Modelling household electricity demand

Euan Enticott, Georgina Mansell, and Conor Newton

May 20, 2021

Abstract

1 Introduction

2 Data

A dataset of residential electricity demand was collected as part of a customer behaviour trial in 2010 in Ireland. Electricity consumption was recorded for 2672 households at half hour intervals over the year.

Figure 1 shows the total electricity demand profile for the households. There is a characteristic pattern of low demand at night, a low peak around 7am and a higher peak around 6pm. There are some seasonal effects where demand is higher in the winter months, likely due to increased heating and lighting. The profile also differs on weekends where there is no dip in demand during typical work hours.

While there is a very clear pattern to the demand at an aggregated level, individual customer profiles are much more varied.

The dataset comes with additional information such as the external temperature, the year the house was built, and whether the house has electric heating.

The dataset is available at `https://github.com/mfasiolo/electBook/blob/master/data/Irish.RData`, and more information about the original study can be found at `http://www.ucd.ie/issda/data/commissionforenergyregulationcer/`.

3 Methods

All code is available on GitHub at `github.com/eenticott/electricity-modelling`.

3.1 Clustering

3.2 Ridge Regression

Ridge regression is a commonly used regularisation technique for least squares regression. In the standard setting we have a dataset of n samples each with a vector of p covariates $\mathbf{x}_i$ and scalar response y_i. The terms are collected in design matrix $\mathbf{X} \in \mathcal{R}^{n \times p}$ and response vector $\mathbf{y} \in \mathbb{R}^n$. Typically the matrix $\mathbf{X}$ is standardised so that each variable has a zero mean and unit variance.

Ordinary least squares aims to find the vector of parameters $\beta \in \mathbb{R}^p$ which minimises the residual sum of squares:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \, ||\mathbf{y} - \mathbf{X}\beta||^2$$
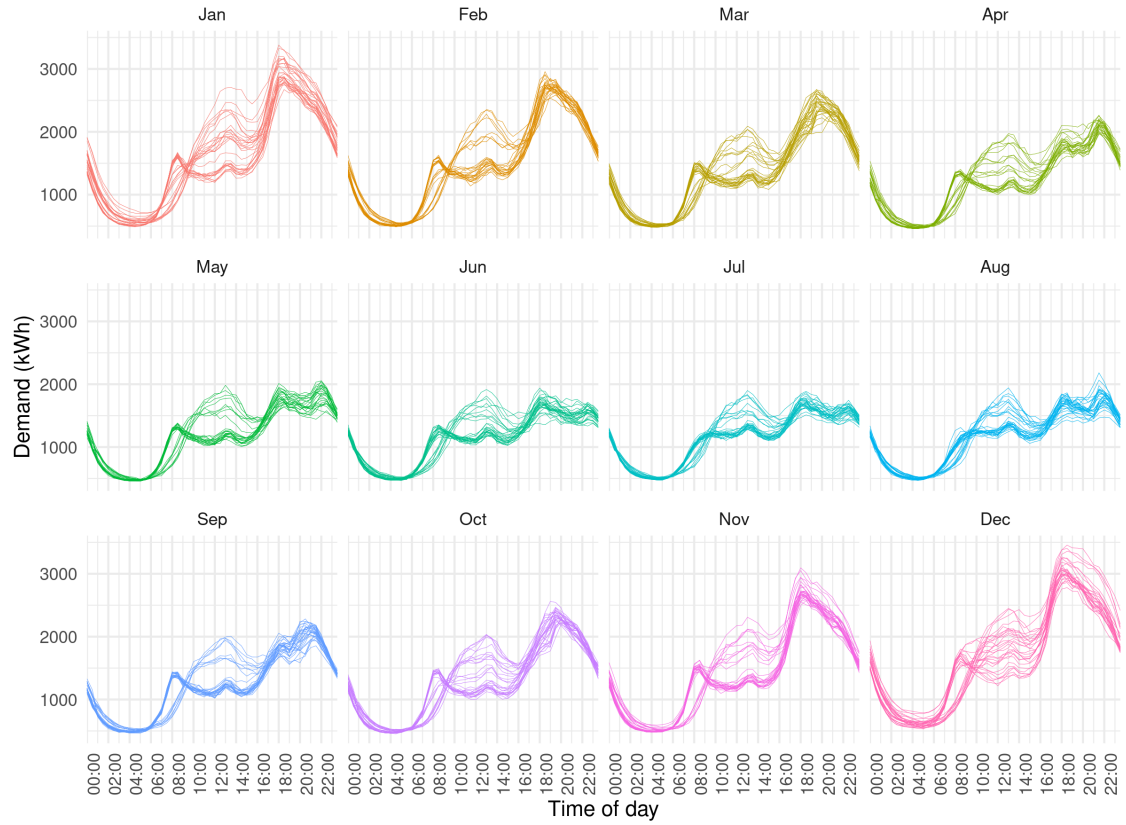
Figure 1: Total electricity demand over all recorded households; each line shows the usage over one day, split over subfigures by month.

However, OLS is susceptible to over-fitting; as the number of variables p increases, so does the model variance. Additionally, if $p > n$ or if there is multicollinearity between variables, the matrix $\mathbf{X}^T\mathbf{X}$ is not invertible.

Ridge regression can overcome both of these issues by placing an L_2 penalty term on the parameter vector β:

$$\hat{\beta} = \underset{\beta}{\mathrm{argmin}}\, ||\mathbf{y} - \mathbf{X}\beta||^2 + \lambda||\beta||^2$$

Here $\lambda \geq 0$ is a hyperparameter which controls the strength of the regularisation and which should be selected through a cross validation procedure.

Solving for β analytically gives:

$$\hat{\beta} = (\lambda\mathbf{I} + \mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$$

- Ridge regression with factor variables?

3.3   Selecting λ

A simple way to select lambda is to calculate the mean square error on a held out portion of the dataset, then select the value of λ which minimises the MSE. An approach that is less sensitive to the specific choice of test data is leave-one-out cross validation (OCV). This is calculated as:

$$\text{OCV} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\mu}_i^{[-i]} \right)^2$$

OCV can also be re-written as:

$$\text{OCV} = \frac{1}{n} \sum_{i=1}^{n} \frac{(y_i - \hat{\mu})^2}{(1 - A_{ii})^2}$$

which is less computationally expensive, as only one model needs to be fit per λ, not n.
    - GCV is apparently better?

3.4   Bayesian Perspective

Ridge regression can also be considered from a Bayesian perspective; we assume that the responses y are generated by the following process:

$$y = \beta \mathbf{x} + \epsilon \quad \text{where} \quad \epsilon \sim \mathcal{N}(0, \sigma^2)$$

Equivalently

$$\mathbf{y} \sim \mathcal{N}(\mathbf{X}\beta, \mathbf{I}\sigma^2)$$

We could t

3.5   R Package

Algorithm 1: Algorithm1

Data: ...
Result: ...
Text
repeat
 | $a = b$
until *convergence*
for *i=1, 2, ...* **do**
 | Initialise b
 | $a = b$
end

---

4   Results

5   Conclusion