

Arranging logger data from HoboLink

Jens Åström

26 June, 2020

Contents

Intro	1
Read in data	1
Package this into a funtion	5
Check the data out	7

Intro

The data exports for the temperature and humidity MX loggers from Hobo needs a bit of data wrangling before it can be used. The different data streams from each logger all get a separate column. Here we develop a script to turn this into a more usable long format.

Read in data

I have a single export with many loggers, as a csv file.

```
inputFile <- "../rawData/All_MX_2020_2020_06_26_09_20_43_UTC_1.csv"

rawDat <- read_csv(inputFile,col_types = cols(.default = "c"))

dat <- rawDat %>%
  select(-"Line#") %>%
  mutate(date = as.POSIXct(Date, format = "%m/%d/%y %H:%M:%S")) %>%
  mutate_if(is_character, as.double) %>%
  select(-Date)
```

```
## Warning: NAs introduced by coercion
```

```
dat
```

```
## # A tibble: 6,889 x 64
##   `Temperature (M~`RH (MX-RH-2 20~`Dew Point (MX--`Temperature (M~
##           <dbl>           <dbl>           <dbl>           <dbl>
## 1             NA             NA             NA             NA
```

```

## 2          NA          NA          NA          NA
## 3          22.3        28.2        3.05        NA
## 4          NA          NA          NA          NA
## 5          24.7        25.6        3.69        NA
## 6          NA          NA          NA          NA
## 7          25.5        24.4        3.72        NA
## 8          NA          NA          NA          NA
## 9          26.1        24.0        3.95        NA
## 10         NA          NA          NA          NA
## # ... with 6,879 more rows, and 60 more variables: `RH (MX-RH-2
## # 20835815:20835815-2),`, `Dew Point (MX-TEMP-2
## # 20835815:20835815-4),`, `Temperature (MX-TEMP-2
## # 20835816:20835816-1),`, `RH (MX-RH-2
## # 20835816:20835816-2),`, `Dew Point (MX-TEMP-2
## # 20835816:20835816-4),`, `Temperature (MX-TEMP-2
## # 20835817:20835817-1),`, `RH (MX-RH-2
## # 20835817:20835817-2),`, `Dew Point (MX-TEMP-2
## # 20835817:20835817-4),`, `Temperature (MX-TEMP-2
## # 20835818:20835818-1),`, `RH (MX-RH-2
## # 20835818:20835818-2),`, `Dew Point (MX-TEMP-2
## # 20835818:20835818-4),`, `Temperature (MX-TEMP-2
## # 20835819:20835819-1),`, `RH (MX-RH-2
## # 20835819:20835819-2),`, `Dew Point (MX-TEMP-2
## # 20835819:20835819-4),`, `Temperature (MX-TEMP-2
## # 20835820:20835820-1),`, `RH (MX-RH-2
## # 20835820:20835820-2),`, `Dew Point (MX-TEMP-2
## # 20835820:20835820-4),`, `Temperature (MX-TEMP-2
## # 20835821:20835821-1),`, `RH (MX-RH-2
## # 20835821:20835821-2),`, `Dew Point (MX-TEMP-2
## # 20835821:20835821-4),`, `Temperature (MX-TEMP-2
## # 20835822:20835822-1),`, `RH (MX-RH-2
## # 20835822:20835822-2),`, `Dew Point (MX-TEMP-2
## # 20835822:20835822-4),`, `Temperature (MX-TEMP-2
## # 20835823:20835823-1),`, `RH (MX-RH-2
## # 20835823:20835823-2),`, `Dew Point (MX-TEMP-2
## # 20835823:20835823-4),`, `Temperature (MX-TEMP-2
## # 20835824:20835824-1),`, `RH (MX-RH-2
## # 20835824:20835824-2),`, `Dew Point (MX-TEMP-2
## # 20835824:20835824-4),`, `Temperature (MX-TEMP-2
## # 20835825:20835825-1),`, `RH (MX-RH-2
## # 20835825:20835825-2),`, `Dew Point (MX-TEMP-2
## # 20835825:20835825-4),`, `Temperature (MX-TEMP-2
## # 20843228:20843228-1),`, `RH (MX-RH-2
## # 20843228:20843228-2),`, `Dew Point (MX-TEMP-2
## # 20843228:20843228-4),`, `Temperature (MX-TEMP-2
## # 20843229:20843229-1),`, `RH (MX-RH-2

```

```
## # 20843229:20843229-2), %, 20843229` <dbl>, `Dew Point (MX-TEMP-2
## # 20843229:20843229-4), *C, 20843229` <dbl>, `Temperature (MX-TEMP-2
## # 20843230:20843230-1), *C, 20843230` <dbl>, `RH (MX-RH-2
## # 20843230:20843230-2), %, 20843230` <dbl>, `Dew Point (MX-TEMP-2
## # 20843230:20843230-4), *C, 20843230` <dbl>, `Temperature (MX-TEMP-2
## # 20843231:20843231-1), *C, 20843231` <dbl>, `RH (MX-RH-2
## # 20843231:20843231-2), %, 20843231` <dbl>, `Dew Point (MX-TEMP-2
## # 20843231:20843231-4), *C, 20843231` <dbl>, `Temperature (MX-TEMP-2
## # 20843233:20843233-1), *C, 20843233` <dbl>, `RH (MX-RH-2
## # 20843233:20843233-2), %, 20843233` <dbl>, `Dew Point (MX-TEMP-2
## # 20843233:20843233-4), *C, 20843233` <dbl>, `Temperature (MX-TEMP-2
## # 20843235:20843235-1), *C, 20843235` <dbl>, `RH (MX-RH-2
## # 20843235:20843235-2), %, 20843235` <dbl>, `Dew Point (MX-TEMP-2
## # 20843235:20843235-4), *C, 20843235` <dbl>, `Temperature (MX-TEMP-2
## # 20843236:20843236-1), *C, 20843236` <dbl>, `RH (MX-RH-2
## # 20843236:20843236-2), %, 20843236` <dbl>, `Dew Point (MX-TEMP-2
## # 20843236:20843236-4), *C, 20843236` <dbl>, `Temperature (MX-TEMP-2
## # 20843238:20843238-1), *C, 20843238` <dbl>, `RH (MX-RH-2
## # 20843238:20843238-2), %, 20843238` <dbl>, `Dew Point (MX-TEMP-2
## # 20843238:20843238-4), *C, 20843238` <dbl>, `Temperature (MX-TEMP-2
## # 20843239:20843239-1), *C, 20843239` <dbl>, `RH (MX-RH-2
## # 20843239:20843239-2), %, 20843239` <dbl>, `Dew Point (MX-TEMP-2
## # 20843239:20843239-4), *C, 20843239` <dbl>, date <dtm>
```

That's quite the number of columns...

We have to pivot this data set to a longer format. We also get rid of the rows with no data.

```
temp <- dat %>%
  pivot_longer(cols = starts_with("Temperature"),
               names_to = "logger",
               values_to = "temperature") %>%
  select(date,
         logger,
         temperature) %>%
  filter(!is.na(temperature))

rh <- dat %>%
  pivot_longer(cols = starts_with("RH"),
               names_to = "logger",
               values_to = "rh") %>%
  select(date,
         logger,
         rh) %>%
  filter(!is.na(rh))
```

```
dew <- dat %>%
  pivot_longer(cols = starts_with("Dew"),
               names_to = "logger",
               values_to = "dew") %>%
  select(date,
         logger,
         dew) %>%
  filter(!is.na(dew))
```

The data now looks like this

```
temp

## # A tibble: 6,889 x 3
##   date                logger                temperature
##   <dtm>              <chr>                <dbl>
## 1 2020-05-14 08:21:51 Temperature (MX-TEMP-2 20843236:2084323~ 20.8
## 2 2020-05-14 08:41:51 Temperature (MX-TEMP-2 20843236:2084323~ 22.2
## 3 2020-05-14 08:54:29 Temperature (MX-TEMP-2 20835814:2083581~ 22.3
## 4 2020-05-14 09:01:51 Temperature (MX-TEMP-2 20843236:2084323~ 24.9
## 5 2020-05-14 09:14:29 Temperature (MX-TEMP-2 20835814:2083581~ 24.7
## 6 2020-05-14 09:21:51 Temperature (MX-TEMP-2 20843236:2084323~ 26.2
## 7 2020-05-14 09:34:29 Temperature (MX-TEMP-2 20835814:2083581~ 25.5
## 8 2020-05-14 09:41:51 Temperature (MX-TEMP-2 20843236:2084323~ 26.6
## 9 2020-05-14 09:54:29 Temperature (MX-TEMP-2 20835814:2083581~ 26.1
## 10 2020-05-14 10:01:51 Temperature (MX-TEMP-2 20843236:2084323~ 27.2
## # ... with 6,879 more rows
```

Time to strip the logger names and merge the tables

```
temp <- temp %>%
  mutate(logger = str_extract(logger,
                              "[^, ]+$"))

rh <- rh %>%
  mutate(logger = str_extract(logger,
                              "[^, ]+$"))

dew <- dew %>%
  mutate(logger = str_extract(logger,
                              "[^, ]+$"))
```

Check to see that the dates are the same for the datasets

```
all(all(temp$date == rh$date),
all(rh$date == dew$date))

## [1] TRUE
```

```

combDat <- temp %>%
  full_join(rh,
    by = c("date" = "date",
           "logger" = "logger")) %>%
  full_join(dew,
    by = c("date" = "date",
           "logger" = "logger")) %>%
  arrange(logger,
    date)

```

```
combDat
```

```

## # A tibble: 6,889 x 5
##   date          logger temperature    rh    dew
##   <dtm>         <chr>         <dbl> <dbl> <dbl>
## 1 2020-05-14 08:54:29 20835814    22.3  28.2  3.05
## 2 2020-05-14 09:14:29 20835814    24.7  25.6  3.69
## 3 2020-05-14 09:34:29 20835814    25.5  24.4  3.72
## 4 2020-05-14 09:54:29 20835814    26.1  24.0  3.95
## 5 2020-05-14 10:14:29 20835814    26.5  23.2  3.83
## 6 2020-05-14 10:34:29 20835814    26.5  23.3  3.91
## 7 2020-05-14 10:54:29 20835814    20.6  19.5 -3.47
## 8 2020-05-14 11:14:29 20835814     5.32 47.9 -4.83
## 9 2020-05-14 11:34:29 20835814     4.53 53.3 -4.13
## 10 2020-05-14 14:39:05 20835814     1.88 79.3 -1.32
## # ... with 6,879 more rows

```

Package this into a funtion

```

longerHobo <- function(inputFile){

  rawDat <- read_csv(inputFile,col_types = cols(.default = "c"))

  dat <- rawDat %>%
    select(-"Line#") %>%
    mutate(date = as.POSIXct(Date, format = "%m/%d/%y %H:%M:%S")) %>%
    mutate_if(is_character, as.double) %>%
    select(-Date)

  dat <- rawDat %>%
    select(-"Line#") %>%
    mutate(date = as.POSIXct(Date, format = "%m/%d/%y %H:%M:%S")) %>%
    mutate_if(is_character, as.double) %>%
    select(-Date)
}

```

```

temp <- dat %>%
  pivot_longer(cols = starts_with("Temperature"),
               names_to = "logger",
               values_to = "temperature") %>%
  select(date,
         logger,
         temperature) %>%
  filter(!is.na(temperature))

rh <- dat %>%
  pivot_longer(cols = starts_with("RH"),
               names_to = "logger",
               values_to = "rh") %>%
  select(date,
         logger,
         rh) %>%
  filter(!is.na(rh))

dew <- dat %>%
  pivot_longer(cols = starts_with("Dew"),
               names_to = "logger",
               values_to = "dew") %>%
  select(date,
         logger,
         dew) %>%
  filter(!is.na(dew))

temp <- temp %>%
  mutate(logger = str_extract(logger,
                              "[^, ]+$"))

rh <- rh %>%
  mutate(logger = str_extract(logger,
                              "[^, ]+$"))

dew <- dew %>%
  mutate(logger = str_extract(logger,
                              "[^, ]+$"))

if(!all(all(temp$date == rh$date),
all(rh$date == dew$date))) stop("Tables datetimes doesn't match")

combDat <- temp %>%
  full_join(rh,
            by = c("date" = "date",
                  "logger" = "logger")) %>%

```

```

full_join(dew,
          by = c("date" = "date",
                 "logger" = "logger")) %>%
arrange(logger,
        date)

return(combDat)
}

```

We can check that it produces the same results as the script.

```

combDat2 <- longerHobo("../rawData/All_MX_2020_2020_06_26_09_20_43_UTC_1.csv")

## Warning: NAs introduced by coercion

## Warning: NAs introduced by coercion
all(combDat == combDat2)

## [1] TRUE

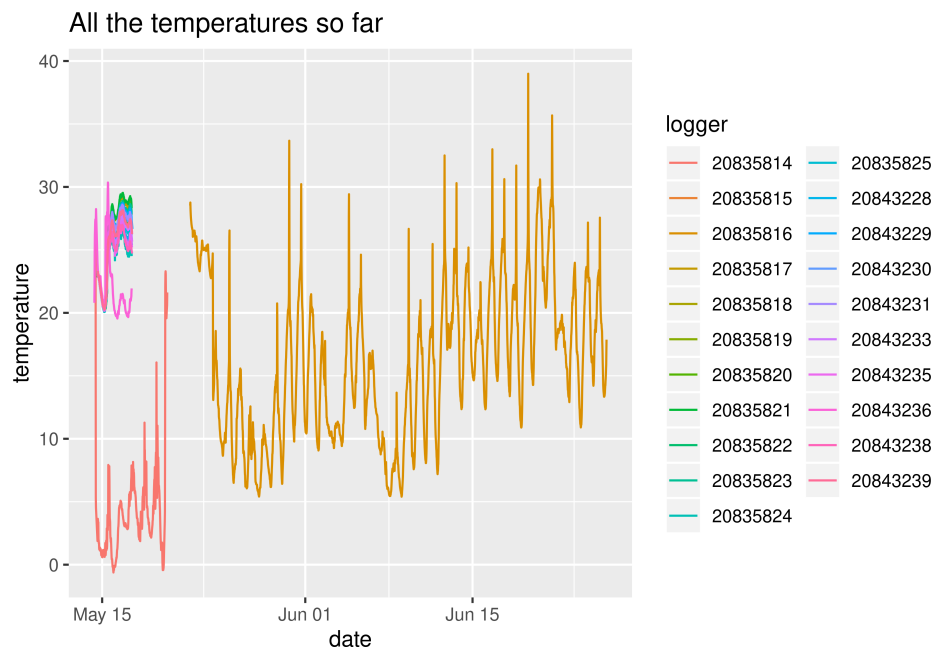
```

Check the data out

```

ggplot(combDat) +
  geom_line(aes(x = date, y = temperature, color = logger)) +
  ggtitle("All the temperatures so far")

```



```
oneLogger <- combDat %>%
  filter(logger == "20835816") %>%
  select(Date = date,
         logger,
         Temperature = temperature,
         Relative_humidity = rh,
         Dew_point = dew) %>%
  pivot_longer(-c(Date, logger),
               names_to = "Data_type",
               values_to = "Values")

ggplot(oneLogger) +
  geom_line(aes(x = Date, y = Values, color = Data_type)) +
  scale_color_nina() +
  ggtitle("All the data from one logger")
```


All the data from one logger

