

# Lab 2

March 21, 2022

## 1 STATS 765 Lab 2

Chase Robertson  
cro873

### 1.1 Task 1: Examine the Data

#### 1.1.1 Problems with the data

The `simulation_output_raw.xlsx` file contains empty leading rows and columns, has some merged cells, occasionally contains a row with a group name like ‘Sensible Cooling’ that should apply to the following rows, and has some missing values.

#### 1.1.2 How to address the problems

I would address the challenges above by reorganising the excel file into a more strictly manageable format. I would start by removing the leading rows and columns. Then I would new columns for the climate type, simulation control group, building size, and building material. There is no need to address the missing values, as they suggest that we will simply not have any rows with both ‘sensible heating’ control group and ‘large’ office size.

### 1.2 Task 2: Clean the data

First let’s load the data using `read_xlsx`

```
[1]: suppressPackageStartupMessages(library(tidyverse))

raw.df <- readxl::read_xlsx('simulation_output_raw.xlsx', skip=7, col_names=F)
head(raw.df)
```

New names:

```
* `` -> ...1
* `` -> ...2
* `` -> ...3
* `` -> ...4
* `` -> ...5
* ...
```

	...1 <chr>	...2 <chr>	...3 <dbl>	...4 <dbl>	...5 <dbl>
A tibble: 6 × 8	Very Hot Humid (Honolulu, HI)	Heating energy [GJ]	0.000	0.000	0.010
	NA	Cooling energy [GJ]	47.070	47.350	639.2
	NA	Total Cosumption [GJ]	162.470	162.750	1792
	NA	Sensible Cooling	NA	NA	NA
	NA	Calculated Design Air Flow [m3/s]	0.256	0.257	3.468
	NA	User Design Air Flow [m3/s]	0.308	0.309	3.468

Nice! `read_xlsx` took care of those empty columns automatically. Now let's name the columns something more sensible.

```
[2]: df <- raw.df
names(df) <- c("weather", "description",
               "small-concrete", "small-curtain_wall",
               "medium-concrete", "medium-curtain_wall",
               "large-concrete", "large-curtain_wall")
names(df)
```

1. 'weather' 2. 'description' 3. 'small-concrete' 4. 'small-curtain\_wall' 5. 'medium-concrete' 6. 'medium-curtain\_wall' 7. 'large-concrete' 8. 'large-curtain\_wall'

```
[3]: x1 <- as.data.frame(matrix(rep(NA, length(df)), nrow=1))
names(x1) <- names(df)
df2 <- bind_rows(x1, df)
head(df2)
```

	weather <chr>	description <chr>	small-concrete <dbl>
A data.frame: 6 × 8	1 NA	NA	NA
	2 Very Hot Humid (Honolulu, HI)	Heating energy [GJ]	0.000
	3 NA	Cooling energy [GJ]	47.070
	4 NA	Total Cosumption [GJ]	162.470
	5 NA	Sensible Cooling	NA
	6 NA	Calculated Design Air Flow [m3/s]	0.256

Great, now we can create a new group column based on the 'Sensible XXXXing' grouping we see in the .xlsx file.

```
[4]: df2 <- df2 %>%
      mutate(group = ifelse(is.na(description), 'overall',
                           ifelse(description=='Sensible Cooling', 'sensible_cooling',
                                   ifelse(description=='Sensible Heating',
                                           'sensible_heating', NA))))
df2$group
```

1. 'overall' 2. NA 3. NA 4. NA 5. 'sensible\_cooling' 6. NA 7. NA 8. NA 9. NA 10. NA 11. NA 12. NA 13. 'sensible\_heating' 14. NA 15. NA 16. NA 17. NA 18. NA 19. NA 20. NA 21. 'overall' 22. NA 23. NA 24. NA 25. 'sensible\_cooling' 26. NA 27. NA 28. NA 29. NA 30. NA 31. NA 32. NA 33. 'sensible\_heating' 34. NA 35. NA 36. NA 37. NA 38. NA 39. NA 40. NA 41. 'overall'

42. NA 43. NA 44. NA 45. 'sensible\_cooling' 46. NA 47. NA 48. NA 49. NA 50. NA 51. NA  
 52. NA 53. 'sensible\_heating' 54. NA 55. NA 56. NA 57. NA 58. NA 59. NA 60. NA 61. 'overall'  
 62. NA 63. NA 64. NA 65. 'sensible\_cooling' 66. NA 67. NA 68. NA 69. NA 70. NA 71. NA  
 72. NA 73. 'sensible\_heating' 74. NA 75. NA 76. NA 77. NA 78. NA 79. NA 80. NA 81. 'overall'  
 82. NA 83. NA 84. NA 85. 'sensible\_cooling' 86. NA 87. NA 88. NA 89. NA 90. NA 91. NA 92. NA  
 93. 'sensible\_heating' 94. NA 95. NA 96. NA 97. NA 98. NA 99. NA 100. NA 101. 'overall' 102. NA  
 103. NA 104. NA 105. 'sensible\_cooling' 106. NA 107. NA 108. NA 109. NA 110. NA 111. NA  
 112. NA 113. 'sensible\_heating' 114. NA 115. NA 116. NA 117. NA 118. NA 119. NA 120. NA  
 121. 'overall' 122. NA 123. NA 124. NA 125. 'sensible\_cooling' 126. NA 127. NA 128. NA 129. NA  
 130. NA 131. NA 132. NA 133. 'sensible\_heating' 134. NA 135. NA 136. NA 137. NA 138. NA  
 139. NA 140. NA 141. 'overall' 142. NA 143. NA 144. NA 145. 'sensible\_cooling' 146. NA 147. NA  
 148. NA 149. NA 150. NA 151. NA 152. NA 153. 'sensible\_heating' 154. NA 155. NA 156. NA  
 157. NA 158. NA 159. NA 160. NA 161. 'overall' 162. NA 163. NA 164. NA 165. 'sensible\_cooling'  
 166. NA 167. NA 168. NA 169. NA 170. NA 171. NA 172. NA 173. 'sensible\_heating' 174. NA  
 175. NA 176. NA 177. NA 178. NA 179. NA 180. NA

And now we can fill the group and weather column's NAs using `na.locf`.

```
[5]: suppressPackageStartupMessages(library(zoo))
```

```
group.vt <- zoo(df2$group)
df2$group <- na.locf(group.vt)
df2$group <- as.character(df2$group)

head(df2)
```

		weather <chr>	description <chr>	small-concrete <dbl>
A data.frame: 6 × 9	1	NA	NA	NA
	2	Very Hot Humid (Honolulu, HI)	Heating energy [GJ]	0.000
	3	NA	Cooling energy [GJ]	47.070
	4	NA	Total Cosumption [GJ]	162.470
	5	NA	Sensible Cooling	NA
	6	NA	Calculated Design Air Flow [m3/s]	0.256

```
[6]: num_na_per_row.vt <- apply(df2, 1, function(x){
      sum(is.na(x))
    })
num_na_per_row.vt
```

1. 8 2. 0 3. 1 4. 1 5. 7 6. 1 7. 1 8. 1 9. 1 10. 1 11. 1 12. 1 13. 7 14. 3 15. 3 16. 3 17. 3 18. 3 19. 3  
 20. 3 21. 8 22. 0 23. 1 24. 1 25. 7 26. 1 27. 1 28. 1 29. 1 30. 1 31. 1 32. 1 33. 7 34. 3 35. 3 36. 3 37. 3  
 38. 3 39. 3 40. 3 41. 8 42. 0 43. 1 44. 1 45. 7 46. 1 47. 1 48. 1 49. 1 50. 1 51. 1 52. 1 53. 7 54. 3 55. 3  
 56. 3 57. 3 58. 3 59. 3 60. 3 61. 8 62. 0 63. 1 64. 1 65. 7 66. 1 67. 1 68. 1 69. 1 70. 1 71. 1 72. 1 73. 7  
 74. 3 75. 3 76. 3 77. 3 78. 3 79. 3 80. 3 81. 8 82. 0 83. 1 84. 1 85. 7 86. 1 87. 1 88. 1 89. 1 90. 1 91. 1  
 92. 1 93. 7 94. 3 95. 3 96. 3 97. 3 98. 3 99. 3 100. 3 101. 8 102. 0 103. 1 104. 1 105. 7 106. 1 107. 1  
 108. 1 109. 1 110. 1 111. 1 112. 1 113. 7 114. 3 115. 3 116. 3 117. 3 118. 3 119. 3 120. 3 121. 8 122. 0  
 123. 1 124. 1 125. 7 126. 1 127. 1 128. 1 129. 1 130. 1 131. 1 132. 1 133. 7 134. 3 135. 3 136. 3 137. 3  
 138. 3 139. 3 140. 3 141. 8 142. 0 143. 1 144. 1 145. 7 146. 1 147. 1 148. 1 149. 1 150. 1 151. 1 152. 1

153. 7 154. 3 155. 3 156. 3 157. 3 158. 3 159. 3 160. 3 161. 8 162. 0 163. 1 164. 1 165. 7 166. 1 167. 1  
168. 1 169. 1 170. 1 171. 1 172. 1 173. 7 174. 3 175. 3 176. 3 177. 3 178. 3 179. 3 180. 3

```
[7]: df2 <- df2[which(num_na_per_row.vt<7),]
```

```
[8]: weather.vt <- zoo(df2$weather)
df2$weather <- na.locf(weather.vt, na.rm=F)
df2$weather <- as.character(df2$weather)
```

Now finish up our cleaning by removing the leading row and group demarcation rows, and pivoting the office\_size/office\_type columns into their own two columns.

```
[9]: clean.df <- df2 %>%
  filter(!is.na(weather), !grepl('^Sensible', description)) %>%
  pivot_longer(3:8,
    names_to=c('office_size', 'office_type'),
    names_sep="-",
    values_to='value',
    values_transform=list(value=as.double),
    values_drop_na=T)

clean.df
```

weather <chr>	description <chr>
Very Hot Humid (Honolulu, HI)	Heating energy [GJ]
Very Hot Humid (Honolulu, HI)	Heating energy [GJ]
Very Hot Humid (Honolulu, HI)	Heating energy [GJ]
Very Hot Humid (Honolulu, HI)	Heating energy [GJ]
Very Hot Humid (Honolulu, HI)	Heating energy [GJ]
Very Hot Humid (Honolulu, HI)	Heating energy [GJ]
Very Hot Humid (Honolulu, HI)	Cooling energy [GJ]
Very Hot Humid (Honolulu, HI)	Cooling energy [GJ]
Very Hot Humid (Honolulu, HI)	Cooling energy [GJ]
Very Hot Humid (Honolulu, HI)	Cooling energy [GJ]
Very Hot Humid (Honolulu, HI)	Cooling energy [GJ]
Very Hot Humid (Honolulu, HI)	Cooling energy [GJ]
Very Hot Humid (Honolulu, HI)	Total Cosumption [GJ]
Very Hot Humid (Honolulu, HI)	Total Cosumption [GJ]
Very Hot Humid (Honolulu, HI)	Total Cosumption [GJ]
Very Hot Humid (Honolulu, HI)	Total Cosumption [GJ]
Very Hot Humid (Honolulu, HI)	Total Cosumption [GJ]
Very Hot Humid (Honolulu, HI)	Total Cosumption [GJ]
Very Hot Humid (Honolulu, HI)	Calculated Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	Calculated Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	Calculated Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	Calculated Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	Calculated Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	Calculated Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	User Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	User Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	User Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	User Design Air Flow [m3/s]
Very Hot Humid (Honolulu, HI)	User Design Air Flow [m3/s]
A tibble: 792 × 6 Very Hot Humid (Honolulu, HI)	User Design Air Flow [m3/s]
Very Cold (International Falls, MN)	Outdoor Humidity Ratio at Peak Load [kgWater/kgAir]
Very Cold (International Falls, MN)	Outdoor Humidity Ratio at Peak Load [kgWater/kgAir]
Very Cold (International Falls, MN)	Calculated Design Air Flow [m3/s]
Very Cold (International Falls, MN)	Calculated Design Air Flow [m3/s]
Very Cold (International Falls, MN)	Calculated Design Air Flow [m3/s]
Very Cold (International Falls, MN)	Calculated Design Air Flow [m3/s]
Very Cold (International Falls, MN)	User Design Air Flow [m3/s]
Very Cold (International Falls, MN)	User Design Air Flow [m3/s]
Very Cold (International Falls, MN)	User Design Air Flow [m3/s]
Very Cold (International Falls, MN)	User Design Air Flow [m3/s]
Very Cold (International Falls, MN)	Thermostat Setpoint Temperature at Peak Load [C]
Very Cold (International Falls, MN)	Thermostat Setpoint Temperature at Peak Load [C]
Very Cold (International Falls, MN)	Thermostat Setpoint Temperature at Peak Load [C]
Very Cold (International Falls, MN)	Thermostat Setpoint Temperature at Peak Load [C]
Very Cold (International Falls, MN)	Indoor Temperature at Peak Load [C]
Very Cold (International Falls, MN)	Indoor Temperature at Peak Load [C]
Very Cold (International Falls, MN)	Indoor Temperature at Peak Load [C]
Very Cold (International Falls, MN)	Indoor Temperature at Peak Load [C]
Very Cold (International Falls, MN)	Indoor Humidity Ratio at Peak Load [kgWater/kgAir]
Very Cold (International Falls, MN)	Indoor Humidity Ratio at Peak Load [kgWater/kgAir]

## 1.3 Task 3: Exploration using {ggplot2}

### 1.3.1 Question 1

For small office with sensible cooling in place, how does the outdoor temperature at peak load differ by weather? Does the pattern vary by office type?

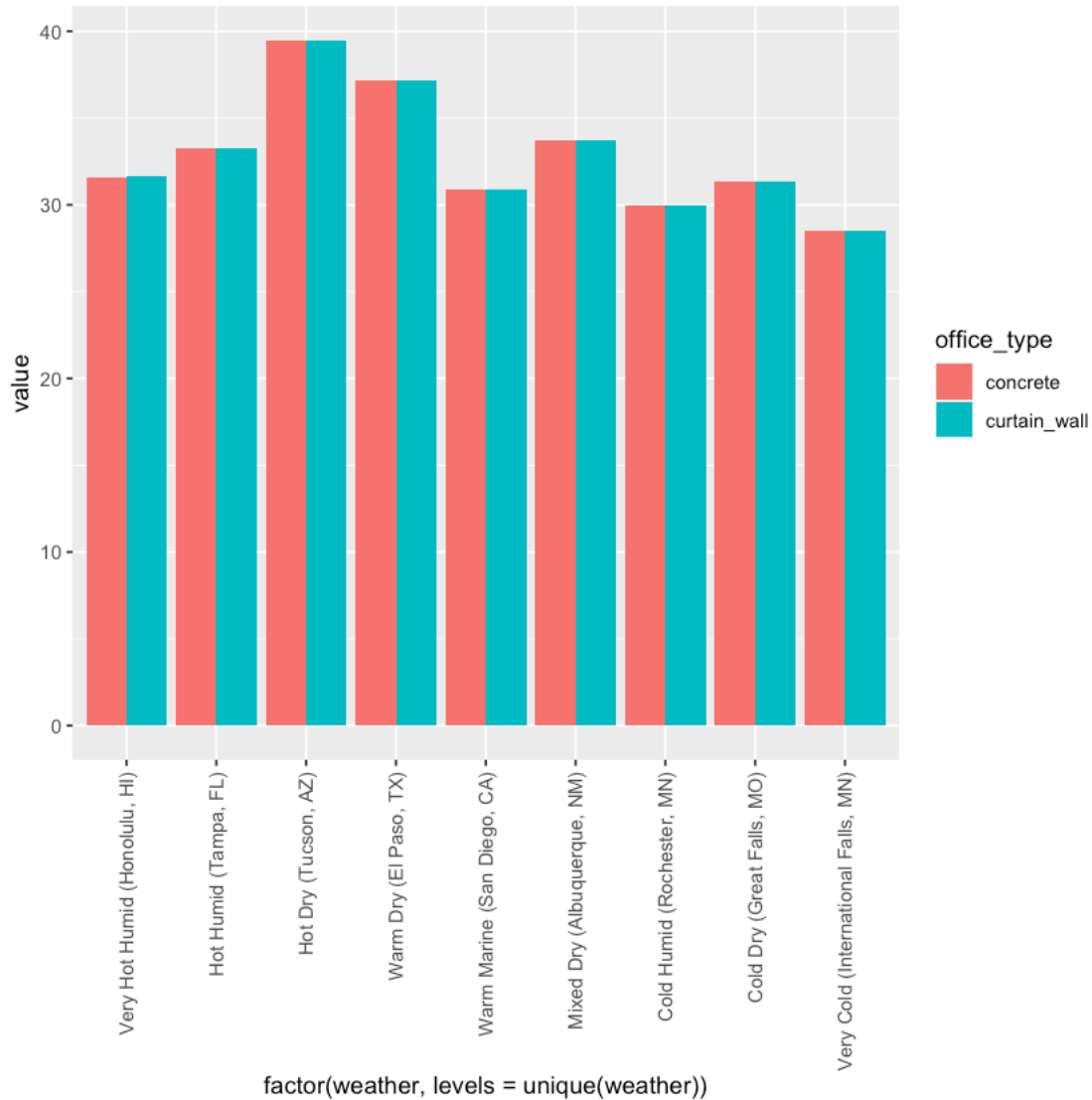
### 1.3.2 Analysis

First we filter down to the rows that apply to our question: only the Outdoor Temperature at Peak Load of small offices in the cooling group.

```
[10]: library(ggplot2)
      sm_cool <- clean.df %>%
        filter(office_size=='small', group=='sensible_cooling',
              grepl('Outdoor Temperature at Peak Load', description))
```

Then we can plot. I chose to plot on a barchart ordered by local climate, with different building materials colored, so that we can easily see at a glance how office type and weather affect our data.

```
[11]: sm_cool %>%
      ggplot(aes(x=factor(weather, levels=unique(weather)),
                y=value, fill=office_type)) +
      geom_bar(position="dodge", stat="identity") +
      theme(axis.text.x=element_text(angle=90, hjust=1, vjust=0.5))
```



We see that the outdoor temperature at peak load differ greatly by climate, with drier and hotter climates hitting especially high maximums. We also see that the office type, or building material, does not have any significant difference.

### 1.3.3 Question 2

Do any of these factors affect the total energy consumption: office\_type; office\_size and weather? If so, which one has the most significant impact?

### 1.3.4 Analysis

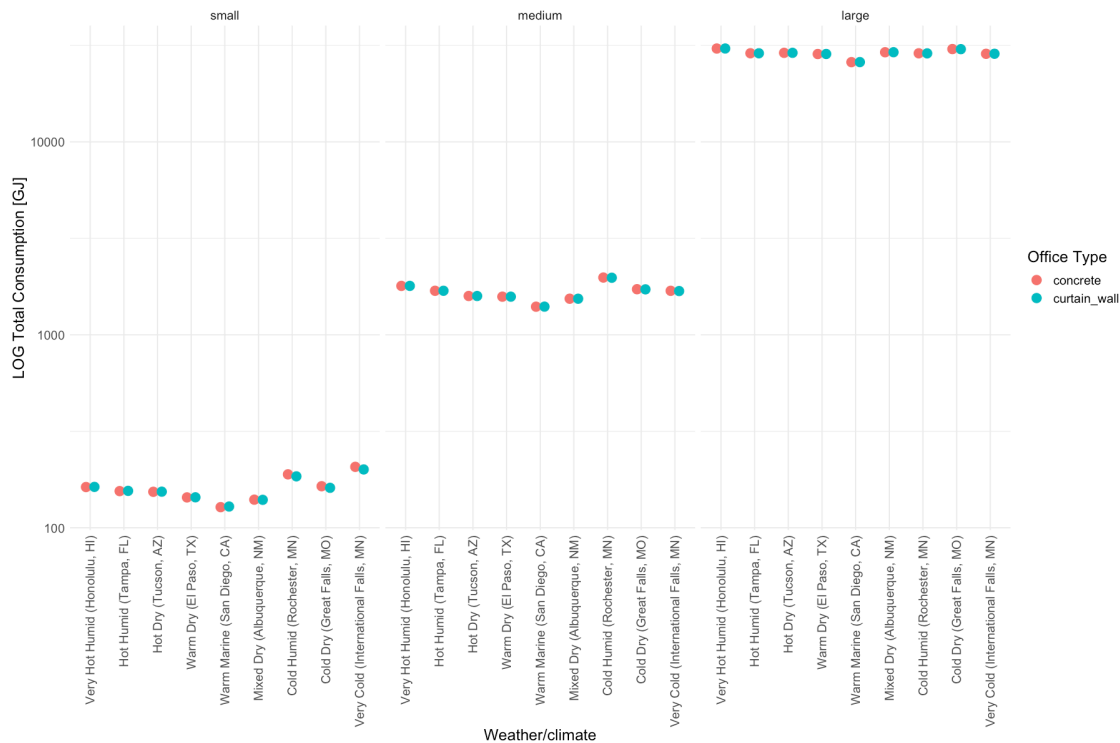
First we filter down to rows that are relevant to the question at hand: those with a Total Consumption [GJ] value.

```
[12]: consumption <- clean.df %>%
      filter(grepl('Total Cosumption', description))
```

Now we create a new chart to illuminate the most significant factors that affect total consumption.

```
[13]: # make the plot wider
options(repr.plot.width=15, repr.plot.height=10)

ggplot(consumption,
      aes(x = factor(weather, levels = unique(weather)),
          y = value, colour = factor(office_type))) +
facet_wrap(~factor(office_size, levels = unique(office_size))) +
# put it on a logarithmic scale!
scale_y_continuous(trans='log10') +
geom_point(size=4, position=position_dodge(0.5)) +
theme_minimal() +
theme(text = element_text(size=15),
      axis.text.x = element_text(angle=90, hjust=1, vjust=0.5)) +
labs(x = 'Weather/climate',
     y = 'LOG Total Consumption [GJ]',
     colour = 'Office Type')
```



It can be seen that the most important factor which affects total energy consumption (by orders of magnitude!) is the size of the office. Please note that the y-axis is on a **logarithmic scale**.



We see also that a combination of cold and humidity in an office's climate has a positive effect on consumption, with the relative effect being strongest on the smallest buildings. Conversely, offices in warm marine climates have the lowest consumption among offices of the same size.

Building material does not have a notable impact on total consumption.

#### **1.3.5 EOF**