NZ GREEN Grid Household Power Demand Profiles: Lighting

Data extraction and preliminary plots

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

Last run at: 2018-06-08 12:47:48

Contents

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
## 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>
```

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

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 Lighting 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: Lighting_2015-04-01_2016-03-31_observations.csv

```
## [1] "/Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/dataExtracts/Lighti
## [1] "# Loaded 10,378,657 rows of data"
```

The following table summarises the Lighting data we have found.

Summary of household grid spy data for: Lighting

hhID

nObs

nHouseholds

nCircuits

meanPower

minDate

maxDate

rf_06

523312

1

1

147.12939

2015-04-01

2016-03-31

rf_22

526097

1

1

289.65706

2015-04-01

2016-03-31

rf_28

79085

1

1

33.51889

2015-04-01

2015-05-26

rf_29

526780

1

1

88.79616

2015-04-01

2016-03-31

rf_30

477491

1

1

98.61175

2015-04-01

2016-03-31

rf_31

526878

1

1

60.91970

2015-04-01

2016-03-31

rf_32

526785

1

1

22.27559

2015-04-01

2016-03-31

rf_33

526863

1

1

33.51870

2015-04-01

2016-03-31

rf_34

526677

1

1

91.04671

2015-04-01

2016-03-31

rf_35

327974

1

1

80.94829

2015-04-01

2015-11-14

rf_36

516242

1

1

47.66327

2015-04-01

2016-03-31

rf_37

526771

1

1

25.13406

2015-04-01

2016-03-31

rf_38

373722

1

1

60.18270

2015-04-01

2015-12-26

rf_39

495806

1

1

270.60561

2015-04-01

2016-03-31

rf_40

338289

1

1

145.15008

2015-04-01

2015-11-22

rf_41

223824

1

1

223.98446

2015-04-01

2015-11-12

rf_42

518179

1

1

350.58808
2015-04-01
2016-03-31
rf_43
288838
1
1
103.17964
2015-04-01
2015-10-18
rf_44
526850
1
1
92.66229
2015-04-01
2016-03-31
rf_45
526110
1
1
73.12970
2015-04-01
2016-03-31
rf_46
950976
1
2
95.92889
2015-04-01
2016-03-31
rf_47
525108
1
1
22.58957

2015-04-01

2016-03-31

This table will have a large number (10,378,657) of obserations caused by the number of different circuit labels as shown by the following table.

Counts of Lighting observations by label and household

rf_06

rf_22

rf_28

rf_29

rf_30

rf_31

rf_32

rf_33

rf_34

rf_35

rf_36

rf_37

rf_38

rf_39

rf_40

rf_41

rf_42

rf_43

rf_44

rf_45

rf_46

rf_47

Lighting (inc heat lamps)$4129

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

518179

0

0

0

0

0

Lighting & 2 Towel Rail$4245

0

0

0

0

0

0

0

0

0

0

0

0

495806

0

0

0

0

0

0

0

0

Lighting$2232

0

526097

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$2244

523312

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4123

0

0

0

0

0

0

0

0

0

327974

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4133

0

0

0

0

0

0

0

0

0

0

0

526771

0

0

0

0

0

0

0

0

0

0

Lighting$4142

0

0

0

0

0

0

0

526863

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4149

0

0

0

0

0

0

0

0

0

0

516242

0

0

0

0

0

0

0

0

0

0

0

Lighting$4153

0

0

0

0

0
0
0
0
0
0
0
0
0
0
0
0
0
0
526850
0
0
0
Lighting$4159
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0

0

0

526110

0

0

Lighting$4165

0

0

0

0

0

0

0

0

0

0

0

0

0

338289

0

0

0

0

0

0

0

0

Lighting$4172

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

525108

Lighting$4176

0

0

0

0

0

0

0

0

0

0

0

0

373722

0

0

0

0

0

0

0

0

0

0

Lighting$4183

0

0

0

526780

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4189

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

223824

0

0

0

0

0

0

Lighting$4197

0

0

0

0

0

0

526785

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4203

0

0

0

0

0

526878

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4212

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

288838

0

0

0

0

Lighting$4218

0

0

79085

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4222

0

0

0

0

0

0

0

0

526677

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4233

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

486982

0

Lighting$4236

0

0

0

0

477491

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

Lighting$4404

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

463994

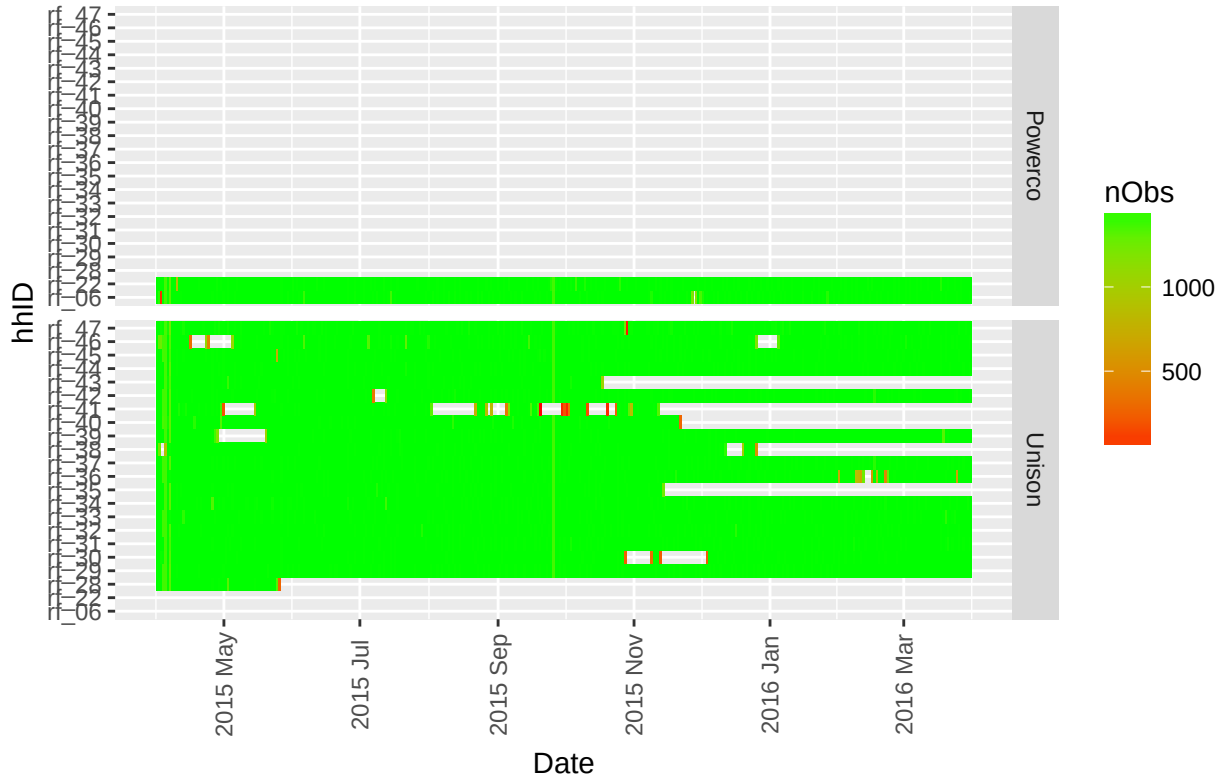
0

Note that some households may have more than one Lighting circuit.

4.3 Test Lighting data

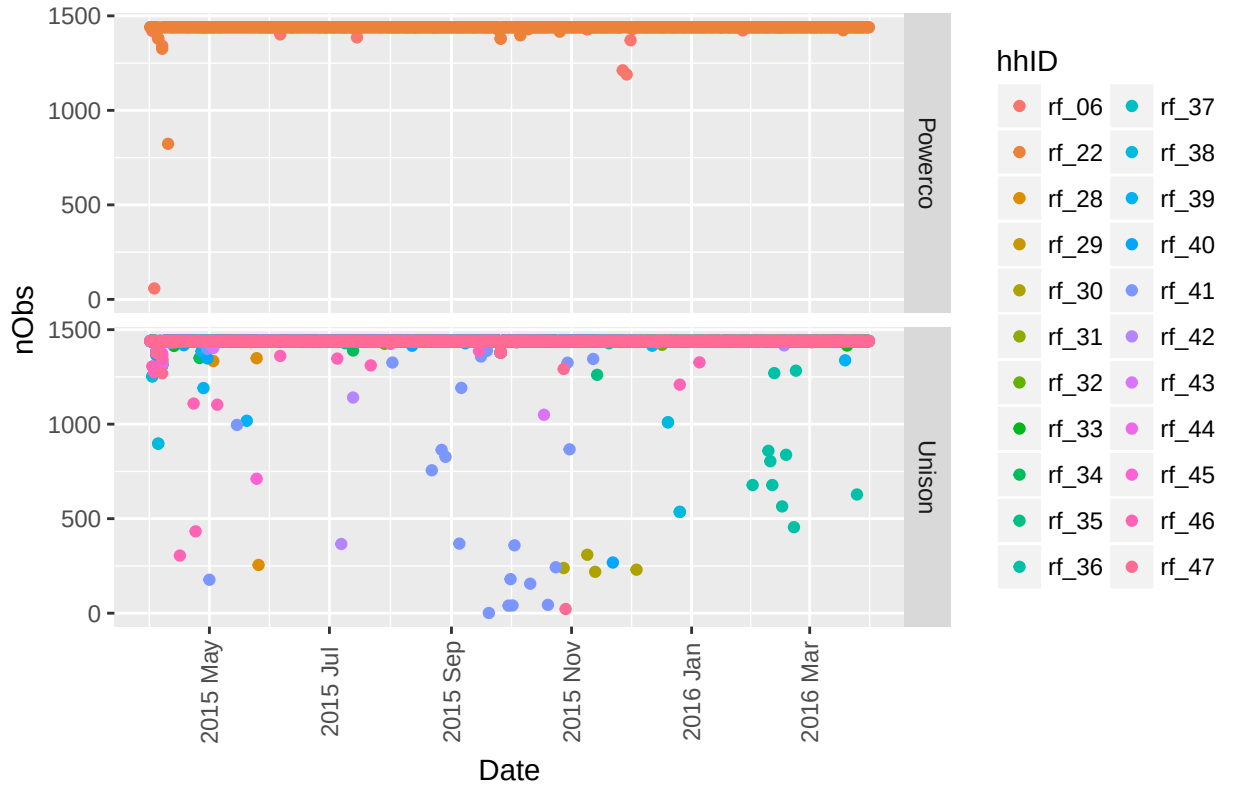
This section tests the availability of Lighting data by replicating the standard data quality checks used for the whole gridSpy dataset.

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



ɹrce: /Volumes/hum−csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/
Circuits: Lighting 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: Lighting 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!

Summary observation stats by hhID (sorted by date last heard from) for: Lighting

hhID

sample

nObs

minDate

maxDate

rf_28

Unison

79085

2015-04-01T00:00:00Z

2015-05-26T04:56:00Z

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_06

Powerco

523312

2015-04-01T00:00:00Z

2016-03-31T23:59:00Z

rf_22

Powerco

526097

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_30

Unison

477491

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_39

Unison

495806

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

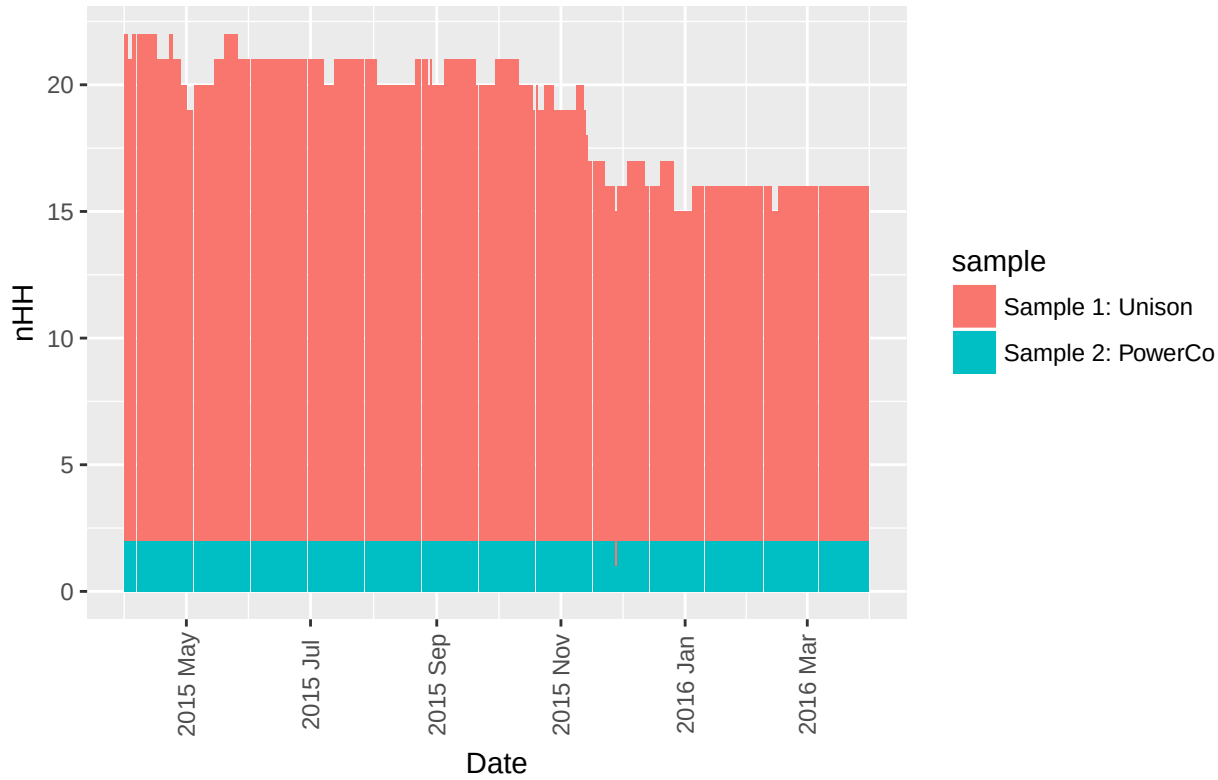
Unison

525108

2015-04-01T00:00:00Z

2016-03-31T23:59:00Z

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



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

The following table summarises the Lighting data. Any surprises?

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

Summary of Lighting circuits

hhID </th>

r_dateTime

circuit

powerW </th>

obsHourMin

Length:10378657

Length:10378657

Length:10378657

Min. :-133.8

Length:10378657

Class :character

Class :character

Class :character

1st Qu.: 0.0

Class :character

Mode :character

Mode :character

Mode :character

Median : 0.0

Mode :character

NA

NA

NA

Mean : 111.5

NA

NA

NA

NA

3rd Qu.: 102.5

NA

NA

NA

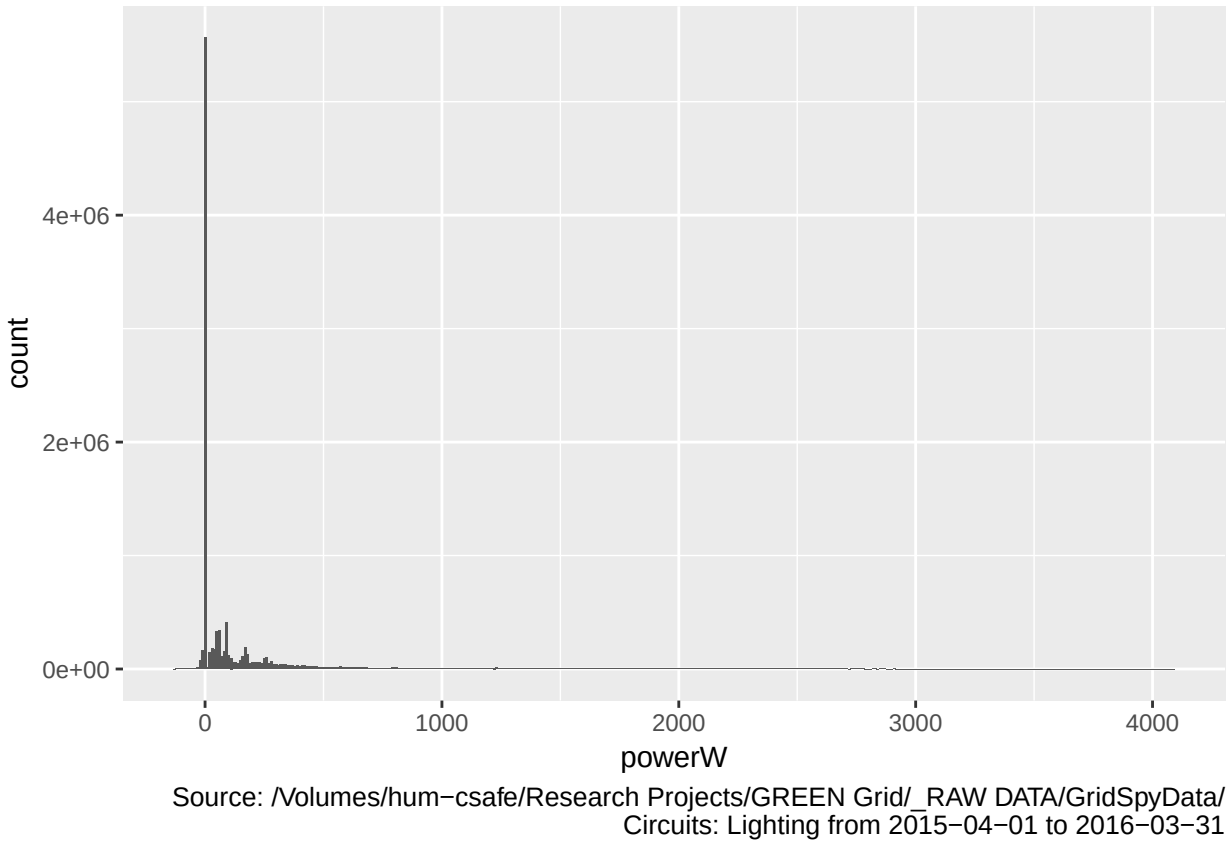
NA

Max. :4087.4

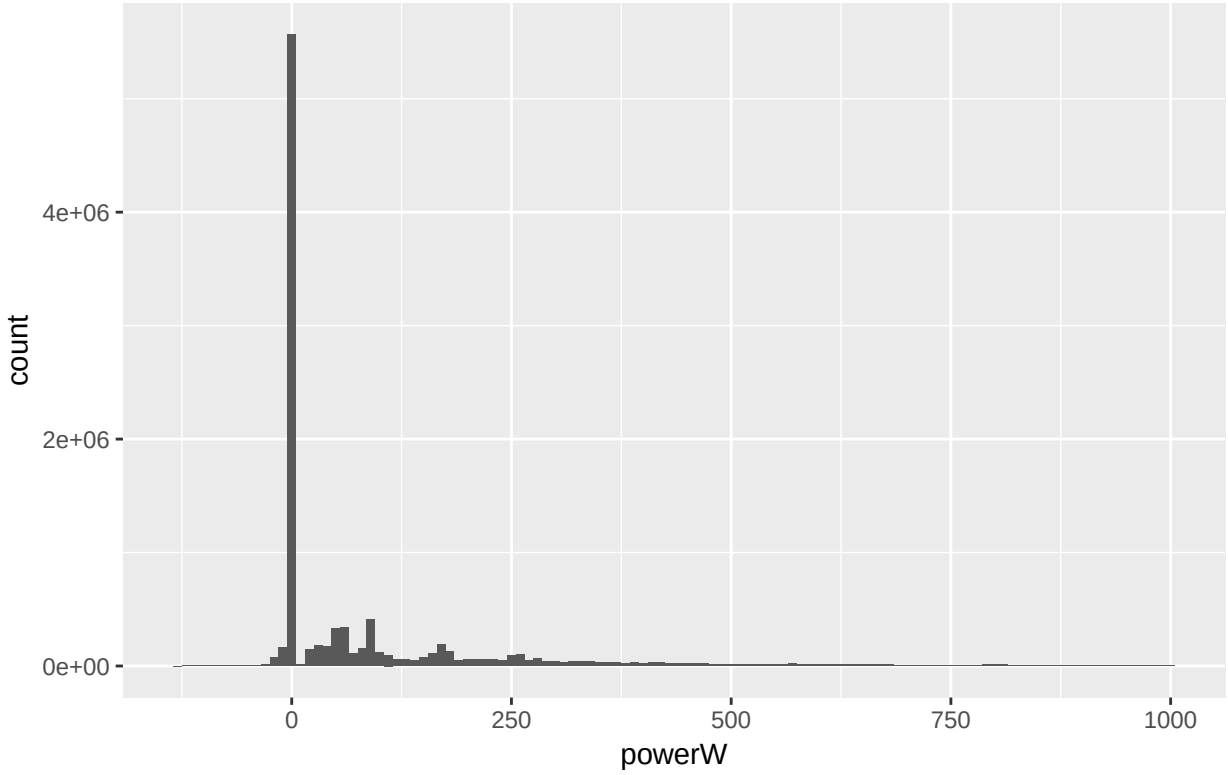
NA

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?

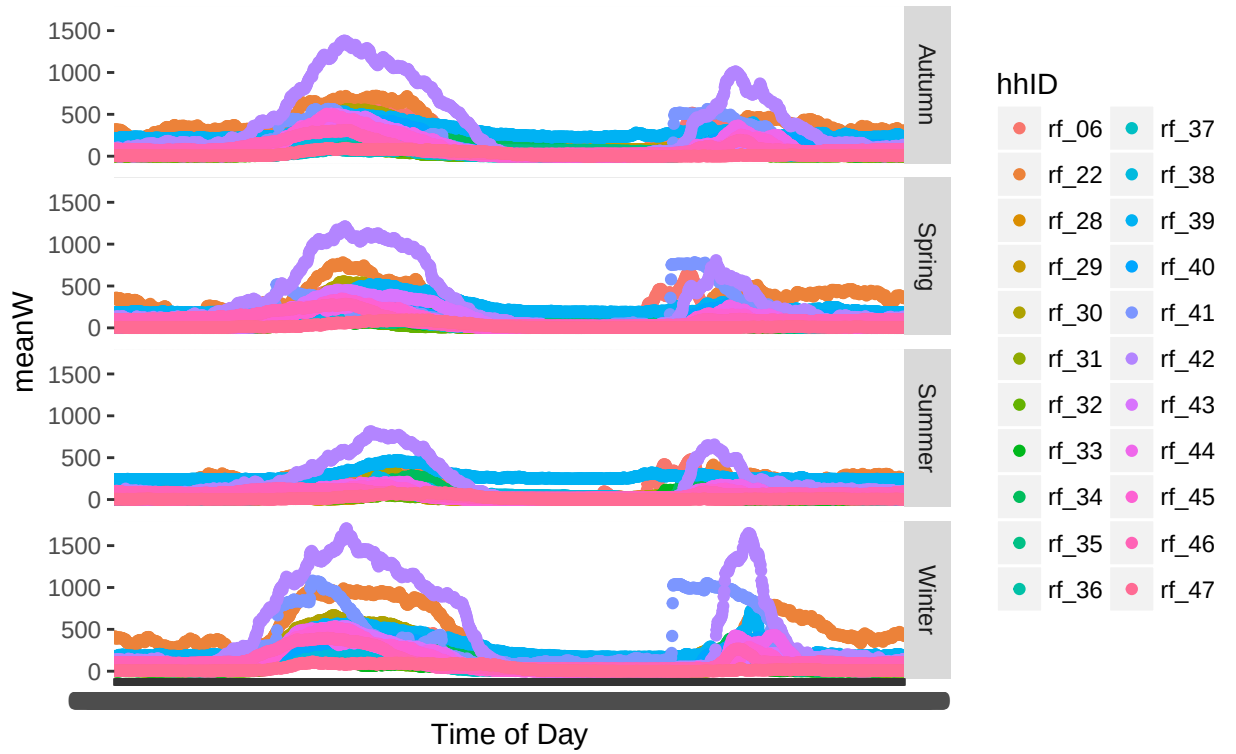
5 Lighting profiles

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: Lighting from 2015−04−01 to 2016−03−31
n households = 22

```
## [1] "Saving plot to Lighting_2015-04-01_2016-03-31_byHouseholdSeasonalProfilePlot.png"
## [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/L
```

Summary of household level mean profiles for Lighting

```
 hhID </th>
```

obsHourMin

```
season </th>
 meanW </th>
```

Length:116640

Length:116640

Length:116640

Min. : 0.00

Class :character

Class :character

Class :character

1st Qu.: 10.87

Mode :character

Mode :character

Mode :character

Median : 46.08

NA

NA

NA

Mean : 112.24

NA

NA

NA

3rd Qu.: 140.39

NA

NA

NA

Max. :1705.55

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

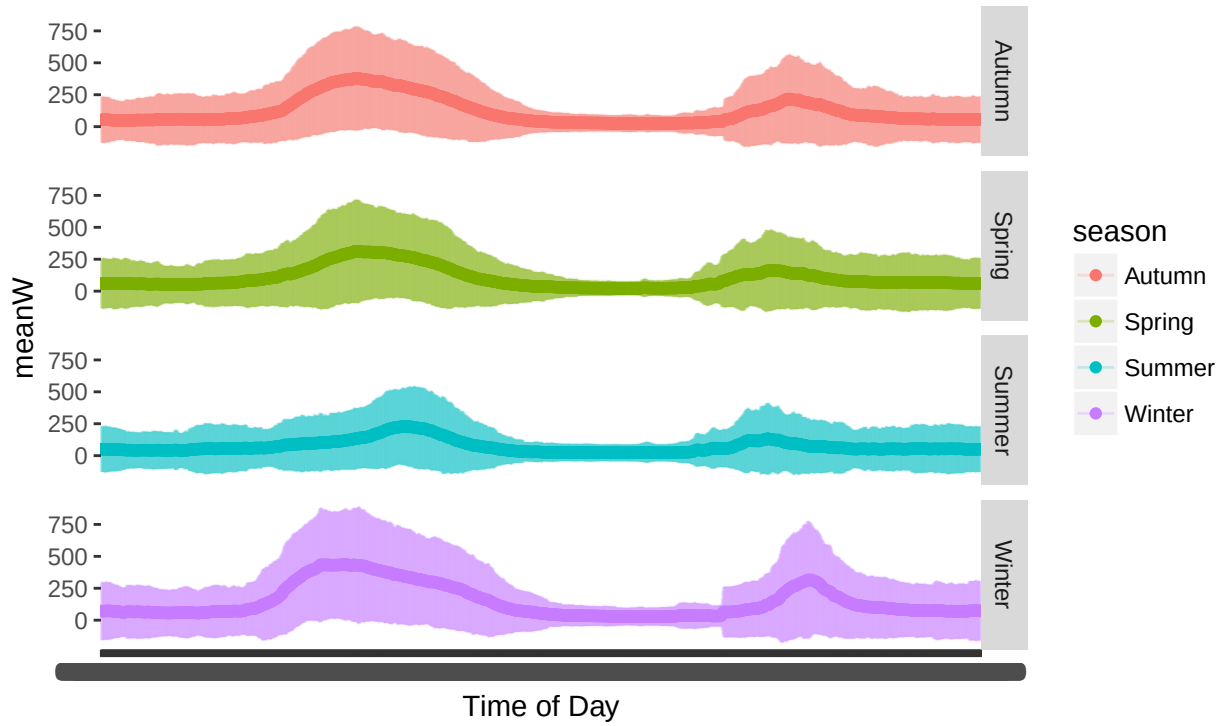
Note that the code saves a high definition version of the plot and the profiles for future re-use.

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`
  or
- `dt <- data.table::as.data.table(readr::read_csv("/Volumes/hum-csafe/Research Projects/GREEN Grid/Clean_data/safe/gridSpy/1min/profiles/Lighting_2015-04-01_2016-03-31_byHouseholdSeasonalProfil`
  if you prefer `data.table`

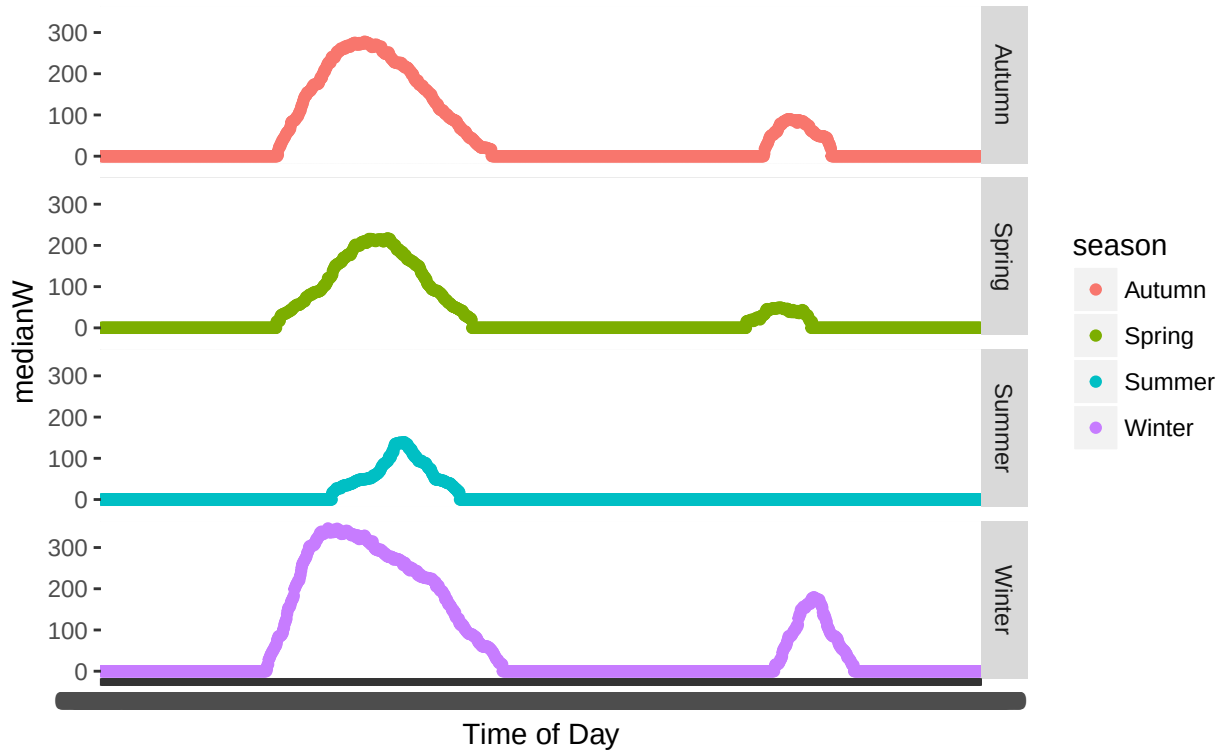
5.2 Profile plots: overall household mean

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: Lighting from 2015−04−01 to 2016−03−31
n households = 22
Error bars = +/− 1 s.d.

```
## [1] "Saving plot to Lighting_2015-04-01_2016-03-31_overallMeanSeasonalProfilePlot.png"
```



e: /Volumes/hum−csafe/Research Projects/GREEN Grid/_RAW DATA/GridSpyData/
Circuits: Lighting from 2015−04−01 to 2016−03−31
n households = 22

```
## [1] "Saving plot to Lighting_2015-04-01_2016-03-31_overallMedianSeasonalProfilePlot.png"
## [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/L
```

Summary of overall profiles for Lighting

obsHourMin

```
season </th>
 meanW </th>
medianW </th>
  nObs </th>
  sdW </th>
```

Length:5760

Length:5760

Min. : 21.72

Min. : 0.00

Min. :1525

Min. : 53.32

Class :character

Class :character

1st Qu.: 49.81

1st Qu.: 0.00

1st Qu.:1734

1st Qu.:147.01

Mode :character

Mode :character

Median : 67.50

Median : 0.00

Median :1836

Median :199.94

NA

NA

Mean :109.21

Mean : 37.48

Mean :1802

Mean :207.87

NA

NA

3rd Qu.:135.72

3rd Qu.: 32.30

3rd Qu.:1900

3rd Qu.:268.93

NA

NA

Max. :437.12

Max. :346.33

Max. :1996

Max. :460.51

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 443.95 seconds (7.4 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: /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] rmarkdown_1.9     kableExtra_0.9.0  knitr_1.20        readr_1.1.1
## [5] ggplot2_2.2.1     dplyr_0.7.5       data.table_1.11.2 nzGREENGrid_0.1.0
##
## loaded via a namespace (and not attached):
##  [1] progress_1.1.2      tinytex_0.5         tidyselect_0.2.4
##  [4] reshape2_1.4.3      purrr_0.2.4         splines_3.5.0
##  [7] lattice_0.20-35     colorspace_1.3-2    htmltools_0.3.6
## [10] viridisLite_0.3.0   yaml_2.1.19         base64enc_0.1-3
## [13] survival_2.42-3     rlang_0.2.0         pillar_1.2.2
## [16] foreign_0.8-70      glue_1.2.0          RColorBrewer_1.1-2
## [19] readxl_1.1.0        bindrcpp_0.2.2      bindr_0.1.1
## [22] plyr_1.8.4          stringr_1.3.1       munsell_0.4.3
## [25] gtable_0.2.0        cellranger_1.1.0    rvest_0.3.2
## [28] htmlwidgets_1.2     evaluate_0.10.1     labeling_0.3
## [31] latticeExtra_0.6-28 htmlTable_1.11.2    highr_0.6
## [34] Rcpp_0.12.17        acepack_1.4.1       checkmate_1.8.5
## [37] backports_1.1.2     scales_0.5.0        Hmisc_4.1-1
## [40] gridExtra_2.3       hms_0.4.2           digest_0.6.15
## [43] stringi_1.2.2       grid_3.5.0          rprojroot_1.3-2
## [46] tools_3.5.0         magrittr_1.5        lazyeval_0.2.1
## [49] tibble_1.4.2        cluster_2.0.7-1     Formula_1.2-3
## [52] pkgconfig_2.0.1     Matrix_1.2-14       xml2_1.2.0
## [55] prettyunits_1.0.2   lubridate_1.7.4     assertthat_0.2.0
```

```
## [58] httr_1.3.1          rstudioapi_0.7      rpart_4.1-13
## [61] R6_2.2.2            nnet_7.3-12         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.

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