Exercise #2

Immanuel Frenzel

12.01.2022

Loaded packages

```
library("lubridate")
```

```
##
## Attache Paket: 'lubridate'
```

```
## Die folgenden Objekte sind maskiert von 'package:base':
##
##     date, intersect, setdiff, union
```

```
library("tidyverse")
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## Warning: Paket 'readr' wurde unter R Version 4.1.2 erstellt
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

```
library("zoo")
```

```
## Warning: Paket 'zoo' wurde unter R Version 4.1.2 erstellt
```

```
##
## Attache Paket: 'zoo'
```

```
## Die folgenden Objekte sind maskiert von 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library("tibbletime")
```

```
## Warning: Paket 'tibbletime' wurde unter R Version 4.1.2 erstellt
```

```
##
## Attache Paket: 'tibbletime'
```

```
## Das folgende Objekt ist maskiert 'package:stats':
##
##     filter
```

1. Quality control procedures (4 QCPs)

```
#import
reimport <- read.csv("C:/Users/Imifr/Documents/Github/hyd_data_management/10610854.csv")

#time in POSIXct
data <- reimport %>%
  mutate(dttm = ymd_hm(reimport$dttm))
```

1.1 Measurement range (Plausible values)

```
data <- data %>%
  mutate(QCP_1 = if_else(temp <= -20 | temp >= 70, 0, 1))
```

Question: How many data points are outside the measurement range?

```
summarise(data, QCP_1 = sum(QCP_1 == 0, na.rm = TRUE))
```

```
##   QCP_1
## 1     0
```

```
summary(data)
```

```
##        id            dttm                          temp             lux
##  Min.   :   1   Min.   :2021-12-13 00:00:00   Min.   :-2.962   Min.   :   0.0
##  1st Qu.:1008   1st Qu.:2021-12-19 23:45:00   1st Qu.: 4.934   1st Qu.:   0.0
##  Median :2014   Median :2021-12-26 23:30:00   Median : 8.132   Median :   0.0
##  Mean   :2014   Mean   :2021-12-26 23:30:00   Mean   : 8.377   Mean   : 146.1
##  3rd Qu.:3020   3rd Qu.:2022-01-02 23:15:00   3rd Qu.:11.916   3rd Qu.:  32.3
##  Max.   :4027   Max.   :2022-01-09 23:00:00   Max.   :16.999   Max.   :5166.7
##                                               NA's   :317      NA's   :75
##      QCP_1
```

```
##  Min.   :1
##  1st Qu.:1
##  Median :1
##  Mean   :1
##  3rd Qu.:1
##  Max.   :1
##  NA's   :317
```

Answer: 0, all 75 Values are NAs

1.2 Plausible rate of change

```
data <- data %>%
  mutate(QCP_2 = ifelse(between(temp - lag(temp), -1, 1), 1, 0))
```

Question: Describe shortly how many data points failed during this QCP and discuss whether there is a certain daytime pattern of failure or not?

Answer: 10 points failed QCP_2. Looks like it often happens before midday, but there is no clear pattern to the data.

```
summarise(data, QCP_2 = sum(QCP_2 == 0, na.rm = TRUE))
```

```
##   QCP_2
## 1     7
```

```
filter(data, QCP_2 == 0)
```

```
##     id                dttm   temp    lux QCP_1 QCP_2
## 1 1355 2021-12-22 09:40:00 -0.774    0.0     1     0
## 2 1356 2021-12-22 09:50:00  1.112  322.9     1     0
## 3 1357 2021-12-22 10:00:00 -0.213  215.3     1     0
## 4 1491 2021-12-23 08:20:00  2.837    0.0     1     0
## 5 2652 2021-12-31 09:50:00 12.497 1248.6     1     0
## 6 2945 2022-01-02 10:40:00 13.173   53.8     1     0
## 7 3382 2022-01-05 11:30:00  6.775  462.9     1     0
```

```
summary(data)
```

```
##        id            dttm                          temp            lux
##  Min.   :   1   Min.   :2021-12-13 00:00:00   Min.   :-2.962   Min.   :   0.0
##  1st Qu.:1008   1st Qu.:2021-12-19 23:45:00   1st Qu.: 4.934   1st Qu.:   0.0
##  Median :2014   Median :2021-12-26 23:30:00   Median : 8.132   Median :   0.0
##  Mean   :2014   Mean   :2021-12-26 23:30:00   Mean   : 8.377   Mean   : 146.1
##  3rd Qu.:3020   3rd Qu.:2022-01-02 23:15:00   3rd Qu.:11.916   3rd Qu.:  32.3
##  Max.   :4027   Max.   :2022-01-09 23:00:00   Max.   :16.999   Max.   :5166.7
##                                               NA's   :317      NA's   :75
##      QCP_1         QCP_2
##  Min.   :1     Min.   :0.0000
##  1st Qu.:1     1st Qu.:1.0000
```

```
##  Median :1     Median :1.0000
##  Mean   :1     Mean   :0.9981
##  3rd Qu.:1     3rd Qu.:1.0000
##  Max.   :1     Max.   :1.0000
##  NA's   :317   NA's   :318
```

1.3 Minimum variability (Persistence)

```
data <- data %>%
  mutate(QCP_3 = ifelse((temp == lag(temp)) +
                        (temp == lag(temp, n = 2)) +
                        (temp == lag(temp, n = 3)) +
                        (temp == lag(temp, n = 4)) +
                        (temp == lag(temp, n = 5)) == 5, 0, 1
                        ))
```

Task: Code in this section should analyses the persistance.

```
filter(data, QCP_3 == 0)
```

```
##      id                dttm   temp   lux QCP_1 QCP_2 QCP_3
## 1   274 2021-12-14 21:30:00  6.064   0.0     1     1     0
## 2   717 2021-12-17 23:20:00  3.472   0.0     1     1     0
## 3   718 2021-12-17 23:30:00  3.472   0.0     1     1     0
## 4   719 2021-12-17 23:40:00  3.472   0.0     1     1     0
## 5   720 2021-12-17 23:50:00  3.472   0.0     1     1     0
## 6   763 2021-12-18 07:00:00  2.943   0.0     1     1     0
## 7   764 2021-12-18 07:10:00  2.943   0.0     1     1     0
## 8   765 2021-12-18 07:20:00  2.943   0.0     1     1     0
## 9   766 2021-12-18 07:30:00  2.943   0.0     1     1     0
## 10  777 2021-12-18 09:20:00  2.837 301.4     1     1     0
## 11  778 2021-12-18 09:30:00  2.837 312.2     1     1     0
## 12  839 2021-12-18 19:40:00  4.519   0.0     1     1     0
## 13  840 2021-12-18 19:50:00  4.519   0.0     1     1     0
## 14 1668 2021-12-24 13:50:00 10.553  32.3     1     1     0
## 15 1669 2021-12-24 14:00:00 10.553  10.8     1     1     0
## 16 1670 2021-12-24 14:10:00 10.553  10.8     1     1     0
## 17 1695 2021-12-24 18:20:00 11.334   0.0     1     1     0
## 18 1765 2021-12-25 06:00:00 12.110   0.0     1     1     0
## 19 1775 2021-12-25 07:40:00 11.916   0.0     1     1     0
## 20 1789 2021-12-25 10:00:00 12.110  53.8     1     1     0
## 21 1790 2021-12-25 10:10:00 12.110  43.1     1     1     0
## 22 2299 2021-12-28 23:00:00 11.528   0.0     1     1     0
## 23 2349 2021-12-29 07:20:00 10.748   0.0     1     1     0
## 24 2350 2021-12-29 07:30:00 10.748   0.0     1     1     0
## 25 2351 2021-12-29 07:40:00 10.748   0.0     1     1     0
## 26 2352 2021-12-29 07:50:00 10.748   0.0     1     1     0
## 27 2353 2021-12-29 08:00:00 10.748   0.0     1     1     0
## 28 2436 2021-12-29 21:50:00 14.325   0.0     1     1     0
## 29 2475 2021-12-30 04:20:00 14.709   0.0     1     1     0
## 30 2484 2021-12-30 05:50:00 14.804   0.0     1     1     0
```

```
## 31 2485 2021-12-30 06:00:00 14.804   0.0     1     1     0
## 32 2486 2021-12-30 06:10:00 14.804   0.0     1     1     0
## 33 2498 2021-12-30 08:10:00 14.804   0.0     1     1     0
## 34 2499 2021-12-30 08:20:00 14.804   0.0     1     1     0
## 35 2911 2022-01-02 05:00:00  8.680   0.0     1     1     0
## 36 2912 2022-01-02 05:10:00  8.680   0.0     1     1     0
## 37 2913 2022-01-02 05:20:00  8.680   0.0     1     1     0
## 38 3045 2022-01-03 03:20:00 12.980   0.0     1     1     0
## 39 3063 2022-01-03 06:20:00 12.401   0.0     1     1     0
## 40 3064 2022-01-03 06:30:00 12.401   0.0     1     1     0
## 41 3070 2022-01-03 07:30:00 12.304   0.0     1     1     0
## 42 3370 2022-01-05 09:30:00  5.962  43.1     1     1     0
## 43 3371 2022-01-05 09:40:00  5.962  53.8     1     1     0
## 44 3428 2022-01-05 19:10:00  5.655   0.0     1     1     0
## 45 3429 2022-01-05 19:20:00  5.655   0.0     1     1     0
```

1.4 Light intensity

```
data <- data %>%
  mutate(SIC = case_when(lux < 0 ~ "NA",
                         lux < 10 ~ "night",
                         lux < 500 ~ "sun_rise_or_set",
                         lux < 2000 ~ "overcast_full",
                         lux < 15000 ~ "overcast_light",
                         lux < 20000 ~ "clear_sky_shady",
                         lux < 50000 ~ "sunshine",
                         lux >= 50000 ~ "sunshine_bright")) %>%
  mutate(QCP_4 = case_when(is.na(SIC) ~ as.double(NA),
                           hour(dttm) < 6 | hour(dttm) >= 18 ~ 1, #set all QCP_4 values for nighttime t
                           lag(SIC, n = 3) == "sunshine_bright" ~ 0, #set 0 if sunshine_bright +- 3 bef
                           lag(SIC, n = 2) == "sunshine_bright" ~ 0,
                           lag(SIC) == "sunshine_bright" ~ 0,
                           SIC == "sunshine_bright" ~ 0,
                           lead(SIC) == "sunshine_bright" ~ 0,
                           lead(SIC, n = 2) == "sunshine_bright" ~ 0,
                           lead(SIC, n = 3) == "sunshine_bright" ~ 0, #set 0 if sunshine +- 1 before or
                           lag(SIC) == "sunshine" ~ 0,
                           SIC == "sunshine" ~ 0,
                           lead(SIC) == "sunshine" ~ 0,
                           TRUE ~ 1))
```

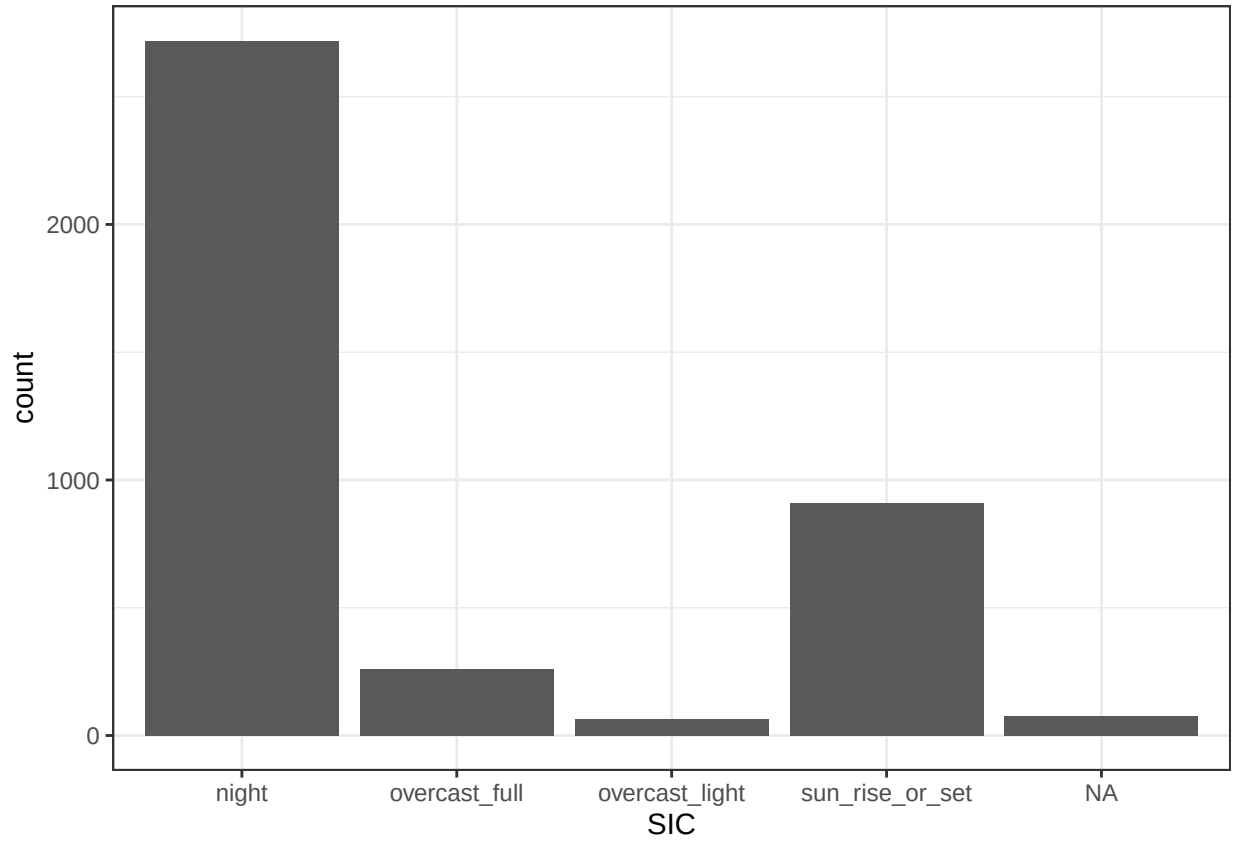
Task: Discuss shortly how often and when during daytime the QCP4 flags bad data. Elaborate on some reasons for your results.

```
summary(data)
```

```
##        id             dttm                          temp             lux
##  Min.   :   1   Min.   :2021-12-13 00:00:00   Min.   :-2.962   Min.   :   0.0
##  1st Qu.:1008   1st Qu.:2021-12-19 23:45:00   1st Qu.: 4.934   1st Qu.:   0.0
##  Median :2014   Median :2021-12-26 23:30:00   Median : 8.132   Median :   0.0
##  Mean   :2014   Mean   :2021-12-26 23:30:00   Mean   : 8.377   Mean   : 146.1
```

```
##  3rd Qu.:3020   3rd Qu.:2022-01-02 23:15:00   3rd Qu.:11.916   3rd Qu.:  32.3
##  Max.   :4027   Max.   :2022-01-09 23:00:00   Max.   :16.999   Max.   :5166.7
##                                               NA's   :317      NA's   :75
##      QCP_1          QCP_2            QCP_3            SIC
##  Min.   :1     Min.   :0.0000   Min.   :0.0000   Length:4027
##  1st Qu.:1     1st Qu.:1.0000   1st Qu.:1.0000   Class :character
##  Median :1     Median :1.0000   Median :1.0000   Mode  :character
##  Mean   :1     Mean   :0.9981   Mean   :0.9879
##  3rd Qu.:1     3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1     Max.   :1.0000   Max.   :1.0000
##  NA's   :317   NA's   :318      NA's   :322
##      QCP_4
##  Min.   :1
##  1st Qu.:1
##  Median :1
##  Mean   :1
##  3rd Qu.:1
##  Max.   :1
##  NA's   :75
```

```
ggplot(data, mapping = aes(x = SIC)) +
  geom_bar() +
  theme_bw()
```

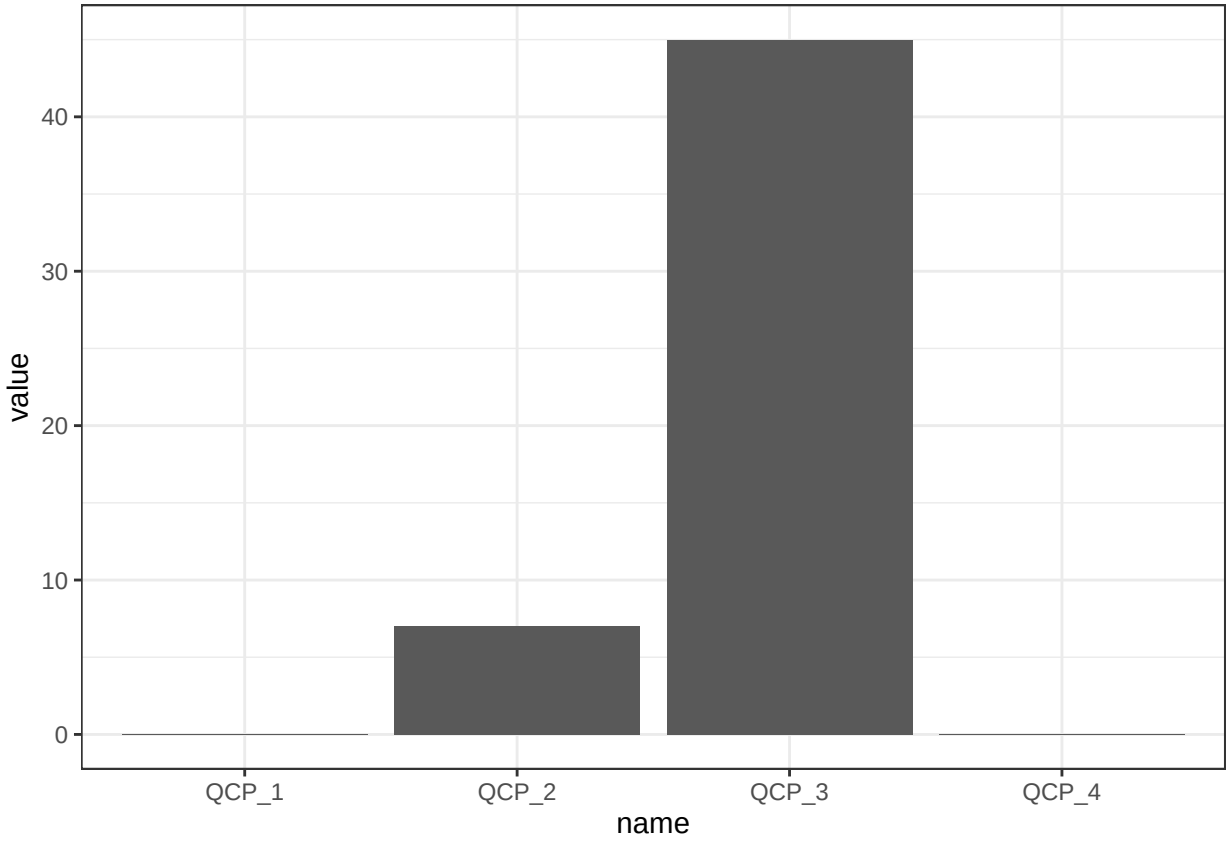


Answer: 0, It seems like there was not much light at the sensor location.

2. Synthesis

```
a <- data %>%
  summarise_at(vars(QCP_1 , QCP_2, QCP_3, QCP_4), ~ sum(.x == 0, na.rm = TRUE)) %>%
  pivot_longer(cols = QCP_1:QCP_4)

ggplot(a, mapping = aes(x = name, y = value)) +
  geom_col() +
  theme_bw()
```




```
summary(data)
```

```
##        id            dttm                          temp             lux
##  Min.   :   1   Min.   :2021-12-13 00:00:00   Min.   :-2.962   Min.   :   0.0
##  1st Qu.:1008   1st Qu.:2021-12-19 23:45:00   1st Qu.: 4.934   1st Qu.:   0.0
##  Median :2014   Median :2021-12-26 23:30:00   Median : 8.132   Median :   0.0
##  Mean   :2014   Mean   :2021-12-26 23:30:00   Mean   : 8.377   Mean   : 146.1
##  3rd Qu.:3020   3rd Qu.:2022-01-02 23:15:00   3rd Qu.:11.916   3rd Qu.:  32.3
##  Max.   :4027   Max.   :2022-01-09 23:00:00   Max.   :16.999   Max.   :5166.7
##                                               NA's   :317      NA's   :75
##      QCP_1         QCP_2            QCP_3            SIC
##  Min.   :1     Min.   :0.0000   Min.   :0.0000   Length:4027
##  1st Qu.:1     1st Qu.:1.0000   1st Qu.:1.0000   Class :character
##  Median :1     Median :1.0000   Median :1.0000   Mode  :character
```

```
##  Mean   :1     Mean   :0.9981   Mean   :0.9879
##  3rd Qu.:1     3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1     Max.   :1.0000   Max.   :1.0000
##  NA's   :317   NA's   :318      NA's   :322
##      QCP_4
##  Min.   :1
##  1st Qu.:1
##  Median :1
##  Mean   :1
##  3rd Qu.:1
##  Max.   :1
##  NA's   :75
```

Task: Present a table or graph to show how many data points fail during the four specific QCPs. Discuss shortly the reasons for failure and compare the different QCPs against each other.

Answer: Reasons for failure were sudden temperature changes (QCP_2, 10 points) and constant temperature over at least one hour (QCP_3, 60 points). All temperature readings were in the measurement-interval (QCP_1). The sensor never experienced more than 5166.7 lux (QCP_4) which was not enough for not passing the checkpoint.

3. Results

3.1 Result (Flagging system: 10-minutes-values)

```
qc_df <- data %>%
  mutate(QCP_total = ifelse(QCP_1 + QCP_2 + QCP_3 + QCP_4 < 4, 0, 1))

head(qc_df)
```

```
##   id                dttm  temp lux QCP_1 QCP_2 QCP_3   SIC QCP_4 QCP_total
## 1  1 2021-12-13 00:00:00 9.077   0     1    NA    NA night     1        NA
## 2  2 2021-12-13 00:10:00 9.077   0     1     1    NA night     1        NA
## 3  3 2021-12-13 00:20:00 8.879   0     1     1    NA night     1        NA
## 4  4 2021-12-13 00:30:00 8.779   0     1     1    NA night     1        NA
## 5  5 2021-12-13 00:40:00 8.779   0     1     1    NA night     1        NA
## 6  6 2021-12-13 00:50:00 8.779   0     1     1     1 night     1         1
```

Task: At the end of the code section above you should generate one! tibble or data.frame named `qc_df` with all time information, all data points (temperature and lux) and your outcomes of the different QCPs.

3.2 Result (Aggregate to hourly series)

```
hobo_hourly <- qc_df %>%
  mutate(hour = cut(dttm, breaks = "hour")) %>%
  group_by(hour) %>%
  summarise(date_time = first(hour), th = round(ifelse(sum(QCP_total) < 6, NA, mean(temp)), digits = 4)
  select("date_time", "th")

head(hobo_hourly)
```

```
## # A tibble: 6 x 2
##   date_time              th
##   <fct>               <dbl>
## 1 2021-12-13 00:00:00 NA
## 2 2021-12-13 01:00:00  8.53
## 3 2021-12-13 02:00:00  8.51
## 4 2021-12-13 03:00:00  8.25
## 5 2021-12-13 04:00:00  7.63
## 6 2021-12-13 05:00:00  7.23
```

Task: At the end of the code section above you should generate one! tibble or data.frame named `hobo_hourly` with averaged temperature values per hour or NA values (if the hour is flagged as bad data). See exercise description for more details.

- First column: YYYY-DD-MM HH:MM:SS
- Second column: Temperature values (4 digits), NA values possible