GREEN Grid Heat Pump Profiles

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

Last run at: 2018-06-06 23:06:14

Contents

1 Status

Test run using reduced data from ~/Data/NZGreenGrid/gridspy/1min_orig/

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
## 1: Unison rf_28              2      <NA>            3(12,8,4)    <NA>
## 2: Unison rf_29              2      <NA>     1 (7 months old)    live
## 3: Unison rf_30              2         0                    0    <NA>
```

```
## 4: Unison rf_31 2 (Plus cousin)      <NA>                 <NA>    live
## 5: Unison rf_32               2      <NA> 2 (7 and 4years old)    <NA>
## 6: Unison rf_33               2 1(14yold)            1 (6yold)    live
```

```
##     sample  hhID Adults Teenagers Children  removed
## 1: Powerco rf_12      1      <NA>     <NA> 3/6/1015
## 2: Powerco  <NA>      1      <NA>     <NA>     <NA>
## 3: Powerco rf_25      1      <NA>     <NA>     <NA>
## 4: Powerco  <NA>     NA      <NA>     <NA>     <NA>
## 5: Powerco  <NA>      1      <NA>   1(5mo)     <NA>
## 6: Powerco  <NA>     NA      <NA>     <NA>     <NA>
```

4.2 Grid Spy data

In this section we load the cleaned data files from ~/Data/NZGreenGrid/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 <- "Heat Pump"
dateFrom <- "2015-04-01"
dateTo <- "2016-03-31"

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

So we are looking for Heat Pump 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.

The file we are looking for is: Heat Pump_2015-04-01_2016-03-31_observations.csv

```
## [1] "~/Data/NZGreenGrid/safe/gridSpy/1min/dataExtracts/Heat Pump_2015-04-01_2016-03-31_observations.
## [1] "# Loaded 14,252,439 rows of data"
```

The following table summarises the Heat Pump data we have found.

This table will have a large number (14,252,439) of obserations caused by the number of different circuit labels as shown by the following table.

Note that some households may have more than one Heat Pump circuit.

4.3 Test Heat Pump data

This section tests the availability of Heat Pump 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 Heat Pump data we have.

Table 1: Meta data for sample

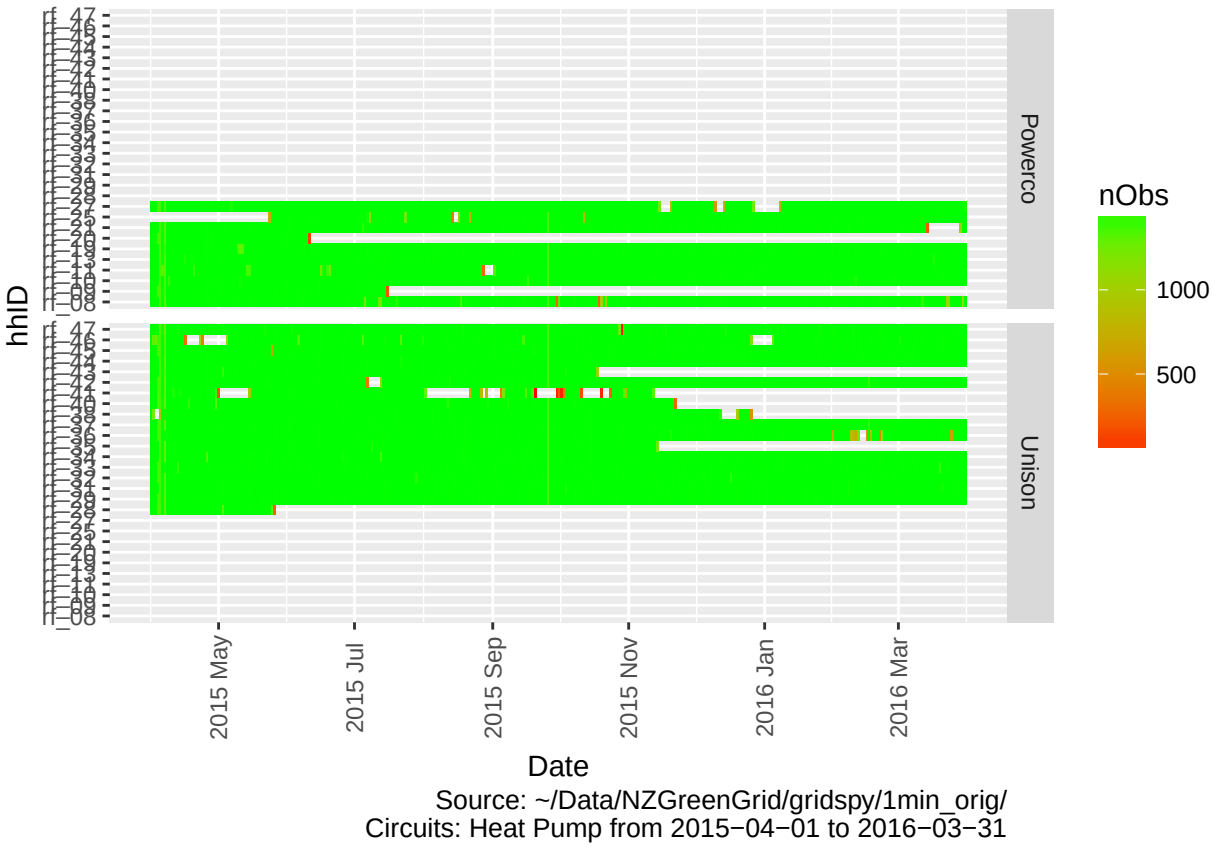
sample	hhID	Adults	Teenagers	Children	removed	nAdults
Powerco	rf_06	2	NA	NA	NA	2
Powerco	rf_07	2	NA	2	NA	2
Powerco	rf_08	2	NA	NA	NA	2
Powerco	rf_09	2	NA	1	42171	2
Powerco	rf_10	2	NA	1(3yo)	NA	3
Powerco	rf_11	NA	NA	NA	NA	1
Powerco	rf_12	1	NA	NA	3/6/1015	1
Powerco	rf_13	2	1(16yo)	1(11)	NA	2
Powerco	rf_14	1	NA	1 (11 yo)	NA	1
Powerco	rf_15	NA	NA	NA	42462	1
Powerco	rf_15_old	1	NA	NA	42019	1
Powerco	rf_16	2	NA	NA	42089	2
Powerco	rf_17 sn_662	NA	NA	NA	NA	1
Powerco	rf_17_oldNo reused	2	1(13yo)	1(11yo)	42457	2
Powerco	rf_18	2	NA	1(1yo)	42532	2
Powerco	rf_19	1	NA	NA	NA	1
Powerco	rf_20	2	NA	2	42166	2
Powerco	rf_21	2	NA	NA	42821	2
Powerco	rf_22	2	NA	NA	NA	2
Powerco	rf_23	1	NA	NA	NA	1
Powerco	rf_24	2	NA	2	NA	2
Powerco	rf_25	1	NA	NA	NA	1
Powerco	rf_26	2	NA	NA	NA	2
Powerco	rf_27	2	1	1	NA	2
Unison	rf_28	2	NA	3(12,8,4)	NA	3
Unison	rf_29	2	NA	1 (7 months old)	live	2
Unison	rf_30	2	0	0	NA	2
Unison	rf_31	2 (Plus cousin)	NA	NA	live	2
Unison	rf_32	2	NA	2 (7 and 4years old)	NA	2
Unison	rf_33	2	1(14yold)	1 (6yold)	live	2
Unison	rf_34	3	NA	NA	NA	1
Unison	rf_35	2	NA	NA	42322	2
Unison	rf_36	1	2 (14 and 12)	NA	live	1
Unison	rf_37	2	NA	NA	live	2
Unison	rf_38	NA	NA	NA	NA	1
Unison	rf_38	2	NA	2 (<12)	NA	2
Unison	rf_39	2	1 (16 YO)	NA	live	2
Unison	rf_40	2	NA	NA	42330	2
Unison	rf_41	2	NA	2 (11 and 8)	live	2
Unison	rf_42	2	NA	3 (<12 yold, 1 10 YO)	NA	3
Unison	rf_43	2	NA	NA	42296	2
Unison	rf_44	2	NA	2 (10 and 7)	NA	2
Unison	rf_45	2	NA	3 (<12 years old)	NA	3
Unison	rf_46	2	NA	1 (4yold-50%)	live	2
Unison	rf_47	3	2	NA	NA	1

Table 2: Summary of household grid spy data for: Heat Pump

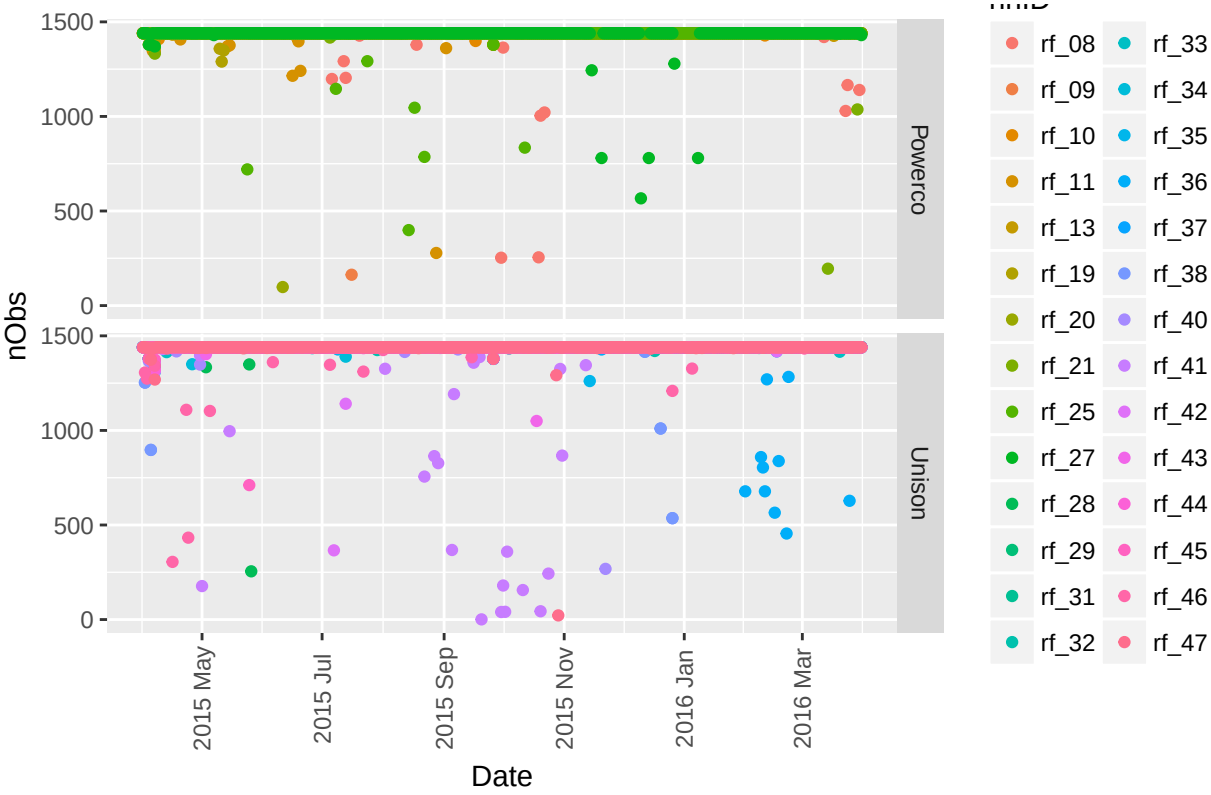
hhID	nObs	nHouseholds	nCircuits	meanPower	minDate	maxDate
rf_08	521843	1	1	65.92309	2015-04-01	2016-03-31
rf_09	152669	1	1	266.37950	2015-04-01	2015-07-16
rf_10	526797	1	1	52.40885	2015-04-01	2016-03-31
rf_11	519185	1	1	236.23474	2015-04-01	2016-03-31
rf_13	1053858	1	2	227.22865	2015-04-01	2016-03-31
rf_17	520028	1	1	31.15216	2015-04-01	2016-03-28
rf_19	1053136	1	2	39.53314	2015-04-01	2016-03-31
rf_20	102188	1	1	140.09662	2015-04-01	2015-06-11
rf_21	505042	1	1	118.65538	2015-04-01	2016-03-31
rf_25	443936	1	1	67.43931	2015-05-24	2016-03-31
rf_27	497806	1	1	168.10637	2015-04-01	2016-03-31
rf_28	79085	1	1	56.43806	2015-04-01	2015-05-26
rf_29	526780	1	1	646.73224	2015-04-01	2016-03-31
rf_31	526878	1	1	95.06807	2015-04-01	2016-03-31
rf_32	526785	1	1	69.66461	2015-04-01	2016-03-31
rf_33	526863	1	1	359.19048	2015-04-01	2016-03-31
rf_34	526677	1	1	174.28675	2015-04-01	2016-03-31
rf_35	327974	1	1	79.08711	2015-04-01	2015-11-14
rf_36	516242	1	1	76.00384	2015-04-01	2016-03-31
rf_37	526771	1	1	75.30394	2015-04-01	2016-03-31
rf_38	373722	1	1	282.16696	2015-04-01	2015-12-26
rf_40	338289	1	1	336.91930	2015-04-01	2015-11-22
rf_41	223824	1	1	82.65447	2015-04-01	2015-11-12
rf_42	518179	1	1	48.25356	2015-04-01	2016-03-31
rf_43	288838	1	1	200.56165	2015-04-01	2015-10-18
rf_44	526850	1	1	109.39986	2015-04-01	2016-03-31
rf_45	526110	1	1	85.44295	2015-04-01	2016-03-31
rf_46	950976	1	2	90.78617	2015-04-01	2016-03-31
rf_47	525108	1	1	132.97838	2015-04-01	2016-03-31

Table 3: Counts of Heat Pump observations by label and household

	rf_08	rf_09	rf_10	rf_11	rf_13	rf_17	rf_19	rf_20	r
Bedroom & Lounge Heat Pumps$2741	0	0	0	0	0	0	526568	0	
Downstairs (inc 1 Heat Pump)$2212	0	0	0	0	526929	0	0	0	
Heat Pump (x2) & Lounge Power$4166	0	0	0	0	0	0	0	0	
Heat Pump & 2 x Bathroom Heat$4171	0	0	0	0	0	0	0	0	
Heat Pump & Bedroom 2$2731	0	152669	0	0	0	0	0	0	
Heat Pump & Kitchen Appliances$4186	0	0	0	0	0	0	0	0	
Heat Pump & Lounge$2590	0	0	0	519185	0	0	0	0	
Heat Pump & Misc$2107	0	0	0	0	0	0	0	102188	
Heat Pump & Washing Machine$2750	0	0	0	0	0	0	0	0	50
Heat Pump$2092	521843	0	0	0	0	0	0	0	
Heat Pump$2148	0	0	0	0	0	520028	0	0	
Heat Pump$2598	0	0	526797	0	0	0	0	0	
Heat Pump$2758	0	0	0	0	0	0	0	0	
Heat Pump$2826	0	0	0	0	0	0	0	0	
Heat Pump$4124	0	0	0	0	0	0	0	0	
Heat Pump$4130	0	0	0	0	0	0	0	0	
Heat Pump$4134	0	0	0	0	0	0	0	0	
Heat Pump$4150	0	0	0	0	0	0	0	0	
Heat Pump$4154	0	0	0	0	0	0	0	0	
Heat Pump$4160	0	0	0	0	0	0	0	0	
Heat Pump$4175	0	0	0	0	0	0	0	0	
Heat Pump$4190	0	0	0	0	0	0	0	0	
Heat Pump$4196	0	0	0	0	0	0	0	0	
Heat Pump$4204	0	0	0	0	0	0	0	0	
Heat Pump$4211	0	0	0	0	0	0	0	0	
Heat Pump$4219	0	0	0	0	0	0	0	0	
Heat Pump$4223	0	0	0	0	0	0	0	0	
Heat Pumps (2x) & Power$4232	0	0	0	0	0	0	0	0	
Heat Pumps (2x) & Power$4399	0	0	0	0	0	0	0	0	
Kitchen Appliances & Heat Pump$4140	0	0	0	0	0	0	0	0	
Theatre Heat Pump$2740	0	0	0	0	0	0	526568	0	
Upstairs Heat Pumps$2211	0	0	0	0	526929	0	0	0	



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.



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!

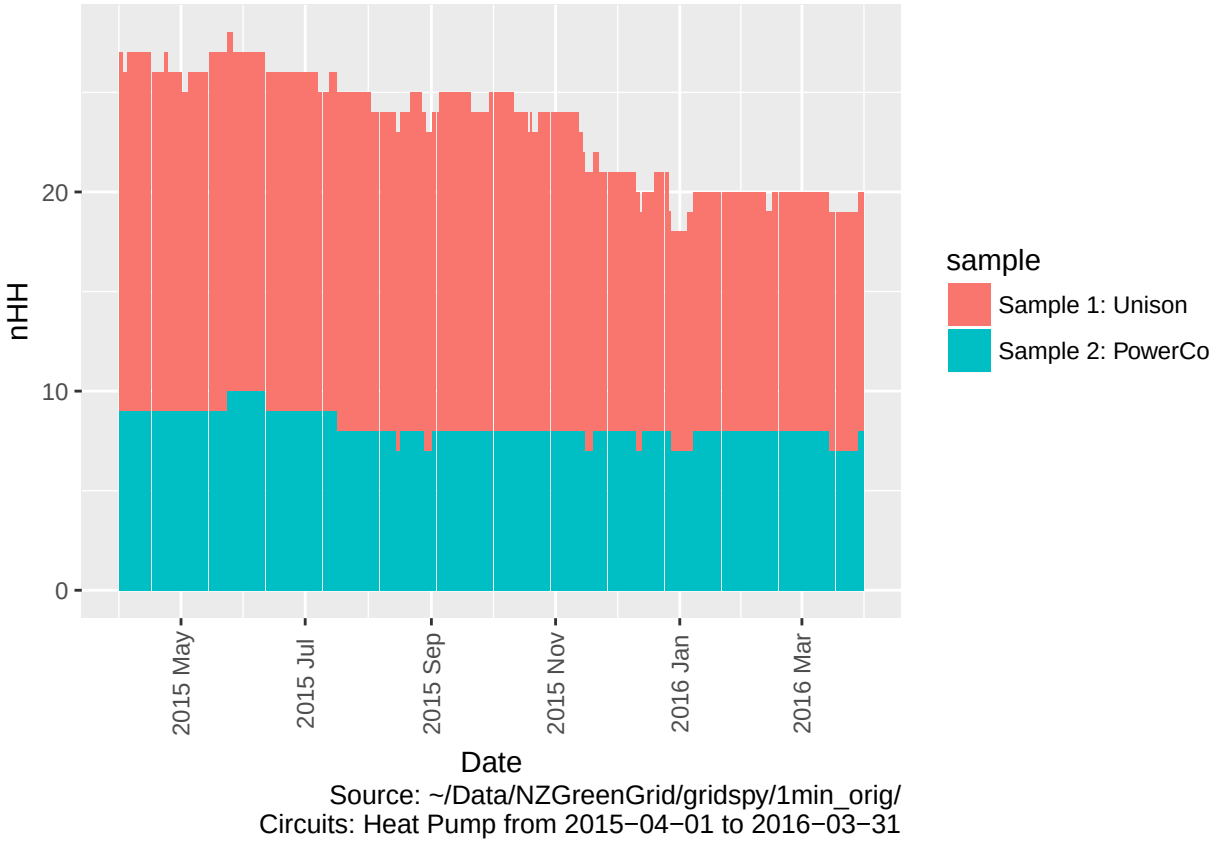
Finally we show the total number of households which we think we have Heat Pump data for.

Table 4: Summary observation stats by hhID (sorted by date last heard from) for: Heat Pump

hhID	sample	nObs	minDate	maxDate
rf_28	Unison	79085	2015-04-01T00:00:00Z	2015-05-26T04:56:00Z
rf_20	Powerco	102188	2015-04-01T00:00:00Z	2015-06-11T01:37:00Z
rf_09	Powerco	152669	2015-04-01T00:00:00Z	2015-07-16T02:42:00Z
rf_43	Unison	288838	2015-04-01T00:00:00Z	2015-10-18T17:29:00Z
rf_41	Unison	223824	2015-04-01T00:00:00Z	2015-11-12T22:24:00Z
rf_35	Unison	327974	2015-04-01T00:00:00Z	2015-11-14T21:00:00Z
rf_40	Unison	338289	2015-04-01T00:00:00Z	2015-11-22T04:27:00Z
rf_38	Unison	747444	2015-04-01T00:00:00Z	2015-12-26T08:55:00Z
rf_08	Powerco	521843	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_10	Powerco	526797	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_11	Powerco	519185	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_13	Powerco	1053858	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_19	Powerco	1053136	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_21	Powerco	505042	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_25	Powerco	443936	2015-05-24T12:00:00Z	2016-03-31T23:59:00Z
rf_27	Powerco	497806	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_29	Unison	526780	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_31	Unison	526878	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_32	Unison	526785	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_33	Unison	526863	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_34	Unison	526677	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_36	Unison	516242	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_37	Unison	526771	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_42	Unison	518179	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_44	Unison	526850	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_45	Unison	526110	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_46	Unison	950976	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z
rf_47	Unison	525108	2015-04-01T00:00:00Z	2016-03-31T23:59:00Z

Table 5: Summary of Heat Pump circuits

	hhID	r_dateTime	circuit	powerW	obsHourMin
	Length:14252439	Length:14252439	Length:14252439	Min. : -655.00	Length:14252439
	Class :character	Class :character	Class :character	1st Qu.: 0.00	Class :character
	Mode :character	Mode :character	Mode :character	Median : 0.00	Mode :character
	NA	NA	NA	Mean : 147.90	NA
	NA	NA	NA	3rd Qu.: 61.23	NA
	NA	NA	NA	Max. :27759.00	NA

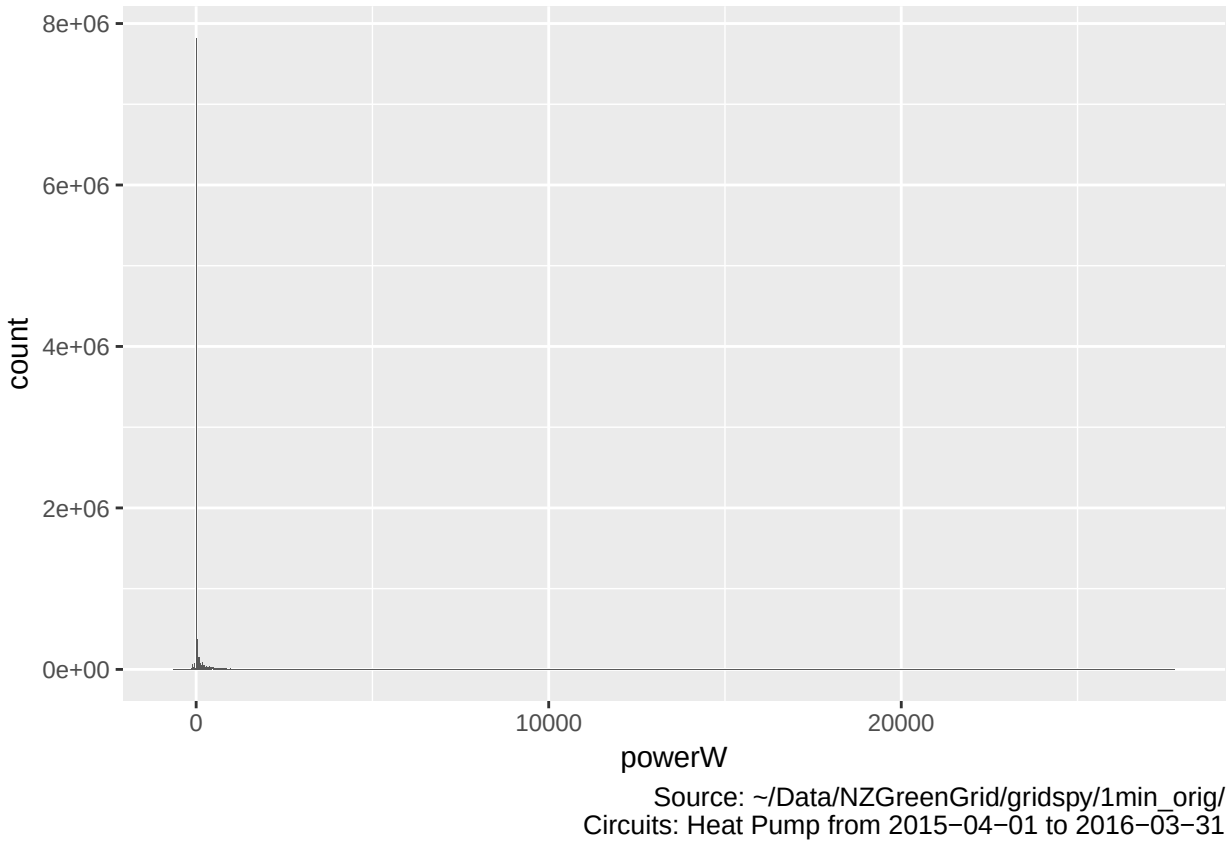


The following table summarises the Heat Pump data. Any surprises?

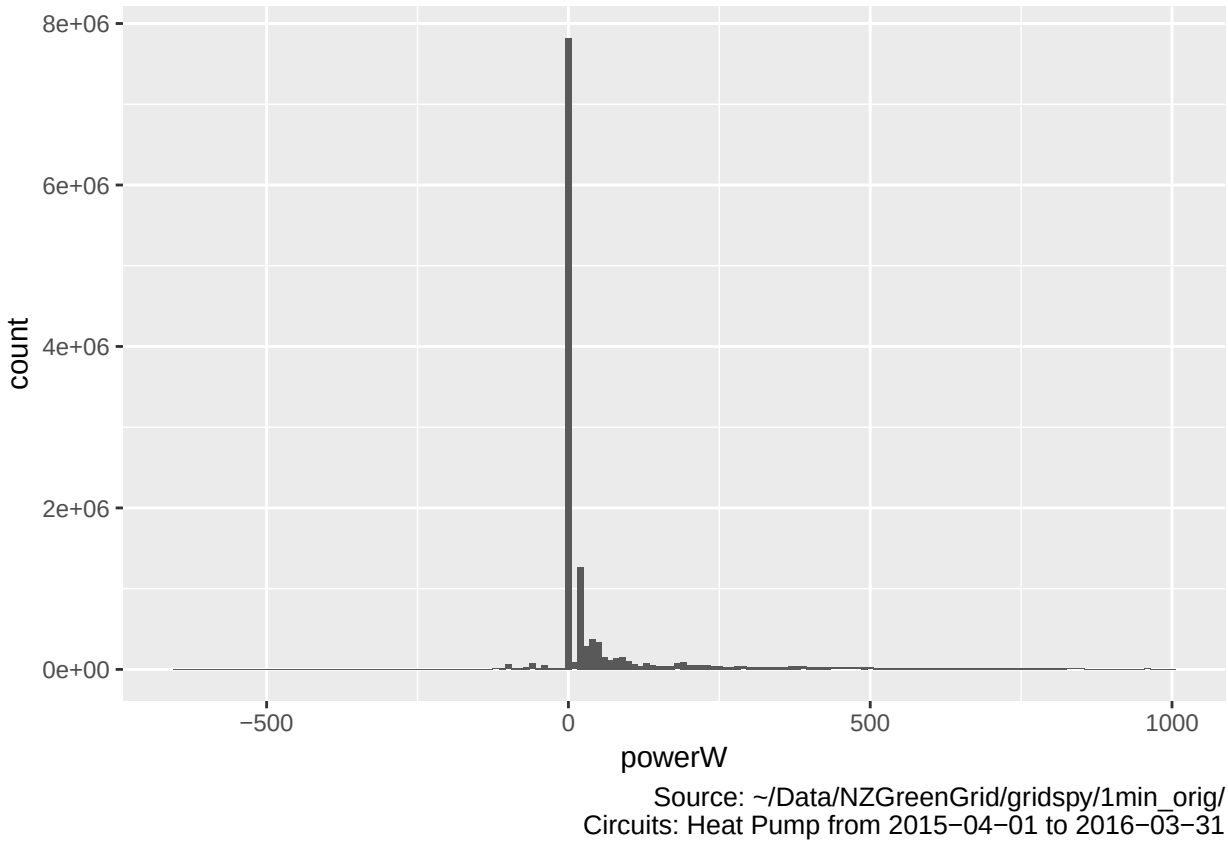
```
t <- summary(gs1MinDT)
kable(caption = paste0("Summary of ", circuitPattern, " circuits"), t)
```

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...



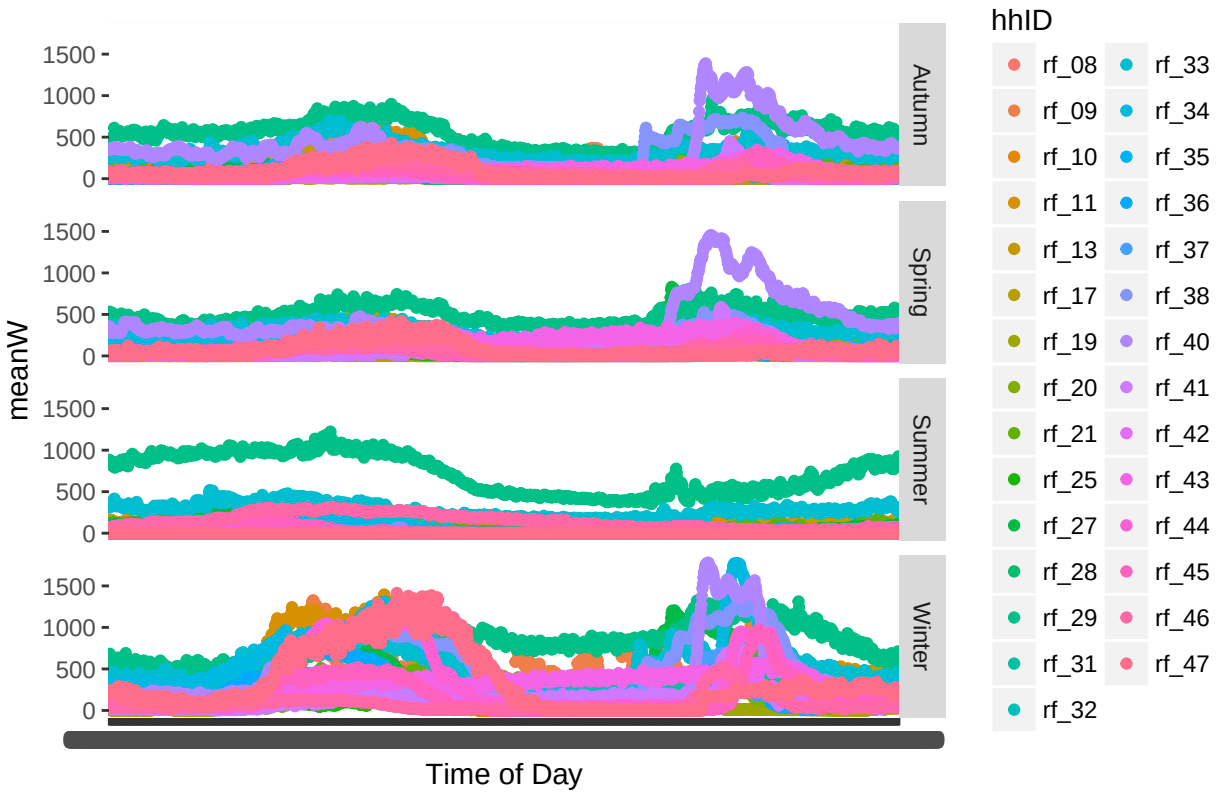
There are a lot of zeros (as we'd expect) but why are there negative values?

5 Heat Pump 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)]
```

Profile plots:




```
## [1] "Saving profile data used to build this plot to: /Users/ben/git.soton/ba1e12/nzGREENGrid/data/Hea
```

```
## [1] "Gzipped /Users/ben/git.soton/ba1e12/nzGREENGrid/data/Heat Pump_2015-04-01_2016-03-31_meanW_hhID_
```

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

Note that the code saves a high definition version of the chart and the average profiles out to the repo (in /Users/ben/git.soton/ba1e12/nzGREENGrid/data/Heat Pump_2015-04-01_2016-03-31_meanW_hhID_profiles_by_season.csv.gz) for future re-use.

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

- `df <- readr::read_csv("/Users/ben/git.soton/ba1e12/nzGREENGrid/data/Heat Pump_2015-04-01_2016-03-31`
  or
- `dt <- data.table::as.data.table(readr::read_csv("/Users/ben/git.soton/ba1e12/nzGREENGrid/data/Heat Pump_2015-04-01_2016-03-31_meanW_hhID_profiles_by_season.csv.gz"))` if you prefer data.table

6 Runtime

Analysis completed in 395.46 seconds (6.59 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)
- kableExtra - for extra neat tables (Zhu 2018)
- 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 10.13.5
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## 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_1.11.2 nzGREENGrid_0.1.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.17      cellranger_1.1.0  pillar_1.2.2
##  [4] compiler_3.5.0    plyr_1.8.4        bindr_0.1.1
##  [7] prettyunits_1.0.2 tools_3.5.0       progress_1.1.2
## [10] digest_0.6.15     viridisLite_0.3.0 lubridate_1.7.4
## [13] evaluate_0.10.1   tibble_1.4.2      gtable_0.2.0
## [16] pkgconfig_2.0.1   rlang_0.2.0       rstudioapi_0.7
## [19] yaml_2.1.19       bindrcpp_0.2.2    xml2_1.2.0
## [22] httr_1.3.1        stringr_1.3.1     hms_0.4.2
## [25] rprojroot_1.3-2   grid_3.5.0        tidyselect_0.2.4
## [28] glue_1.2.0        R6_2.2.2          readxl_1.1.0
## [31] rmarkdown_1.9     purrr_0.2.4       reshape2_1.4.3
## [34] magrittr_1.5      backports_1.1.2   scales_0.5.0
## [37] htmltools_0.3.6   rvest_0.3.2       assertthat_0.2.0
## [40] colorspace_1.3-2  labeling_0.3      stringi_1.2.2
## [43] lazyeval_0.2.1    munsell_0.4.3
```

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.

Zhu, Hao. 2018. *KableExtra: Construct Complex Table with 'Kable' and Pipe Syntax*. https://CRAN.R-project.org/package=kableExtra.