NZ GREEN Grid Household Power Demand Profiles: Hot Water

Data extraction and preliminary plots

Ben Anderson (b.anderson@soton.ac.uk, `@dataknut`)

Last run at: 2018-07-26 17:40:22

Contents

1 Status **2**

2 Citation **2**

3 Introduction **3**
- 3.1 Purpose . . . 3
- 3.2 Requirements: . . . 3
- 3.3 History . . . 3
- 3.4 Support . . . 3

4 Load data files **3**
- 4.1 Grid Spy metadata . . . 3
- 4.2 Grid Spy data . . . 4
- 4.3 Test Hot Water data . . . 5

5 Hot Water profiles **9**
- 5.1 Profile plots: means per household . . . 9
- 5.2 Profile plots: overall household mean . . . 11

6 Runtime **13**

7 R environment **13**

References **14**

1 Status

Full run using all data from /Volumes/hum-csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/

2 Citation

If you wish to use any of the material from this report please cite as:

- Anderson, B. (2018) GREEN Grid Heat Pump Profiles, University of Otago: Dunedin, NZ.

3 Introduction

Report circulation:

- Restricted to: NZ GREEN Grid project partners and contractors.

3.1 Purpose

This report is intended to:

- load and clean the project electricity power data (Grid Spy)
- select the Heat Pump circuits (via their labels)
- build exploratory demand profiles

3.2 Requirements:

- cleaned and safe grid spy 1 minute data processed via https://git.soton.ac.uk/ba1e12/nzGREENGrid/blob/master/dataProcessing/processNZGGElecCons1minData.Rmd

3.3 History

Generally tracked via our git.soton repo:

- history
- issues

3.4 Support

This work was supported by:

- The University of Otago
- The New Zealand Ministry of Business, Innovation and Employment (MBIE)
- SPATIALEC - a Marie Skłodowska-Curie Global Fellowship based at the University of Otago's Centre for Sustainability (2017-2019) & the University of Southampton's Sustainable Energy Research Group (2019-202).

We do not 'support' the code but if you have a problem check the issues on our repo and if it doesn't already exist, open one. We might be able to fix it :-)

4 Load data files

4.1 Grid Spy metadata

In this section we load metadata from /Users/ben/Syncplicity Folders/Green Grid Project Management Folder/Gridspy/Master list of Gridspy units.xlsx to link to the power data.

sample	hhID	Adults	Teenagers	Children	removed	nAdults
Powerco	rf_06	2	NA	NA	NA	2
Powerco	rf_07	2	NA	2	NA	2

sample	hhID	Adults	Teenagers	Children	removed	nAdults
Powerco	rf_08	2	NA	NA	NA	2
Powerco	rf_09	2	NA	1	42171	2
Powerco	rf_10	2	NA	1(3yo)	NA	3
Powerco	rf_11	NA	NA	NA	NA	1

4.2 Grid Spy data

In this section we load the cleaned data files from /Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/data/. If we loaded all the data at once and then filtered out what we want we might run out of memory so we filter as we load. Set the filters here:

```
circuitPattern <- params$circuitPattern
dateFrom <- params$dateFrom
dateTo <- params$dateTo

plotCaption <- paste0("Source: ", fpath,
                      "\nCircuits: ", circuitPattern, " from ", dateFrom, " to ", dateTo)
```

So we are looking for Hot Water circuits between 2015-04-01 and 2016-03-31. We do this by checking to see if the extract file has already been created. If so we load it. If not, we create it.

Loading: /Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/dataExtracts/Hot Water_2015-04-01_2016-03-31_observations.csv.gz (needs to exist!)

The following table summarises the Hot Water data we have found.

	hhID	r_dateTime	circuitLabel	circuitID	powerW
	Length:14493877	Min. :2015-04-01 00:00:00	Length:14493877	Min. :1574	Min. :-1110.0
	Class :character	1st Qu.:2015-06-22 16:46:00	Class :character	1st Qu.:2248	1st Qu.: 0.0
	Mode :character	Median :2015-09-18 21:32:00	Mode :character	Median :4135	Median : 0.0
	NA	Mean :2015-09-22 19:59:55	NA	Mean :3421	Mean : 283.8
	NA	3rd Qu.:2015-12-20 17:19:00	NA	3rd Qu.:4184	3rd Qu.: 0.0
	NA	Max. :2016-03-31 23:59:00	NA	Max. :4400	Max. : 4076.0

This table will have a large number (14,493,877) of obserations caused by the number of different circuit labels as shown by the following table.

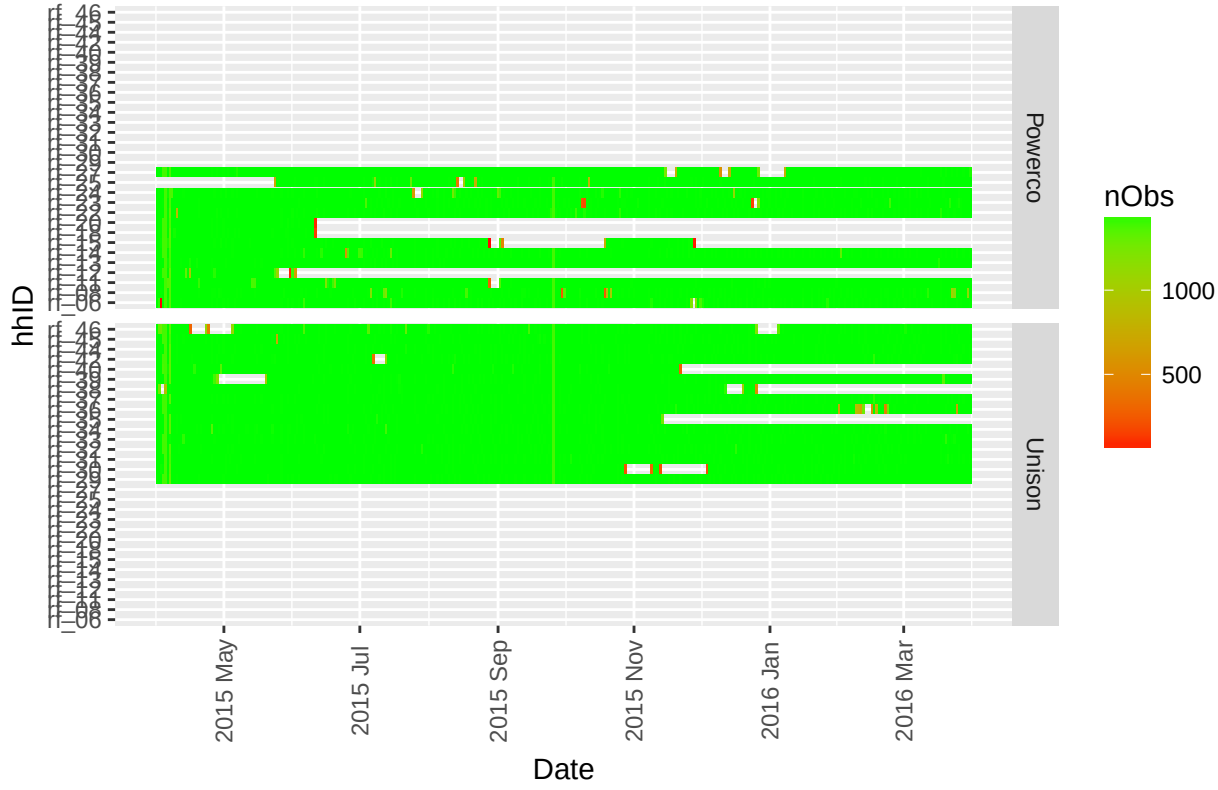
Var1	Freq
Hot Water	354795
Hot Water (2 elements)	495697
Hot Water - Controlled	10635830
Hot Water - Controlled (HEMS)	519808
Hot Water - Uncontrolled	1888644
Hot Water Cpbd Heater- Cont	519082
Incomer 1 - Hot Water - Cont	80021

Note that some households may have more than one Hot Water circuit.

4.3 Test Hot Water data

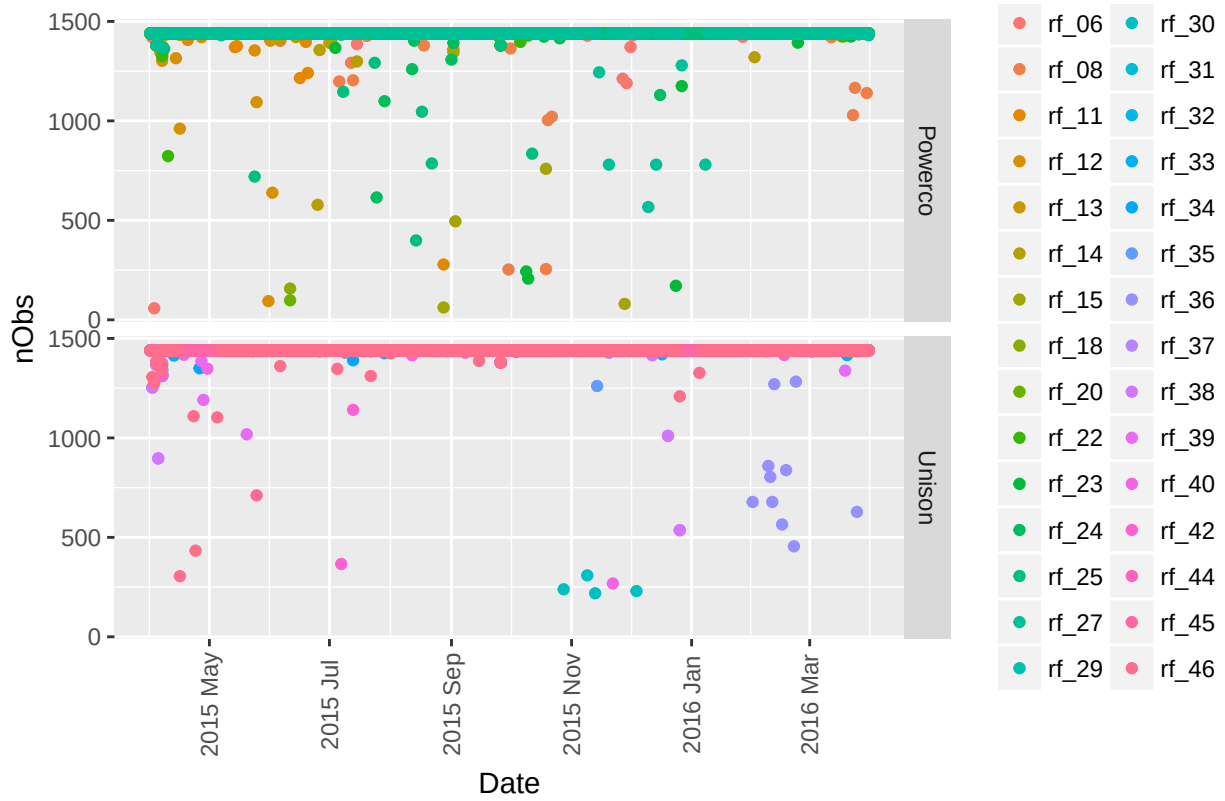
This section tests the availability of Hot Water data by replicating the standard data quality checks used for the whole gridSpy dataset.

The following plot shows loaded data observation plots - just to confirm what Hot Water data we have.



ırce: /Volumes/hum−csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/
Circuits: Hot Water from 2015−04−01 to 2016−03−31

The next plot shows the same data but as a dot plot to highlight those households and dates where we did not receive 60 * 24 = 1440 observations per day.



umes/hum−csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/
Circuits: Hot Water from 2015−04−01 to 2016−03−31

The following table shows the min/max observations per day and min/max dates for each household. As above, we should not see:

- dates before 2014 or in to the future (indicates date conversion errors)
- more than 1440 observations per day (indicates potentially duplicate observations)
- non-integer counts of circuits as it suggests some column errors

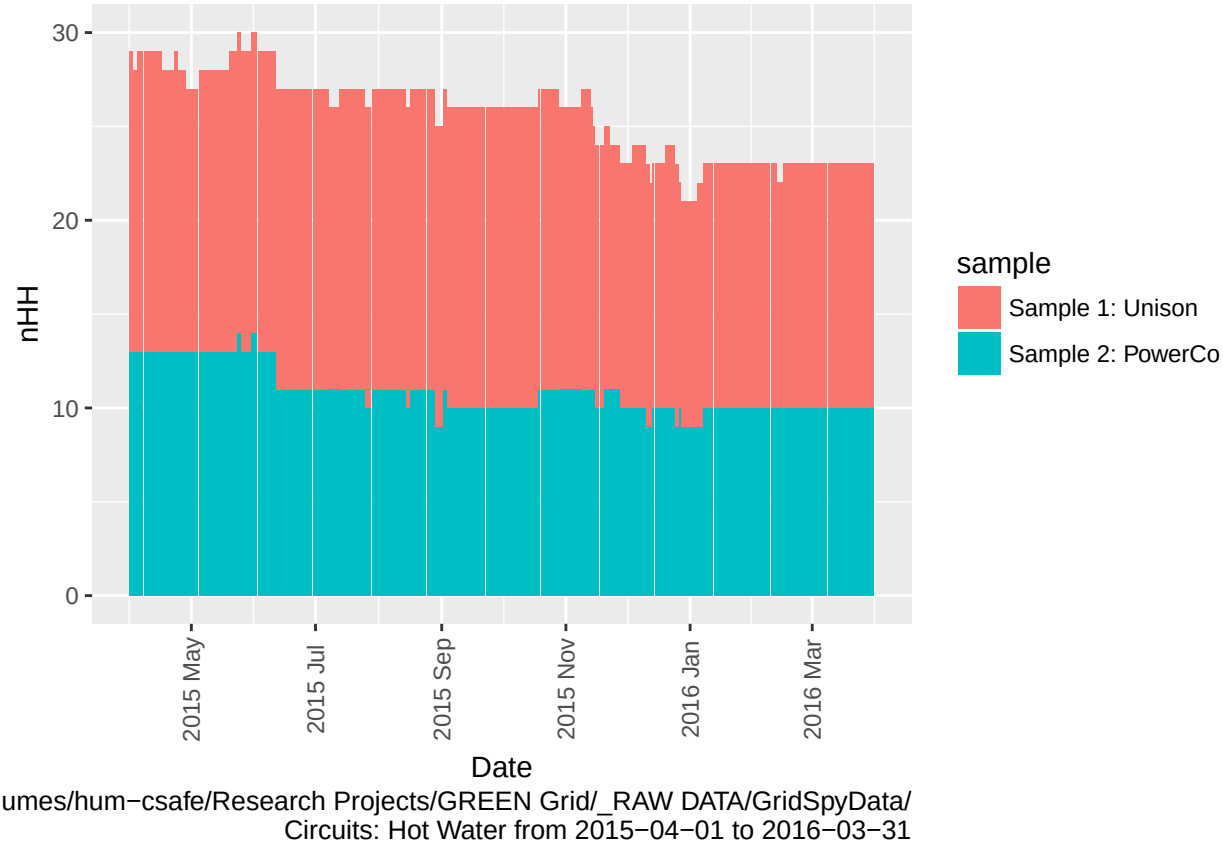
We should also not see NA in any row (indicates date conversion errors).

If we do see any of these then we still have data cleaning work to do!

hhID	sample	nObs	minDate	maxDate
rf_12	Powerco	80021	2015-04-01 00:00:00	2015-06-02 20:07:00
rf_20	Powerco	102139	2015-04-01 00:00:00	2015-06-11 01:37:00
rf_18	Powerco	102153	2015-04-01 00:00:00	2015-06-11 02:36:00
rf_35	Unison	327862	2015-04-01 00:00:00	2015-11-14 21:00:00
rf_40	Unison	338208	2015-04-01 00:00:00	2015-11-22 04:27:00
rf_15	Powerco	273202	2015-04-01 00:00:00	2015-11-28 01:19:00
rf_38	Unison	747324	2015-04-01 00:00:00	2015-12-26 08:55:00
rf_06	Powerco	523192	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_08	Powerco	521771	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_11	Powerco	519082	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_13	Powerco	526848	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_14	Powerco	525400	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_22	Powerco	526015	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_23	Powerco	519808	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_24	Powerco	520436	2015-04-01 00:00:00	2016-03-31 23:59:00

hhID	sample	nObs	minDate	maxDate
rf_25	Powerco	443882	2015-05-24 12:00:00	2016-03-31 23:59:00
rf_27	Powerco	497686	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_29	Unison	526661	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_30	Unison	477431	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_31	Unison	526781	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_32	Unison	526680	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_33	Unison	526771	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_34	Unison	526566	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_36	Unison	516142	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_37	Unison	526651	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_39	Unison	495697	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_42	Unison	518074	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_44	Unison	526749	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_45	Unison	526010	2015-04-01 00:00:00	2016-03-31 23:59:00
rf_46	Unison	950796	2015-04-01 00:00:00	2016-03-31 23:59:00

Finally we show the total number of households which we think we have Hot Water data for.



umes/hum−csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/
Circuits: Hot Water from 2015−04−01 to 2016−03−31

The following table summarises the Hot Water data. Any surprises?

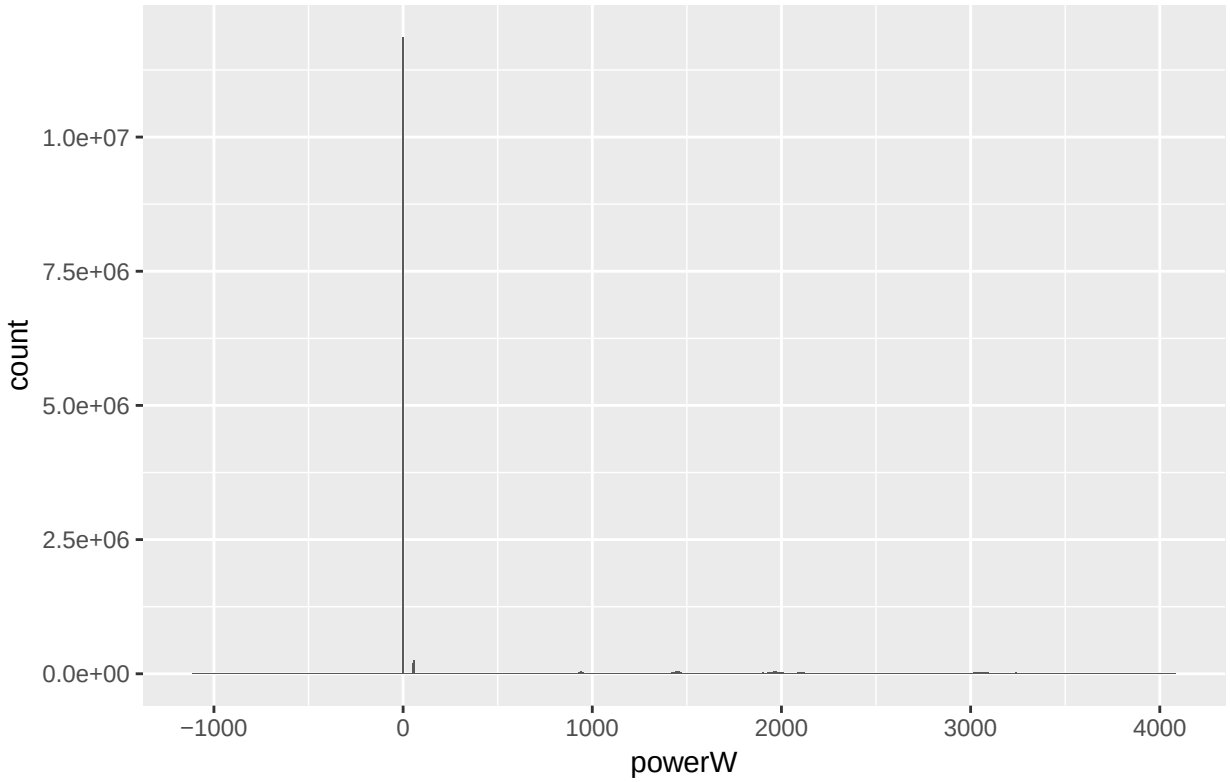
```
t <- summary(gs1MinDT)
knitr::kable(caption = paste0("Summary of ", circuitPattern, " circuits"), t)
```

	hhID	r_dateTime	circuitLabel	circuitID	powerW
	Length:14493877	Min. :2015-04-01 00:00:00	Length:14493877	Min. :1574	Min. :-1110.0

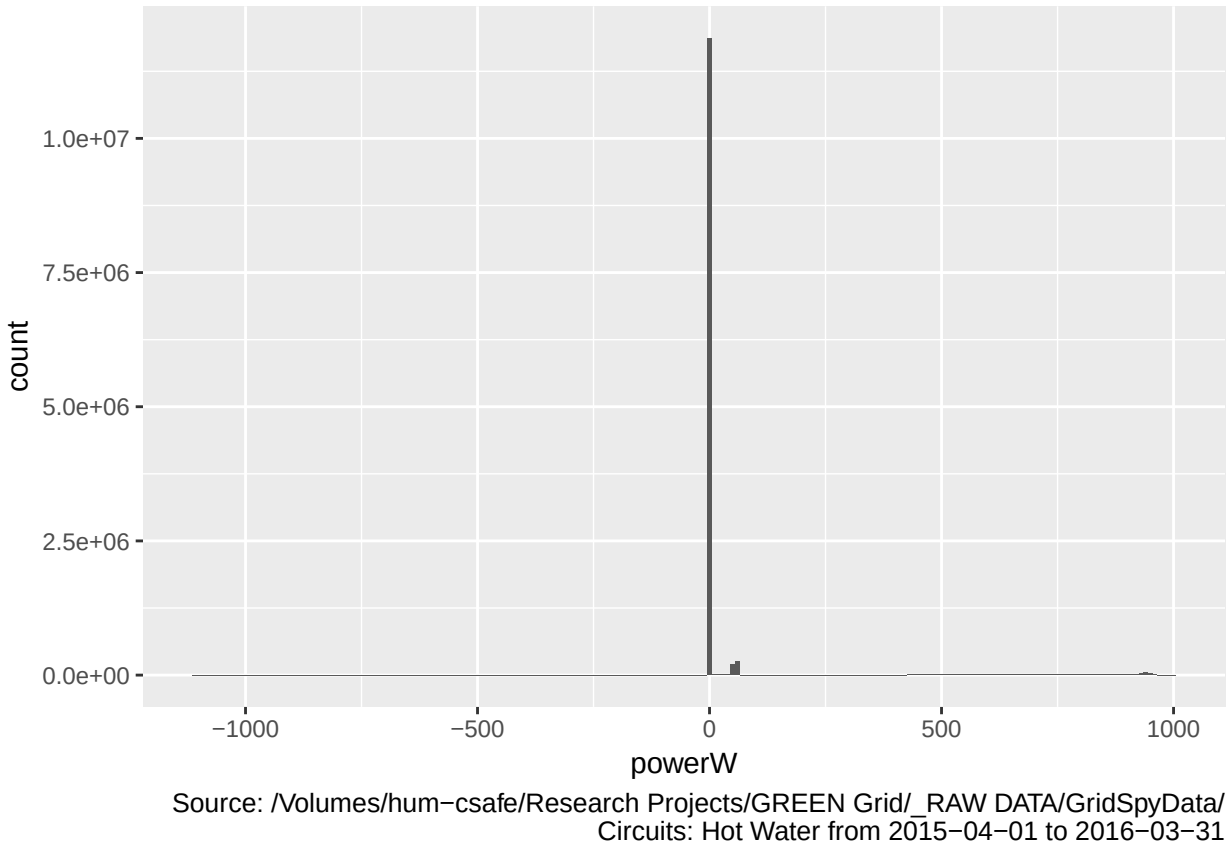
hhID	r_dateTime	circuitLabel	circuitID	powerW
Class :character	1st Qu.:2015-06-22 16:46:00	Class :character	1st Qu.:2248	1st Qu.: 0.0
Mode :character	Median :2015-09-18 21:32:00	Mode :character	Median :4135	Median : 0.0
NA	Mean :2015-09-22 19:59:55	NA	Mean :3421	Mean : 283.8
NA	3rd Qu.:2015-12-20 17:19:00	NA	3rd Qu.:4184	3rd Qu.: 0.0
NA	Max. :2016-03-31 23:59:00	NA	Max. :4400	Max. : 4076.0

We seem to have some negative powerW values and at least one very large power value.

Nasty surprises often lurk in histograms... The following histogram shows all observations.



The next shows the histogram for powerW < 1000W...



There are a lot of zeros (as we'd expect) but why are there negative values?

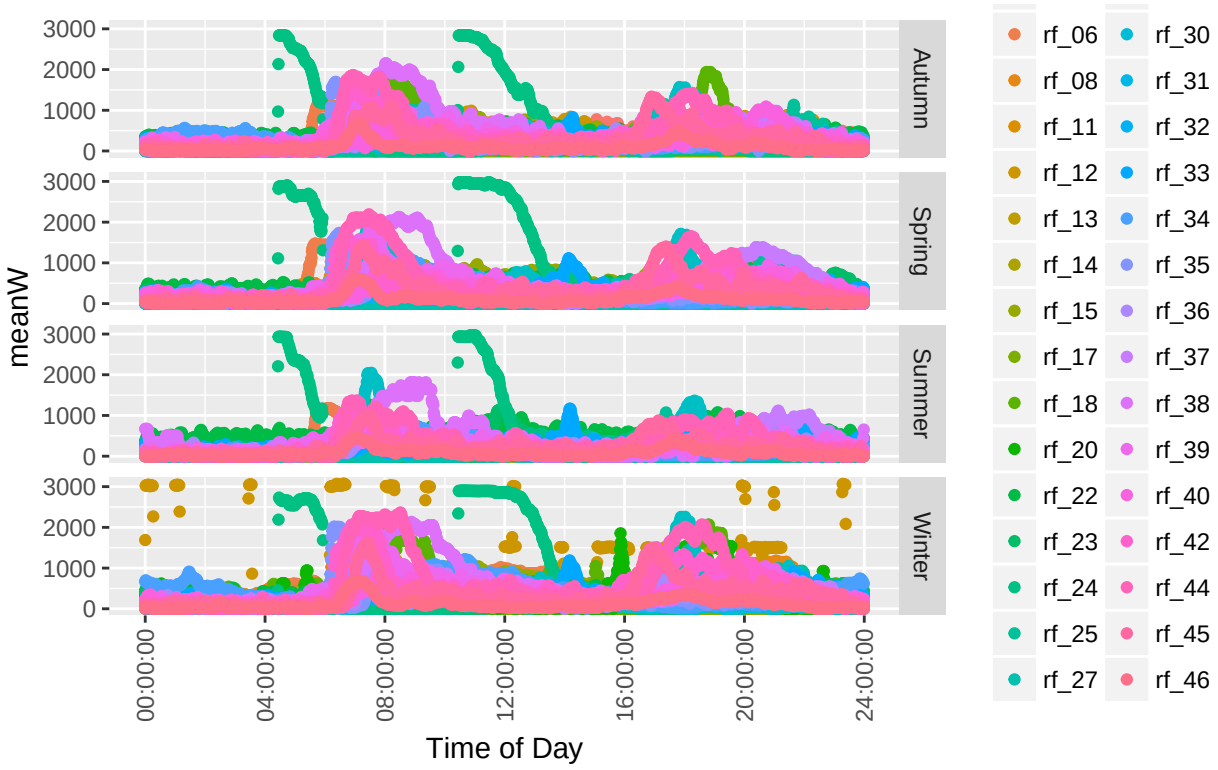
5 Hot Water profiles

This section produces the profiles as one for each HH but averaged over each season. Data is kept at 1 minute intervals. Note definition of season below...

```
# add season
gs1MinDT <- gs1MinDT[, month := lubridate::month(r_dateTime, label = TRUE)]
gs1MinDT <- gs1MinDT[, season := "Summer"]
gs1MinDT <- gs1MinDT[, season := ifelse(month == "Mar" |
                                        month == "Apr" |
                                        month == "May", "Autumn", season)]
gs1MinDT <- gs1MinDT[, season := ifelse(month == "Jun" |
                                        month == "Jul" |
                                        month == "Aug", "Winter", season)]
gs1MinDT <- gs1MinDT[, season := ifelse(month == "Sep" |
                                        month == "Oct" |
                                        month == "Nov", "Spring", season)]
```

5.1 Profile plots: means per household

This section shows a plot of mean profiles for each household by season.



umes/hum−csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/
Circuits: Hot Water from 2015−04−01 to 2016−03−31
n households = 32

```
## [1] "Saving plot to /Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/pro
```

```
## [1] "Saving profile data used to build this plot to: /Volumes/hum-csafe/Research Projects/GREEN Grid/
```

```
## [1] "Gzipped /Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/profiles/H
```

hhID	obsHourMin	season	meanW
Length:167011	Length:167011	Length:167011	Min. : -8.279
Class :character	Class1:hms	Class :character	1st Qu.: 69.255
Mode :character	Class2:difftime	Mode :character	Median : 193.417
NA	Mode :numeric	NA	Mean : 293.940
NA	NA	NA	3rd Qu.: 389.015
NA	NA	NA	Max. :3072.650

As we can see there is considerable variation between households in both the level and timing of heat pump demand.

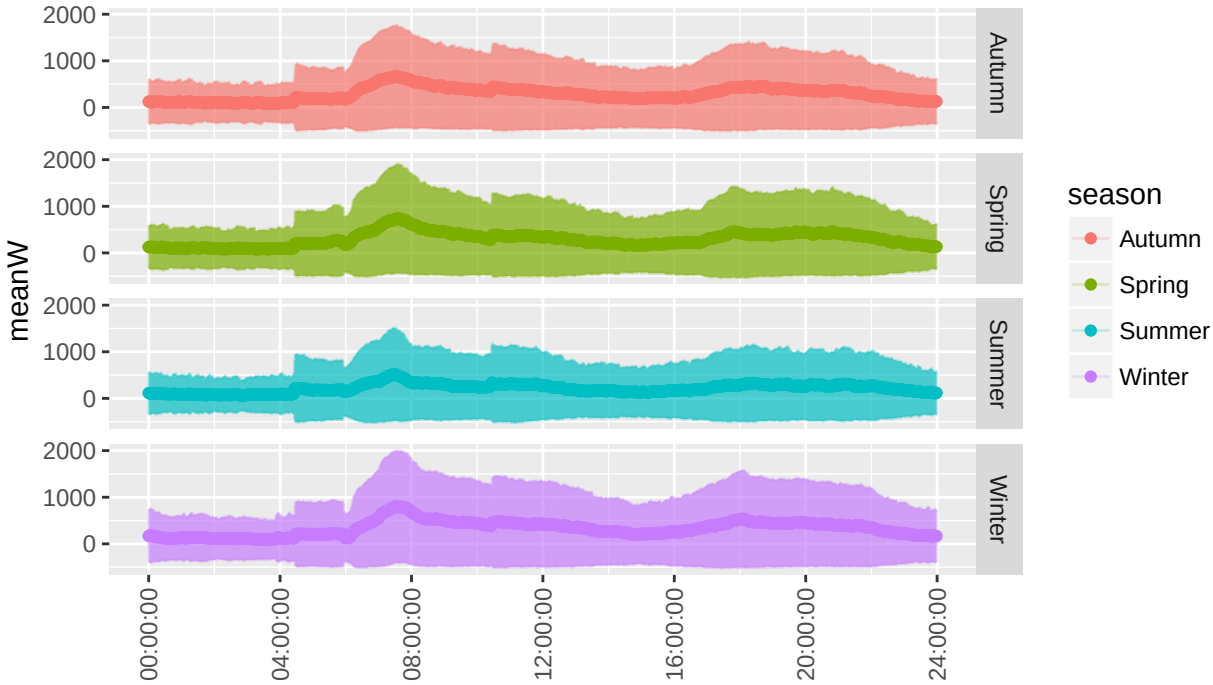
Note that the code saves a high definition version of the plot and the profiles for future re-use.

The .csv.gz file can be loaded using the following code:

- `df <- readr::read_csv("/Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1mi Water_2015-04-01_2016-03-31_byHouseholdSeasonalProfiles.csv.gz")` or
- `dt <- data.table::as.data.table(readr::read_csv("/Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/profiles/Hot Water_2015-04-01_2016-03-31_byHouseholdSeasonalProfi` if you prefer `data.table`

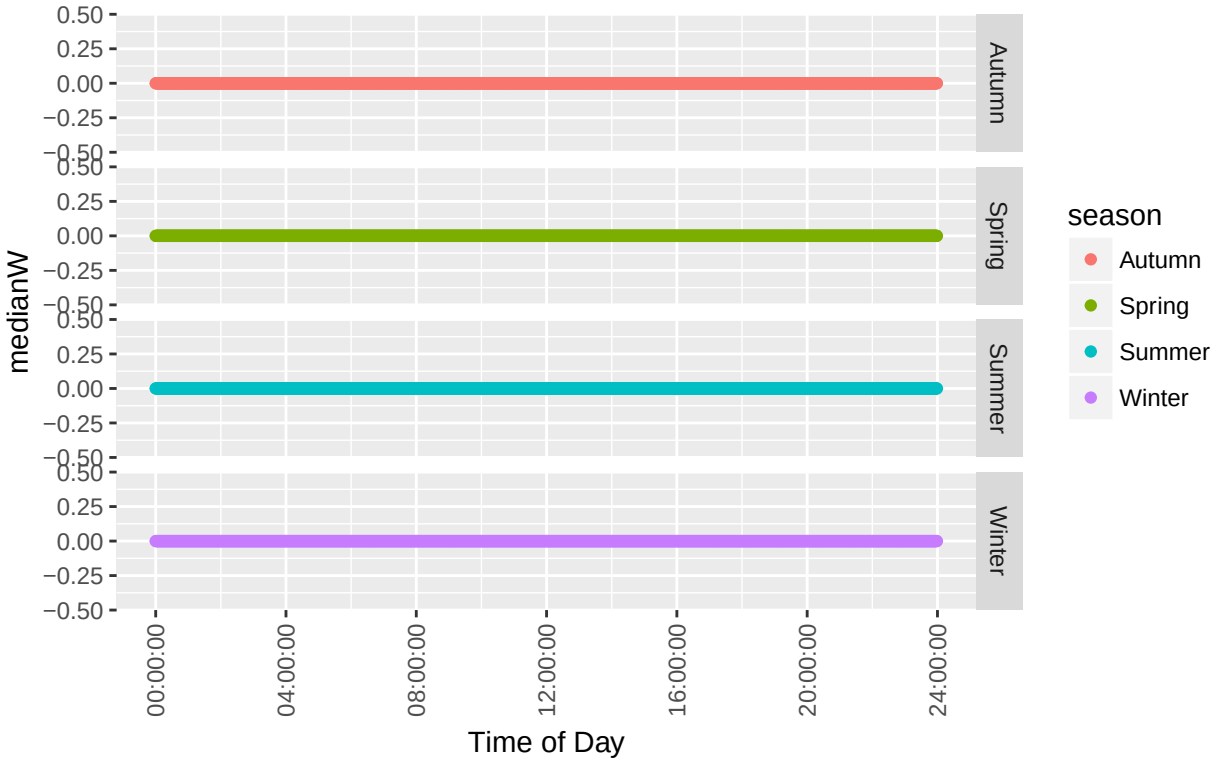
5.2 Profile plots: overall household mean

This section shows a plot of mean and median profiles across all household by season. The mean profile also shows the level of variance by plotting error bars at +/- 1 s.d.



e: /Volumes/hum−csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/
Circuits: Hot Water from 2015−04−01 to 2016−03−31
n households = 32
Error bars = +/− 1 s.d.

```
## [1] "Saving plot to /Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/pro
```



e: /Volumes/hum−csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/
Circuits: Hot Water from 2015−04−01 to 2016−03−31
n households = 32

```
## [1] "Saving plot to /Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/proc
## [1] "Saving profile data used to build this plot to: /Volumes/hum-csafe/Research Projects/GREEN Grid/
## [1] "Gzipped /Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/profiles/Ho
```

obsHourMin	season	meanW	medianW	nObs	sdW
Length:5760	Length:5760	Min. : 58.62	Min. :0	Min. :2230	Min. : 320.4
Class1:hms	Class :character	1st Qu.:170.19	1st Qu.:0	1st Qu.:2430	1st Qu.: 576.9
Class2:difftime	Mode :character	Median :261.01	Median :0	Median :2569	Median : 725.2
Mode :numeric	NA	Mean :281.21	Mean :0	Mean :2516	Mean : 721.3
NA	NA	3rd Qu.:386.90	3rd Qu.:0	3rd Qu.:2657	3rd Qu.: 876.0
NA	NA	Max. :805.88	Max. :0	Max. :2671	Max. :1208.5

The difference between the mean and median plots is instructive - it suggests that the mean plots for summer are skewed by a few higher heat pump-using households.

The plots could be repeated or re-facted e.g. by household size.

As before, the code saves a high definition version of the plot and the profiles for future re-use.

6 Runtime

Analysis completed in 130.32 seconds (2.17 minutes) using knitr in RStudio with R version 3.5.0 (2018-04-23) running on x86_64-apple-darwin15.6.0.

7 R environment

R packages used:

- base R - for the basics (R Core Team 2016)
- data.table - for fast (big) data handling (Dowle et al. 2015)
- lubridate - date manipulation (Grolemund and Wickham 2011)
- ggplot2 - for slick graphics (Wickham 2009)
- readr - for csv reading/writing (Wickham, Hester, and Francois 2016)
- dplyr - for select and contains (Wickham and Francois 2016)
- progress - for progress bars (Csárdi and FitzJohn 2016)
- knitr - to create this document & neat tables (Xie 2016)
- nzGREENGrid - for local NZ GREEN Grid project utilities

Session info:

```
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS High Sierra 10.13.6
##
## Matrix products: default
## BLAS: /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Versions
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_GB.UTF-8/en_GB.UTF-8/en_GB.UTF-8/C/en_GB.UTF-8/en_GB.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] kableExtra_0.9.0  knitr_1.20        readr_1.1.1       ggplot2_2.2.1
## [5] dplyr_0.7.5       data.table_1.11.4 rmarkdown_1.10    nzGREENGrid_0.1.0
##
## loaded via a namespace (and not attached):
##  [1] tinytex_0.5       tidyselect_0.2.4  reshape2_1.4.3
##  [4] purrr_0.2.5       lattice_0.20-35   colorspace_1.3-2
##  [7] htmltools_0.3.6   viridisLite_0.3.0 yaml_2.1.19
## [10] utf8_1.1.4        rlang_0.2.1       pillar_1.2.3
## [13] glue_1.2.0        sp_1.3-1          readxl_1.1.0
## [16] bindrcpp_0.2.2    jpeg_0.1-8        bindr_0.1.1
## [19] plyr_1.8.4        stringr_1.3.1     cellranger_1.1.0
## [22] munsell_0.5.0     gtable_0.2.0      rvest_0.3.2
## [25] RgoogleMaps_1.4.2 mapproj_1.2.6     evaluate_0.10.1
## [28] labeling_0.3      highr_0.7         proto_1.0.0
## [31] Rcpp_0.12.17      geosphere_1.5-7   openssl_1.0.1
## [34] scales_0.5.0      backports_1.1.2   rjson_0.2.20
## [37] hms_0.4.2         png_0.1-7         digest_0.6.15
## [40] stringi_1.2.3     grid_3.5.0        rprojroot_1.3-2
```

```
## [43] cli_1.0.0         tools_3.5.0       magrittr_1.5
## [46] maps_3.3.0        lazyeval_0.2.1    tibble_1.4.2
## [49] crayon_1.3.4      pkgconfig_2.0.1   xml2_1.2.0
## [52] lubridate_1.7.4   assertthat_0.2.0  httr_1.3.1
## [55] rstudioapi_0.7    R6_2.2.2          ggmap_2.6.1
## [58] compiler_3.5.0
```

References

Csárdi, Gábor, and Rich FitzJohn. 2016. *Progress: Terminal Progress Bars*. https://CRAN.R-project.org/package=progress.

Dowle, M, A Srinivasan, T Short, S Lianoglou with contributions from R Saporta, and E Antonyan. 2015. *Data.table: Extension of Data.frame*. https://CRAN.R-project.org/package=data.table.

Grolemund, Garrett, and Hadley Wickham. 2011. "Dates and Times Made Easy with lubridate." *Journal of Statistical Software* 40 (3): 1–25. http://www.jstatsoft.org/v40/i03/.

R Core Team. 2016. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley. 2009. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. http://ggplot2.org.

Wickham, Hadley, and Romain Francois. 2016. *Dplyr: A Grammar of Data Manipulation*. https://CRAN.R-project.org/package=dplyr.

Wickham, Hadley, Jim Hester, and Romain Francois. 2016. *Readr: Read Tabular Data*. https://CRAN.R-project.org/package=readr.

Xie, Yihui. 2016. *Knitr: A General-Purpose Package for Dynamic Report Generation in R*. https://CRAN.R-project.org/package=knitr.