreadsheets, there were columns for 48 demand figures, one for each half-hour of the day. The demand figures were approximately around 700 to 800. The units were unknown (these values seemed highly unlikely to be MW or GW).

3.2 Further Data

Maximum-Illumination Data
Maximum Illumination (MI) is a log-transformed relative indication of the hypothetical amount of sunlight that would reach the surface on a cloudless day. The log transform is appropriate since human vision and light sensors - both of which can lead to the switching on and off of electric lights - tend to have logarithmic response curves.

Actual demand depends in part on the proportion of MI actually reaching the surface, following attenuation by clouds and haze. For the competition, we have no data on cloudiness, but possibly MI information would be better than nothing. The training/regression algorithm would be able to "decide" its usefulness, provided we made it available.

MI is in principle computable from Sun/Earth geometry and earth atmospheric characteristics [4]. However, it was further noted that we were lucky enough to have a simpler approach: Slovakia is at about the same latitude as the UK, so we were able to simply re-use our existing look-up table for the UK, where the look-up is by local time-of-day.

3.3 Data Selection

The earlier this was done the better, since then we would have less data (and hence fewer special cases such as holidays) to think about and act upon.

Which Days-of-Year?
The competition was merely to forecast demand for January. Therefore, in principle, there was no advantage in training a year-round demand model. On the other hand, we wanted enough data to capture as many possible combinations of time (in all senses) and weather to train a good general

model. The least amount of data we would want to choose would be all the January data. February is fairly similar to January and has no holidays[5], so it was an attractive potential way to augment this data. Restricting to these two months avoided the complication of the clock-change.

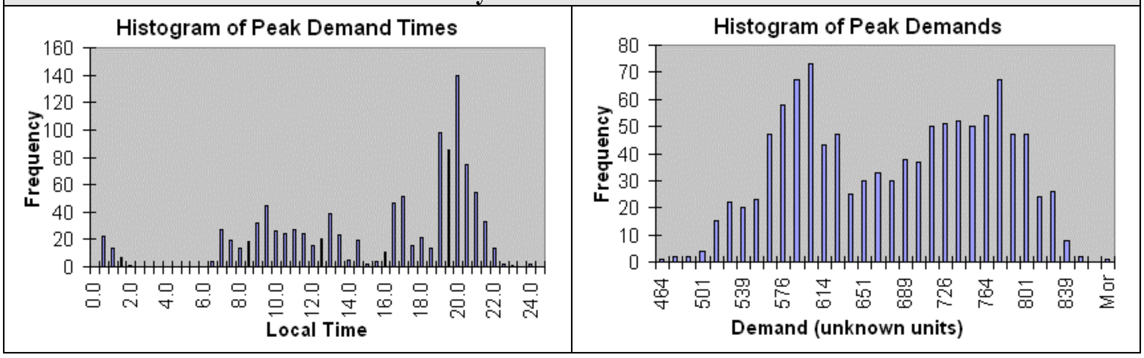
Which Times-of-day?
In principle we only needed to model the times of day where the peaks occurred. However it was known from experience that ALNs benefit from being allowed to form general continuous models, so if the problem happened to be a smooth one, then data adjacent to the peaks could nevertheless contribute to the production of a good model for the peaks. It was believed that there was no downside to this (other than taking slightly longer to train) - it was known that on previous non-smooth problems, ALNs had been effective at partitioning the data space.

3.4 Data Distributions

Next the data was analyzed graphically to get an impression of the main patterns of distribution and the ranges of values. The data were analyzed in Excel97 - by writing VBA "Macros" and using Excel's (plug-in) Data Analysis tools.

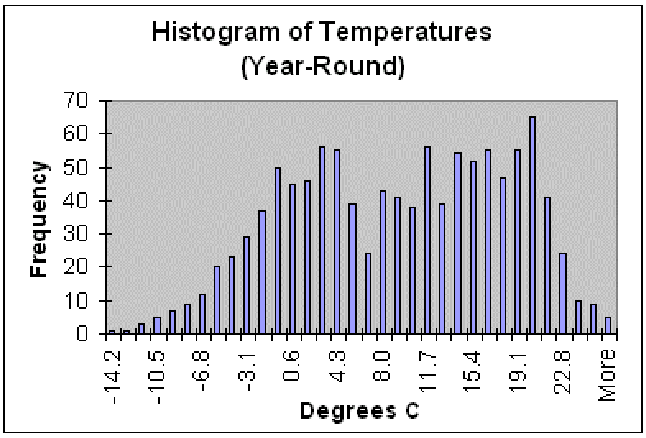
<u>**First, Analyze the All-Year-Round Data**</u>**:**
Although it had been decided not use the whole year of data, it was a good starting-point for analysis.

Analysis of Demand Peaks



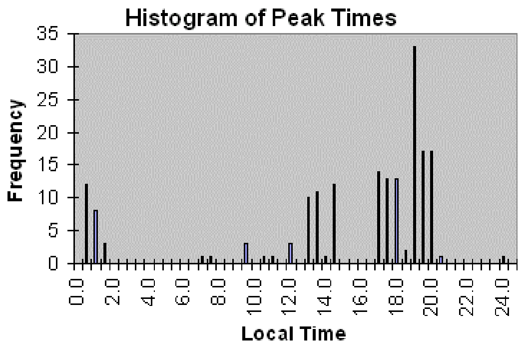
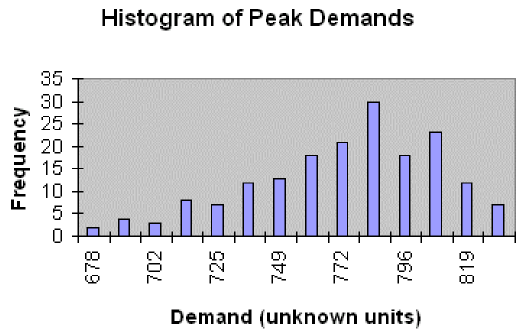
It can be seen that that over the period, the peak demands can occur at most times of the day. The only significant exception is between about 2am and 6am. This confirms that, for half-hourly demand modelling, we would obtain negligible benefit from excluding non-peak times of day. There are four discernible periods where peak demands can occur: midnight, 10am, 4pm and 8pm. The after-midnight peak is unlike anything the author has seen in the UK national demand profile, though it is reminiscent of off-peak (cheap rate) demand switching in some regions. Also, both it and the 8pm have an obviously asymmetric distribution.

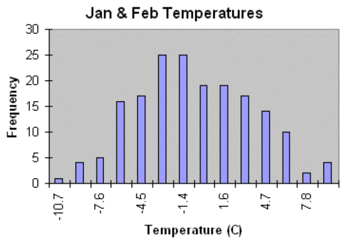
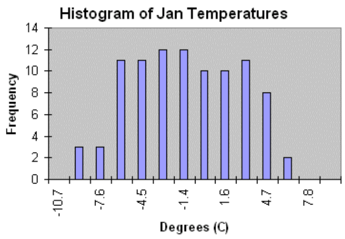
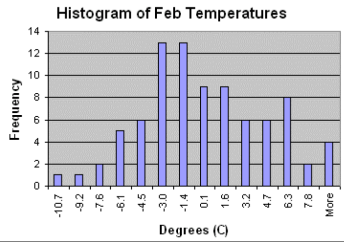
It is interesting that there are two most-frequent levels of peak demand - around 600 and around 780.



[5] Incidentally, if there had been any holidays in February, these records could simply have been excluded (since we are only to be assessed against January of 1999).

Next, Analyze the January and February Data Only:
This is the subset of the data which had been selected for training.

<table>
<tr><th colspan="2">Histograms for Peak Demands (Jan & Feb Only)</th></tr>
<tr><th>Histogram of Peak Times</th><th>Histogram of Peak Demands</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>Again, the peaks can occur over a broad range of times-of-day.</td><td>Peak demands tend to be around 780. The distribution is not symmetrical.</td></tr>
</table>

<table>
<tr><th colspan="3">Histograms for Temperatures (Jan & Feb Only)
(temperature ranges are the same in each case)</th></tr>
<tr><th>Jan & Feb</th><th>Jan Only</th><th>Feb Only</th></tr>
<tr><td>
</td><td>
</td><td>
</td></tr>
<tr><td colspan="3">As had been hoped for, the temperatures are not too different in January and February.</td></tr>
</table>

3.5 Data Checks

A number of questions arose about the data. Here are some of the main ones, together with answers, obtained following data analysis in Excel:

- *Are the data complete and consistent (e.g. any missing or "label-status" data)?*
  Yes (as far as January and February are concerned[6]).

- *Are the values, variances and time-patterns reasonable?*
  Yes.

- *Are the results consistent? Are there outliers?*
  Fairly consistent. Comparing similar day-types, one can see demand inversely proportional (in some way) to temperature.
  There are very few outliers.

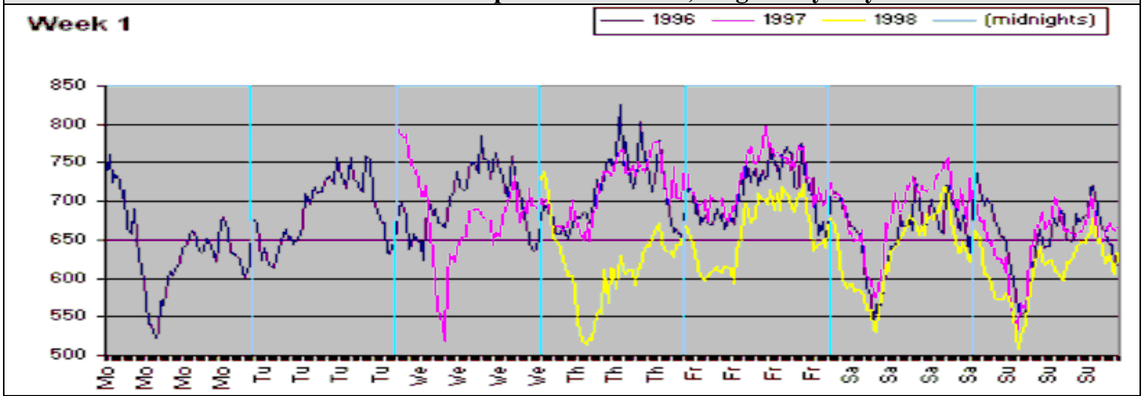
- *Do some assumed ordinary days look a bit like holidays (and vice versa)*
  The Jan 6 holiday looks almost identical to a Saturday at that time of year. This suggests we could possibly encode this holiday as a pseudo-Saturday.

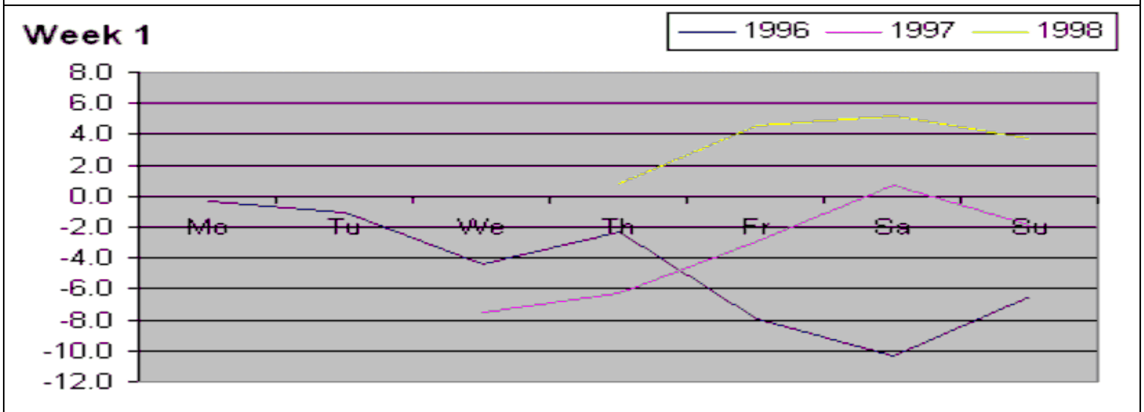
[6] Load data for 31/03/96 for 1st two half-hours was 0 (possibly due to clock-change, but it was not present in the 1997 and 1998 data). Not an issue since we will not be using this period for training.

- *Is there a regular timeseries pattern to the data?*
  Yes. Graphs were plotted (below) showing ordinary days, aligned by day-of-week, and also of holidays, aligned by day-of-year (i.e. date).

- *Is* there a load growth effect?
  No significant growth was discernable.

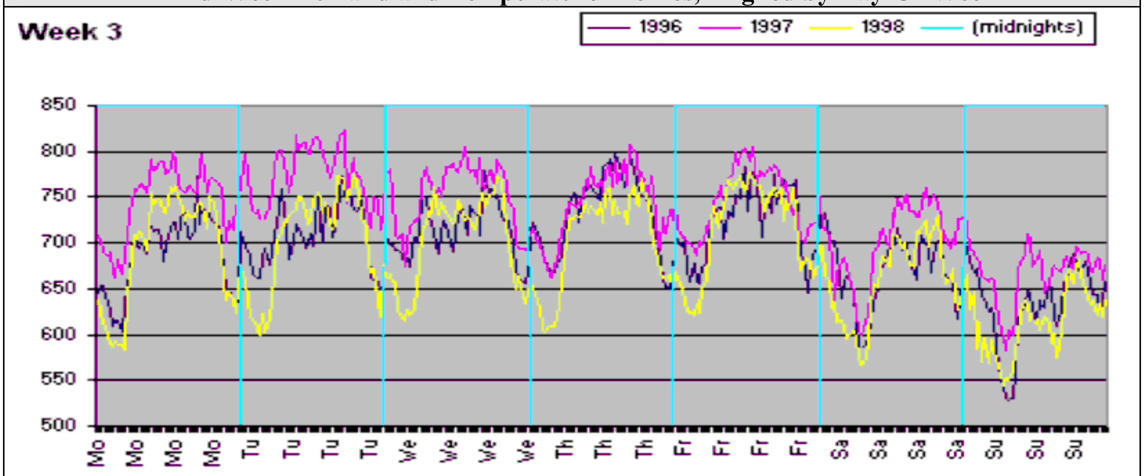
First-Week Demand and Temperature Profiles, Aligned by Day-Of-Week

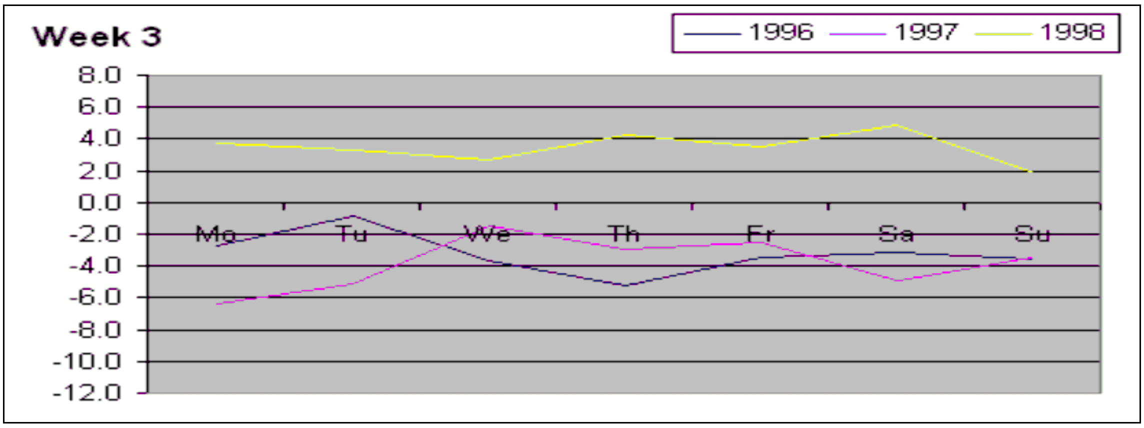




The warmest year (1998, yellow) has the lowest demand.

Third-Week Demand and Temperature Profiles, Aligned by Day-Of-Week





The warmest year (1998, yellow) again has the lowest demand. At the beginning, the coldest year (1997, pink) has the highest demand.

The patterns do indeed appear to be well-formed. The time-series pattern is repetitive and the effect of temperature on demand is reasonably consistent.

4 Experimental Procedure

Next, an experimental procedure was followed, to explore the "search space" identified in Section 2.3 to some reasonable extent in the limited time available (about 1.5 man-weeks).

Initially it had been hoped to at least partially automate the "search" through the use of a genetic algorithm "wrapper" around the ALN training software. In the event, this was not possible due to low-level software "teething" problems. Thus there was no alternative but for repetitive manual training and testing. Unfortunately during this process, some errors were made affecting the preprocessing of some data. However, for those experiments where there was time to make corrections and re-run them, there were no changes to the general conclusions drawn.

In order to measure progress towards an improved model, both the half-hourly and daily-peak results against January 1996 were monitored and compared. This was performed in Excel using VBA macros.

5 Best Method

Here we list the set of decisions that, following the experimentation of Section 4, appeared to work best - at least as judged according to the January 1996 data (half-hourly and peaks). Some new ideas also occurred, indicated by "+" symbols.

- **Big Decisions:**
  - Predict half-hours and take the maximum of these over the day.
  - Training period: January & February of 1997 and 1998.
  - Model the whole training period (no separate model for 1st week of January).
  + Replicate the special days four times over. Each replicate to be re-assigned a "pseudo day of week", to demonstrate to the ALN that day-of-week is irrelevant to (date-bound) special days. Details are given later in this section.

- **Input Selection / Preprocessing:**
  - Day-of-Year: Integer
  - Time-of-Day: Quadrature (Sin & Cos)
  - Temperature[7]: current day and previous day
  - Input *Maximum Illumination*: Include it (debatable - see later).
  - Holidays: 6th Jan as a "pseudo Saturday", flag for Jan 1st, separate flag for Jan 2nd.

---

[7] Possibly a smoothed transform would have been better but there was no time to investigate this. Instead, each half-hour of the day simply has the average temperature for that day.

- Holiday Records: Replicated with diverse pseudo day-of-week values.

- **ALN Training Parameters**
  - Tolerance: 20 power units.
  - \+ (would have liked to also try modelling variable tolerance with time-of-day etc., but there was not sufficient time for this).
  - No jitter (small random changes of the inputs).
  - Learning rate: Phase 1 at 0.1, Phase 2 ("polishing") at 0.001 (for 998 epochs).
  - Number of training epochs: 1000 maximum, but early termination when RMS error falls below 16 power units.
  - ALN training software: A locally customized version of new ALN training software, "ALNfit", was used.

Replication of Special Days
Some early experimental models did not model the first week as well as expected, especially as regards the special days. This was imagined to result from the fact that there were insufficient data for a non-parameterized model to learn from. To address this issue it was proposed to *replicate* these special days, four times in each case. In principle, for each special day, the replicates would each be assigned to a different "pseudo day of week". The purpose of this was to demonstrate to the ALN training process that day-of-week was irrelevant to the (date-bound) special days.

In practice, this approach was only implemented for Jan 1st. The selection of pseudo days-of-week was made to include actual occurrences of this day in 1996-1999:

Monday (as in 1996), Wednesday (as in 1997), Thursday (as in 1998), Friday (as in 1999)

The model inputs were then as follows:

Specification of Model "ALN3"				
Name	**Type**	**Min**	**Max**	**Remarks**
Day of Year	Integer	1	31	1 = Jan 1st
Jan 1 Flag	Integer	0	1	1 if Jan 1st else 0
Jan 2 Flag	Integer	0	1	1 if Jan 2nd else 0
Sin-Day	Real	-1.0	1	Sin(2*PI*n/48), n is half-hr, 0-47
Cosine-Day	Real	-1.0	1.0	Cos(ditto)
Day of Week or Jan 6th	Integer	0	6	0=Mon, 6=Sun; except Jan 6th=5 regardless of day-of-week.
MI	Real	0.0	200	Maximum Illumination
Temp-1	-20		20	Previous day's average temp
Temp	-20		20	Target day's average temp.

Example data, including *target output* for training purposes:

Example Training Data									
1	1	0	0	1	0	0	-3.9	-0.4	740
1	1	0	0.130526	0.991445	0	0	-3.9	-0.4	761
1	1	0	0.258819	0.965926	0	0	-3.9	-0.4	725
1	1	0	0.382683	0.92388	0	0	-3.9	-0.4	734

6 Local Assessment Of Best Model

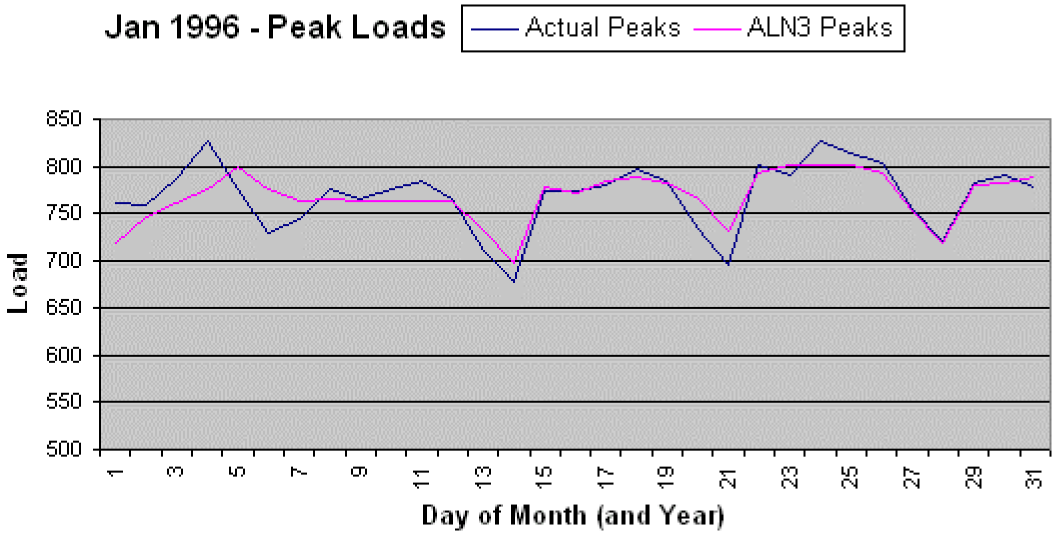
Having established a "best technique", a model was produced according to it. The model was called "ALN3", and was rendered[8] into Visual Basic for Applications (VBA) for convenient use within Excel. In retrospect, it would have been much better to use a "bagging" approach, where several ALNs are trained and the average of their results taken. This did not happen due to shortage of time.

[8] The ALN is initially generated in a formal descriptive format. A separate tool can then be used to render this into VBA, C++ or Java as well as an analysis-friendly form.

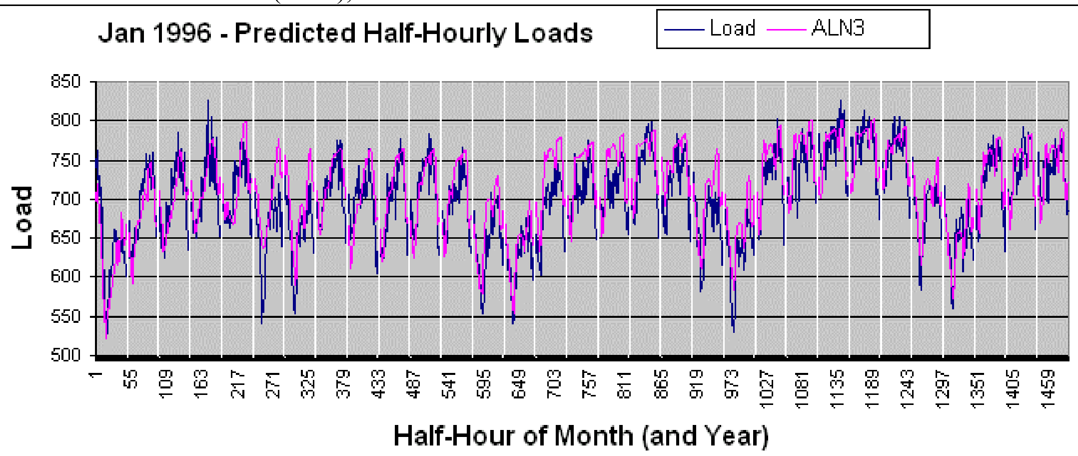
6.1 Validation Against January 1996

As stated at the beginning, the main form of assessment was to compare predictions against January 1996, as regards both half-hourly and daily-peak demand values.

Local Assessment - Against January 1996



Max Abs Error: 50 = 6.3% (Jan 4), Mean Abs Error: 16 = 2%.



Max Abs Error: 107 = 19% (missed "valley" in Jan 6th), Mean Abs Error: 25 = 3.7%

Note that differences here should initially be regarded as disagreements rather than errors. We have no prior way of knowing how representative January 1996 is of January 1999. What we can do however is to compare January 1996 to the other Januaries, in the Training data subset (1997 and 1998), to see how consistent it is with them, bearing in mind the possible effects of temperature differences.

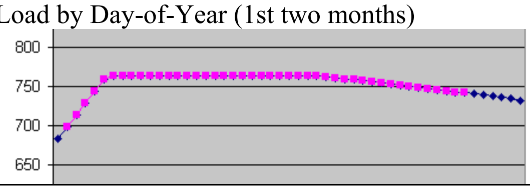
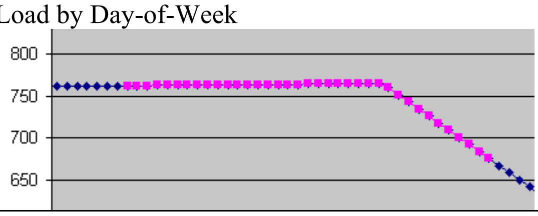
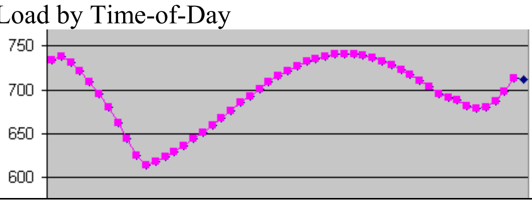
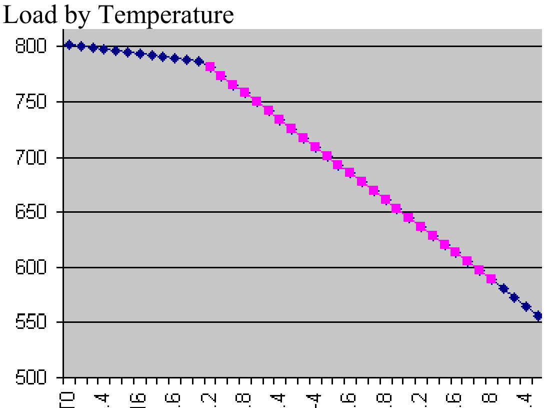
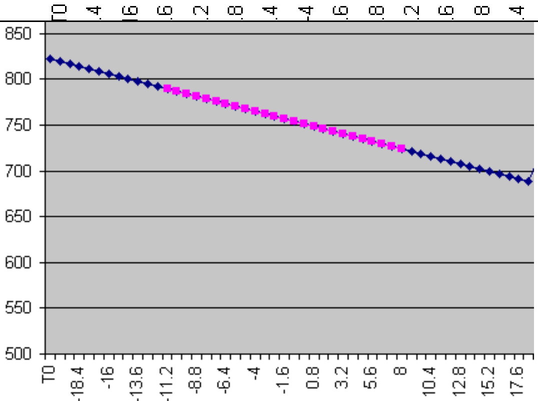
Jan 1 of 1996 has a reasonably typical profile. However Jan 6 of 1996 an exceptional peaking micro-feature around 14:00, which establishes itself as the daily peak itself. No such micro-feature is present in Jan 6 of the other years, where the main peak of the day is instead at around 20:00. As a result the daily peak for 6 Jan 1996 is about 10 units greater than it would have been without this feature. Also, 1996 had the lowest average daily temperature for Jan 6.

For comparison, the Naive model of Section 2.2, managed to achieve a mean absolute percentage error (against the January 1996 peaks) of 2.9%. So the additional effort of experimentation had managed to reduce the mean error by about a third.

This is the general experience of load modelling problems - further reduction of error is subject to diminishing (though not zero) returns.

6.2 Model Cross-Section Analysis

An alternative form of assessment is to examine "cross sections" through the model. Here, we vary one input while all others are held at some (strategically chosen) fixed values. This was carried out using a purpose-written "Macro" in Excel.

Cross-Section Graphs through Model for Different Variables	
Partial Function	**Fixed Settings**
Load by Day-of-Year (1st two months) 	Temp = Temp-1 = -2.93, MI=68, Time =11am, Day=Thurs.
Load by Day-of-Week 	Temp = Temp-1 = -2.93, MI=68, Time =11am, Day-of-Year=28
Load by Time-of-Day 	Slightly misleading since here MI is fixed (i.e. the sun shines even at night!) - a current limitation of the Excel macro. Temp = Temp-1 = -2.93, MI=68, DayOfYear=1, Day=Thurs,
Load by Temperature 	MI=68, Time =11am, Day-of-Year=1, Day=Thurs Here the load sensitivity to temperature is great - about (782-589) / (9.4 - -10.7) i.e. 193/20.1 i.e. around 9.6 load units per degree C.
	As above but Day-of-Year = 23. Here the load sensitivity to temperature is much lower - about 66/20 = 3.3 load units per degree C

- The steeper temperature gradient of the first week is possibly an artefact of the limited data (i.e. if it just happened to be cold on high-demand 1st weeks of January whereas there was high-demand for some other, unexplainable, reason).

- MI was revealed to have only a tiny effect (hence no graph shown). For example, at high temperatures and illumination levels, increases in it would very slightly reduce demand. Possibly the experiments that suggested it had been useful were misleading (most likely due to experimental error). Or possibly, it served some useful transient purpose during training (a temporary "stick" for the part-trained model to lean upon)

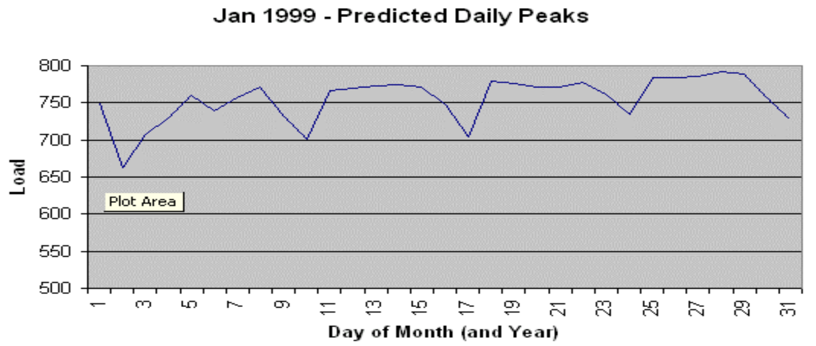
7 Final Results - January 1999

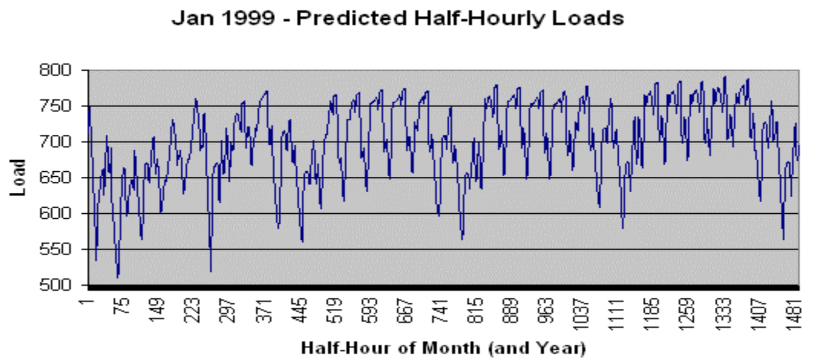
A single ALN was produced according to the "Best" method outlined in the previous section. Since no temperature data was provided for 1999, it had to be estimated for the target period (January 1999). A simple policy was chosen: for each day-of-year, use its average temperature over 1996 to 1998. Since one of the required inputs was the previous day's temperature, it was also necessary for Jan 1 to use Dec 31 from 1995 to 1997. In retrospect, one could perhaps have made more thorough use of the available data, but by this point in the exercise, there was little time left, so this simple straightforward method was used. Results were as follows:

Main Results - January 1999

Daily Peak Loads
750
663
707
730
760
740
757
771
731
702
766
769
773
774
771
748
705
779
776
772
771
778
761
734
783
784
785
792
788
758
728

Graphs





Broadly speaking, the daily peaks graph can be matched up against the top edge of the half-hourly loads graph. Bear in mind however that there is potentially more than one peak per day.

The accuracy of this final result is, of course, unknown at the time of writing.

8 Further Ideas

- More thorough exploration of the "search space", for example using a genetic algorithm or possibly an "almost-exhaustive" systematic search.
- Training of the ALN with adaptive tolerance, as a function of estimated uncertainty (e.g. noise) in the data. The ALN software debatably does not require this, but it would be worth trying as the procedure and algorithms for this are already established.
- Day-Type classification using Kohonen's Self-Organizing Map (for example).

- Make use of bagging - where several ALNs are trained then in execution, the average of their results is taken. This tends to reduce the effects of noise in data or the training process.
- Revisit the treatment of special days - e.g. Jan 6th is not *exactly* like a Saturday.
- Temperature variables - investigate different kinds of moving average etc.

9 Conclusions

The problem of predicting peak daily demands was approached by developing a model from time and weather information to half-hourly electrical demand, then computing the maximum of its results for each day. This was found more successful than directly modelling the peaks. The East Slovak load profile is quite different from that of the UK, featuring contending peaks at different times of day, including one just after midnight.

The goal was to estimate daily peak loads in January 1999. Available data available consisted of year-round average daily temperatures from 1995 to 1998 as well as half-hourly loads from 1996 to 1998. The January 1999 temperature data, required by the half-hourly model, were estimated as a simple average of the 1996-1998 data. Only a small, restricted, use was made of the 1995 data.

The modelling technique used was a non-parametric nonlinear modelling method called Adaptive Logic Network (ALN). The ALN training software used was from Dendronic Decisions Limited, of Canada. The training data used was half-hourly load and daily average temperature for January and February of 1997 and 1998. The input variables also included various transformations of time and, experimentally, an indication of relative cloud-free sunlight levels[9].

As for any non-parametric nonlinear models, especially growing ones, there was a need for some method of determining when the training data has been optimally fitted by an ALN. For this purpose the January 1996 data was used as the basis for a “benchmark” test. The degree of fit for each prototype model was controllable through the Tolerance setting of the ALN training software. The same data was also used as a basis for comparing the various experimental models as regards accuracy on both half-hourly and daily-peaks.

Initially, within two man-days of the exercise, a small variety of half-hourly models were produced for various proportions of the year and “best guess” selections of inputs and training conditions. Around two man-weeks were then spent exploring the relative merits of variations on these initial approaches, such as selecting different inputs and their transformations and varying the training parameters and training algorithms used to produce the ALNs.

The initial models attained a Mean Absolute Percentage Error (MAPE) of around 3%, while the final and best model attained about 2%. It is felt that one or two further weeks of experimentation would probably have reduced it still further.

The accuracy of the Jan *1999* prediction by this model cannot be known at the time of writing, but its daily-peak accuracy against the Jan 1996 data, not involved in the training, was as follows:

Performance Against January 1996	
Mean Absolute Error: 16 = 2%.	**Max. Absolute Error:** 50 = 6.3% (Jan 4)

Regardless of the outcome of the competition, this has been a useful exercise. It has demonstrated the flexibility of the ALN-based modelling method to accommodate new kinds of load pattern. Also it has been a useful learning exercise as regards the treatment of special days – an ongoing subject of our research.

---

9 Computed from Sun-Earth geometry etc.

Acknowledgements

The paper is published with the permission of the National Grid Company plc. in the UK (www.nationalgrid.com).

In addition, the author would like to thank National Grid for supporting this exercise as well as for the opportunity to conduct prior research in this general field.

Thanks also to Dendronic Decisions Limited, 3624 - 108 Street NW, Edmonton, Alberta, Canada, T6J 1B4 (http://www.dendronic.com) for their general advice on how best to apply ALNs.

References

1. Armstrong, W. W. and Gorodnichy, D. O., “Breaking Hyperplanes to fit Data with Applications to 3D World Modeling and Oil Sand Data Analysis”, Proc. ICSC Symposium on Neural Computation, NC 2000, Berlin, May 23 - 26, 2000, CD-ROM publ. by ICSC Academic Press, Int'l Comp. Sci. Conv., Canada/Switzerland ISBN 3-906454-21-5

2. Armstrong, W. “On Networks of Adaptive Boolean Logic Elements and their Application to Pattern Recognition”, Proc. Fourth Princeton Conference, 1968, p. 326.

3. Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, E., Winkler, R., “The Forecasting Accuracy of Major Time Series Methods”, publ. John Wiley & Sons, ISBN 0 471 90327 2, 1984.

4. Cox, Arthur N., 2001, "Allen's Astrophysical Quantities", ISBN: 038795189X,: Springer-Verlag New York, Incorporated, Publish Date: 01/01/2001.