Weather Analysis

Hirley Dayan Lourenço da Silva & Marcia Parmigiani

2019/03/10

Data Loading

The dataset contains a series of weather measurements collected since March 2014, handled by the **Center of Meteorological and Climate Research Applied to Agriculture - Cepagri** at **Unicamp** - as part of the research program of agrometeorology and remote sensing applied to agriculture and ecophysiology.

The following lines read the data from the URL and creates a dataset with the read values:

```
# Cepagry dataset URL:
# file <- url("https://ic.unicamp.br/~zanoni/cepagri/cepagri.csv")
file <- "cepagri.csv"

# Reading the dataset from the URL:
cepagri <- read.table(file, header = FALSE, fill = TRUE, sep = ";",
                      colClasses="character")
```

The dataset contains **259706** observations and **5** features.

The data observations are composed of temperature (`temperature`), in centigrade (°C), wind speed (`wind`), in kilometer per hour (Km/h), humidity (`humidity`), in percentage (%), and thermal sensation (`sensation`), in centigrade (°C), collected every 10 minutes.

The feature names are set as follows as part of the preparation, as the source dataset does not contain the feature names:

```
# Feature names:
feature.names <- c("datetime", "temperature", "wind", "humidity", "sensation")
names(cepagri) <- feature.names
```

Data Cleansing

The data is acquired from deployed sensors and errors are expected during the collection. The string `[ERRO]` may occur in the collected observations in case faults occur during the acquisition of the data. The table following shows a few `[ERRO]` occurring observations from the dataset:

	datetime	temperature	wind	humidity	sensation
26491	12/09/2014-17:50	[ERRO]			
26673	14/09/2014-00:10	[ERRO]			
26674	14/09/2014-00:20	[ERRO]			
26675	14/09/2014-00:30	[ERRO]			

There are **2320** observations that contain the string `[ERRO]` that must be removed from the dataset.

The string `[ERRO]` might be followed by spaces and in order to facilitate the removal of those lines, `sapply` is used first for removing any spaces from the observation values knowing that the dataset was imported in `character` format.

```
# Removal of entry spaces
cepagri <- as.data.frame(apply(cepagri,2,function(x)gsub('\\s+', '',x)),
                         stringsAsFactors = FALSE)
```

`[ERRO]` string can now easily be removed once the spaces have been removed from the observation values by doing:

```
# Replace any ERRO entry with NA value
cepagri[cepagri=="[ERRO]"] <- NA
```

For a proper interpretation of the date and time values of the feature `datetime`, the values have to be converted into `POSIXlt` format:

```
cepagri$datetime <- as.POSIXct(cepagri$datetime, format="%d/%m/%Y-%H:%M")
```

Additionally, the `character` features can also be converted into `double` with:

```
cepagri[,-1] <- as.data.frame(sapply(cepagri[,-1], as.double))
```

After the cleaning, there are **0** observations that contain the string `[ERRO]` in the dataset.

The data period chosen for this study was from **January 1st, 2015** to **December 31rd, 2018** as follows:

```
# Selecting initial and final working dates:
data.period <- c(as.POSIXct("2015-01-01 00:00:01"),
                 as.POSIXct("2018-12-31 23:59:59"))
cepagri <- cepagri[cepagri$datetime >= min(data.period) &
                       cepagri$datetime <= max(data.period),]
```

The filtered dataset for the desired period contains **208274** observations and **5** features.
Besides having features converted into the correct format, and any known incorrect strings properly replaced with `NA` values, the next step is to handle lines with incomplete samples.

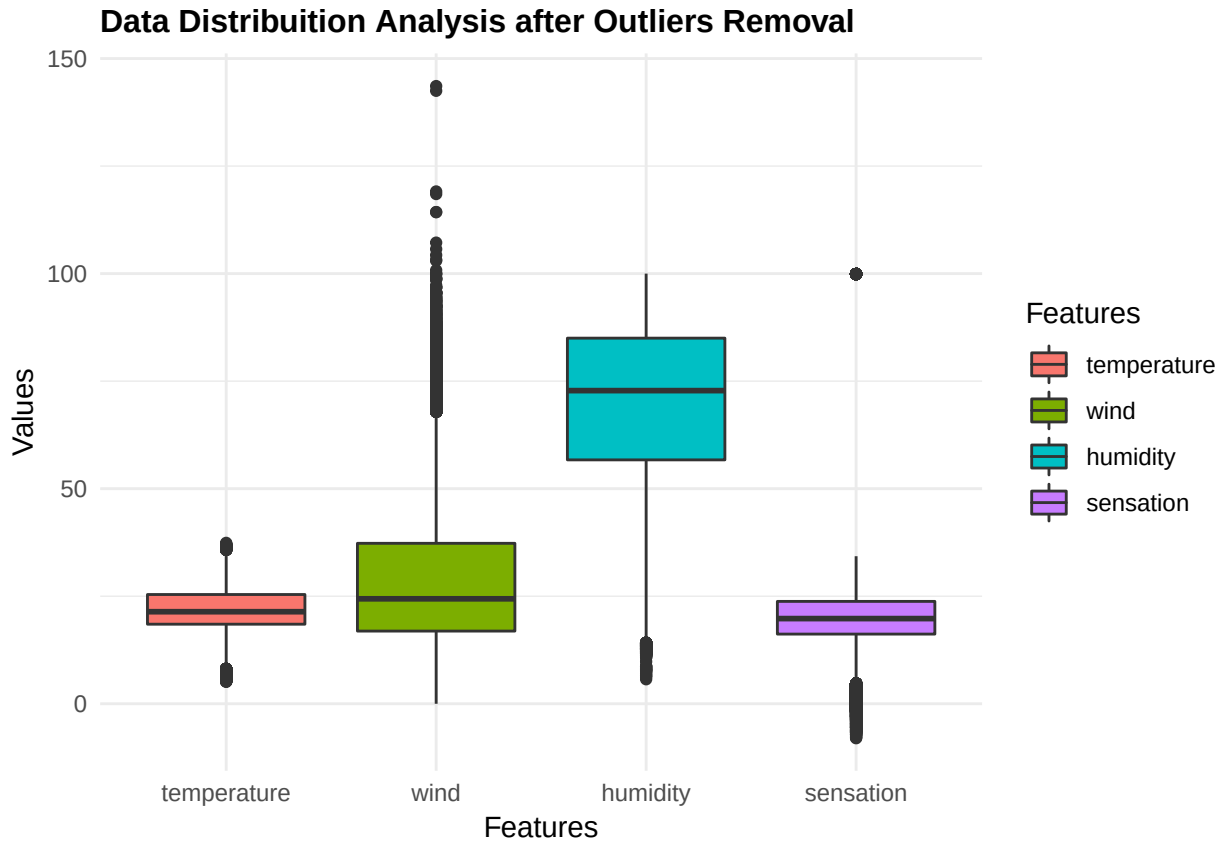
After the removal of **0** incomplete observations, a new dataset is created with a total of **206227** observations.

Data Analysis

The `summary` that follows shows a quick overview of the cleansed dataset:

	temperature	wind	humidity	sensation
Min.	5.10000	0.00000	5.70000	-8.00000
1st Qu.	18.50000	16.90000	56.70000	16.20000
Median	21.40000	24.40000	72.80000	19.80000
Mean	21.85756	28.17909	69.90768	19.60679
3rd Qu.	25.40000	37.30000	85.00000	23.80000
Max.	37.40000	143.60000	100.00000	99.90000

The data distribution can also be evaluated by the plot that follows:



For the removal of the outliers, which are values that differ considerably from the majority of a set of data, different techniques are available. In this study, outlier removal is performed using the *capping* technique, as described in [4], by replacing values outside the $1.5 * IQR$ limits, with the lower limit replaced by the **5th** percentile and the bigger limit replaced by the **95th** percentile.

The quantiles are calculated as follows:

```r
cepagri.quantiles <-
    as.data.frame(sapply(cepagri[,-1],
                         function(x){quantile(x, c(0.05,0.25,0.75,0.95))}))
```

The lower and upper limits are calculated by the equations:

$$IQR = Q3 - Q1 \tag{1}$$
$$Lower = Q1 - 1.5 * IQR \tag{2}$$
$$Upper = Q3 + 1.5 * IQR \tag{3}$$

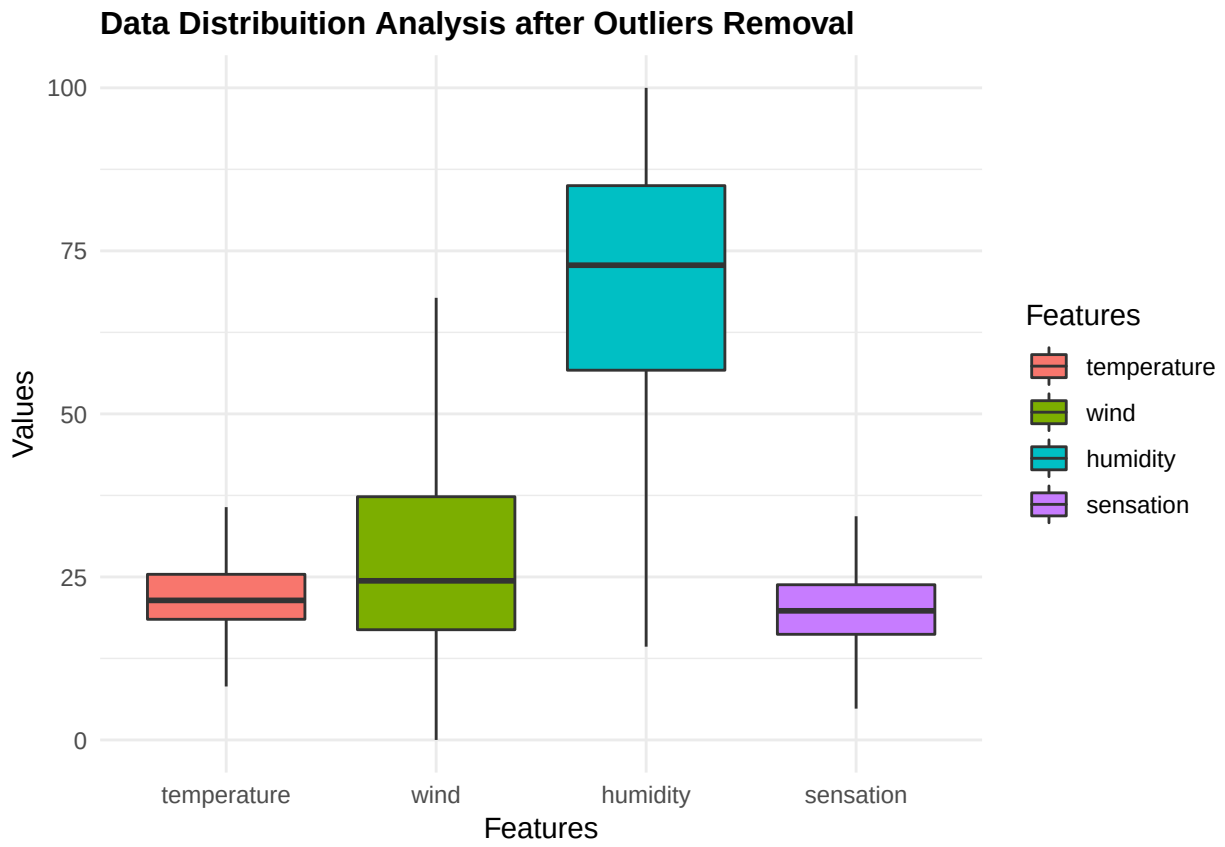
Once the limits are calculated, outliers below **5th** percentile and above the **95th** percentile are replaced by both ***Lower*** and ***Upper*** limits, respectively, with the algorithm:

```r
# Replace outliers using capping method, by replacing values below 5th percentile
# and above 95th percentile by both lower and upper limits, respectively, where:
# lower = Q1 - 1.5*IQR
# upper = Q1 - 1.5*IQR
for(n in names(cepagri[,-1])){
    cepagri[!is.na(cepagri[, n]) &
                cepagri[, n] < cepagri.limits["lower", n], n] <-
        cepagri.quantiles["5%", n]
    cepagri[!is.na(cepagri[, n]) &
                cepagri[, n] > cepagri.limits["upper", n], n] <-
        cepagri.quantiles["95%", n]
}
```

The `summary` that follows shows a quick overview of the prepared dataset, after the removal of the outliers:

	temperature	wind	humidity	sensation
Min.	8.20000	0.00000	14.30000	4.80000
1st Qu.	18.50000	16.90000	56.70000	16.20000
Median	21.40000	24.40000	72.80000	19.80000
Mean	21.86566	27.93708	69.92965	19.72076
3rd Qu.	25.40000	37.30000	85.00000	23.80000
Max.	35.70000	67.80000	100.00000	34.30000

Additionally, the plot that follows brings another perspective for the data distribution:



Duplicated observation values may happen and must be handled accordingly. The automatic removal of duplicates can only be done for adjacent observations as it is not a natural weather behavior to stand still all its indicators for a length of time.

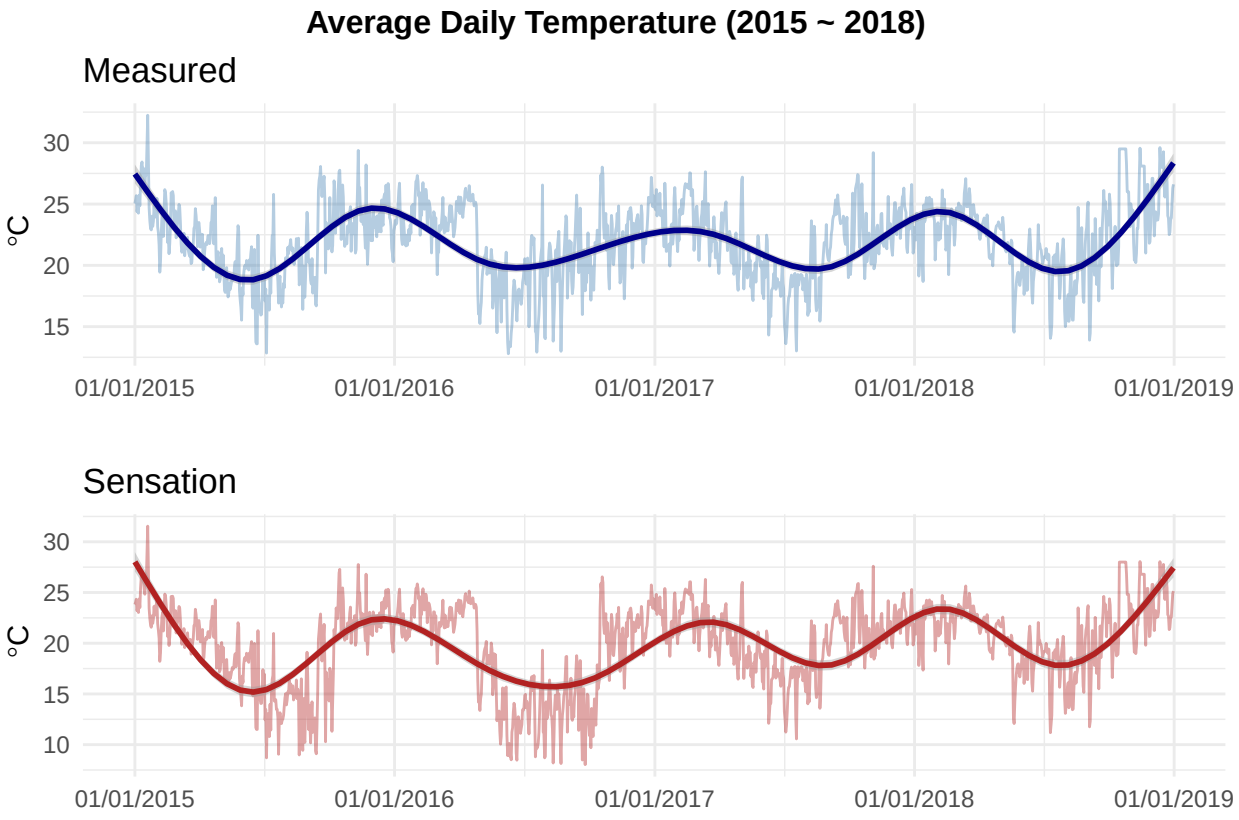
The following table shows adjacent occurrences with the same measurement values:

	datetime	temperature	wind	humidity	sensation
42130	2015-01-01 07:50:00	23.4	16.1	88.6	22.3
42131	2015-01-01 08:00:00	23.4	16.1	88.6	22.3

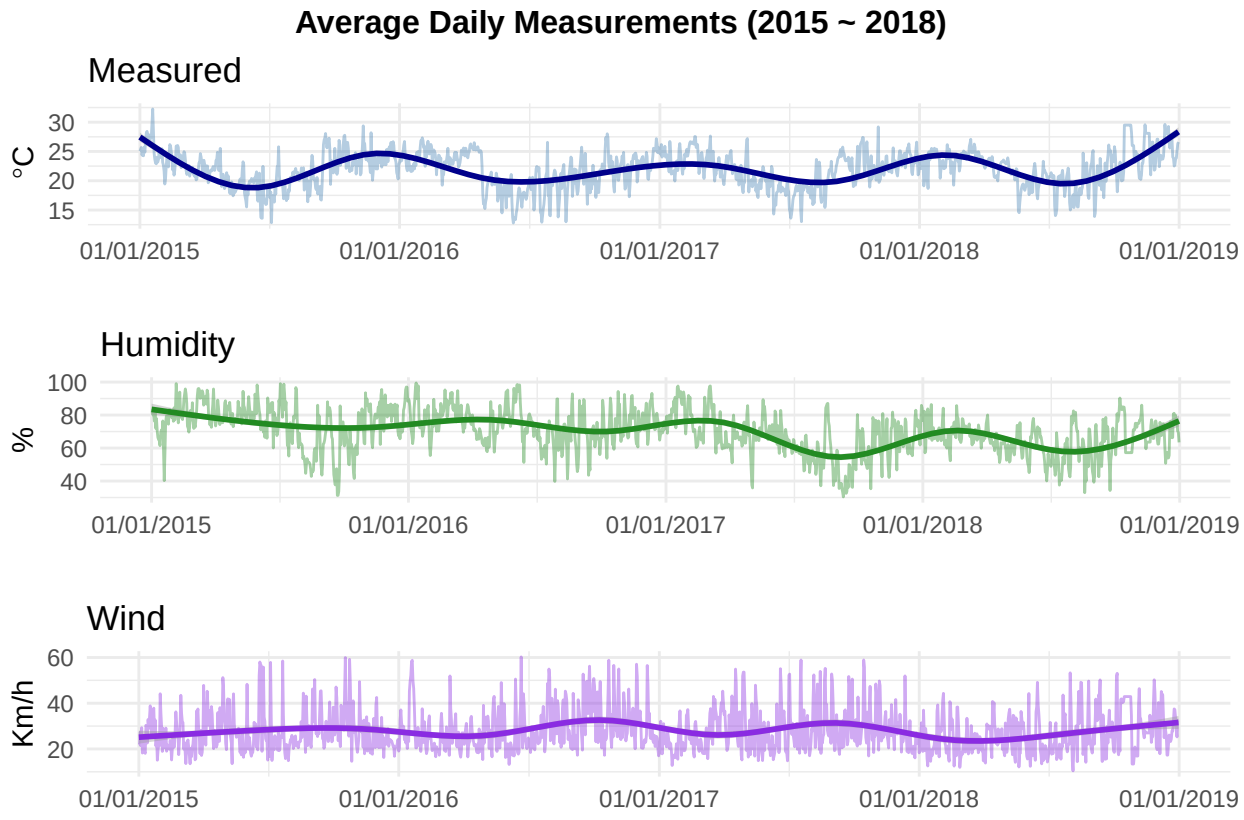
After the removal of **91** duplicated observations, a new dataset is created containing **206136** observations.

Meteorological Analysis

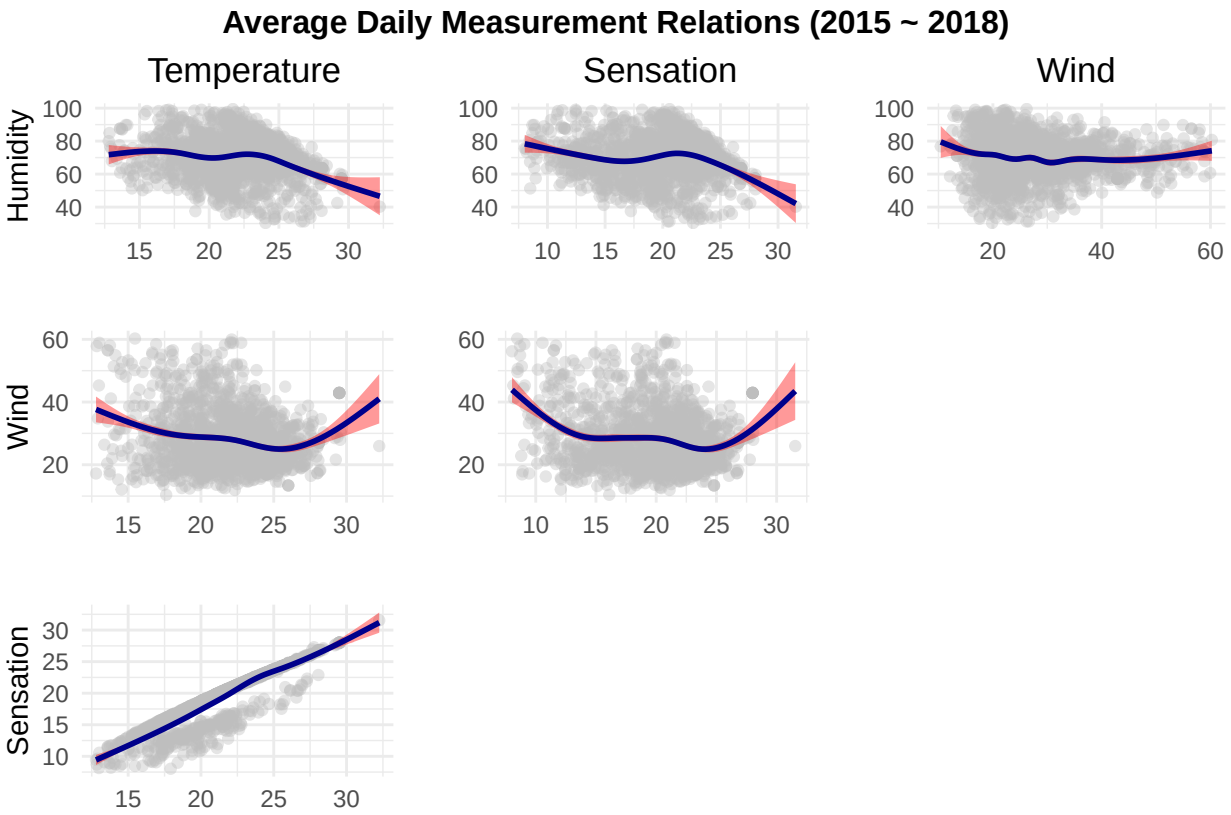
The following graph shows the measured temperature and the temperature sensation for the period from 2015 to 2018:



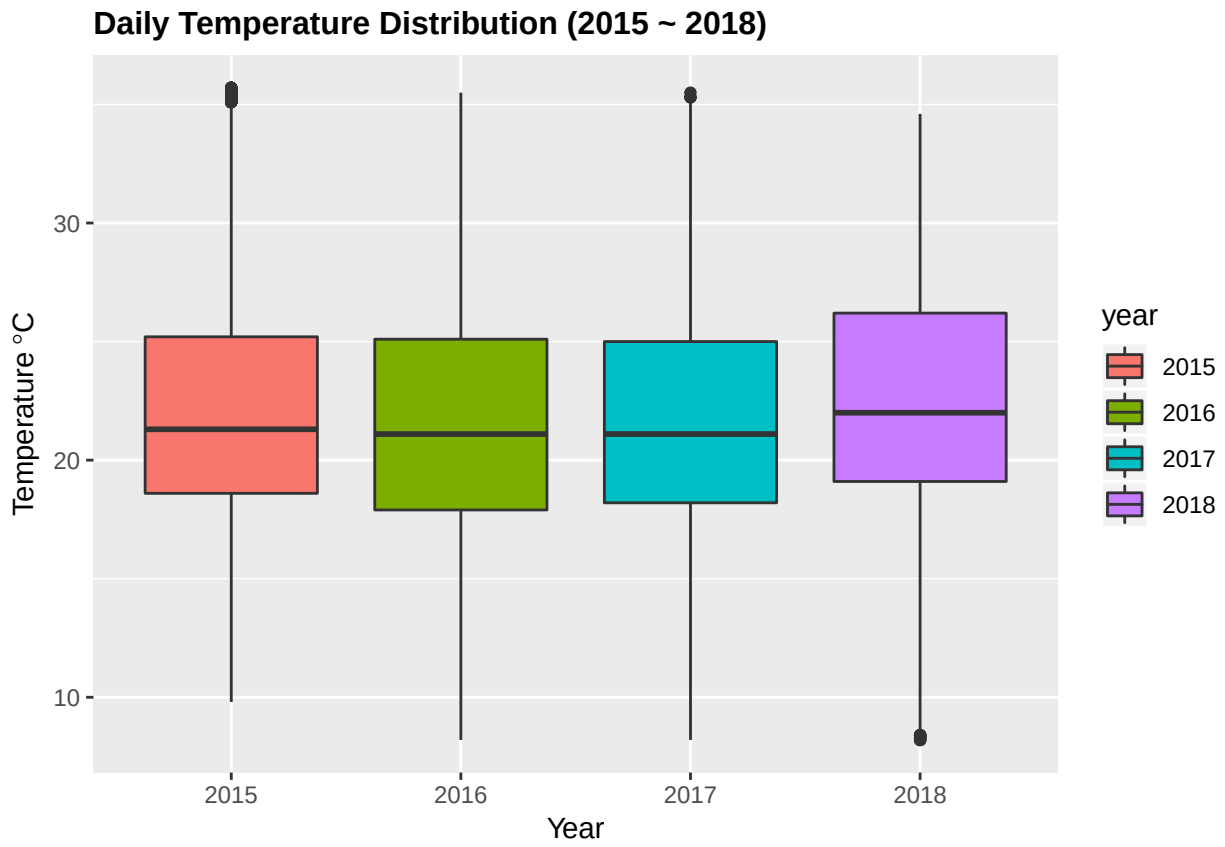
The following graph shows the measured temperature, humidity and wind speed for the period from 2015 to 2018:



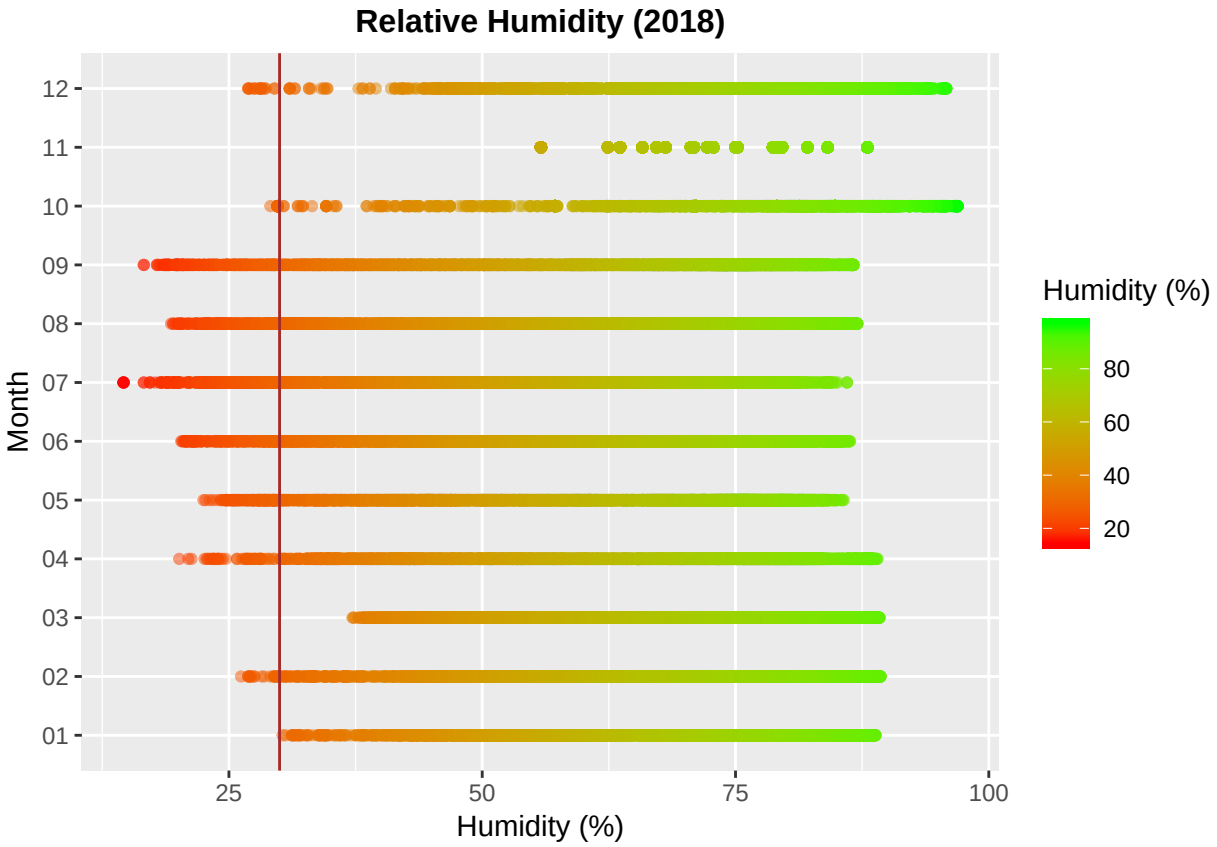
The following graph matrix brings a cross analysis overview of all sensors, showing how they related to each other.



The following graph shows the daily temperature distribution between 2015 and 2018. As revealed by the graph, the temperature oscillated around the same range in the period, without presenting significant variations.



The next graph shows the relative humidity in 2018. According to the Cepagri classification indicator (CGE) [5], the humidity level below 30% is not good for health, which is indicated in the graph by the vertical red line.



References

[1] De Jonge, E., & Van Der Loo, M. (2013). An introduction to data cleaning with R. Retrieved from www.cbs.nl/information

[2] Wickham, H. (2014). Tidy Data. Journal of Statistical Software, 59(10). http://doi.org/10.18637/jss.v059.i10

[3] Beck, C. (2012). Handling date-times in R. Retrieved from http://biostat.mc.vanderbilt.edu/wiki/pub/Main/ColeBeck/datestimes.pdf

[4] Outlier detection and treatment with R | R-bloggers. (n.d.). Retrieved March 8, 2019, from https://www.r-bloggers.com/outlier-detection-and-treatment-with-r

[5] Umidade Relativa do Ar - CGE. (n.d.). Retrieved March 9, 2019, from https://www.cgesp.org/v3/umidade-relativa-do-ar.jsp