Establishing a Solution Strategy for Electrical Demand Forecasting in Ireland

Damien Fay[a*] John V. Ringwood[b] Marissa Condon[a] Michael Kelly[c]

[a] Dublin City University, Glasnevin, Dublin 9, Ireland
[b] NUI Maynooth, Maynooth, Co. Kildare, Ireland
[c] Electricity Supply Board, Dublin 2, Ireland

Abstract

Electrical demand is driven by economic and human activity, which has obvious daily, weekly and yearly cycles as well as a long-term trend and special periods such as bank holidays, Christmas etc., all of which are reflected in load data. These characteristics of electrical demand must inevitably be incorporated into any demand-forecasting model. However, with the exception of a few papers, the vast bulk of the literature on electrical demand forecasting is concerned with forecasting techniques. This paper proposes several methods with which to quantify the characteristics of Irish electrical load data prior to modelling.

1. Introduction

This paper describes preliminary analysis of Irish electrical load data, which is important for model building. Electrical load data, unlike a stationary time series, changes over time and has embedded cycles.

The amount, time-scale and type of data available has obvious importance in model building. This information is detailed in Section 2.

In the long term, load is an evolving variable. With increased economic activity over the last few years, electrical demand has increased considerably. It is important to quantify the rate of increase, the effect it may have on the shape of the load and to ascertain if these characteristics are deterministic. Such characteristics are examined in Section 3.

The embedded cycles in the load occur at weekly and yearly time-scales. The shape of the daily load curve as well as the level of the load is affected by these cycles. Classification of the shapes is examined in Section 4. The level of the load is effected mainly by temperature and this relationship is examined in Section 5 to determine if it is non-linear or can be linearised.

Section 6 establishes a general strategy to model the data in view of the characteristics. Finally, conclusions are drawn in Section 7.

2. Data Set Details

There are two sources of the data used in this thesis: the Electricity Supply Board (ESB) supplies load data (Table 1), while the Meteorological Office of Ireland (MOI) provides dry bulb temperature data gathered at their station at Dublin airport. The range and time-scale for the available data is given in Table 1.

Table 1. Data time-scale and range.

Range	29th December 1986 – 31st March 2000
Timescale	Hourly
No. of data points	4842 Days (116208 hours)

3. Trend and Variability

Figure 1 shows the growth in electrical demand in Ireland in recent years. To gain initial insight into the nature of the underlying trend a quadratic curve of the form (1) is fitted to the data:

* Corresponding author. Email: fayd@eeng.dcu.ie

$$d(t) = at^2 + bt + c + e(t) \quad (1)$$

where t is the time in hours since the start of the data (1986), $d(t)$ is the trend at time t, $e(t)$ is an error term, and a, b, c are coefficients calculated via least squares. Note that a,b,c are positive (Table 2).

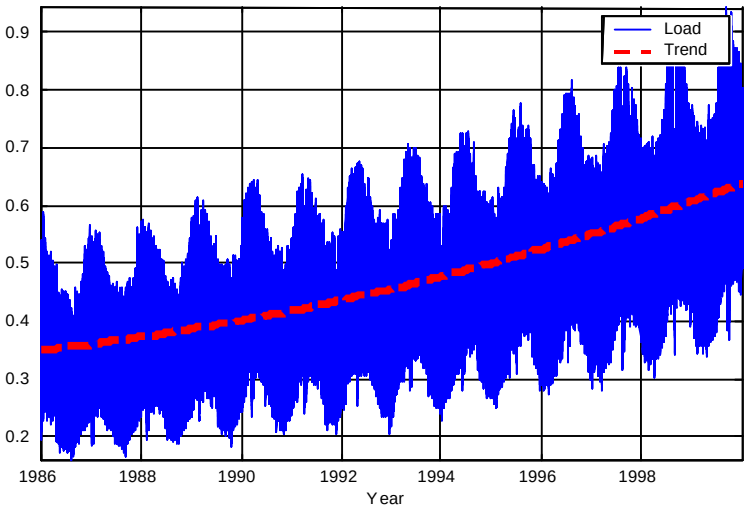

Figure 1. Load with approximate trend curve.

Table 2.Co-efficients of trend curve (normalised).

Co-efficient	Value
a	1.1360 e-011
b	1.1640 e-006
c	0.3498

In addition to an underlying trend there is also a growing *variability v*. The variability of load for year j is defined as

$$v_j = y_{\max,j} - y_{\min,j} \quad (2)$$

where $y_{max,j}$ and $y_{min,j}$ are the maximum and minimum loads in year j, respectively. This rising variability is shown in Figure 2(b).

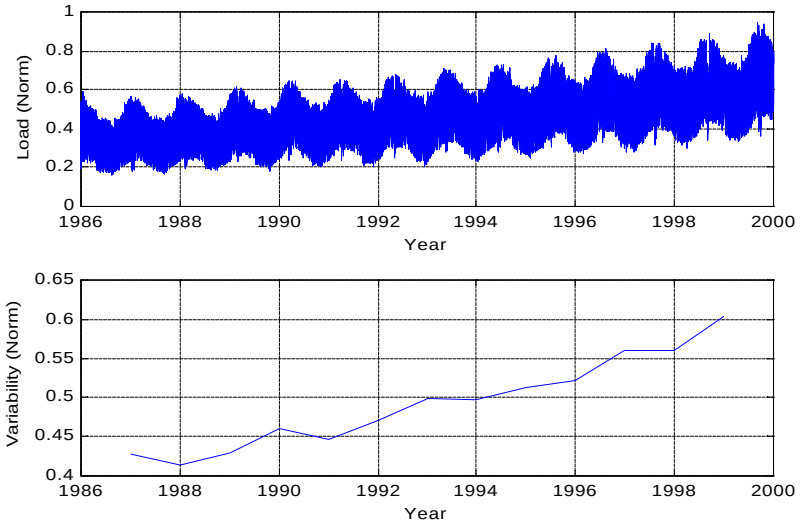

Figure 2 (a) Hourly load data and (b) its variability

The trend and the variability can be related by the use of the Box Cox transform [1]. They proposed a transform for *equalising* (to make constant) the variance of a time series where the variance, at a given time *t*, is related to the value of the series, at that time *t*. This transform can also be applied to equalise the variability of time-series whose variability is a function of the trend [2].

The data is transformed according to a parameter λ [1] as:

$$y'(t) = \begin{cases} (y(t)^{l} - 1)/l & \text{if } l \neq 0 \\ \log(y(t)) & \text{if } l = 0 \end{cases} \quad (3)$$

where $y'(t)$ is the transformed load *t* hours from the start of the data (1986) and $y(t)$ is the original load.

In the current study electrical load data is transformed using the Box-Cox transform with λ = 0.3 (Figure 3). As the purpose of this section is merely to establish that a relationship between trend and variability *exists*, λ is determined empirically (as suggested by Cryer, [3]). The transformed load data exhibits an approximately equalised variability (Figure 3), showing that a definite relationship exists between the trend and variability of the load.

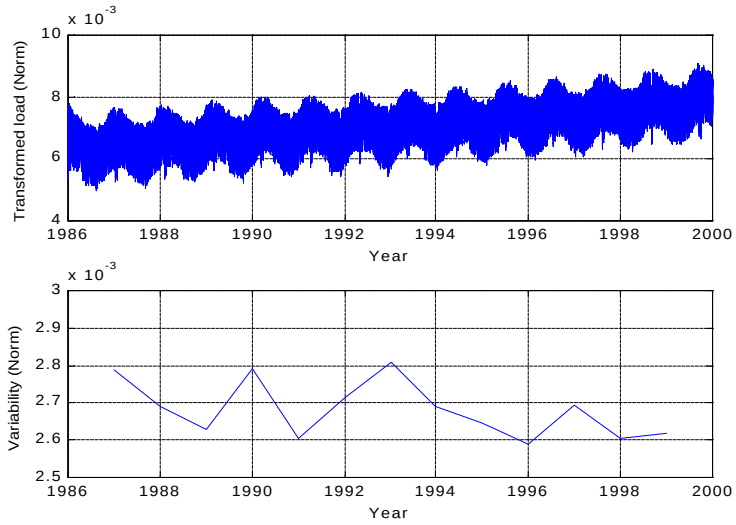

Figure 3. (a) Transformed load and (b) variability of transformed load

4. Day-Type Identification

Daily load data can be disaggregated into distinct groups (called day-types) each of which have common characteristics. As can be seen in Figure 4 there is, for example, an obvious difference between the *shape* of the load on a typical Sunday and Monday due to decreased economic activity on a Sunday. Furthermore, there is a distinct difference between the *shape* of a typical Winter day and Summer day (Figure 5). A typical Winter day exhibits a higher peak at 6pm relative to a Summer day, due to increased lighting needs in Winter, among other things.

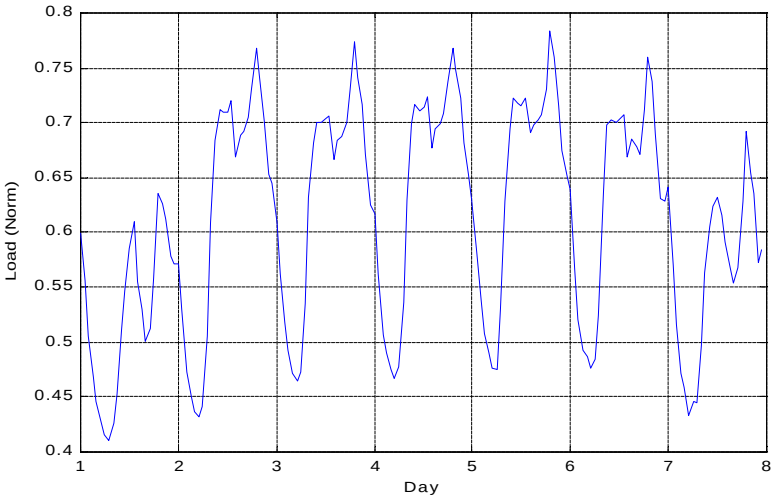

Figure 4 A typical weekly load (Day 1=Sunday)

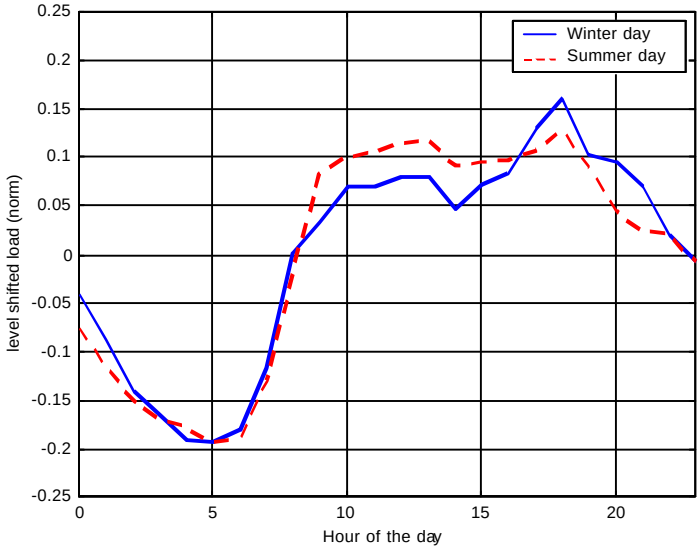

Figure 5 Typical shape of a Winter (18/11/1998) and Summer working day (17/06/1998) (The mean load for each day has been subtracted)

The existence of several different day-types has been shown by several researchers [4,5,6]. However, the level of disaggregation in day-type selection is, to a large extent, subjective and dependant on the judgement of the forecaster.

As pointed out by Hubele and Cheng [7], the application of a separate load forecasting model for different seasons (for example Summer, Autumn, Winter and Spring) has the advantage that the models need not incorporate seasonal information. Further disaggregation of the load by day of the week (for example Summer Sunday, Winter Sunday, Summer Monday etc.) reduces further the amount of information that the model need incorporate. Such approaches have been implemented successfully by Srinivasan *et al* [8] and Mastorocostas *et al* [9], to mention but a few.

Where a single model is used for all the data, the day-type information is often incorporated as an additional input (two examples are Chen *et al* [10], and Lertpalangsunti and Chan [11]). In either case the day-types must, however, be identified.

The selection of day-types can be guided by analytical techniques. Three candidate techniques were considered for day-type identification:

1. Interviews with system operators (for example [12]),
2. Clustering algorithms (for example [5]) and
3. Self-organising feature maps (for example [6]).

Lonergan [13] presented interviews with Irish system operators. The interviews indicate that there are two seasons in the year: Winter and Summer. There is no dissagregation by day of the week and so the total number of day-types identified is two.

Clustering algorithms group data together into *clusters* (sets). Often a cluster can be assigned to a single unusual data point while another cluster represents two close but distinct groups [14]. These algorithms are best applied when the *a-priori* existence of the number of clusters is known.

The self-organising feature map or Kohonen map [15] is the preferable option for day-type identification as the number of day-types is not pre-specified and the proximity of the identified day-types is known.

The Kohonen map can be implemented for day-type identification in several different ways (examples are [4,5,6]); however differences in the results are insignificant in most cases thus the algorithm used by Hsu and Yang [6] was chosen. The Kohonen map structure is diagrammatically shown in Figure 6 below.

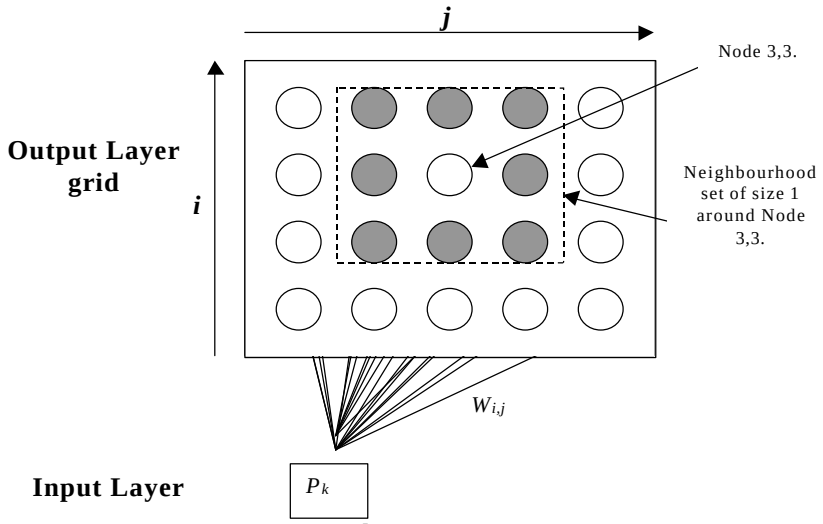

Figure 6 Kohonen map structure.

The network consists of a grid of output nodes connected to the inputs via a set of weights. When presented with the k^{th} input vector $\boldsymbol{P}_k$, $\varepsilon\ R^{1\times n}$, the network calculates the *activation* of each node by $\boldsymbol{P}_k$ as:

$$a_{i,j,k} = \boldsymbol{W}_{i,j}\boldsymbol{P}_k \tag{4}$$

where $a_{i,j,k}$ and $\boldsymbol{W}_{i,j}$ are the activation of, and weight ($\varepsilon\ R^{n\times 1}$) connecting $\boldsymbol{P}_k$ to, node i,j respectively. $\boldsymbol{P}_k$ is said to be *mapped* onto the node with the highest activation. After several inputs have been presented, similar inputs are mapped to the same or adjacent nodes, i.e. within a small *neighbourhood*. A neighbourhood of size Nc around node i,j is defined as nodes $i \pm Nc$ to $j \pm Nc$.

$\boldsymbol{P}_k$ for the current study is formed in two steps. Initially, the daily load curve is extracted from each day to give a set of load curves that have a minimum value of zero and a maximum value of one [6]:

$$\boldsymbol{Y}'(i)_k = \frac{\boldsymbol{Y}(i)_k - \min(\boldsymbol{Y}_k)}{\max(\boldsymbol{Y}_k) - \min(\boldsymbol{Y}_k)} \qquad i = 1,2,3...24 \tag{5}$$

where $\boldsymbol{Y}(i)_k$ and $\boldsymbol{Y}(i)_k$ are the i^{th} elements (hour) of the load curve $\boldsymbol{Y}_k$, $\varepsilon R^{1\times 24}$, and actual load $\boldsymbol{Y}_k$, $\varepsilon R^{1\times 24}$, of day k respectively. The load curves are then normalised to give them unity length:

$$\boldsymbol{P}(i)_k = \boldsymbol{Y}'(i)_k \Bigg/ \left(\sum_{j=1}^{24} \boldsymbol{Y}'(j)^2{}_k \right)^{1/2} \qquad i = 1,2,3,...,24 \tag{6}$$

where $\boldsymbol{P}(i)_k$ is the i^{th} element of $\boldsymbol{P}_k$. The weights are initialised [6] as:

$$\boldsymbol{W}_{i,j} = \left\| \left[m_p(1), m_p(2), ..., m_p(24) \right] + 5u \left[s_p(1), s_p(2), ... s_p(24) \right] \right\| \tag{7}$$

where $m_p(i)$ and $s_p(i)$ are the mean and standard deviation of $p(i)$ over all k, u is a uniformly distributed random number in the range -0.5 to 0.5 and $\boldsymbol{W}_{i,j}$ is normalised to unit length as in (6). The weights are not initialised randomly but initialised around the mean of the inputs as the inputs are all similar and thus restricted to a small portion of the space [6].

During training the inputs are presented one by one and the weights of the *triggered* node (the node to which the inputs is mapped) and nodes in its neighbourhood are updated [6] via:

$$\boldsymbol{W}_{i,j}(m+1) = \boldsymbol{W}_{i,j}(m) + \alpha(m)[\boldsymbol{P}_k - \boldsymbol{W}_{i,j}(m)] \tag{8}$$

Where α is the adaptation gain, with $0 < \alpha < 1$, and m is the iteration number. This has the effect of increasing the activation of the triggered node and it's neighbours. In a single iteration all the inputs are presented and the weights adapted. After several iterations, the neighbourhood size is reduced by one and so on until zero, i.e. the triggered node only is adapted.

The parameters used in the Kohonen map are:

- Initial neighbourhood size: 4,
- Adaptation gain: .002,
- Number of iterations after which *Nc* is reduced: 10 and
- Number of output nodes: 18×18 (324 in total).

These values are similar to those used by Hsu and Yang [6]. The value of α affects the rate at which the network adapts to each input; if the value is too large the network overreacts to each input while the network will not converge if the value is too small. The value of α may be adapted from iteration to iteration however in simulations conducted by the author little difference was found; this was also the case reported by Hsu and Yang [6]. As was found by Hsu and Yang [6], a wide range of values from .001 to .007 was found to give satisfactory results.

For the present study, the trials used all of the last 2 full years of data, 1998 and 1999. Figures 7 to 10 below show which nodes are triggered and the number of inputs that mapped to that node.

Although the weights are initialised to aid in spreading the activation across the complete output grid, after a single iteration the inputs map to several nodes in close proximity to each other (Figure 7). This is because the neighbourhood is large and the adaptation is acting *globally* (the weights of many of the output nodes are adapted by each input).

After fifteen iterations the outputs have segregated into four groups, the groups with the lower number of mappings representing mostly Sundays and Saturdays and the other groups representing mainly weekdays (Figure 8). At this point the neighbourhood size has reduced to 3.

After thirty iterations the data has been spread across the grid of output nodes with different day-types occupying different parts of this grid (Figure 9). The neighbourhood size is now 2 and the weights are

being adapted *locally* (only the weights associated with nodes a distance of 2 away from the mapped node are being adapted). This causes similar day-types in the four groups to subdivide and trigger adjacent nodes. Graphically, this means that new parts of the grid are being triggered in Figure 9 relative to Figure 8. An example is shown in Figure 8 and Figure 9 of the subdivision of the winter weekday group.

Iteration fifty is the last iteration as the neighbourhood size has now reduced to zero. The final iterations from thirty to fifty refine the day-types already identified (Figure 10).

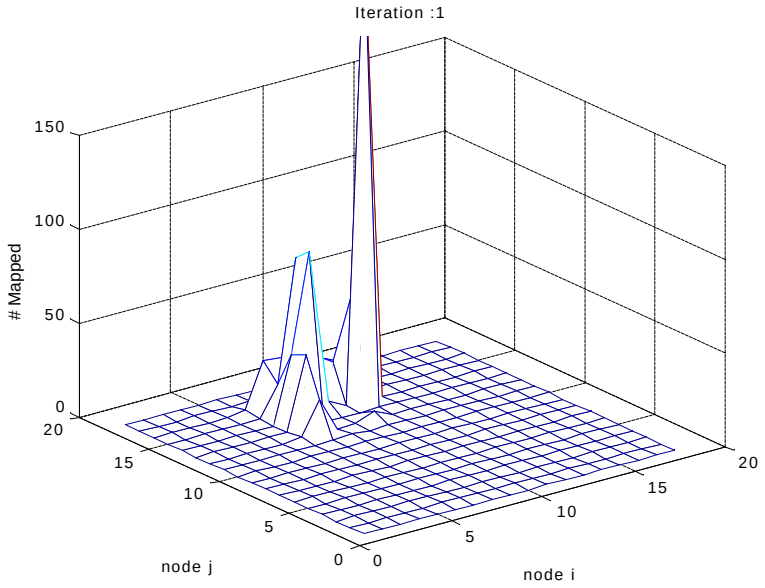

Figure 7. Number of inputs that map to each node (iterations:1, *Nc: 4*).

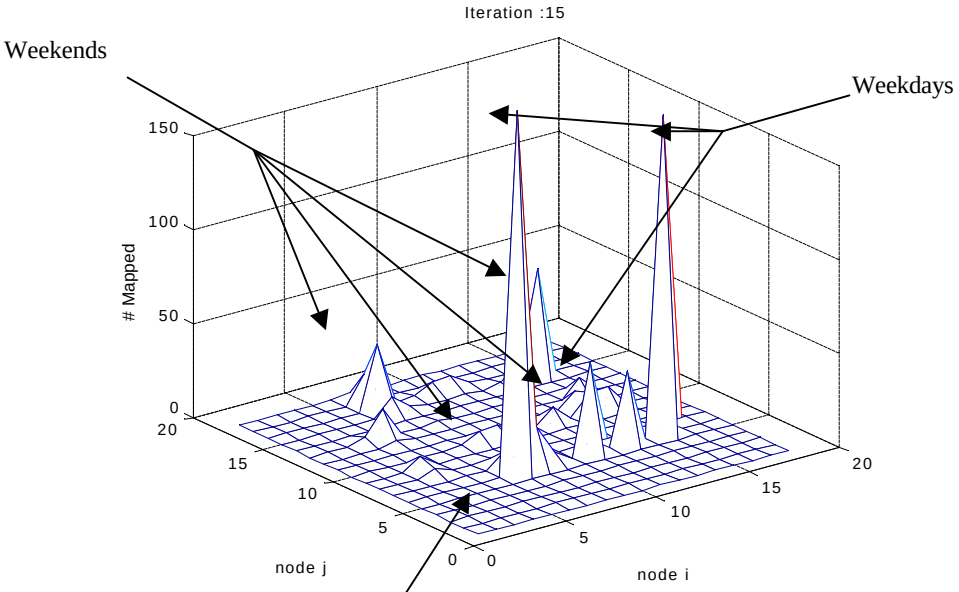

Figure 8. Number of inputs that map to each node (iterations:15, *Nc:3*).

Subdivision of the winter weekday group

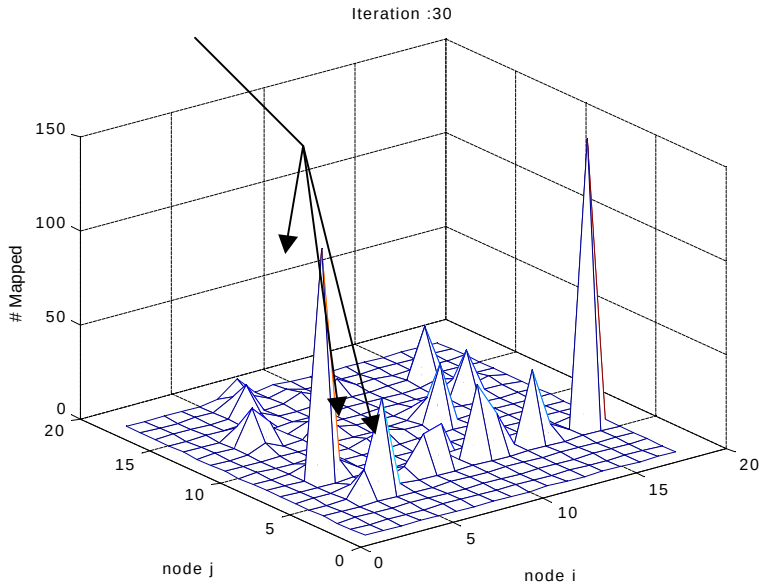

Figure 9 Number of inputs that map to each node (iterations:30, *Nc:2*).

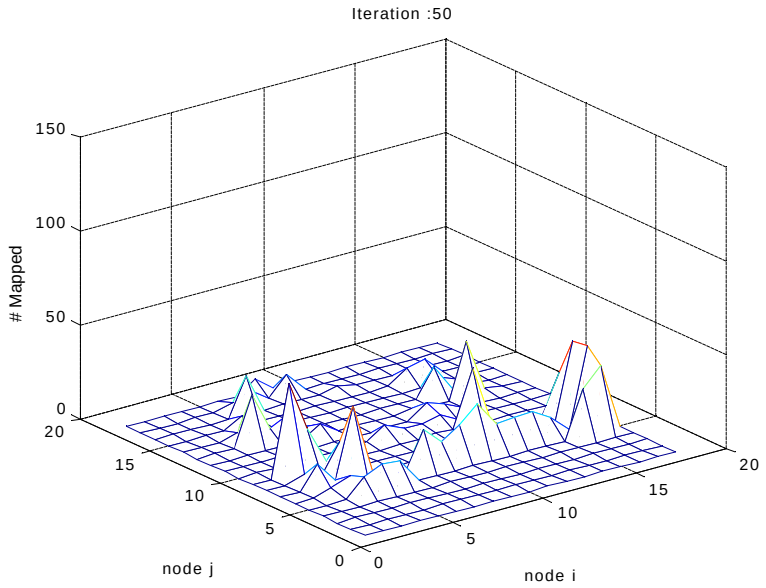

Figure 10. Number of inputs that map to each node (iterations:50, *Nc:0*).

From the above analysis, it can be seen that the algorithm has operated as expected. Initially, the inputs are spread across the output grid according to large differences in the inputs. Subsequently each of the large groups is refined locally into any sub-groups that may exist.

The next step is to assign day-types to the triggered nodes in Figure 10. Figure 11 shows the nodes that are triggered from Monday to Friday. As can be seen the nodes that are triggered for these days occupy the same parts of the grid. Note the only exception, that several Monday loads trigger nodes around (i=13, j=13). This is explained later in this section.

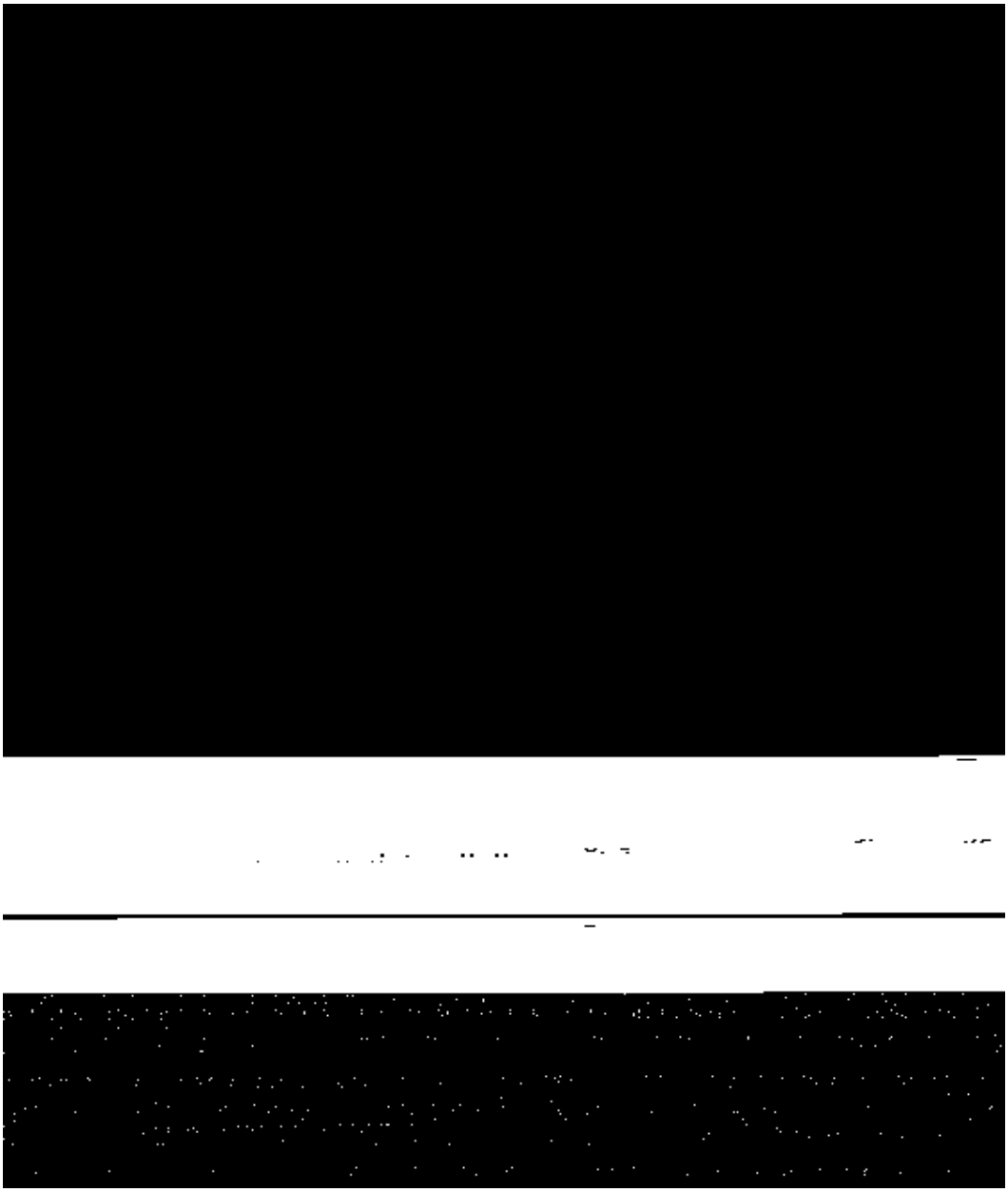

Figure 11. Nodes triggered by Monday-Friday loads (iterations:50, *Nc:0*).

Sunday, Saturday and Monday loads trigger different parts of the grid showing the difference between these days (Figure 12). The one exception as pointed out previously is that several Monday inputs trigger the nodes around (i=13,j=13). The inputs responsible for this are Monday bank holidays and it is interesting to note that they are mapped to the same nodes as some of the Sunday loads (Figure 12).

On closer inspection, it is seen that the summer months (May to September) are mapped to the right-hand side of the grid while the winter months (excluding Christmas) are mapped to the left-hand side. Christmas loads with the exception of Saturdays occupy a separate node indicating that Christmas loads are different to others (Figure 12). It is by co-incidence that the triggered nodes are aligned on the grid such that loads on different days of the week changes with the *y*-axis and the time of year with the *x*-axis (Figure 12). Other tests, performed by this author, have shown similar results with the triggered nodes aligned in other directions e.g. diagonally. Finally, with the exception of Summer bank holidays, the spikes in Figure 11 for working day loads can be assigned similar day-types to those for Mondays (Figure 12).

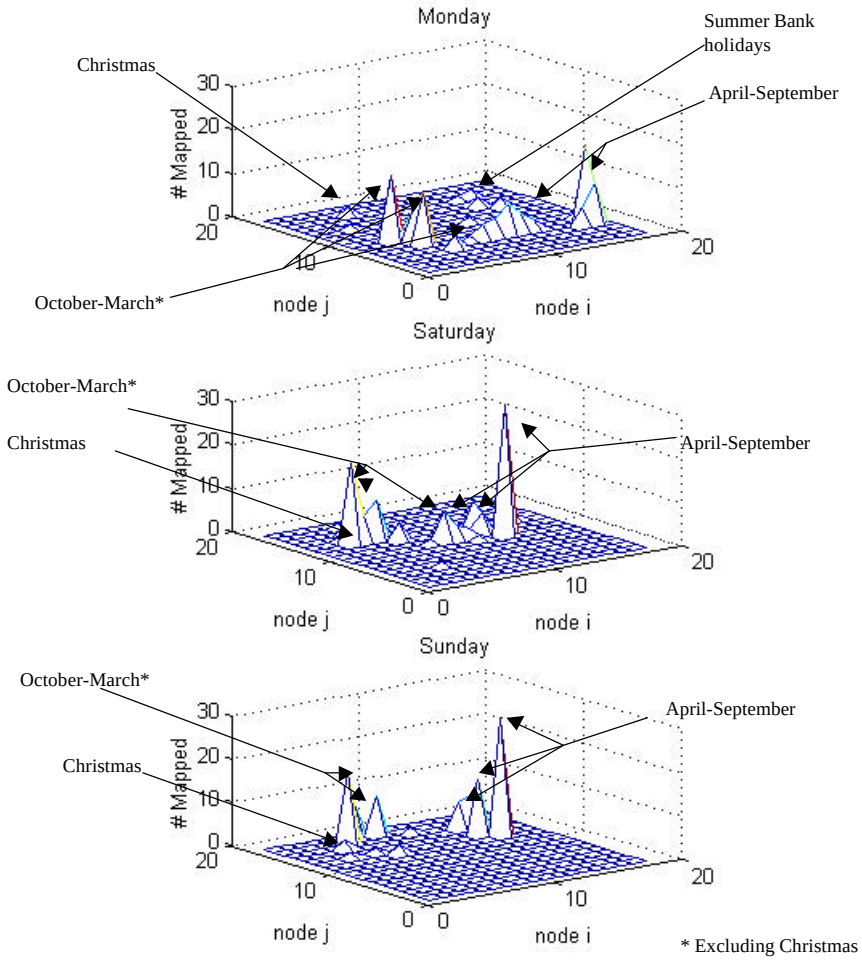

Figure 12. Nodes triggered by Monday, Saturday and Sunday loads. (iterations:50, *Nc:0*).

The day-types identified are collated in Table 3. The October to March day-types are split into two groups (early and late Winter) to reflect the fact that Christmas lies at the centre of this range.

Table 3. Day-types identified for Irish load.

Day-type	Range
Early Winter Sundays	October-Christmas
Summer Sundays & bank holidays	April-September & bank holidays
Late Winter workings days	January-March
Early Winter working days	October-Christmas
Summer working days	April-September
Late Winter Saturdays	January-March
Early Winter Saturdays	October-Christmas
Summer Saturdays	April-September
Late Winter Saturdays	January-March
Christmas days	Christmas

The bank holidays in the Irish calendar are:

- St. Patrick's day March 17th (or closest Monday),
- Good Friday (Lunar calendar),
- Easter Monday (Lunar calendar),
- May day, 1st Monday in May,
- 1st Monday in June,
- 1st Monday in August and
- Last Monday in October.

The transitions between these day-types are considered in Section 6 after the relationship of temperature to the load in each day-type has been examined.

5. Temperature-Load Relationship

This section examines the relationship between load and temperature, the dominant causal variable for load (as pointed out by Murray [16], Hyde and Hodnett [17], for Irish load data and Lu *et. al*. [18], Chen and Kao [19], for other systems to mention but a few). In Section 4 different day-types were identified within the load data. The relationship between temperature and load within each day-type is of prime importance in modelling that day-type. It is important to know if the relationship is consistent within the day-type or if the relationship is a function of the time of year. Consider, for example, the late Winter working day day-type, which spans the months January to March. If the load-temperature relationship in January is significantly different to that in March then the inclusion of time of year information is warranted.

Pre-Processing of Electrical Data to Remove Trend and Variability

The relationship of the most recent load data (year 2000 data) to temperature is of prime importance in predicting future loads. As the load is segmented for analysis into several day-types, the amount of data for the year 2000 available for that analysis reduces significantly and previous years of data must be used (1986-2000). As pointed out in Section 3, Irish load data has an underlying trend and rising variability. These characteristics are not present in the temperature readings for the same period (1986-2000) (Figure 13) and so the relationship between load and temperature is obscured. This section details how the growing trend and variability are removed from the data, so that the fundamental relationship between the load and temperature is revealed. This is achieved in three steps.

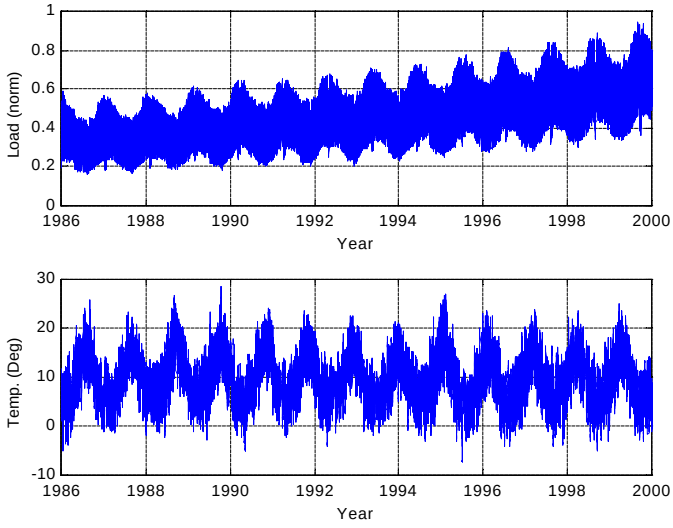

Figure 13. Load and temperature.

As pointed out in Section 3 the Box-Cox transform can be used to remove the variability of a time series if the variability is related to trend. This is the case with Irish electric load, so the Box-Cox transform is applied as in Section 3.

Secondly, the trend of the transformed load is removed using a quadratic curve of the form (1). This results in the transformed and *de-trended* (the trend is removed) load series shown in Figure 14.

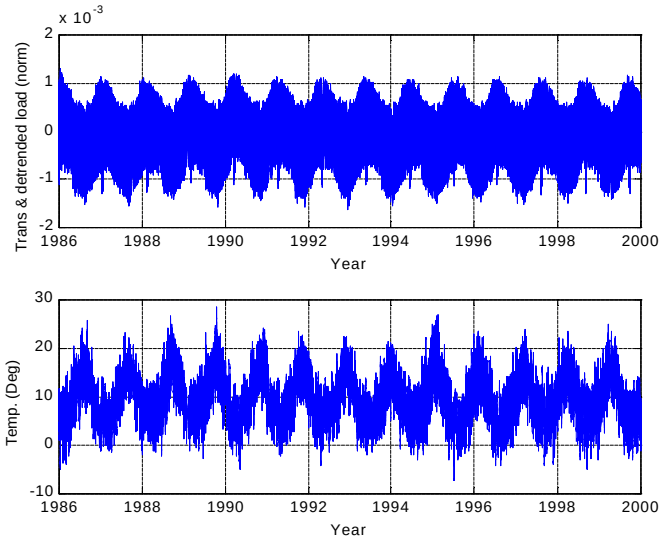

Figure 14. Temperature and transformed and de-trended load.

Finally, the transformed trend value at the end of year 2000 (the trend value of prime importance) to all the data. The resultant series has now got the equalised variability and the trend value of the year 2000 (Figure 15). This is known as *level shifting*.

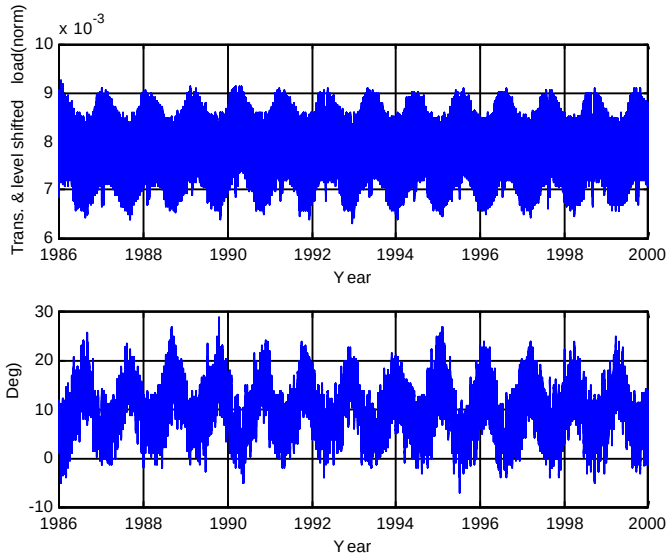

Figure 15 Temperature and transformed re-trended load.

The inverse Box-Cox transform (9) is then applied to the load series in Figure 15 to produce an equalised, level shifted and de-trended load series (Figure 16). The inverse Box-Cox transform can be easily derived from (3) as:

$$z'(t) = \begin{cases} (l\,z(t)+1)^{1/l} & \text{if } l \neq 0 \\ \exp(z(t)) & \text{if } l = 0 \end{cases} \qquad (9)$$

where $z'(t)$ is the inverse-transformed series at time t and $z(t)$ is the transformed series which has been further manipulated (de-trended and level shifted).

It should be noted that the *pre-processed* (equalised and de-trended) load in Figure 16 has the same value as the original load at the last data point (31st March 2000). The pre-processed load is an approximation to Irish load in a system without growth. Thus, the relationship between this and temperature is not obscured.

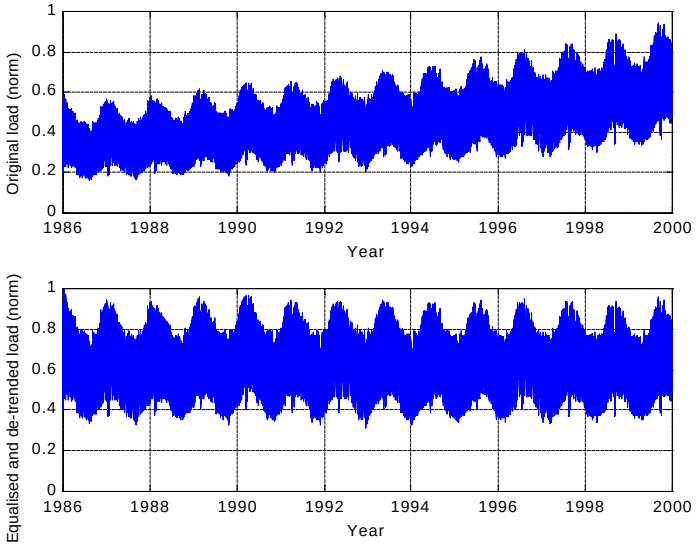

Figure 16. Original and equalised de-trended load (normalised).

Pre-Processing of Temperature and load

The temperature at a particular hour is rarely the only temperature input used when forecasting the load at that hour (Dash et. al., 1997, is an example of one exception). The load is typically aggregated in some fashion. Chen *et. al.* [20], for example, use the average temperature for the day in question while Hyde and Hodnett [17] use several different non-linear transformations of the current and previous temperatures, to name but a few. Before explaining the reason for using more than one temperature in predicting the load, a measure called *coherence* must be defined.

The coherence between a time series $x(t)$ and $y(t)$ is the correlation of the power in series x at frequency f with the power in series y at frequency f ([21]):

$$C_{xy}(f) = \frac{\left|E\left[P_{xy}(f)\right]\right|^2}{E\left[P_{xx}(f)\right]E\left[P_{yy}(f)\right]} \tag{10}$$

where $C_{xy}(f)$ is the coherence of x with y at frequency f, $P_{xx}(f)$ and $P_{yy}(f)$ are the power spectral densities of x and y respectively, $P_{xy}(f)$ is the cross power spectrum of x and y , E denotes the expectation operator and | | denotes the absolute value operator. The coherence can be used to indicate if certain frequencies in one series are related to those frequencies in another.

The coherence function (10) of temperature with pre-processed load demonstrates that pre-processed load is only correlated to temperature at daily and yearly frequencies (17). The high coherence at a 12-hourly period is due to the *twin peak shape* of the daily load curve caused by the daily maximum and minimum (4) as noted by Moutter *et. al.* [22] presence of a 12-hourly cycle in both the temperature and the load. The *high frequency* (time periods less than 12 Hours) components of temperature are not correlated with high frequency components of load. This indicates that high frequency components of temperature are not particularly useful in forecasting load on any (including hourly) basis, and justifies the use of some form of low-pass filtering or aggregation of temperature data.

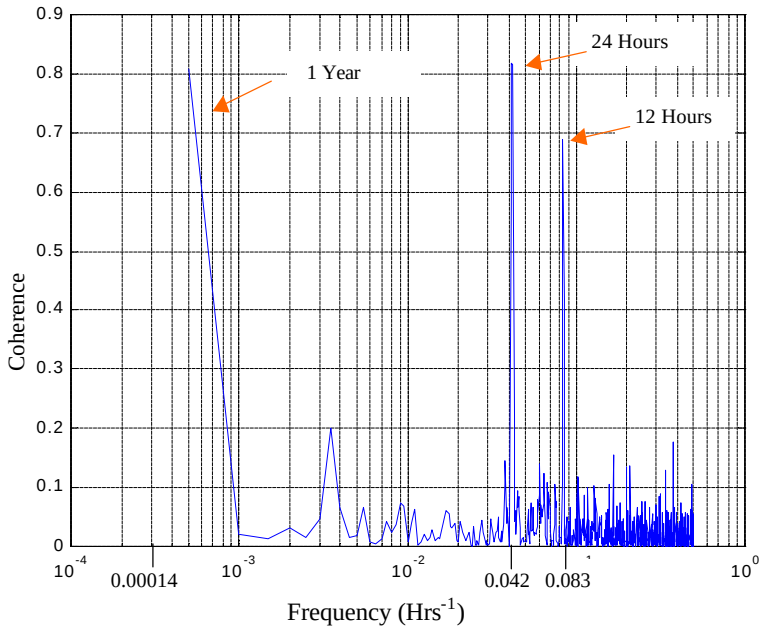

Figure 17. Coherence between pre-processed load and temperature.

The pre-processed temperature used in this analysis is the average temperature $t_{av}(k)$:

$$t_{av}(k) = \frac{1}{24}\sum_{i=0}^{23} t_{hr}(i,k) \tag{11}$$

where $t_{hr}(i,k)$ is the temperature at hour i on day k.

Similarly, the load is aggregated as:

$$z_{av}(k) = \frac{1}{24}\sum_{i=0}^{23} z'(i,k) \tag{12}$$

where $z_{av}(k)$ is the average load for day k and $z'(i,k)$ is the pre-processed load for hour i on day k.

Characterisation of the Temperature Load-Relationship

Typically, for an electrical system, the relationship between load and temperature is similar to that shown in Figure 18 [18,23]. At low temperatures (below 18 degrees for Ireland as discovered by Murray, [16]) the correlation between load and temperature is negative due to heating requirements. Above 18 degrees there is a *deadband* or comfort zone in which the load is largely unresponsive to the temperature [23]. Beyond this deadband, as the temperature rises, the load again increases due to air-conditioning [23]. As the temperature in Ireland rarely exceeds the mid-twenties Murray [16] did not identify any cooling effect.

It should, however, be noted that the temperature-load relationship is not the same for all electrical systems. Murray [16] pointed out that the relationship was significantly different for a regional power board in Northern New Zealand where temperature was a non-dominant input.

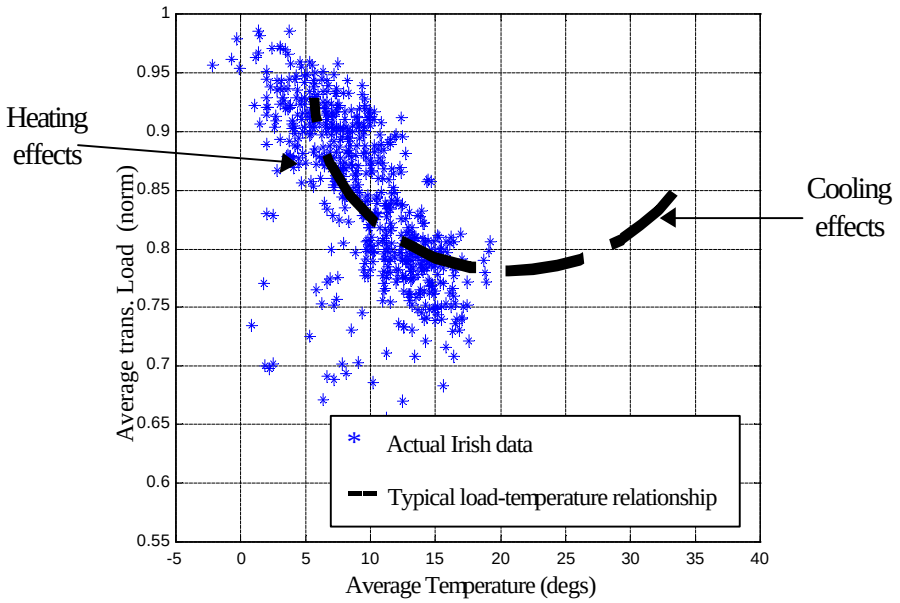

Figure 18 Typical and actual scatter plot of temperature-load relationship (Working days).

Figure 19 shows the load-temperature relationship for working days with data points for each month of the year highlighted. The data points for each month group into clusters with a few exceptions due to Christmas, bank holidays and exceptional days (Figure 19).

In order to compare the load-temperature relationship at different times of the year regression lines are employed (Figure 19). As pointed out by Murray [16], Hyde and Hodnett [17] and Fan and McDonald [23], among others, the temperature-load relationship is non-linear and so it should be noted that the use of regression lines is an approximation. However, the approximation is considered sufficient to identify significant differences in the load-temperature relationship for different months of the year.

In calculating the regression lines, Christmas, bank holidays and exceptional days are excluded to avoid them influencing the line parameters. Several algorithms exist for selecting which points to include and exclude for calculating regression lines (several techniques are listed in [24]). The BACON (Blocked Adaptive Computationally Efficient Outlier Nominators) method proposed by Billor *et. al* [25] was chosen for its fast convergence.

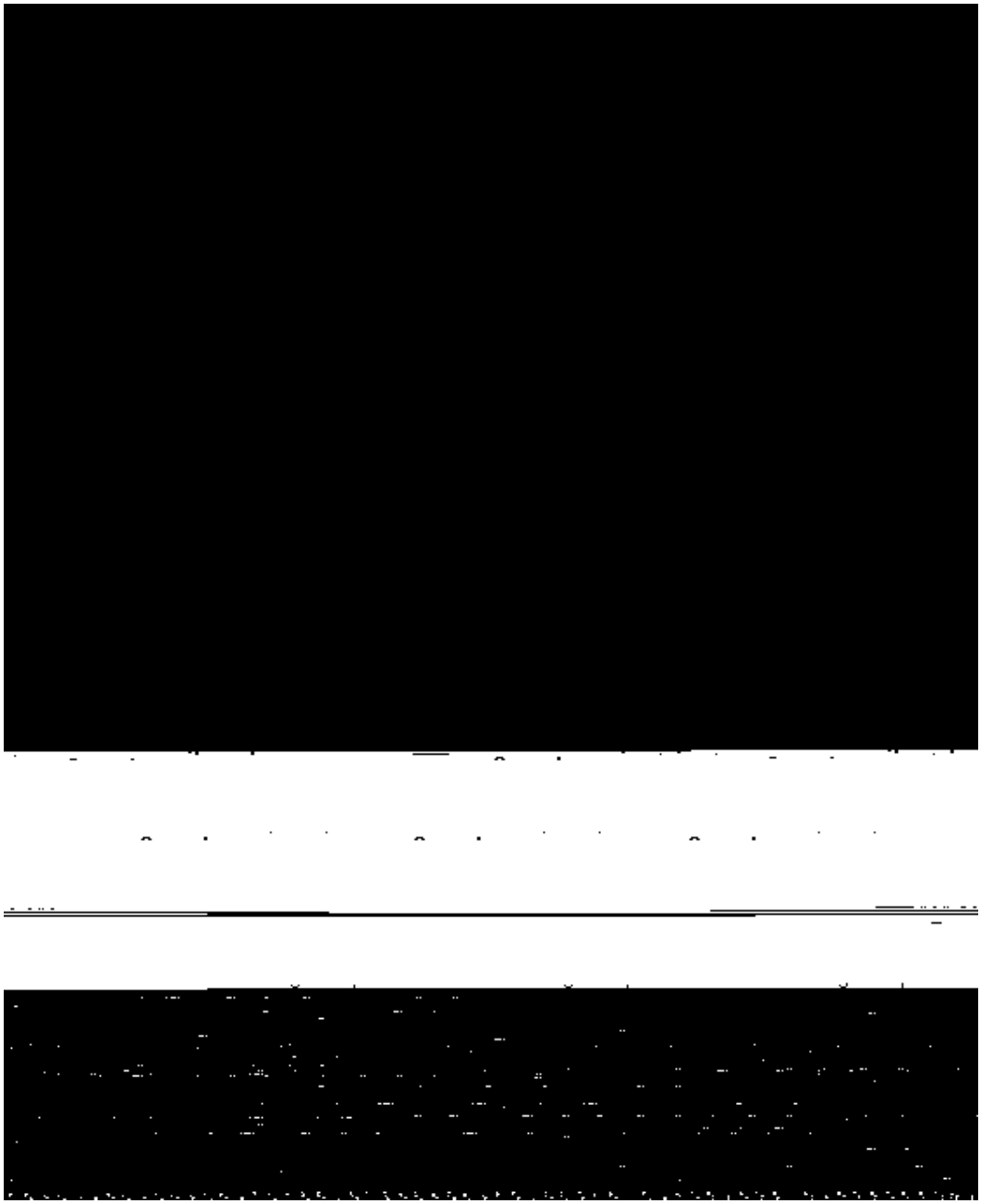

Figure 19. Scatter plot with regression lines for temperature-load relationship by month (Working days).

The load-temperature relationship is different for each month of the year (Table 5). For January to April, the slopes of the regression lines are negative and similar. From May to June the slopes approach zero. The low correlation co-efficients for July to October indicate very little correlation between temperature and load in those months. The regression lines and correlation co-efficents for November and December are similar to those in January. With respect to the day-types identified in Section 4, this indicates that there is no significant difference in the temperature-load relationship within the periods spanned by the late Winter working day day-type (October to March). The Summer working day day-type (spanning April to September), however, has a significantly different temperature-load *sensitivity* (defined as the slope of the regression line) in April and May than in the period June-September. The early Winter working day day-type (spanning October to December, excluding Christmas) also has a different sensitivity in October than in November-December.

The load-temperature sensitivities for Saturday and Sunday day-types show a similar result to that of working days (Table 4).

Table 4. Temperature-load regression line co-efficients and correlations.

		Jan.	Feb.	Mar.	Apr.	May	June	July	Aug.	Sept.	Oct.	Nov.	Dec.
Working Days	Slope (β_1)*	-5.42 E-03	-6.20 E-03	-5.72 E-03	-5.69 E-03	-2.19 E-03	0.559 E-03	1.81 E-03	-10.9 E-03	-0.823 E-03	1.15 E-03	-3.60 E-03	-6.13 E-03
	Intercept (β_2)*	9.63 E-01	9.55 E-01	9.39 E-01	9.00 E-01	8.29 E-01	7.75 E-01	7.43 E-01	9.23 E-01	8.13 E-01	8.31 E-01	9.26 E-01	9.72 E-01
	Correlation between temperature and load	-0.53	-0.49	-0.65	-0.48	-0.14	0.25	-0.20	0.02	-0.11	-0.07	-0.63	-0.49
Sundays	Slope (β_1)*	-5.94 E-03	-4.39 E-03	-6.77 E-03	-5.02 E-03	-2.40 E-03	5.94 E-03	1.87 E-03	-1.33 E-03	1.18 E-03	-2.29 E-03	-4.37 E-03	-3.78 E-03
	Intercept (β_2)*	8.00 E-01	7.67 E-01	7.70 E-01	7.30 E-01	6.87 E-01	5.56 E-01	5.86 E-01	6.47 E-01	6.34 E-01	7.02 E-01	7.80 E-01	8.05 E-01
	Correlation between temperature and load	-0.28	-0.65	-0.37	-0.04	-0.56	-0.38	0.26	-0.18	-0.35	-0.15	-0.42	-0.39
Saturdays	Slope (β_1)*	-2.74 E-03	-4.40 E-03	-4.86 E-03	-3.04 E-03	-2.71 E-03	-1.55 E-03	1.95 E-03	-1.69 E-03	5.34 E-04	-1.36 E-03	-2.06 E-03	-2.83 E-03
	Intercept (β_2)*	8.47 E-01	8.48 E-01	8.32 E-01	7.59 E-01	7.58 E-01	7.25 E-01	6.56 E-01	7.18 E-01	7.07 E-01	7.68 E-01	8.31 E-01	8.66 E-01
	Correlation between temperature and load	-0.32	-0.38	-0.45	-0.61	-0.83	-0.19	-0.64	0.20	0.46	-0.18	-0.73	-0.68

6. Establishment of a Solution Strategy

This section lays out the blueprint for a load-forecasting *package* (a set of models that can be used to forecast all day-types) in light of the analysis in Sections 3,4,5. As pointed out in Section 4, by modelling each day-type separately the information that each model must incorporate is reduced. This aids in the modelling task. There is however a trade-off. As pointed out by Hippert *et al*. [26] in a study of neural networks applied to the load-forecasting problem, if the data is subdivided into too many day-types then the resulting data sets are too small to permit adequate model training.

Segmentation of Data for Model Building

The segmentation of the data set (by the day-types chosen in Section 4) is shown in Figure 20 with the approximate number of days in each set shown. As the boundaries between day-types is not always *hard* (with no overlap), the number of days in each day-type is approximated.

* For normalised data.

The amount of data required to allow modelling is difficult to quantify as it is influenced by the type of model used and the amount of noise in the data, among other things. To determine if the amount of data in each subset is sufficient to allow modelling, comparisons are drawn with other studies. Hippert *et al*. [26] gives a list of the data set sizes used in twenty-two studies by various authors. The size of the *total* data sets used in each study varies from thirty-four days to four years. The *total* size of the data set in the current study is thirteen years, which is relatively large. This allows a greater level of segmentation, if desired.

The amount of data in *each day-type*, for the current study, is given in Figure 20 and shows that for the day-types chosen, the amount of data is similar to that used by Sharaf and Lie [27], Srinivasan *et al*[8] to mention but a few. Sharaf and Lie [27] used three months of data for each of their day-types and Srinivasan *et al*. [8] used two years of data broken into 3 day-types with approximately three hundred days in the working and Saturday day-type and fifty days in the Sunday day-type.

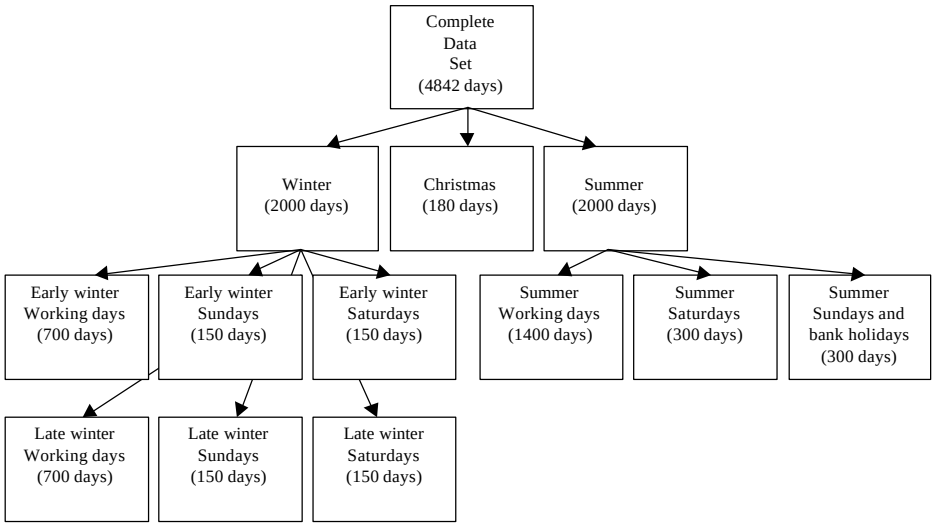

Figure 20. Segmentation of data set with approximate number of days in each subset.

Interpolation between Model Boundaries

[28] demonstrated that for Irish weekly demand there is a transitional period between Winter and Christmas. In Figure 12, the nodes triggered by Summer and Winter-Saturday loads in the current study (Section 4) clearly show an overlap.

Analytical techniques, such as clustering algorithms [19] and Kohonen maps [4], have been used to determine the transitions between day-types. However, Kohonen maps are not ideal for this task. Consider a day lying in the transition between Summer and Winter. This day will trigger a node in between the Summer and Winter regions of the output nodes. However, as pointed out by Song and Hopke [30] the relative position of two nodes in a Kohonen Map is not an exact measure of the proximity between the inputs that trigger those nodes.

Cloarec *et al.* [28] and Rahman and Bhatnagar [31] determine the transition between two seasons using model forecast errors. The load in question (weekly in [28] study and hourly in [31]) is forecast using both seasonal models and the ratio of model errors is used to determine the transition. This technique is desirable for any online system, as it tracks the errors of that system. However, it is dependant on the performance of load forecasting models and not the characteristics of the data.

Cluster analysis methods are used mainly as a first step in designing a fuzzy logic system based on the data as opposed to *a priori* information [14]. There are several types of clustering algorithms such as K-means clustering, nearest neighbour, polynomial interpolants and adaptive vector (for an overview see [32]). Other algorithms such as the adaptive fuzzy classification algorithm suggested by

Bretschneider *et al*. [5] can be adaptive and change with the dynamic behaviour of the data. By far, the most popular clustering algorithm is the Fuzzy C-Means (FCM) algorithm [33]. As pointed out by Jain and Dubes [14], there are no guidelines for *a-priori* choice of the correct algorithm based on the data. As the FCM algorithm is the most popular it is chosen in this study.

The FCM algorithm seeks to break the data into c_n clusters. Each cluster is characterised by a *cluster centre* $\boldsymbol{C}_{i,} \in \mathbf{R}^{1\times Q}$, which represents a point at the centre of the cluster and Q is the dimension of the inputs. Given an input point $\boldsymbol{P}_{k,} \in \mathbf{R}^{1\times Q}$, instead of assigning it to a single cluster its proximity to all clusters is quantified by a *potential* defined as:

$$u_{j,k} = L(\boldsymbol{P}_k, \boldsymbol{C}_j)^{\frac{-2}{m-1}} \Big/ \sum_{i=1}^{c_n} L(\boldsymbol{P}_k, \boldsymbol{C}_i)^{\frac{-2}{m-1}} \qquad j = 1...c_n \tag{13}$$

where $u_{j,k,} \in [0,1]$, is the potential of $\boldsymbol{P}_k$ to cluster centre $\boldsymbol{C}_j$, m is a scalar which determines the level of *fuzziness* (explained below) of the resulting potentials, and $L(\boldsymbol{P}_k,\boldsymbol{C}_j)$ is the distance between $\boldsymbol{P}_k$ and $\boldsymbol{C}_j$ defined as:

$$L(\boldsymbol{P}_k, \boldsymbol{C}_j) = \sqrt{\sum_{i=1}^{Q} \left(\boldsymbol{P}_k(i) - \boldsymbol{C}_j(i)\right)^2} \tag{14}$$

where $\boldsymbol{P}_k(i)$ and $\boldsymbol{C}_j(i)$ are the i^{th} elements of $\boldsymbol{P}_k$ and $\boldsymbol{C}_j$ respectively.

The level of *fuzziness* determines the degree of overlap between clusters. A value of one for m leads to no overlap. As m is increased above one the overlap is increased.

In addition to Equations 13 and 14, a further condition is that the sum of potentials for $\boldsymbol{P}_k$ equals unity:

$$\sum_{i=1}^{c_n} u_{i,k} = 1 \tag{15}$$

It should be noted that although $u_{j,k}$ has similar characteristics as the probability that $\boldsymbol{P}_k$ is a member of cluster $\boldsymbol{C}_j$ (i.e. $u_{j,k,} \in [0,1]$ and Equation 2.18), the two measures are not the same [14]. The potential represents the *membership* (degree to which an input can be classified as a member of a cluster) of an input to a cluster. While an input can have membership of two or more clusters the probability that it is a member of a cluster implies that it belongs to one cluster alone. Thus, the potential is an ideal measure of the membership of a day to a certain day-type.

To apply the algorithm, the cluster centres, $\boldsymbol{C}_{j,}$ and level of fuzziness, m, must be determined. The aim is to minimise overall proximity of the data to the cluster centres. This is achieved by minimising the cost function:

$$J = \sum_{l=1}^{N} \sum_{i=1}^{c_n} (u_{i,l})^m L(\boldsymbol{P}_l, \boldsymbol{C}_i) \tag{16}$$

where J is the cost function to be minimised. The algorithm is initialised using random potentials. The cluster centres are then calculated via:

$$C_j = \sum_{i=1}^{N} u_{j,i}{}^m P_i \Big/ \sum_{i=1}^{N} u_{j,i}{}^m \tag{17}$$

The potentials are then updated *via* (13) and the new cluster centres and value of J are calculated via (17). This process continues until the value of J reaches a pre-specified level or the number of iterations reaches a pre-specified maximum.

In the current study, the inputs $\boldsymbol{P}_k$ are composed of the twenty-four hours of data, normalised using (6). For the current analysis, only data from the years 1999 and 2000 are used. This to avoid the clustering algorithm identifying the differences between years as opposed to day-types. The desired level of fuzziness depends on the accuracy of load-forecasting models. However, at this stage, a value of two for m was found to give a good indication of the transitions between the day-types.

As the data has already been segmented in Section 4, the existence of the clusters is already known. The data is thus segmented into pairs of known clusters, as only two day-types overlap at any time. FCM was then used to determine the transition between the two. These segmented pairs are:

- Early winter working day and Christmas day-types,
- Christmas and late Winter working day day-types,
- Late winter and Summer working day day-types and
- Summer and Early Winter working day day-types.

The transition between early Winter working day day-type and Christmas day-types occurs over the period of several days, however the two periods are distinct (Figure 21).

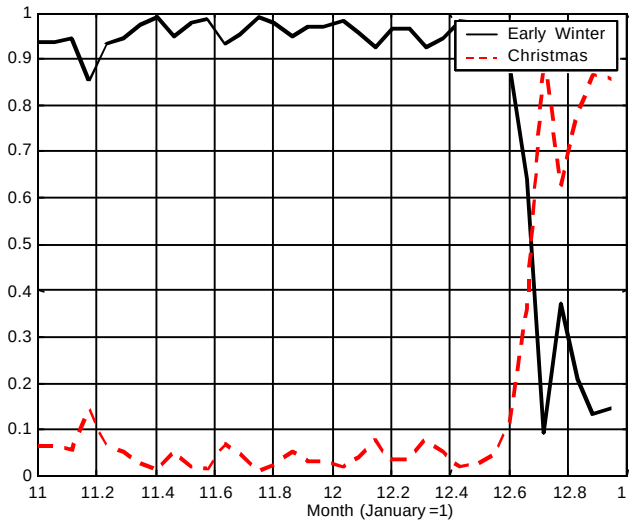

Figure 21 Early Winter (working day) day-type to Christmas day-type transition.

Recall that a Winter day-type is identified in Section 4. However, as the Winter day-type is disjoint due to the Christmas period, the early and late Winter day-types were created. Figure 22 shows the transitions between *Winter day-type* data and Christmas. It is interesting to note, as expected, that the transition into Christmas corresponds to that in Figure 22. The transition between Christmas and late Winter working day day-types also occurs over several days (Figure 22).

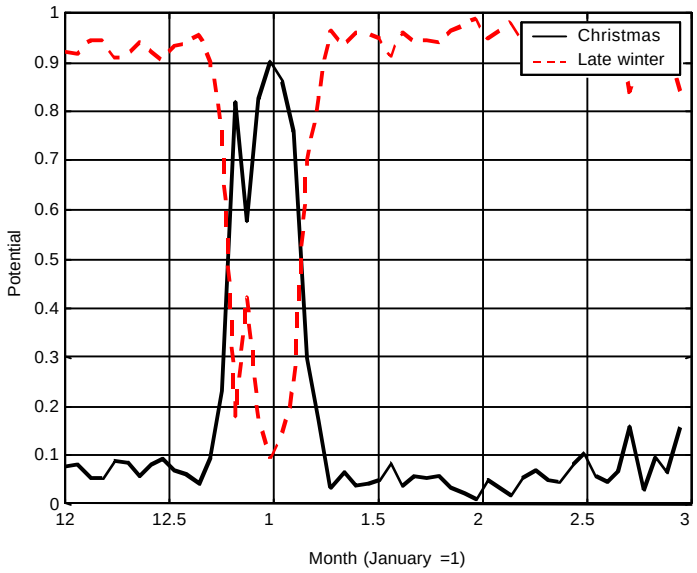

Figure 22. Christmas day-type to late Winter (working day) day-type transition.

The transition from late Winter to Summer occurs over the hourly changes for daylight savings. Figure 23 shows the load for Friday 25th March 1999 (before the change) and Tuesday the 30th March (after the change). The load profiles for these days are significantly different due to the fact that there is a lower lighting requirement on Tuesday 30th March.

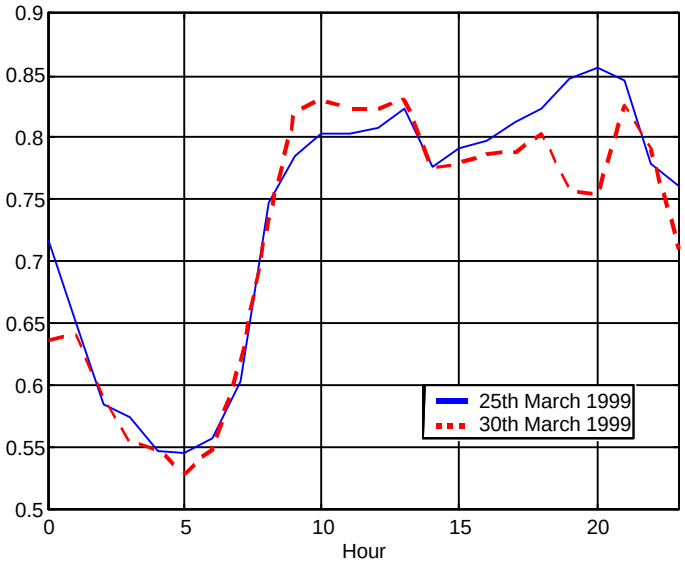

Figure 23. Differing load profiles due to day-light savings changes.

As a consequence of the changes for day-light savings the transition between late Winter working day day-types and Summer working day day-types is hard (Figure 24).

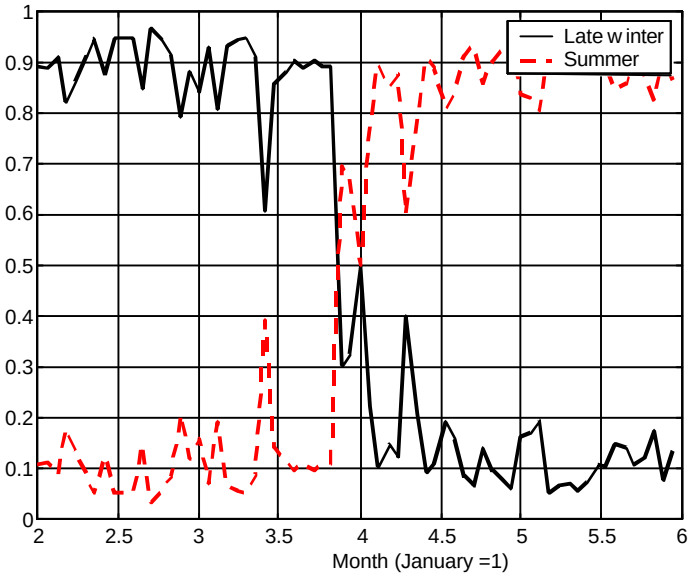

Figure 24. Late Winter to Summer (working day) day-type transition.

Finally the transition between Summer and early Winter working day day-types is shown in Figure 25. This transition is fuzzy demonstrating that there is a significant overlap between the two periods. The day-light savings change does not have the same effect here as lighting requirements are not influential in Summer.

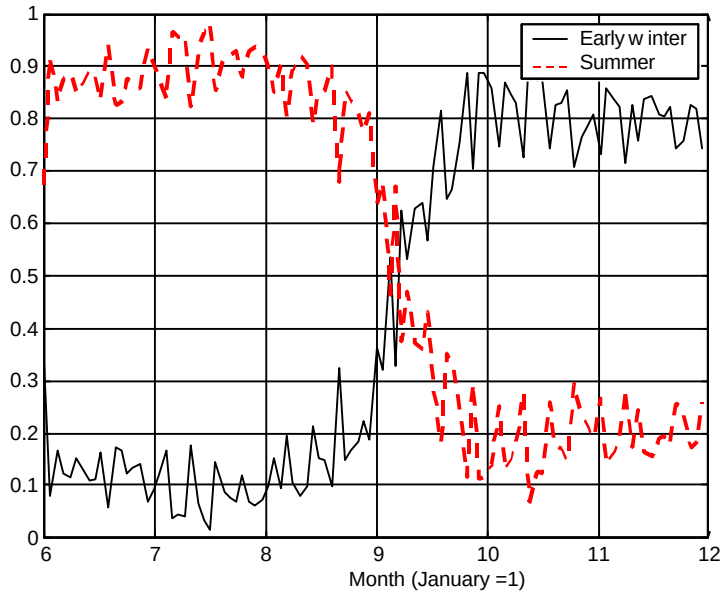

Figure 25. Summer to early Winter (working day) day-type transition.

7. Conclusion

This paper demonstrates that Irish electrical load data has distinct characteristics, which have implications for model building. The blueprint for a load-forecasting package is laid out in consideration of these characteristics without biasing the choice of load forecasting models used.

Firstly, the data has a trend that increases at an increasing rate. Also, the variability of the load increases and is *related* to the trend. This indicates that load is a dynamic series which changes over time but in a *predictable* way. Indeed, the simple methods used to remove the trend and variability of the load in Section 3, although not perfect, are quite effective. The forecast horizon that is the objective of in this body of research is three days. Relative to this, the rate of change of the approximated trend is very low at 1.13 e-011 Mw/hour* (Table 2) and removing the trend and variability can be achieved with a high level of accuracy.

The *shape* of the load on different days was examined in Section 4 and it was found that load can be classified into different day-types. These day-types correspond to the days of the week in different seasons of the year. Exceptional days, such as bank holidays, are found to have a similar shape to Sundays while the Christmas period is treated separately. The fact that the daily load shape is not consistent is important for modelling as it requires that the model must incorporate shape information.

The relationship of temperature (the dominant causal variable to load) to load is examined in Section 5. The results (Table 4) show that the load-temperature relationship for early Winter day-types is constant within the range of those day-types. Thus, when forecasting the load, the operating point on the load-temperature curve is not significant. In contrast, this is not true for the other day-types. For these other day-types, the operating point on the load-temperature curve is significant and must be considered when modelling these loads.

Section 6 outlines the consequences of applying a different model to each day-type. While this reduces complexity in that the shape of the load in each day-type is consistent, the data is more segmented, resulting in several smaller data sets. In this respect, it is the only modelling decision made in this Chapter. However, similar studies (see Section 6) have shown that the segmented data set sizes are not too small to restrict the type of model.

Finally, the transitions between the different day-types is highlighted in Section 6. For some day-types there is a significant overlap indicating that some days are members of two day-types. As the data has already been segmented and the number of models decided, the load forecast in these overlapping days is required to be composed of the output of the two corresponding models.

References

[1] Box, G.E.P., Cox, D.R., 1964, An analysis of transformations, *Journal of the Royal Statistical Society*, 26 (B), 211-243.

[2] Franses, P.H., Koehler, A.B., 1998, A model selection strategy for time series with increasing seasonal variation, *International Journal of Forecasting*, 14, 405-414.

[3] Cryer, J.D., 1986, Time Series Analysis, USA, Boston: Duxbury Press.

[4] Muller, H., Petrisch, G., 1998, Energy and load forecasting by fuzzy-neural networks. *In*: Jurgen, H., Zimmermann, H.J., eds., *Proceedings, European Congress on Intelligent Techniques and Soft Computing, Aachen, Germany, September* 1998. Aachen: Elite foundation, 1925-1929.

[5] Bretschneider, P., Rauschenbach, T., Wernstedt, J., 1999, Forecast using an adaptive fuzzy classification algorithm for load, *6th European Congress on Intelligent Techniques & Soft Computing*, Vol.3, pp 1916-1919.

[6] Hsu, Y.Y. , Yang, C.C., 1991, Design of artificial neural networks for short-term load forecasting Part I: Self-organising feature maps for day type identification, *IEE Proceedings-C*, 138(5), page 407-413.

[7] Hubele, N.F., Cheng, C.S., 1990, Identification of seasonal short-term forecasting models using statistical decision functions, *IEEE Transactions on Power Systems*, 5 (1), 40-45.

[8] Srinivasan, D., Tan, S. S., Chang, C. S., Chan, E. K., 1999, Parallel neural network-fuzzy expert system for short-term load forecasting: system implementation and performance evaluation, *IEEE Transactions on Power Systems*, 14 (3), 1100-1106.

[9] Mastorocotas P.A., Theocharis, J.B., Bakirtzis, A.G., 1999, Fuzzy modelling for short term load forecasting using the orthogonal least squares method, *IEEE Transactions on Power Systems*, 14 (1), 29-35.

[10] Chen, S.T., Yu, D.C., Moghaddamjo, A.R., 1992, Weather sensitive short-term load forecasting using non-fully connected artificial neural network, *IEEE Transactions on Power Systems*, 7 (3), 1098-1104.

* This figure is normalised relative to a maximum load of 1Mw.

[11] Lertpalangsunti, N., Chan, C.W., 1998, An architectural framework for the construction of hybrid intelligent forecasting systems: application for electricity demand prediction., *Engineering Applications of Artificial Intelligence*, 11, 549-565.

[12] Ho, K.L., Hsu, Y.Y., Liang, C.C., Lai, T.S., 1990, Short term load forecasting of Taiwan power system using a knowledge-based expert system, *IEEE Transactions on Power Systems*, 5 (4), 1214-1221.

[13] Lonergan, T., Ringwood, J.V., 1995, Linguistic modelling of electricity consumption using fuzzy modelling techniques, *In: proceedings, Irish DSP and Control Colloquium, Belfast, UK, June* 1995.

[14] Jain, A.K, Dubes, R.C., 1998, *Algorithms for Clustering Data*, N.J.: Prentice Hall.

[15] Kohonen , T., 1990, The self-organising map, *Proceedings IEEE*, 78 (9), September 1990.

[16] Murray, F., 1996, Forecasting methodologies for electricity supply systems, Dublin: *Ph.D. thesis*, School of Electronic Engineering, Dublin City University, Ireland.

[17] Hyde, O., Hodnett, P.F., 1997a, An adaptable automated procedure for short-term electricity load forecasting, *IEEE Transactions on Power Systems*, 12 (1), 84-94.

[18] Lu, Q.C., Grady, W.M., Crawford, M.M., Anderson, G.M., 1989, An adaptive nonlinear predictor with orthogonal escalator structure for short-term load forecasting, *IEEE Transactions on Power Systems*, 4 (1), 158-164.

[19] Chen, S.L., Kao, F.C., 1996, An efficient algorithm to model and forecast hourly weather sensitive load, *Journal of the Chinese Institute of Electrical Engineering*, 3(3), 231-243.

[20] Chen, D., Chen, B., Li, T., 1991, An expert system for short-term load forecasting. *In: Proceedings, IEE Conference on Advance in Power System Control, Operation and Management, Hong Kong, November* 1991. Hong Kong: IEE, 330-334.

[21] Brockwell, P.J., Davis, R.A., 1987, *Time Series: Theory and Methods*, New York USA: Springer Verlag Inc.

[22] Moutter S.P., Bodger, B.E., Gough, P.T. 1986, Spectral decomposition and extrapolation of variations in electricity loading, *IEE Proceedings*, 113 (C), 247-255.

[23] Fan, J.Y., McDonald, J.D., 1994, A real-time implementation of short-term load forecasting for distribution power systems, *IEEE Transactions on Power Systems*, 9 (2), 988-994.

[24] Wisnowski, J.W., Montgomery, D.C., Simpson, J.R., 2001, A comparative analysis of multiple outlier detection procedures in the linear regression model, *Computational Statistics and Data Analysis*, 36, 351-382.

[25] Billor, N., Hadi, A.S., Velleman, P.F., 2000, BACON: blocked adaptive computationally efficient outlier nominators, *Computational Statistics and Data Analysis*, 34, 279-298.

[26] Hippert, S.H., Pedriera, C.E., Souza, R.C., 2001, Neural networks for short-term load forecasting: a review and evaluation, *IEEE Transactions on Power Systems*, 16 (1), 44-55.

[27] Sharaf, A.M., Lie, T.T., 1995, A novel neuro-fuzzy based self-correcting online electric load forecasting model, *Electric Power and System Research*, 34, 121-125.

[28] Cloarec, G.M., Ringwood, J.V., Kelly, M.V., 1998, Neuro-fuzzy forecasting of weekly electricity consumption. *In*: Jurgen, H., Zimmermann, H.J., eds., *Proceedings, European Congress on Intelligent Techniques and Soft Computing, Aachen, Germany, September 1998*. Aachen: Elite foundation, 1939-1939.

[29] Imai, H., Tanaka, A., Miyakoshi, M., 1998, A method of identifying influential data in fuzzy clustering, *IEEE Transactions on Fuzzy Systems*, 6 (1), 90-101.

[30] Song, X.H., Hopke, P.K., 1996, Kohonen neural network as a pattern recognition method based on weight interpretation, *Analytica Chimica Acta*, 334, 57-64.

[31] Rahman, S., Bhatnagar, R., 1988, An expert system based algorithm for short-term load forecasting, *IEEE Transactions on Power Systems*, 3 (2), 392-399.

[32] Harris, C.J., Moore, C.G., Brown, M., 1993, *Intelligent control, aspects of fuzzy logic and neural nets*, World scientific series in robotics and automated systems.

[33] Dunn, J.C., 1974, A fuzzy relative of the ISODATA process and its use in detecting compact well separated clusters, *Journal of Cybernetics*, 3 (3), 35-57.