Experimental bioclimatology Term Paper

Simone Massaro and Cheyenne Rueda

15 September 2021

Contents

Preface

This is the Term Paper for Experimental bioclimatology course (summer semester 2021) at Uni Göttingen, prof. Alexander Knohl.

The term paper is a group work between Simone Massaro and Cheyenne Rueda.

All the code is available at https://github.com/mone27/bioclimatology-cheyenne-simone. This term paper is also available in html format at https://bookdown.org/massaro_simone_it//

1 Shortwave radiation

1.1 Motivation

The sun is the source of all energy reaching the Earth. It arrives under the form of electromagnetic radiation, in particular of light in the shortwave range.

The incoming shortwave radiation heats up the surface and the atmosphere with different intensity. This is the core reason behind numerous phenomena on the earth, from seasons to winds. Moreover a small fraction of shortwave radiation is captured by the plants through photosynthesis, its converted to chemical energy and it is the basis for the whole biosphere.

1.2 Background

Electromagnetic radiation are oscillations of a magnetic and electric field. Electromagnetic radiation is formed by waves, that have a characteristics wavelength and frequency.

Electromagnetic radiation is a form of energy, hence it is measured in Joule (J) and its intensity Watts ($J/s = W$).

Every body emits electromagnetic radiation, the intensity and the wavelength of the emitted radiation depends on its temperature and is described by Planck's law.

Due to its temperature the sun emits short-wave radiation, that have a wavelength in the range $(0.2 - 4)\mu m$. Solar radiation is composed by ultraviolet radiation with a wavelenght lower than $0.4\mu m$, visible radiation $(0.4 - 0.7)\mu m$ and near-infrared $> 0.5\mu m$. In contrast with sun radiation, the radiation emitted by objects in earth will not surpass the range of $3 - 100\mu m$, and this ratio is known as long wave radiation or infrared.

Stefan-Boltzmann law describes the total amount of radiation emitted from bodies at a specific temperature. This law consists on the calculation of emittance (L) in (W/m^2) produced equal the product of the Stefan-Boltzmann constant ($\sigma = 5.6710^{-8} \frac{W}{m^2} K^{-4}$) and the emissivity ($\varepsilon$);

$$L = \varepsilon \sigma T^4$$

The emissivity as ε "*the ratio of the actual emittance to the blackbody emittance*". A blackbody is described as the body that absorbs and emits all the radiation (Bonan 2019)

The solar radiation reaches the ecosystems and it is divided in 3 components: reflected, absorbed and transmitted. The amount of radiation in each of this component depends on the properties of the canopy and on the wavelength (λ). The following equation is true:

$$1 = \sigma(\lambda) + r(\lambda) + t(\lambda)$$

where:

- Adsorption $\sigma(\lambda)$ is the fraction of light that it blocked and absorbed by the medium.
- Reflectivity $r(\lambda)$ is the fraction of electromagnetic wave that is reflected backward. The reflectivity of a surface is also called albedo (Perkins 2019).
- Transmissivity $t(\lambda)$ is the fraction of energy going through a medium without experimenting any change.

Incoming shortwave radiation is formed by two different components:

- Direct solar radiation (S): is the incident radiant flux density falling into a horizontal surface which will differ with the position of the sun.
- Diffuse solar radiation (D): the incident radiant flux density arriving to the surface of earth after scattered and reflected during its way by other molecules as well present in air.

Global radiation (G): is the sum of diffuse and direct solar radiation.

$$G = S + D$$

1.3 Sensors and measuring principle

Shortwave solar radiation can be measured with the use of different instruments that use different priciples. The main ones are:

- **Pyrheliometer**: allows the direct measurement of shortwave radiation coming directly from the sun. It is disposed by a sensor that reacts at wavelengths around 0.2-4 micrometers. It needs to be set close to a sun tracker, this way it is possible to follow the movement of sun along day.
- **Pyranometer**: this instrument measured the global shortwave radiation including the reflected shortwave radiation. It is based on the use of a sensor sensible at same wavelengths as the pyrheliometer, although in this case it must be parallel to the soil surface (horizontal). If the shortwave reflected needs to be measure, the sensor would need to be turn downwards facing the surface. In the case of Pyranometer, it is also used to measure diffused shortwave radiation. For this function, a shadow ring is set, normally shifting towards the suns azimuth angle during time. The main principle is the calculation of voltage differences with wires made of different methals. Normally these voltages are small, this is why the application of a thermopile. In order to decrease the error that winds and air temperature may induce, a glass is situated around the detector.
- **PAR Quantum sensor**: this sensor is used for the measurement of Photosynthetically Active Radiation, it is more sensitive than the other with a range between 400-700 nm. It is formed by a filter on top of the sensor and a photodiode semiconductor made out of silicon. This way, incident light makes react the semiconductor, current measured in voltages with the used of resistance.
- **Campbell-Stokes sunshine autograph**: this instrument is used to measure the sunshine duration of days. A circumference made of glass burns a paper where the sunshine time period is register.

1.4 Analysis

```
library(tidyverse)
library(ggplot2)
library(lubridate)
library(here)


shortwavedata <- read.csv(
  here("1_shortwave/Exercise_1st_lect/Shortwave_incoming_diffuse_W_m-2_Hainich_2020.csv"),
  header = TRUE, sep = ",")
shortwavedata$Date<- as.POSIXct(shortwavedata$Date, format= "%Y-%m-%d %H:%M")

calib_factorsdata <- read.csv(
  here("1_shortwave/Exercise_1st_lect/Shortwave_incoming_outgoing_belowcanopy_mV_Hainich_2020.csv"))
calib_factorsdata$Date<- as.POSIXct(calib_factorsdata$Date
                                    , format= "%Y-%m-%d %H:%M")

sw_b <- data.frame(
  Date = calib_factorsdata$Date,
  incoming = calib_factorsdata$Shortwave_incoming_mV * 86.5801,
  #mV *( W m-2 )/mV  = W m-2
  outgoing =  calib_factorsdata$Shortwave_outgoing_mV * 86.5801,
  #mV *( W m-2 )/mV  = W m-2
  belowcanopy = calib_factorsdata$Shortwave_incoming_bewlow_canopy_mV*194.9318
  #mV *( W m-2 )/mV = W m-2
)

# This is a quick hack to remove incorrect data when the radiation is low
sw_b[sw_b$incoming<=10 | sw_b$outgoing <=10 | sw_b$belowcanopy <= 10,] <- NA
sw_b <- drop_na(sw_b)
```

1.4.1 Albedo

The albedo is the fraction of incoming solar radiation that is reflected upwards by the earth surface (Figure 1). During the winter there are some peaks in the albedo and can be explained by measurement errors due to the low amount of radiation. The albedo is relatively constant across the year, in the range 10-17%. During the summer the albedo is higher, due to the higher reflectance of the green canopy compared to the dark ground.

```r
sw_b <- mutate(sw_b,
               albedo = outgoing/incoming,
               transmitted = belowcanopy/incoming,
               absorbed = 1 - albedo)

sw_b_d <- sw_b %>%
  group_by(yday(Date)) %>%
  summarise_all(mean)
```

```r
ggplot(sw_b_d, aes(x=Date))+
  geom_line(aes(y=albedo)) +
  labs(y = "Albedo", x = "Time")
```

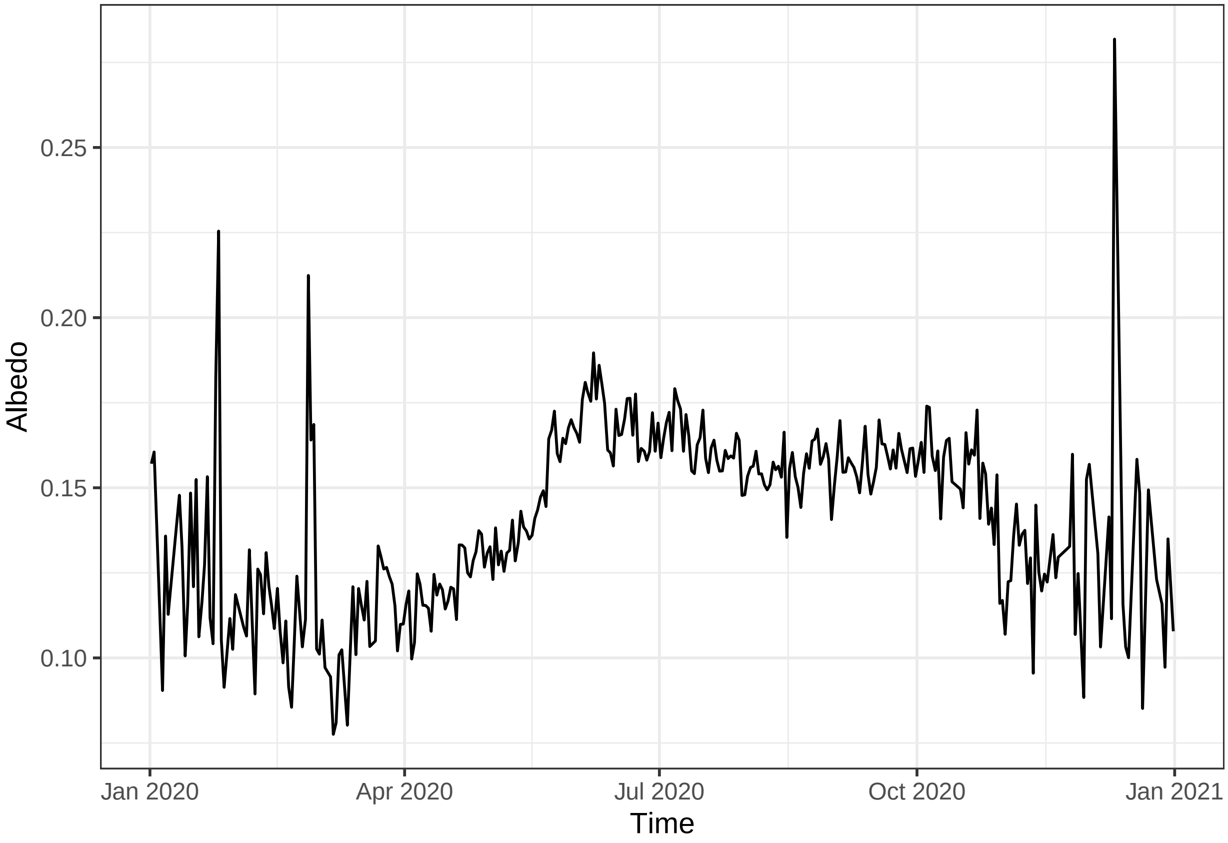

Figure 1: Surface albedo over the year. Data is averaged daily. Data from Hainich national park 2020.

1.4.2 Absorption coefficient

The Absorption coefficient is the fraction of incoming radiation that is absorbed by the ecosystems (Figure 2).

The absorption is quite constant, but during winter is higher (up to 90%) to then decrease in summer when there are leaves (83%). The peaks during the winter can be connected to measurement errors due to the low amount of radiation.

```
ggplot(sw_b_d, aes(x=Date))+
  geom_line(aes(y=absorbed))+
  labs(y = "Absorption coefficient", x = "Time") +
  ylim(0,1)
```

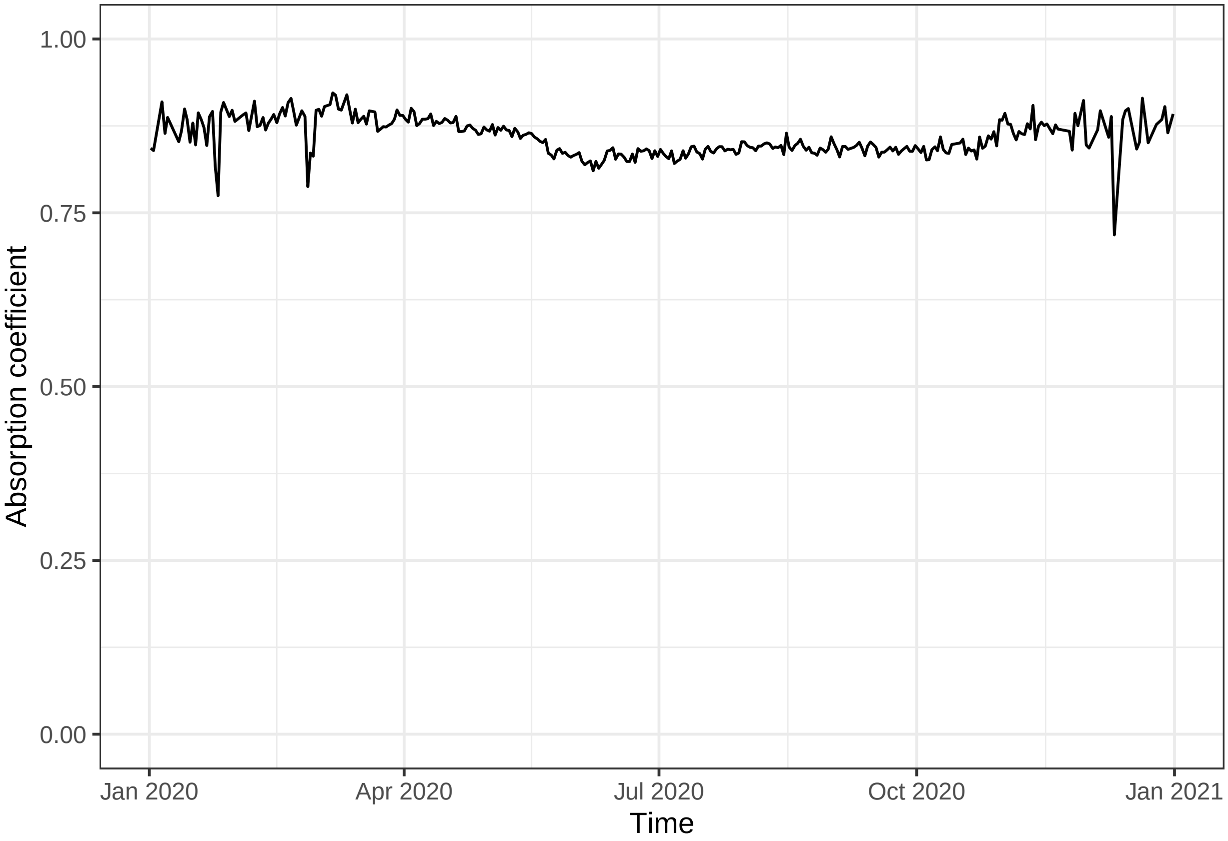

Figure 2: Absorption coefficient over the year, daily average. Data from Hainich national park 2020.

1.4.3 Transmission coefficient

The Transmission coefficient is the fraction of light that is transmitted through the canopy and reach the ground below (Figure 3).

During the year there is a clear difference between summer and winter. During winter the transmission is around 30% while during summer it is around 5%. This shows how effective are the trees in capturing available light.

```r
ggplot(sw_b_d, aes(x=Date))+
  geom_line(aes(y=transmitted))+
  labs(y = "Trasmission coefficient", x = "Time") +
  ylim(0,.5)
```

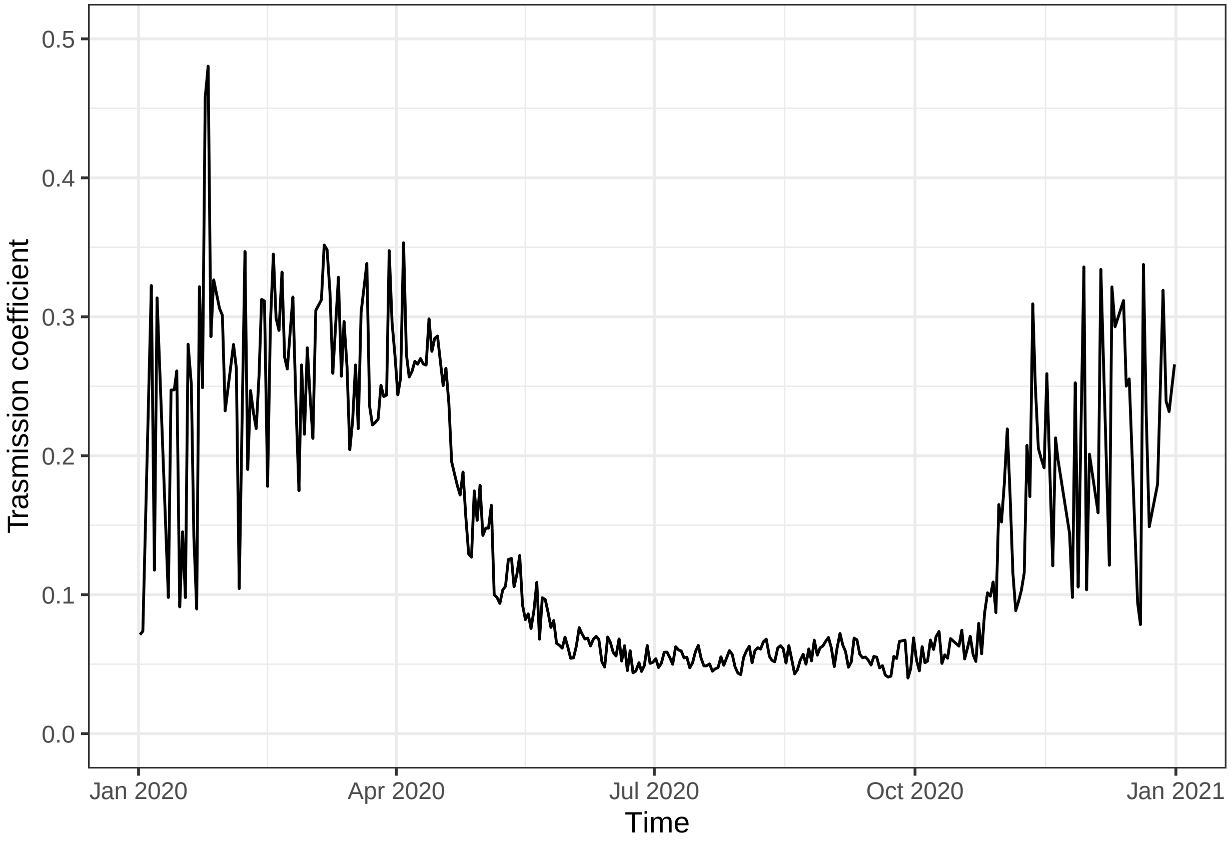

Figure 3: Absorption coefficient over the year, daily average. Data from Hainich national park 2020.

During summer days there is a daily cycle of the transmission coefficient (Figure 4). It is lower during the morning and evening and higher in the middle of the day. This can be explained because when the sun is low on the horizon there is more reflection and the light penetrates less in the canopy, thus resulting in a reduced transmission coefficient.

```
sw_b %>%
  filter(between(Date, as_datetime("2020-07-1"), as_datetime("2020-07-2"))) %>%
ggplot(aes(x=Date))+
  geom_line(aes(y=transmitted))+
  labs(y = "Trasmission coeficient", x = "Time")
```

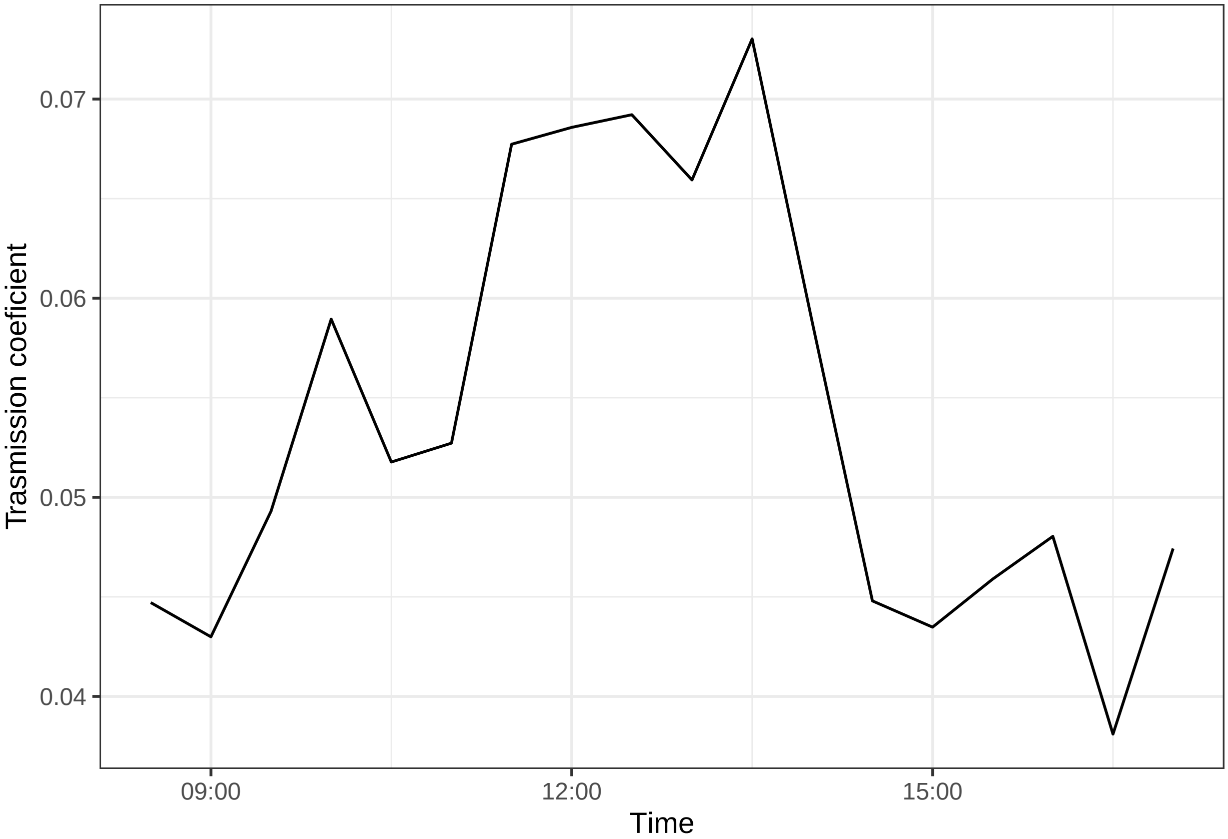

Figure 4: Daily cycle trasmittion coefficient. Data every 30 mins. Data from Hainich national park 1st July 2020.

1.4.4 Radiation below canopy

The absolute value of the radiation below the canopy (Figure 5) is relatively constant between summer and winter with the notable exception of spring. There are no leaves in the canopy yet, but the solar radiation is getting stronger. This can explain the why in beech forest there is a some undergrowth only during spring.

```r
ggplot(sw_b_d, aes(x=Date))+
  geom_line(aes(y=belowcanopy))+
  labs(y = "Transmitted radiation [W m-2]", x = "Time")
```

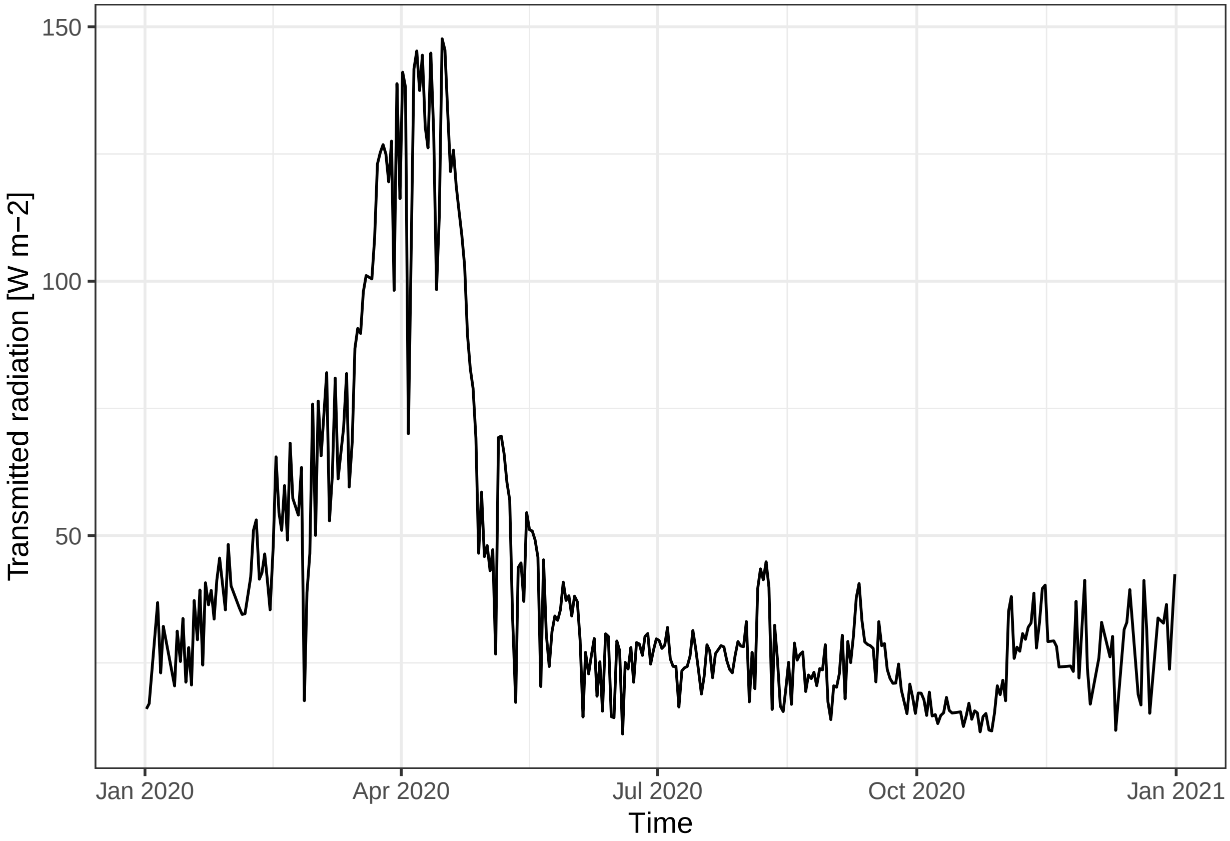

Figure 5: Radiation trasmitted below the canopy over the year. Data is averaged daily. Data from Hainich national park 2020.

1.4.5 Diffuse and direct radiation

The total radiation is sum of the diffuse and direct (Figure 6). The amount of direct radiation changes a lot during the year. You can see that April and September were relatively sunny months (direct and total radiation are similar), while July and August were cloudy (big difference between direct and total) The diffuse radiation has a its own pattern across the year.

```
shortwavedata <- mutate(shortwavedata,
                        direct = Shortwave_incoming_W_m.2 - Shortwave_incoming_diffuse_W_m.2)
```

```
shortwavedata%>%
  group_by(week(Date)) %>%
  summarise_all(mean) %>%
  gather("type", "rad", Shortwave_incoming_W_m.2,
         direct, Shortwave_incoming_diffuse_W_m.2, factor_key = T)  %>%
  ggplot(aes(x=Date, y=rad, color=type)) +
    geom_line() +
    scale_color_colorblind(labels=c("total", "direct", "diffuse")) +
    labs(y="Radiation [W m-2]", x="Time", col="Radiation")
```

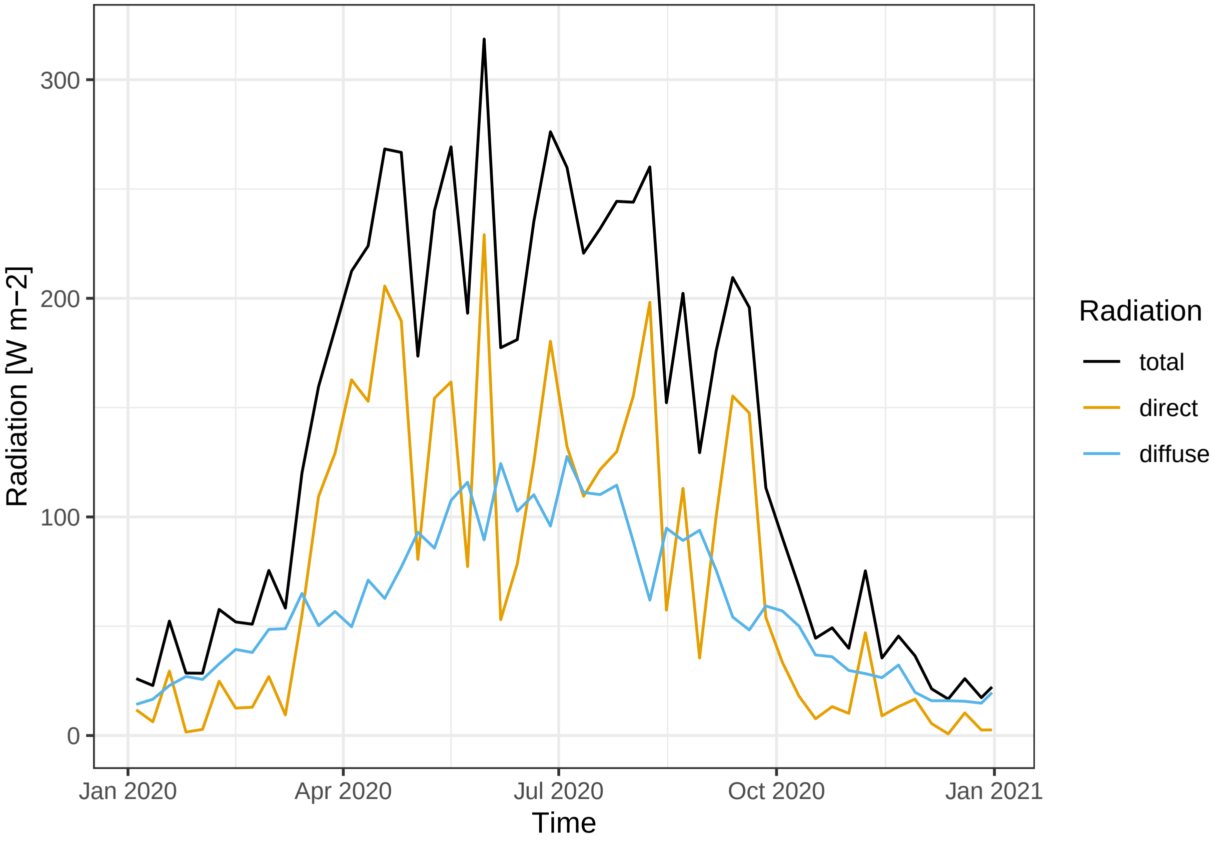

Figure 6: Direct and diffuse component of shortwave radiation over a year. Data is averaged weekly. Data from Hainich national park 2020.

Figure 7 shows a summer week, where it is possible to clearly see the difference between cloudy and sunny day. The 8th of July (first day) was cloudy with virtually no direct radiation, while the 13th of July was very sunny with almost all radiation direct.

```
shortwavedata%>%
  filter(week(Date)==28) %>%
  gather("type", "rad", Shortwave_incoming_W_m.2, direct, factor_key = T)  %>%
  ggplot(aes(x=Date, y=rad, color=type)) +
    geom_line() +
  scale_color_colorblind(labels=c("total", "direct")) +
  labs(y="Radiation [W m-2]", x="Time", col="Radiation")
```

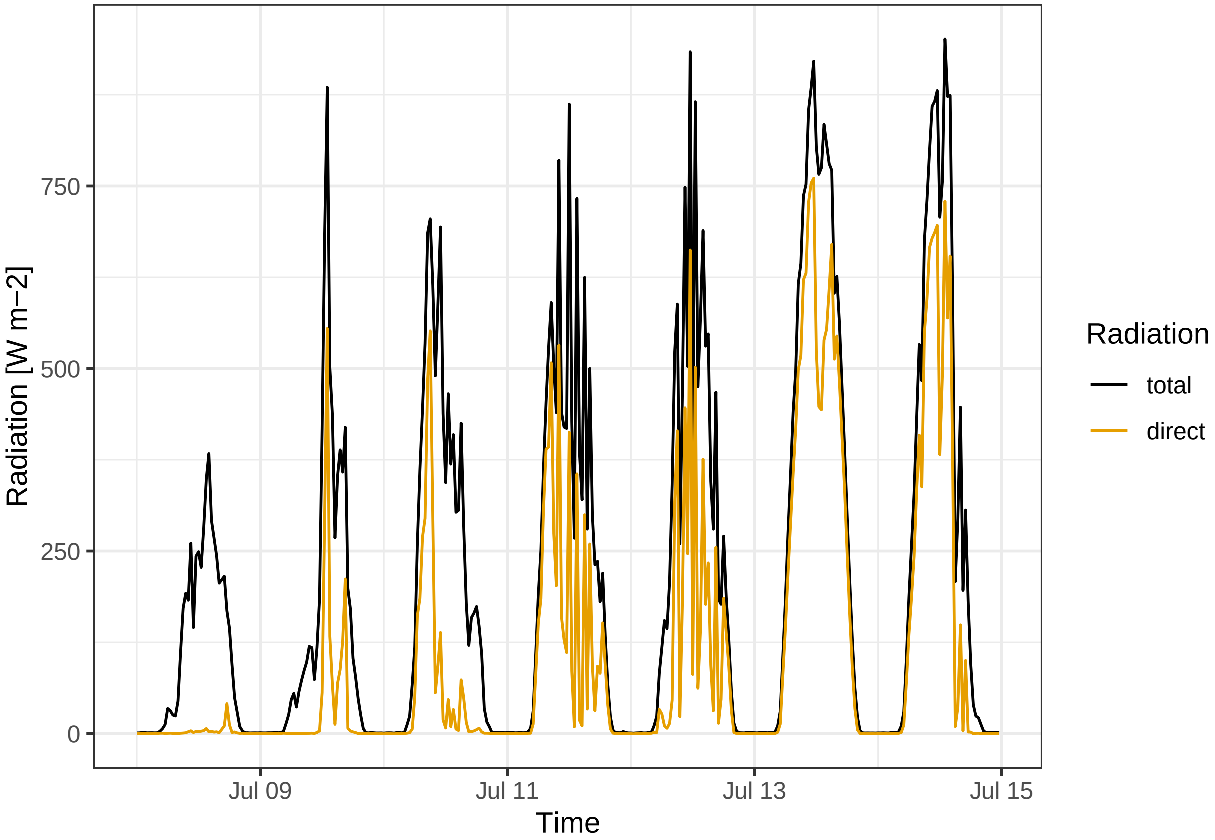

Figure 7: Total and direct component of shortwave radiation over a week. Data is averaged over 30 min. Data from Hainich national park second week July 2020.

2 Longwave radiation

2.1 Motivation

All bodies with a temperature above the absolute zero emit electromagnetic radiation. The wavelength and intensity of this radiation depends on the temperature of the body. Within the temperature range of the earth surface, the emitted radiation has a wavelength between 3 and 100 mm and it is defined as longwave.

Emitting longwave radiation is the only way for the plant to cool itself down, making it a crucial component in the overall energy balance of the earth. In fact climate change is caused by green house emissions, which capture part of the longwave radiation emitted by the planet surface and then re-emit in back towards the Earth. This results in a bigger amount of radiation reaching the earth surface, hence increasing its temperature (Harries 1996).

Longwave is a constantly present in terrestrial ecosystem for all the day. Energy loss in form of longwave radiation can have a substantial impact on the surface temperature, especially during night.

2.2 Background

Longwave radiation is electromagnetic radiation with a wavelength between 3 and 100 mm, hence it falls into the Infrared section of the spectrum.

The overall amount of radiation emitted by a body depends on its temperature following the Stefan-Boltzmann law (Bonan 2019).

$$E = \varepsilon \sigma T^4$$

Where:

- E is the radiation intensity in W/m^2
- ε an adimensional coefficient that represents the emissivity of the body. This depends on the material, a perfect black body has a ε of 1, while other materials have a lower emissivity
- σ is the Steffan-Boltzman constant $5.67 \times 10^{-8} W m^{-2} K^{-4}$
- T is the body temperature in K

As a consequence of Stefan-Boltzmann law the amount of longwave radation emitted by ecosystems depents on their temperature.

The incoming longwave radiation depends on the temperature of the sky and it is partly absorbed by the ecosystems and partly reflected (Stephens et al. 2012) according to the following formula:

$$L_{w,relf} = (1 - \varepsilon) L_{w,in}$$

The amount of longwave radiation incoming depends on the temperature of the sky, hence in cloudy days there is an higher incoming longwave radiation compared to sunny days. This phenomenon is particularly important at nights, where the longwave balance is a major driver of the overall temperature.

The net longwave radiation is summarized by this equation:

$$L_{w,net} = L_{w,in} - L_{w,refl} - L_{w,emit} = \varepsilon(\sigma \varepsilon_{sky} T_{sky}^4) - \sigma \varepsilon T^4$$

The overall net radiation includes also the shortwave component:

$$R_{w,net} = S_{w,in} - S_{w,out} + L_{w,in} - L_{w,out}$$

2.3 Sensors and measuring principle

The longwave radiation is estimated by measuring the change of temperature of a body exposed to the radiation. Compared to shortwave radiation the measured data required further processing as the sensor itself emits longwave radiation, hence there is the need include it the final measurement.

$$L_{tot} = L_{net} + \sigma T_{sensor}^4$$

Moreover the longwave sensors need to filter the incoming radiation to avoid measuring the shortwave component, therefore they usually have a filter that allows only infrared radiation between of 4.5 and 40 micrometers. This principle is used by **pyrgeometers** .

Longwave radiation can also be measured together with shortwave by **net radiation** sensors that don't filter the incoming radiation based on wavelength.

Finally **pyrometer**, or infrared thermometers, measure the temperature of a body using the emitted longwave radiation. The use of the longwave radiation permits to have high frequency measures and more importantly to measure the temperature from a distance. However the emissivity of the body needs to be estimated to correctly

2.4 Analysis

```r
rad <- read_csv(here("Data_lectures/2_Longwave_radiation/LW_SW_TSoil_BotGarten.csv"))
names(rad) <- c("datetime", "t_sens", "sw_in", "sw_out", "lw_in_sens", "lw_out_sens", "t_soil")
```

```r
# Utlitity funcs
sigma <- 5.67e-8

lw2temp <- function(lw) (lw/ sigma)^(1/4)
temp2lw <- function(temp)  return (sigma * temp^4)

c2k <- function(c) c + 273.15
k2c <- function(k) k - 273.15
```

```r
#calculate from input data the real lw and the soil/surface temperature
rad <- rad %>%
  drop_na() %>%
  mutate(
    lw_sens = temp2lw(c2k(t_sens)),
    lw_in = lw_in_sens + lw_sens,
    lw_out = lw_out_sens + lw_sens,
    t_sky = lw2temp(lw_in) %>% k2c,
    t_surface = lw2temp(lw_out) %>% k2c,
    net_rad = lw_in - lw_out + sw_in - sw_out,
    net_sw = sw_in - sw_out,
    net_lw = lw_in - lw_out
  )
```

```r
# for making aggregation easier we are going to consider data only for one calendar year
rad <- rad %>%
  filter(datetime < as_datetime("2020-12-31"))
```

```r
# weekly average data
rad_w <- rad %>%
  as_tsibble(index = datetime) %>%
  index_by(week = ~ yearweek(.)) %>%
  summarise_all(mean, na.rm = TRUE)
```

```
# daily average data
rad_d <- rad %>%
  mutate(yday = yday(datetime)) %>%
  group_by(yday) %>%
  summarize_all(mean, na.rm = TRUE)
```

2.4.1 Surface and Sky temperature

The surface and the sky temperature are plotted for one year using a weekly (Figure 8) and daily aggregation (Figure 9).

The sky temperatures is always lower than the surface one. The surface temperature ranges from -2 C to 25 C, while the sky temperature has a bigger range from -23 C to 17 C. The temperature of the sky mainly depends on the cloud cover and the temperature of the air.

During the last week of march there is biggest different in temperature, with the sky temperature plummeting to -20 °C, while the surface temperature remains above 0 °C. This is probably due to snow cover that insulates the surface from the cold air. For the rest of the year the temperature difference is relatively constant.

```
rad_w %>%
  gather(key="type", value="temp", t_surface, t_sky, factor_key = T) %>%
ggplot(aes(x=datetime, y=temp, colour=type)) +
  geom_line() +
  scale_color_colorblind() +
  labs( y="Temperature [°C]", x="Time")
```

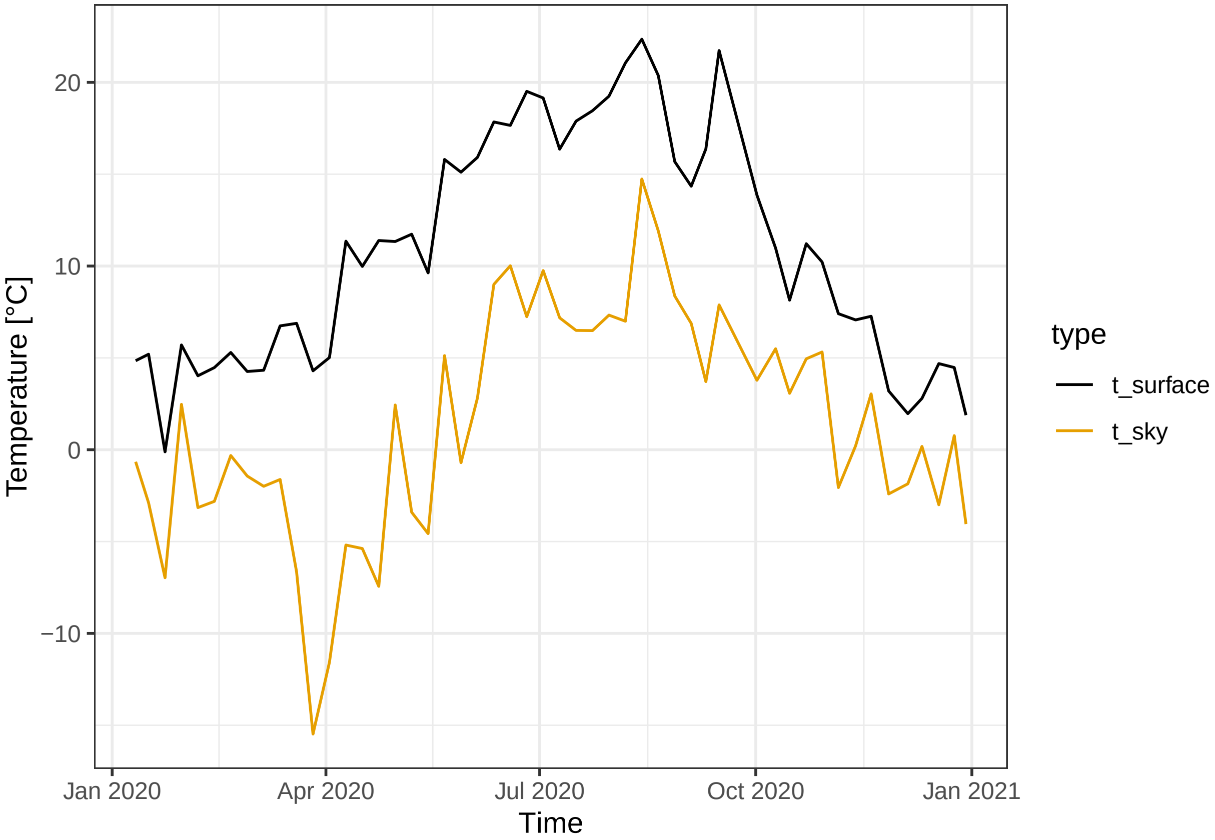

Figure 8: Weekly average of sky and surface temperatures over one year. Data from botanical garden 2020.

```
rad_d %>%
  gather(key="type", value="temp", t_surface, t_sky, factor_key = T) %>%
ggplot(aes(x=datetime, y=temp, colour=type)) +
  geom_line()  +
  scale_color_colorblind() +
  labs(y="Temperature [°C]", x="Time")
```

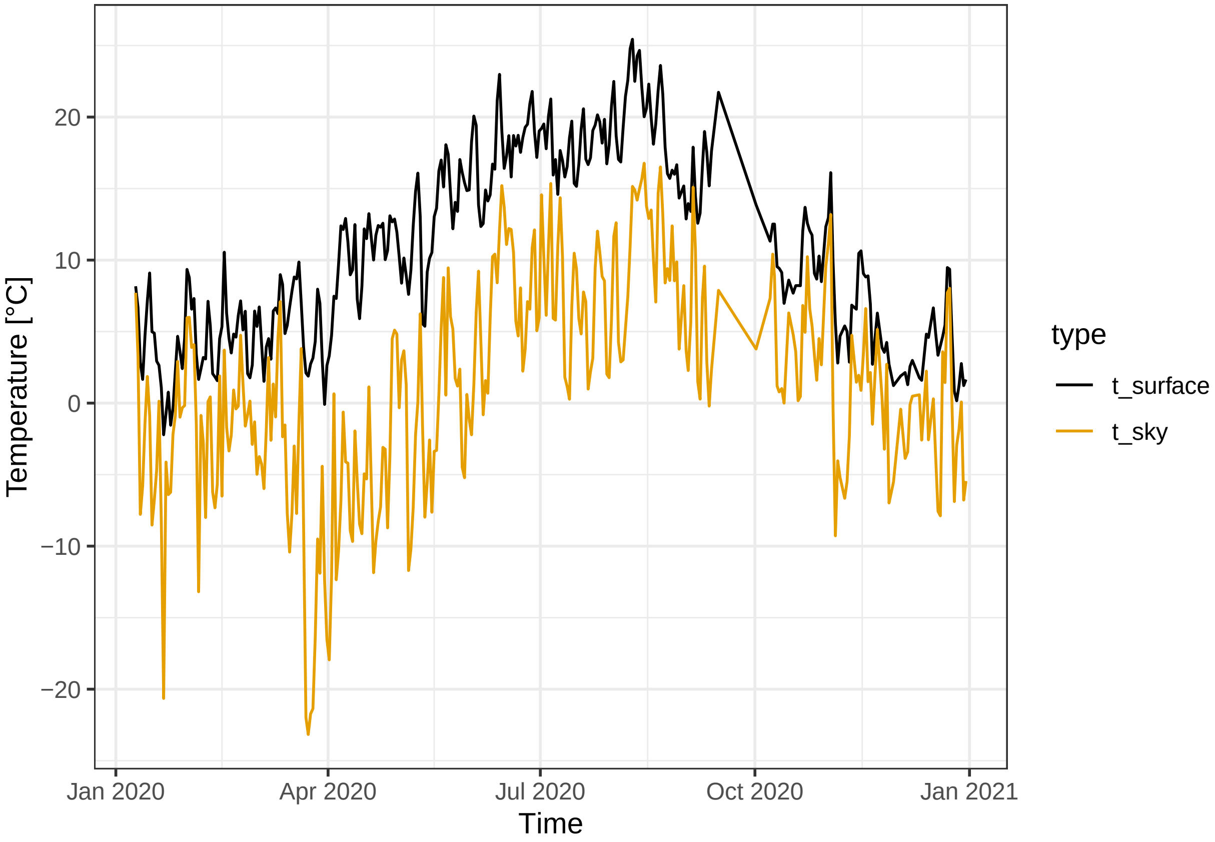

Figure 9: Daily average of sky and surface temperatures over one year. Data from botanical garden 2020.

The difference between the sky at the surface temperature is also analyzed using high frequency data (10 minutes) for a month (Figure 10) and a week (Figure 11).

The surface temperature has a clear day cycle and in the month of July for the majority of the time oscillates in the 10°C - 30°C range.

On the other hand the sky temperature has no daily cycle, but over the month has still important variations from -5°C to 20 °C.

Moreover, it can be clearly seen how during cloudy days (eg. 9th of July) there is a high sky temperature, but a low surface temperature. Conversely, on sunny days (eg. 7th of July) the surface temperature is higher, but the sky temperature is low.

```r
rad %>%
  filter( month(datetime) == 7 ) %>%
  gather(key="type", value="temp", t_surface, t_sky, factor_key = T) %>%
ggplot(aes(x=datetime, y=temp, colour=type)) +
  geom_line()  +
  scale_color_colorblind() +
  labs(y="Temperature [°C]", x="Time", col="") +
  theme(legend.position = "bottom")
```

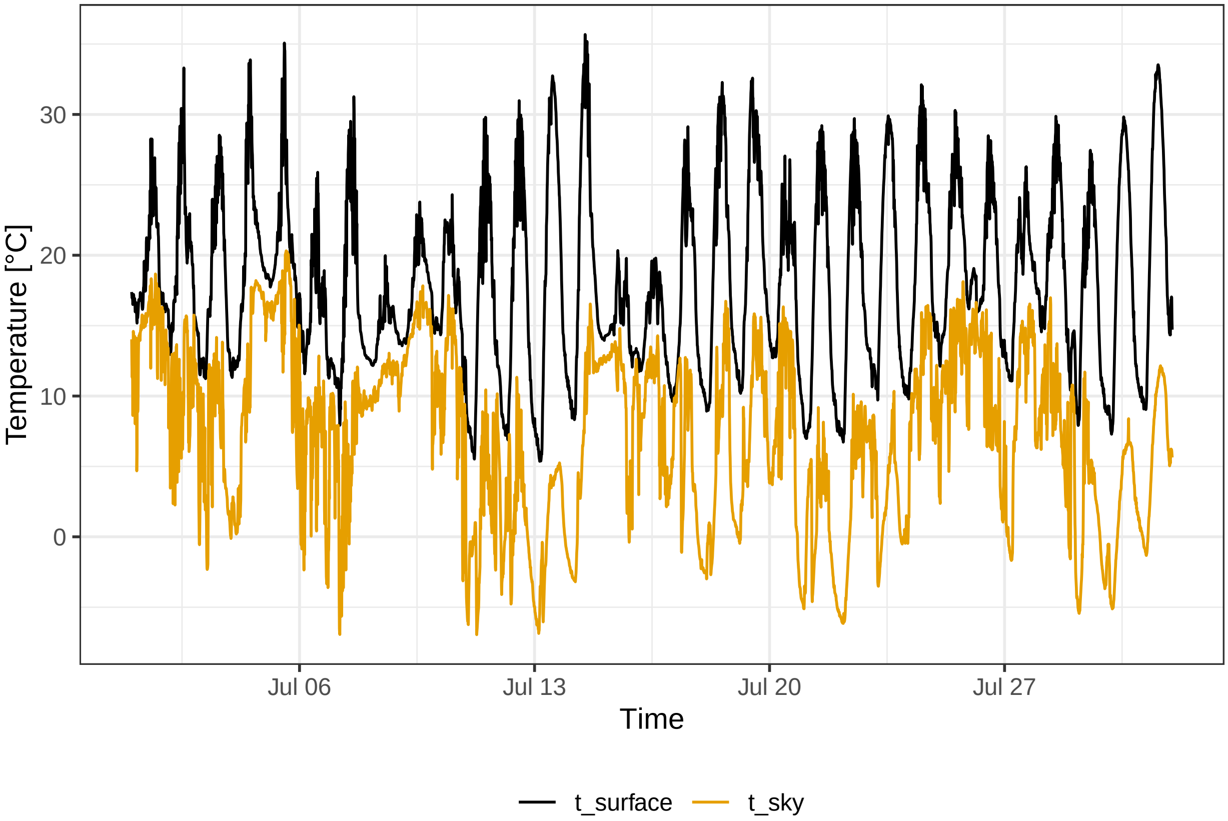

Figure 10: Sky and surface temperatures during July 2020. Data frequency 10 minutes. Data from botanical garden.

```
filter(rad, between(datetime, as_datetime("2020-07-03"), as_datetime("2020-07-12"))) %>%
  gather(key="type", value="temp", t_surface, t_sky, factor_key = T) %>%
ggplot(aes(x=datetime, y=temp, colour=type)) +
  geom_line()  +
  scale_color_colorblind() +
  labs(y="Temperature [°C]", x="Time", col="") +
  theme(legend.position = "bottom")
```

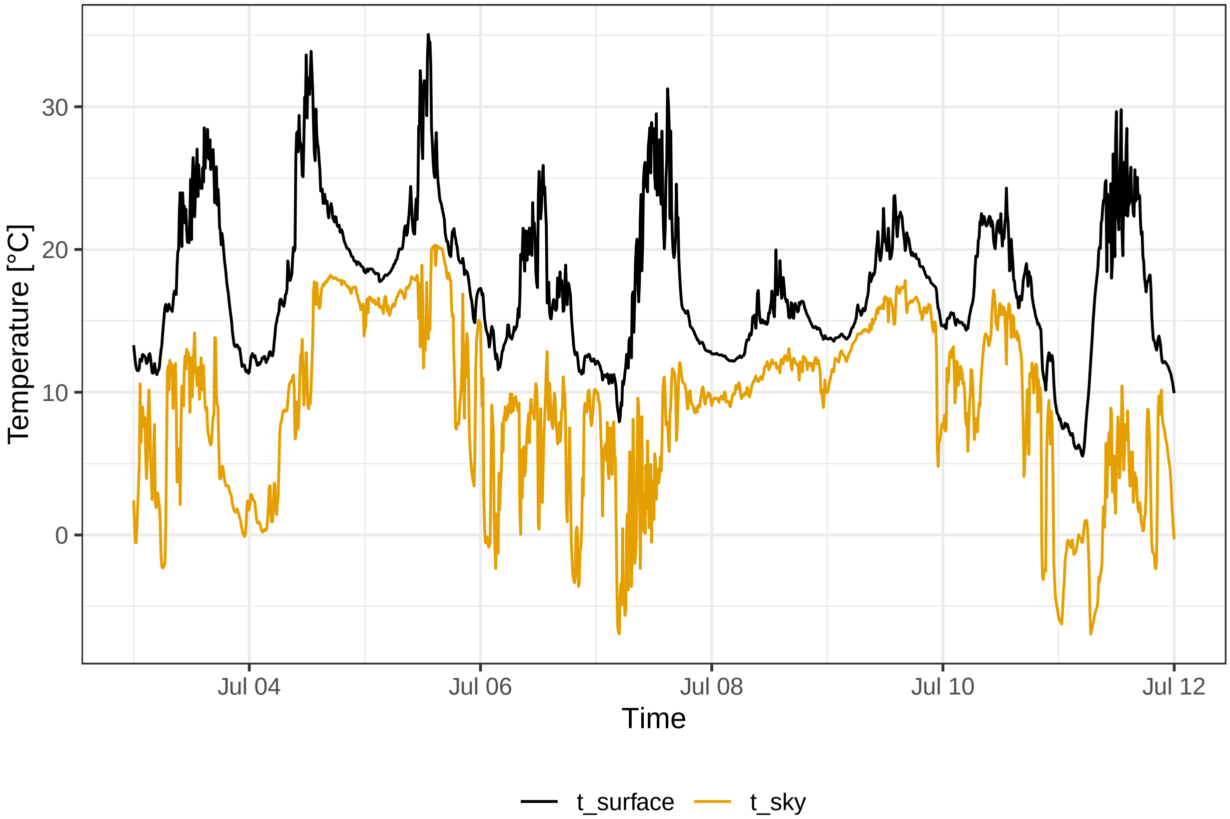

Figure 11: Sky and surface temperatures during first weeek of July 2020. Data frequency 10 minutes. Data from botanical garden.

2.4.2 Net radiation

The longwave and shortwave components are merged to calculate the overall net radiation for one year in figure 12.

The net radiation has a yearly cycle. During the summer it has a relatively constant value at around $100W/m^2$, then it decrease and reach slightly negative values in January. The biggest driver of this yearly cycle is the incoming shortwave radiation, which during summer is much higher than in winter. The radiation from the sun has smooth variations, while the variation on the incoming shortwave during the summer can be explained by the different amount of cloud cover. You would expect a clearer peak of the shortwave radiation during the summer, Moreover the net radiation has an high peak in mid late September. This behavior can probably be explained by different amount of cloud cover.

The outgoing shortwave is the component with the smallest absolute value, it also has a yearly cycle being virtually zero in January but quickly reaching the max value during the spring and then remaining quite flat. Regarding the longwave the outgoing radiation is always bigger than the incoming, due to the higher temperature of the surface compared to the sky. The longwave components have a much smaller change during the year.

There is a notable low peak of incoming longwave in the last week of march, that is probably explained by clear skies but still low air temperature.

```
rad_w %>%
  gather(key="type", value="radiation", lw_out, lw_in, sw_in, sw_out, net_rad,
         factor_key = T) %>%
ggplot() +
  geom_line(aes(x=datetime, y=radiation, colour=type)) +
  scale_color_colorblind() +
  labs(y="Radiation [W m-2]",
       x="Time", caption = "Weekly average", colour="Radiation")
```

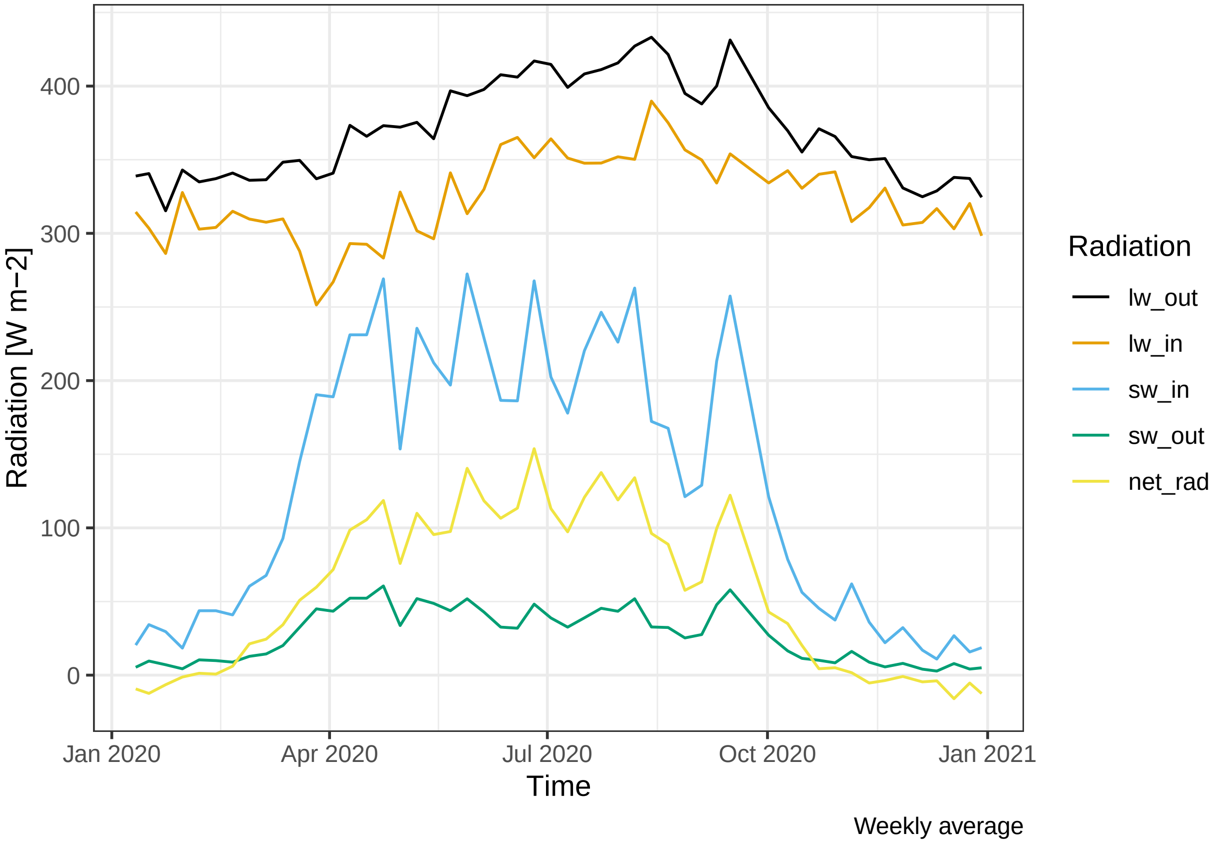

Figure 12: Net radiation over the year. The four components of the radiation (shortwave incoming, shortwave outgoing, longwave incoming, longwave outgoing) are also showed. Data averaged over a week. Data from botanical garden 2020.

2.4.3 Change emissity in the sensor

In the field activity we tried to measure the temperature of the surface by using different emissivity settings in the sensor and see how that could influence the readings. However, there have been some issues with the sensor, so the data has been generated using the formula from the theory

In this virtual experiment the real temperature is set to 19 °C and the emissivity is changed, resulting in different temperature estimates.

```r
t_0 <- 19 # temperature with emissivity 1
rad_0 <- c2k(t_0) %>% temp2lw # connected radiation

temps <- tibble(
  em = seq(1, .1, -.05),
  t = (rad_0 / (em * sigma))^(1/4)  %>% k2c
)
```

```r
ggplot(temps, aes(em, t)) +
  geom_line() +
  scale_x_reverse() +
  labs(x="Emissivity", y="Temperature [°C]")
```

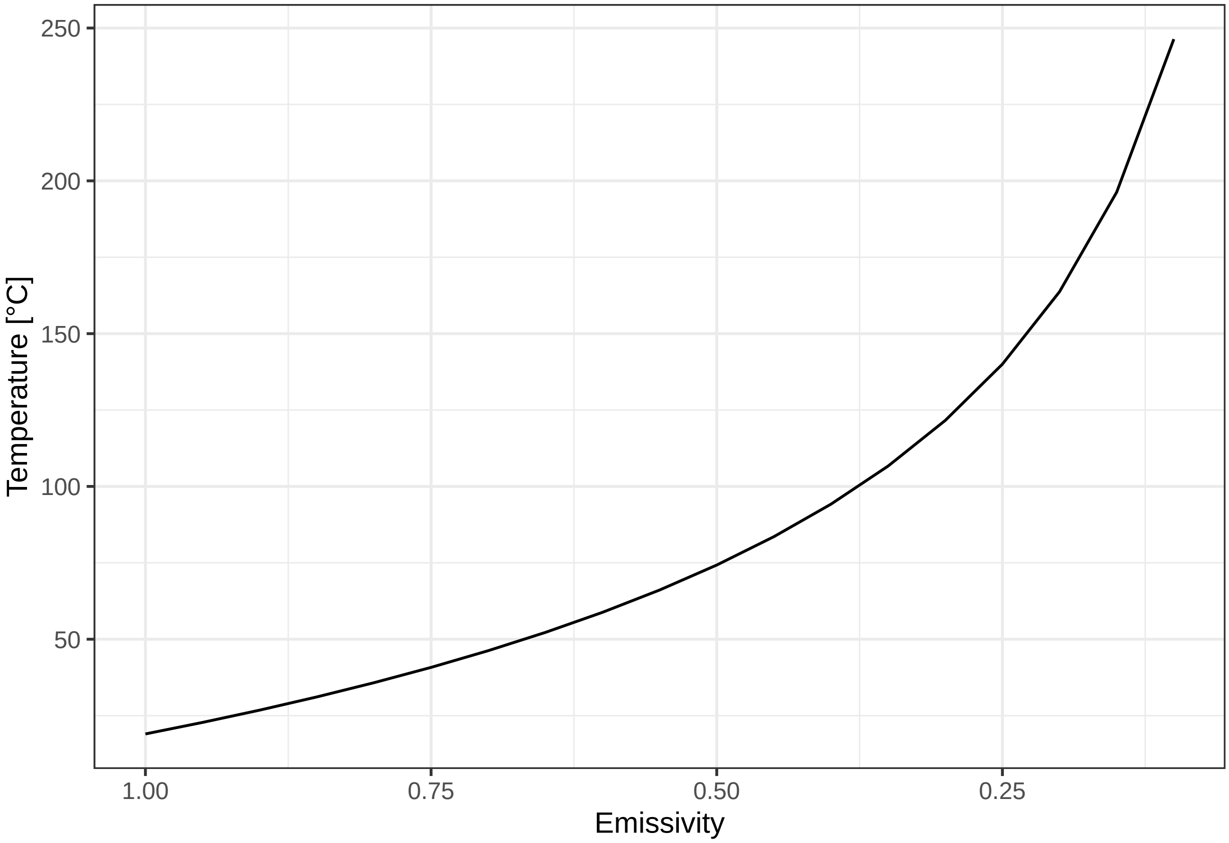

Figure 13: Estimanted temperature measured by infrared themometer for different emissivity. Temperature at emissivity 1 is 19 °C

It can be seen that the emissivity has a big influence on the temperature estimate. The emissivity of material can change drastically from 0.03 for aluminum foil to 0.97 for ice. This clearly shoes the importance of a correct estimation of the emissivity for temperature measurements.

3 Air temperature

3.1 Motivation

Temperature is arguably the environmental variable with the biggest impact of ecosystems. The speed of all chemical reactions is strongly influenced by temperature in a non-linear way, therefore processes like photosynthesis and respiration shows an strong response to temperature changes. Moreover, thermal energy is the main way ecosystem can store energy, hence the temperature constantly change to maintain energy balance. Finally the temperature often influences water availability.

3.2 Background

Temperature is defines as a state variable, which describes the mean kinetic energy of molecules. The SI unit to measure temperature is the Kelvin degree K, that use a reference point the absolute zero, the temperature when there is no motion of molecules.

Often other measurement units are used mainly degrees Celsius (°C) and Fahrenheit (°F).

In a gas the temperature is related to the pressure by the ideal gas law:

$$pV = nRT$$

Therefore in the troposphere there is a drop in temperature with height, as the pressured is reduced with height.

3.3 Sensors and measuring principle

Temperature is measured using different principles, for analog sensors the the measures are connected to thermal expansion of materials. The increase in temperature results in an increase of volume or length and its magnitude on the materials properties. In particular the most common sensors are:

- **Mercury Thermometer**. The sensor has a bulb filled with mercury (or other liquid) that expands or contracts with the change in temperature, that is then connect to a graded pipe where is possible to make the reading.
- **Bimetal Thermometer**. The sensor is based on the different expansion coefficient of metals, hence by putting two different metals (eg. iron and copper) next to each other the temperature can be measured by the amount of the bent.
- **Thermocouple**. Thermocouples measure the voltage at the junction of two metals, which in turn depends on temperature.
- **Resistance**. They measure the change in resistance due to the change of temperature. There are two types of sensors with diffenrt type of responses,
  - *Positive temperature coefficient (PTC).* Increase Usually made of platin.
  - *Negative temperature coefficient (NTC).* Decrease the resistance with an increase in temperature, they are usually made of semi conductors

3.4 Analysis

3.4.1 Hainich time series

```
temp <- read_csv(here("Data_lectures/3_Air_temperature/Hainich_T_air_soil_degC.csv")) %>%
  mutate(diff_canopy = TA_44m - TA_2m) %>%
  drop_na()
```

The temperature changes at different height of the canopy, due to the different incoming solar radiation and emitted longwave radiation. Similarly the top soil temperature can be significantly different from the air temperature just above the soil, this is due to the higher heat capacity and lower conductivity of the soil compared to the air. In this analysis the air temperature at 2 meters is compared with the temperature 2 cm below the soil.

The soil temperature daily variation is limited compared to the air one (Figure 14 and 15). In one day the air temperature can change up to 10 °C, while the soil temperature only a few degrees. However, over the year the total variation of air and soil temperature are similar (Figure 14). Moreover also the yearly mean are similar (Table **??**).

In general the air and the soil temperature follow a similar pattern, but the daily oscillation in the air temperature are bigger (Figure 16).

```
temp %>%
  group_by(Date = round_date(Date, "1d")) %>%
  summarize_all(mean) %>%
  ggplot(aes( x = Date)) +
  geom_line(aes(y=Tsoil_002m_degC, col="Soil 2 cm")) +
  geom_line(aes(y=TA_2m, col="Air 2 meters")) +
  scale_color_colorblind() +
  labs(y = "Temperature [°C]", col="Heigth")
```

```
temp %>%
  filter(between_dates(Date, "1 May 2019", "30 May 2019")) %>%
  ggplot(aes(Date)) +
  geom_line(aes(y=Tsoil_002m_degC, col="Soil 2 cm")) +
  geom_line(aes(y=TA_2m, col="Air 2 meters")) +
  scale_color_colorblind() +
  labs(y = "Temperature (°C)", col="Heigth")
```

```
temp %>%
  group_by(Date = round_date(Date, "1d")) %>%
  summarize_all(mean) %>%
  ggplot(aes(TA_2m, Tsoil_002m_degC)) +
  geom_point() +
  scale_color_colorblind() +
  geom_smooth(method = "lm", se=F, colour = colorblind_pal()(2)[2]) +
  labs(x="Temperature air 2m [°C]", y = "Temperature soil 2cm [°C]")
```

```
tribble( ~"Variable", ~"Value",
  "Correlation air and soil temperature", cor(temp$TA_2m, temp$Tsoil_002m_degC),
  "Mean temperature air", mean(temp$TA_2m),
  "Mean temperature soil",  mean(temp$Tsoil_002m_degC)) %>%
  kable(booktabs=T, caption="Summary soil and air temperature") %>%
  kable_styling(latex_options = "hold_position")
```

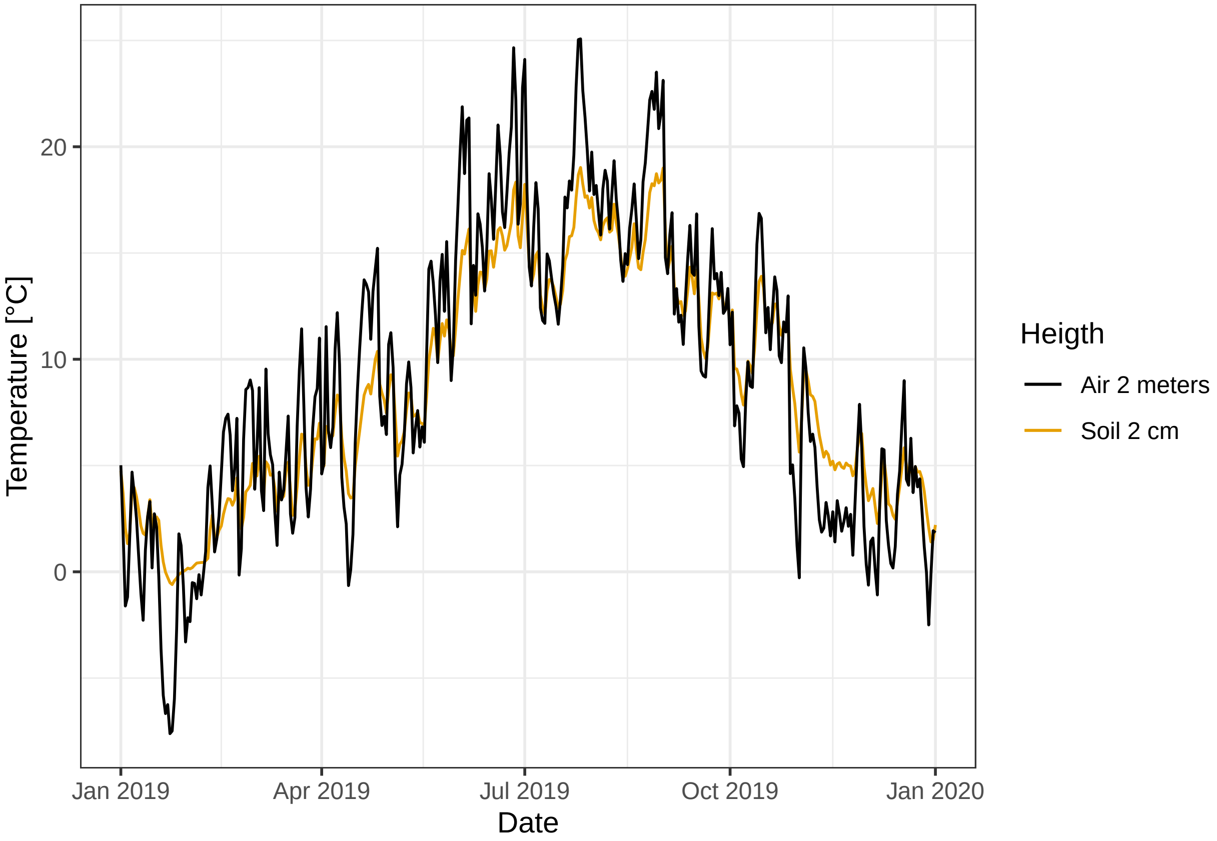

Figure 14: Comparison time series temperature at 2 m and 2 cm in the soil for 2019. Measurement averaged over 1 day. Data from Hainich national park

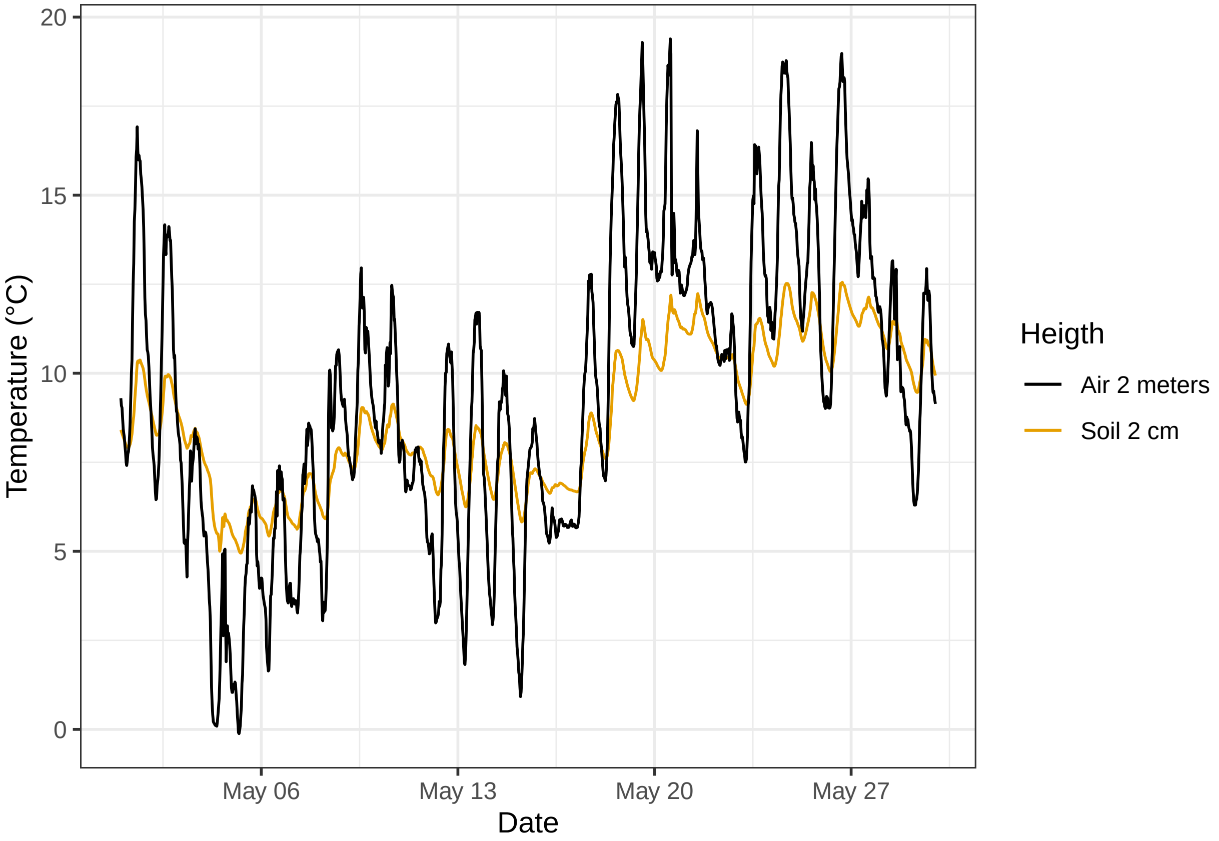

Figure 15: Comparison time series temperature at 2 m and 2 cm in the soil for the month of May 2019. measurement frequency 30 min. Data from Hainich national park

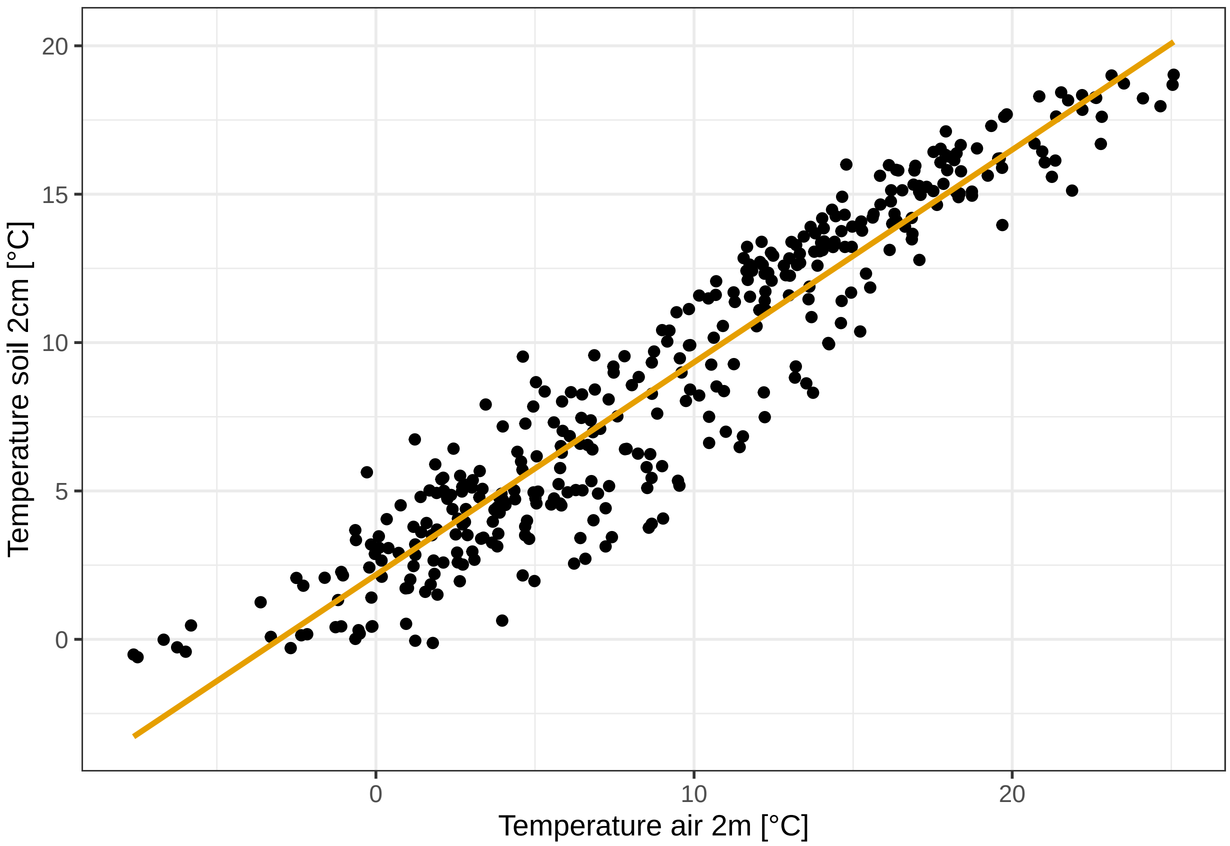

Figure 16: Scatter plot with regression line between temperature at 2 m and 2 cm in the soil for 2019. measurement averaged over 1 day. Data from Hainich national park.

Table 1: Summary soil and air temperature

Variable	Value
Correlation air and soil temperature	0.9287618
Mean temperature air	9.1323011
Mean temperature soil	8.7184121

3.4.2 Temperature sensors response time

The goal of the field experiment is to estimate the response time of a resistance thermometer and a mercury one.

To estimate the response time we can start from the following equation that describes the temperature decrease over time. Then this can be inverted and used to estimated by τ fitting a linear model.

$$T_t = T_a + (T_0 - T_a)e^{\frac{T}{-\tau}}$$

In the field the sensors where heated up and then the temperature measured every 10 seconds while they cooled down. The air temperature was also recorded for reference.

```
res_wire <- 3.7 # Omega. This has been measured in the field
r_0 <- 100 # Omega at 100 degrees
a <- 4e-3

#' converts resistance to temperature
get_temp <- function(resistance) {
  (((resistance - res_wire) / 100) - 1) / 4e-3
}
```

```
temp_res <- read.csv(here("3_air_temperature/resistance_sensor_cooling response.csv"),
                     header = T)
```

```
# add the time (in seconds) and the temperature after conversion from resistance
temp_res <- temp_res %>%
  mutate(time = 0:(nrow(temp_res)-1) * 10,
         temp = get_temp(resistance) )
```

```
T_a <- 13.0 # calculations done in the field
T_0 <- temp_res$temp[1] # first measure

temp_res <- temp_res %>%
  mutate(log_t = log((T_0 - T_a) / (temp - T_a)) )
```

```
model_res <- lm(time ~ log_t, data=temp_res)
```

```
ggplot(temp_res, aes(time, temp)) +
  geom_line() +
  labs(y = "Temperature [°C]", x = "Time [s]")
```

```
temp_merc <- read.csv(here("3_air_temperature/mercury_termometer_cooling_response.csv"),
                      header = T)
```

```
T_a <- 7.0 # ambient temperature
T_0 <- temp_merc$temp[1] # first measure

temp_merc <- mutate(temp_merc, log_t = log((T_0 - T_a) / (temp - T_a)) )
```

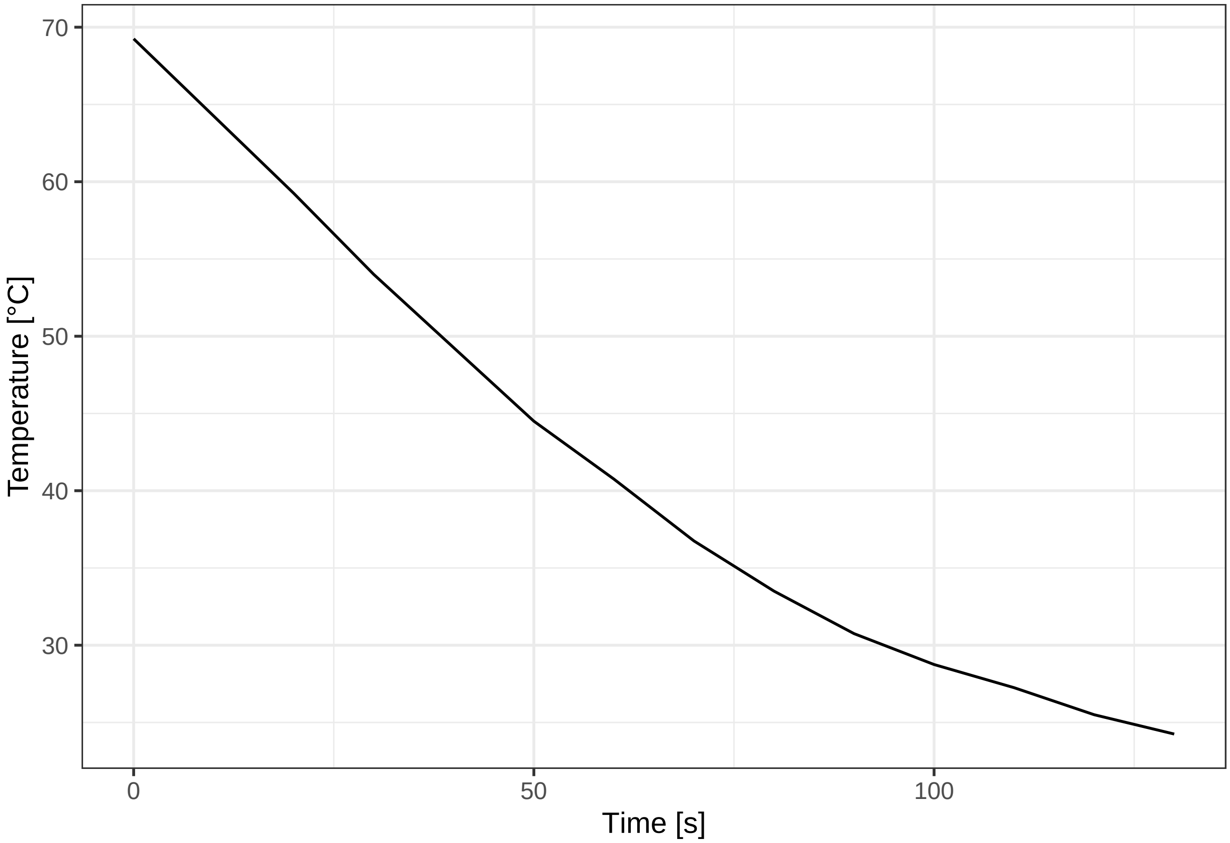

Figure 17: Temperature decrease over time for resistance thermometer.

```
model_merc <- lm(time ~ log_t, data=temp_merc)
```

```
ggplot(temp_merc, aes(time, temp)) +
  geom_line() +
  labs(y = "Temperature [°C]", x = "Time [s]")
```

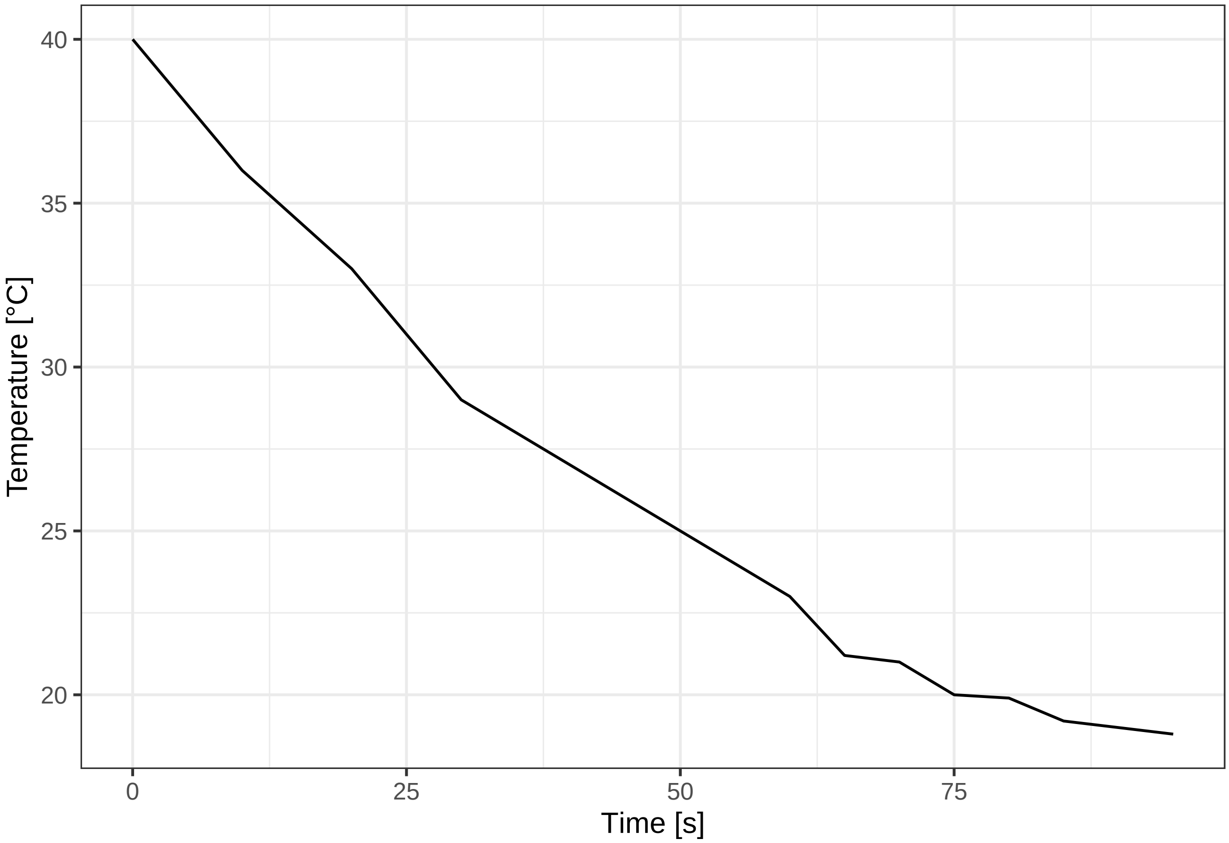

Figure 18: Temperature decrease over time for mercury thermometer.

The time response rate for the resistance sensor is 77 seconds.

The time response rate for the mercury sensor is 88 seconds.

The mercury thermometer has an higher response time than the resistance one.

This is against the expectations as the mass, and therefore the thermal inertia, of the mercury sensor is smaller than the resistance one.

4 Air humidity

4.1 Motivation

Air is a mixture of different gases and water vapour is a small fraction but still plays a crucial role for ecosystems.

The movement of water into the plant is driven by the transpiration, which happens as water leaving the plant from the leaf stomata. The features of this phenomenon are dependent on the amount of water present in the atmosphere, the lower the humidity the higher the stomata conductivity.

Water vapour is also important in the energy balance of ecosystems , as an significant fraction of the sun energy is converted is latent heat flux, or water evaporating goes from the surface.

4.2 Background

The behaviors of water in the air are complex and depending on the contest the are different variable used to measure the air humidity and connected parameters. They are:

- Actual vapor pressure $[hPa]$

The actual vapour pressure is the partial vapour pressure of water, which is the fraction of the total air pressure that is due to water moleculs. It is defined using the ideal gas law as following: It is one of the most common measurement of the air water content.

$$e_a = \rho_v R_v T$$

- Saturation vapour pressure $[hPa]$

The amount of water vapour that can be contained into the air is limited, the saturation vapour pressure is the water vapour pressure that a saturated air would have. It is strongly dependent on temperature increasing exponentially with incrase in temperatures. It can be estimated with the following formula:

$$e_s = 6.1078e^{\frac{17.08085T_a}{234.175\text{ř}C+T_a}}$$

- Relative humidity $[\%]$

The relative humidity is ratio between the actual vapur pressure and the saturation vapour pressure expressed in percentage. It is a common way of describing humidity.

- Dewpoint temperature $[°C]$

The dew point temperature is the temperature when the water in the air would start condensing. This happens when the actual vapur pressure becomes equal to the saturation one. It can be estimated with the following formula:

$$T_d = \frac{(len(e_a) - ln(6.1708)) \cdot 234.17}{17.08085 - ln(e_a) + ln(6.1078)}$$

- Absolute humidity $[Kg \quad m^{-3}]$

The absolute humidity represent the actual mass of water present in a unit volume of air. It is defined as the density of water vapour:

$$\rho_v = \frac{e_a}{(R_v T)}$$

- Specific humidity $[Kg/Kg]$

The specific humidity (q) is the ratio between the density of water vapour of the density of moist air

$$q = \frac{e_a}{p - 0.378 e_a}$$

- Mixing ratio $[Kg/Kg]$

The specific humidity (q) is the ratio between the density of water vapour of the density of dry air

$$q = \frac{e_a}{p - 0.378 e_a}$$

- Vapor pressure deficit $[hPa]$

The vaopur pressure deficit is the difference between the saturation vapour pressure and the actual vapour pressure. It indicates which is the amount of water that can still be added in the atmosphere before reaching saturation.

- Equivalent temperature $[°C]$

The equivalent temperature is the temperature that would be reached if the energy release by the condesation of all water is used to heat up the air mass. Therefore by definition it is always higher than the actual air temperature.

Can be estimated with the following formula:

$$T_{eq} = T_a + \frac{L_v}{C_p} m \approx T_a + 2.5m$$

where:

- $L_v = 2.510^6 JKg^{-1}$ is the latent heat of vaporization
- $C_p = 1004.6 JKg^{-1}K^1$ is the specific heat capacity of dry air
- m = mixing ratio in g/Kg
- T_a = air temperature in K

4.3 Sensors and measuring principle

There are numerous principles used to measure variable related to air humidity.

- **Condensation**. The condensation hygrometers measure the dew point temperature, when the moisture in the air start to condense.
  - **Dew point mirror**. There is a mirror that is cooled down to the dew point temperature. Small droplets on the mirror are then detected on the top of the mirror by a reflected LED light.
  - **LiCL dew point hygrometer**. The is a solution of LiCl, that is hygroscopic and conducts electricity only when wet. This is connected to an heating system is used to estimate the dew point temperature.
- **Hygroscopic**. The basic principle is the change of dimension due to the change in humidity. The most common sensor to use this principles is the **Hair hygrometer**, that uses a series of proerply treated horse hairs connected to a spring that change their size in relation to the relative humidity.
- **Spectroscopric**. Different gas absorbs certain infrared light at specific wavelengths and water vapour has it own characteristic asbosption bands. **Infrared gas analyzer** measure the amount of IR light that is absorbed by the air on some specific wavelenght and can use this information to estimate the number of water molecules. This sensors are used in Eddy Covariance setups and can provide high frequency and high accuracy measurements.
- **Capacitative**. Some material change their dialectic properties with the change of humidity, hence by measuring the change of capacity it is possible to estimate the air humidity. This sensors are fast, accurate and have long-term stability making them common.

- **Psychrometric**. The humidity is estimated by measuring the normal air temperature and the air temperature of a wet themometer. The wet thermometer will have a lower temperature as water evaporation substract energy. The amount of evaporation is related to the air humidity, hence by knowing the entity of the temperature reduction of the wet thermometer it is possible to calculate the air humidity. This principle is used by the Assmann and spinning Psychrometers, which allows to have an analong measurement of air humidity with a good accuracy.

4.4 Analysis

Using the field meaurement of relavitve humidity, air temperature and air pressure from the Hainich national park the following variables were calculated and used of the subsquent analysis.

1. Actual vapor pressure
2. Saturation vapor pressure
3. Dewpoint temperature
4. Absolute humidity
5. Specific humidity
6. Mixing ratio
7. Vapour pressure deficit
8. Equivalent temperature

```
hum <- read_csv(here("Data_lectures/4_air_humidity/04_Air_humidity_TA_RH_PA_NP_Hainich.csv"))
names(hum) <- c("datetime", "ta", "rh", "pa")
```

```
c2k <- function(c) c + 273.15
k2c <- function(k) k - 273.15

Rv <- 461.47 # J K -1 kg -1 ] - gas constant of water vapour
get_es <-  function(ta)  6.1078 * exp((17.08085 * ta) / (234.175 + ta))
get_td <- function(e_a) ((log(e_a) - log(6.1708)) * 234.17) / (17.08085 - log(e_a) + log(6.1078))
get_rh <- function(e_a, e_s) e_a/e_s * 100
rh2ea <- function(rh, e_s) rh/100 * e_s

# need to convert e_a from hPa to Pa and the temperature to degrees Kelvin
# convert the output in g/Kg
get_abs_hum <- function(e_a, ta) (e_a * 100 / (Rv * c2k(ta)) ) * 1000

#here there is no need to convert to Pa
# because the pressures is present both at numerator and denominator
get_spec_hum <- function(e_a, p) 0.622 * e_a / (p - 0.378 * e_a) * 1000 # g/Kg
get_mix_ratio <- function(e_a, p) 0.622 * e_a / (p - e_a) * 1000 # g/Kg


get_p_def <- function(ea, es) es - ea
get_t_eq <- function(ta, mix_ratio) k2c(c2k(ta) + 2.5 * mix_ratio)

get_ea_dry <- function(es_wet, t_dry, t_wet, p){
  a <- (p * 1004.6) / (0.622 * 2.5061e6)
  return(es_wet - a * (t_dry - t_wet))
}

# utility func for plotting
remove_x_axis <- function() theme(
    axis.text.x = element_blank(),
    axis.title.x = element_blank(),
    axis.ticks.x = element_blank()
  )
```

```
# add all the variables to the humidity dataframe
hum <- mutate(hum,
    es = get_es(ta),
    ea = rh2ea(rh, es),
    td = get_td(ea),
    abs_hum = get_abs_hum(ea, ta),
    spec_hum = get_spec_hum(ea, pa),
    mix_ratio = get_mix_ratio(ea, pa),
    p_def = get_p_def(ea, es),
    t_eq = get_t_eq(ta, mix_ratio)
  )

hum_d <- hum %>%
  group_by(week=yday(datetime)) %>%
  summarize_all(mean, na.rm=T)

hum_w <- hum %>%
  group_by(week=week(datetime)) %>%
  summarize_all(mean, na.rm=T)
```

4.4.1 Actual and saturation vapour pressure

The saturation vapour pressure and the actual vapour pressure are compared during one year (Figure 19) and during one week time period (Figure 20).

```
hum_d %>%
  gather("type", "val", es, ea, factor_key = T) %>%
  ggplot(aes(datetime, val, colour=type))+
    geom_line()+
  labs(x="Time", y="Vapour pressure [hPa]", colour="Vapur pressure") +
  scale_color_colorblind(labels=c("saturation", "actual")) +
  theme(legend.position = "bottom")
```

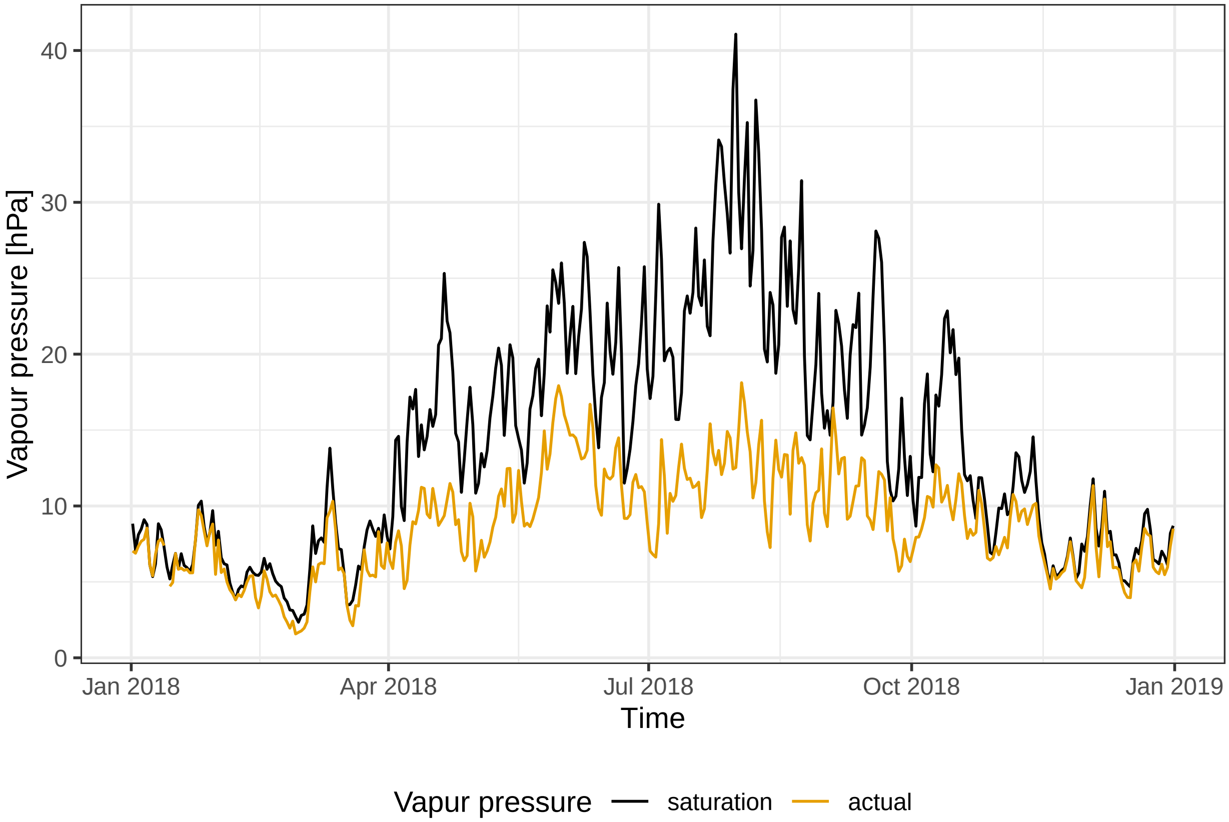

Figure 19: Comparison between saturation vapour pressure and actual vapour pressure for one year. Data averaged on a day. Data from Hainich national park 2018.

```
hum %>%
  filter(week(datetime) ==1) %>%
  gather("type", "val", es, ea, factor_key = T) %>%
  ggplot(aes(datetime, val, colour=type))+
    geom_line()+
    labs(x="Time", y="Vapour pressure [hPa]", colour="Vapur pressure") +
    scale_color_colorblind(labels=c("saturation", "actual"))
```

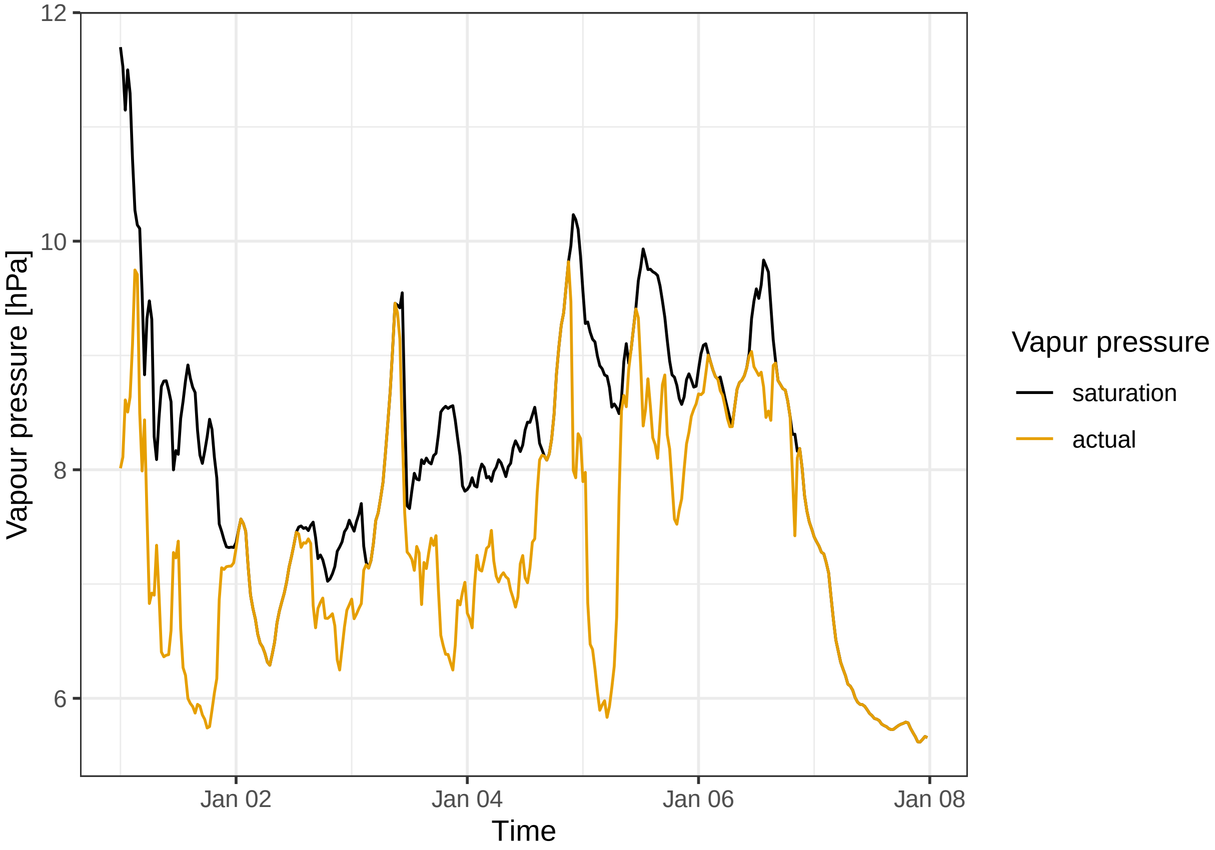

Figure 20: Comparison between saturation vapour pressure and actual vapour pressure for the first week of the year. Data averaged over 30 minutes. Data from Hainich national park 2018.

During the year the saturation water vapour ranges from around 5 hPa in the winter to more than 40 hPa in August. There is a clear yearly cycle that is directly connected to the change of air temperature.

The variation in actual vapour pressure are much smaller, but still show a seasonal pattern.

During the summer there is the biggest different between the actual and the saturation vapour pressure, due to the the high saturation vapour pressure but relatively limited water availability. Conversely during winter the saturation vapour pressure is much smaller and the water availability higher resulting in a small pressure deficit.

4.4.2 Air and dew point temperature

The relative humidity and the air and dew point temperature are compared (Figure 21).

```
p_td <- hum_d %>%
  gather("type", "val", ta, td) %>%
  ggplot(aes(datetime, val, colour=type))+
    geom_line()+
  labs(x="Datetime", y="Temperature [°C]", colour="Temperature") +
  scale_color_colorblind(labels=c("air", "dew point")) +
  remove_x_axis()

p_rh <- ggplot(hum_d, aes(datetime, rh))+
    geom_area() +
  geom_hline(yintercept = 100, linetype="dashed", size=.2) +
  labs(y="Rel. humidity [%]", x="Time")

p_td / p_rh +
  plot_layout(heights = c(4, 2)) +
  plot_annotation(tag_levels = "a")
```

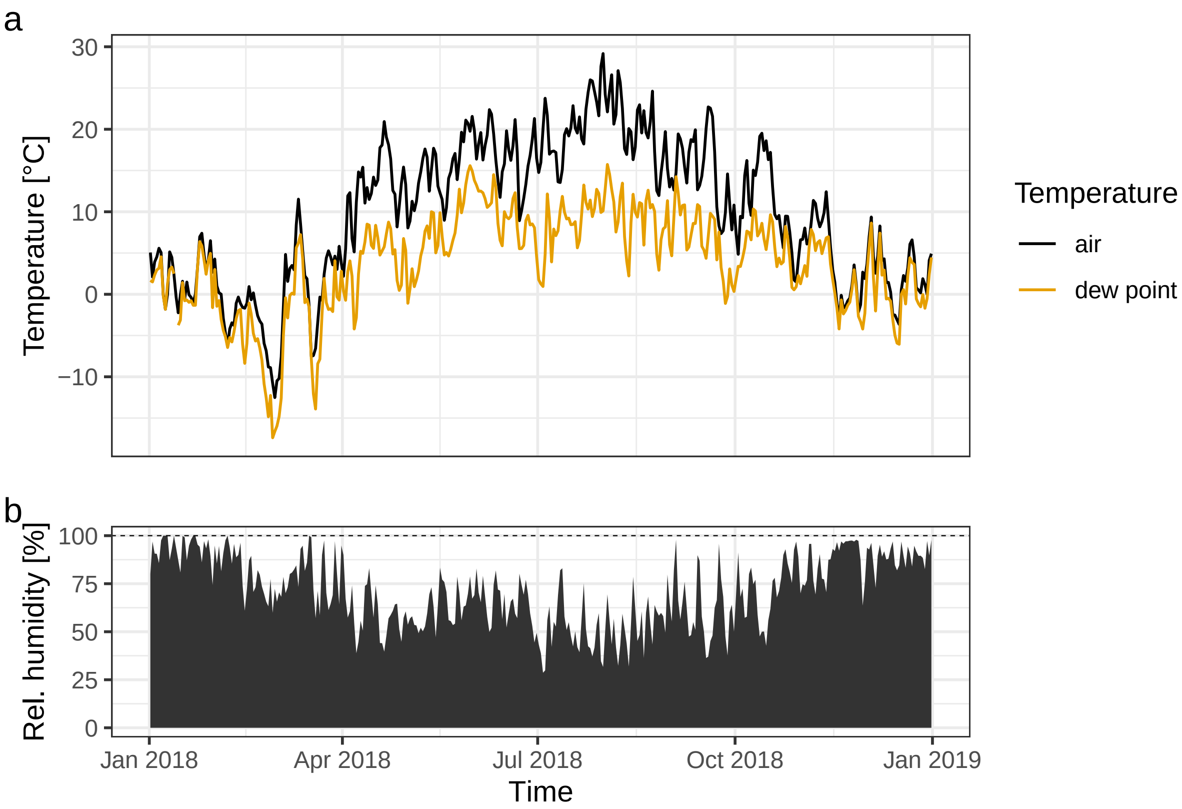

Figure 21: Comparison of air temperature with dew point temperature (fig. a), showed in relation with the relative humidity (fig. b). Data averaged over one day. Data from Hainich national park 2018.

With the analysis of the graphs, a direct relationship between a lower amount of relative humidity during summer and wider difference between the graphs of Tair and Tdewpoint can be concluded. This means, that the temperature of air reaches its peak in summer due to less relative humidity in air, while at the beginning and end of the year more similarities can be seen.

4.4.3 Absolute humidity

The absolute humidity is largest during the summer compared to winter (Figure 22). This may seem counterintuitive as during winter the relative humidity is much higher than during summer, however the amount of water that can be hold in the air during summer is much higher than during winter. In fact the is a strong relation with temperature as can be seen in the last week of March where there is a sudden drop in the the absolute humidity.

```r
hum_d %>%
  ggplot(aes(datetime, abs_hum))+
    geom_line() +
  labs(x="Datetime", y="Absolute humidity [g/kg]")
```

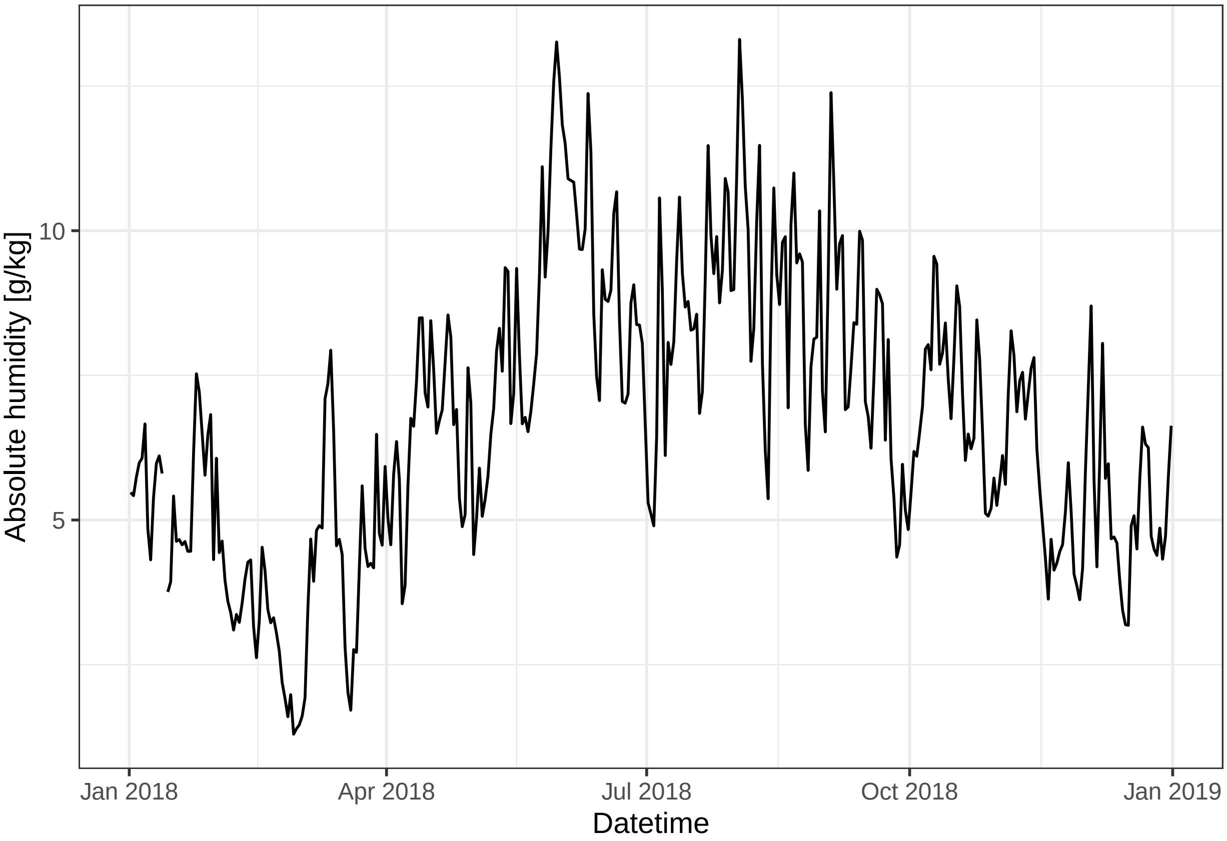

Figure 22: Absolute humidity over the year. Data averaged over one day. Data from Hainich national park 2018.

4.4.4 Mixing ratio and specific humidity

The mixing ratio and the specific are very similar (Figure 23). The only different between them is that the first if the density of water is calculated over dry air in the former and moist in the latter. It is possible to see that with higher values of mixing ratio, and therefore water content, the difference between the two variable is bigger to to the bigger difference in density between wet and dry air.

```r
hum %>%
  filter(week(datetime) ==30) %>%
  gather("type", "val", spec_hum, mix_ratio) %>%
  ggplot(aes(datetime, val, colour=type))+
    geom_line() +
    labs(y="Water content [Kg/Kg]", colour="", x="Time") +
    scale_color_colorblind(labels=c("Mixing ratio", "Specific humidity"))
```

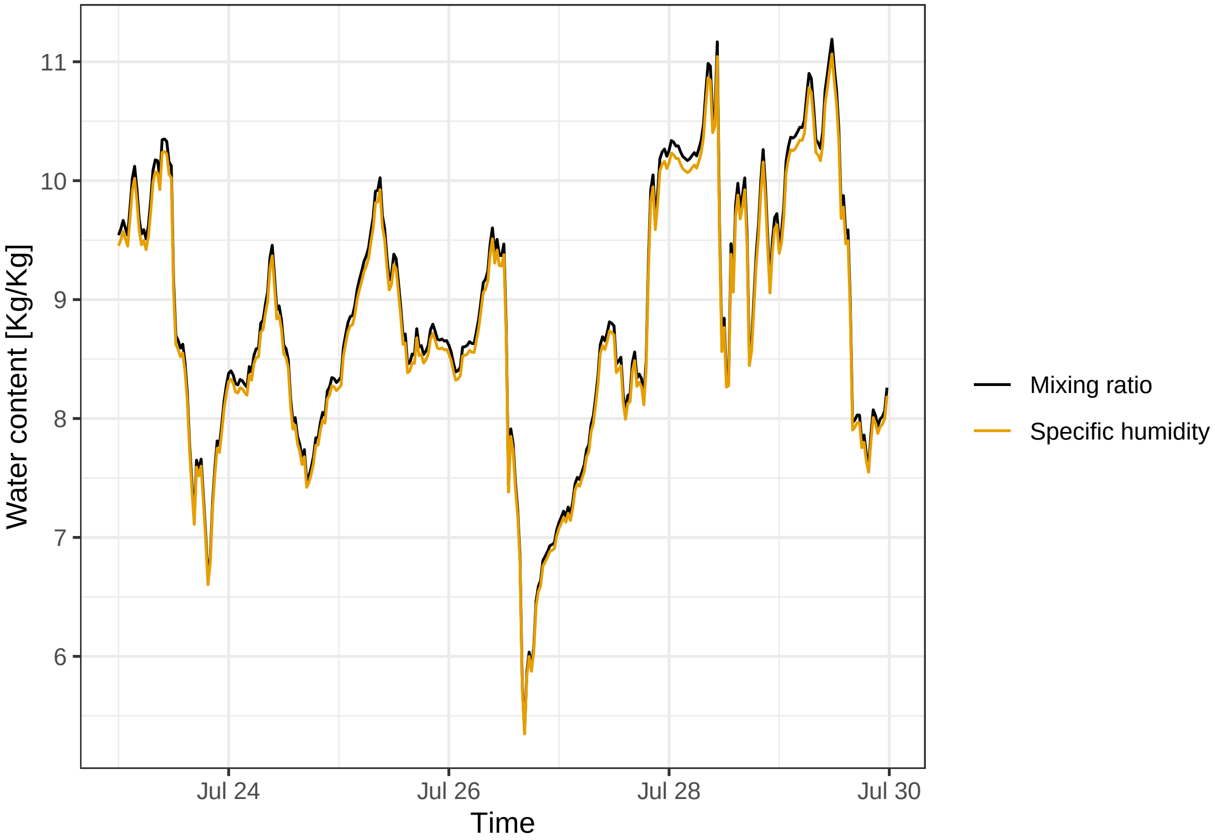

Figure 23: Comparison of mixing ratio and specific humidity. Data averaged at 30 minutes from last week of July 2018. Data from Hainich national park.

4.4.5 Vapour pressure deficit

The vapour pressure deficit has a yearly cycle (Figure 24), it is almost zero during the winter and reaches its maximum during the summer.

```
hum_d %>%
   ggplot(aes(datetime, p_def)) +
   geom_line() +
   labs(y="Pressure deficit [hPa]", x="Time")
```

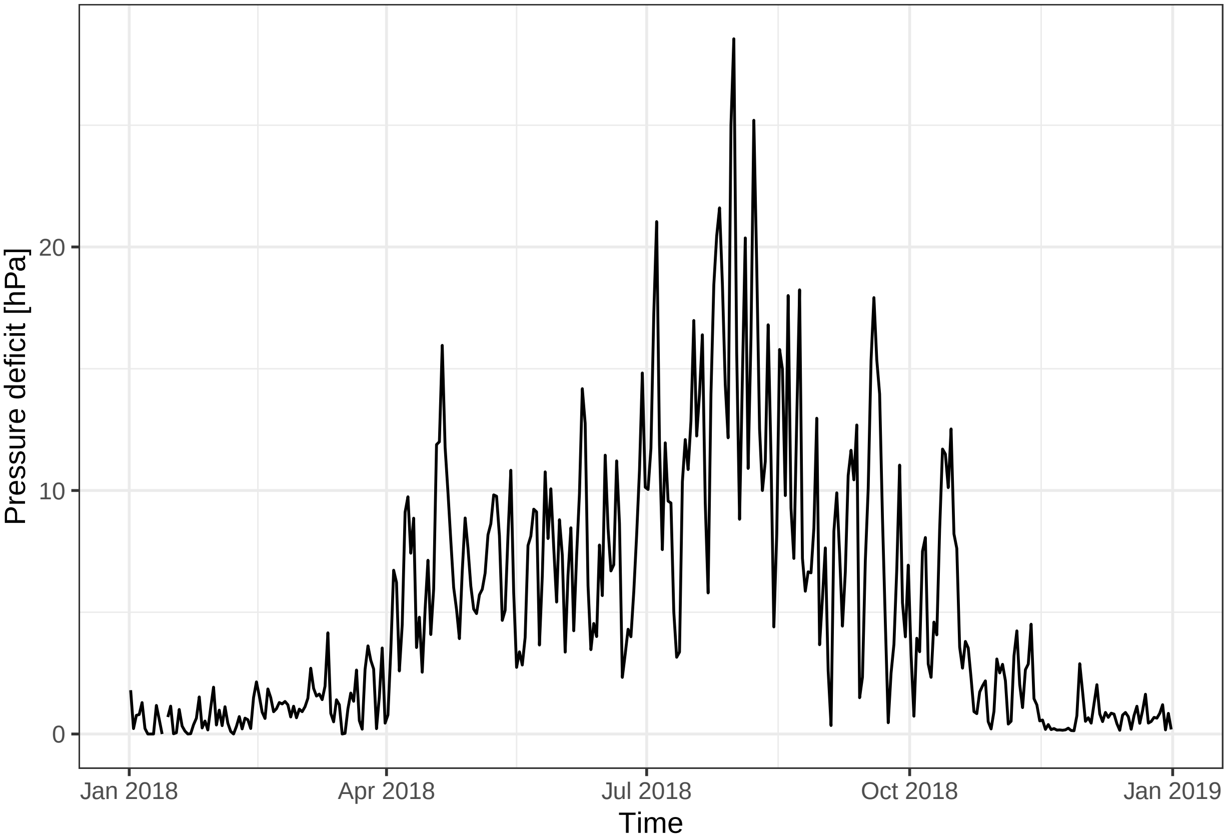

Figure 24: Vapur pressure deficit over the year. Data averaged over one day. Data from Hainich national park 2018.

The vapour pressure deficit has also a daily cycle (Figure 25), with higher values during the afternoon where the temperature is still high but the water reserves have been depleted during the day. It then reaches a minimum in the early hours of the morning, mainly due to the low temperatures.

```
hum %>%
  filter(week(datetime) == 30) %>%
   ggplot(aes(datetime, p_def)) +
   geom_line() +
   labs(y="Pressure deficit [hPa]", x="Time")
```

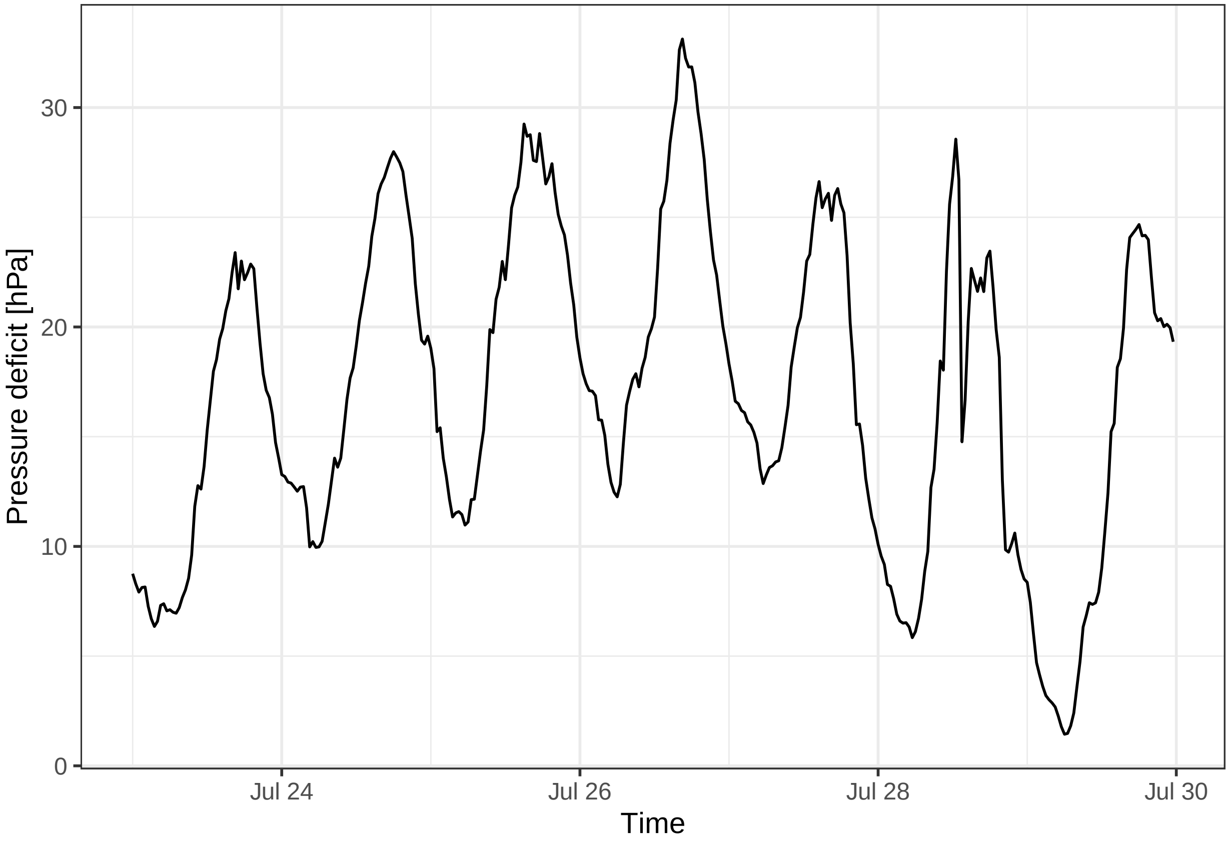

Figure 25: Vapur pressure deficit over one week. Data averaged over 30 mins. Data from Hainich national park last week july 2018.

4.4.6 Equivalent temperature

The equivalent temperature is of course strongly dependent on the base air temperature, but the difference is bigger when the absolute humidity is higher, hence it has a yearly cycle (Figure 26). During the summer the equivalent temperature is roughly the double of the air temperature.

```
hum_d %>%
  gather("type", "val", t_eq, ta, factor_key = T) %>%
  ggplot(aes(datetime, val, colour=type))+
    geom_line() +
    scale_color_colorblind(labels = c("Equivalent", "Air")) +
    labs(y="Temperature [°C]", x="Time", colour="Temperature")
```

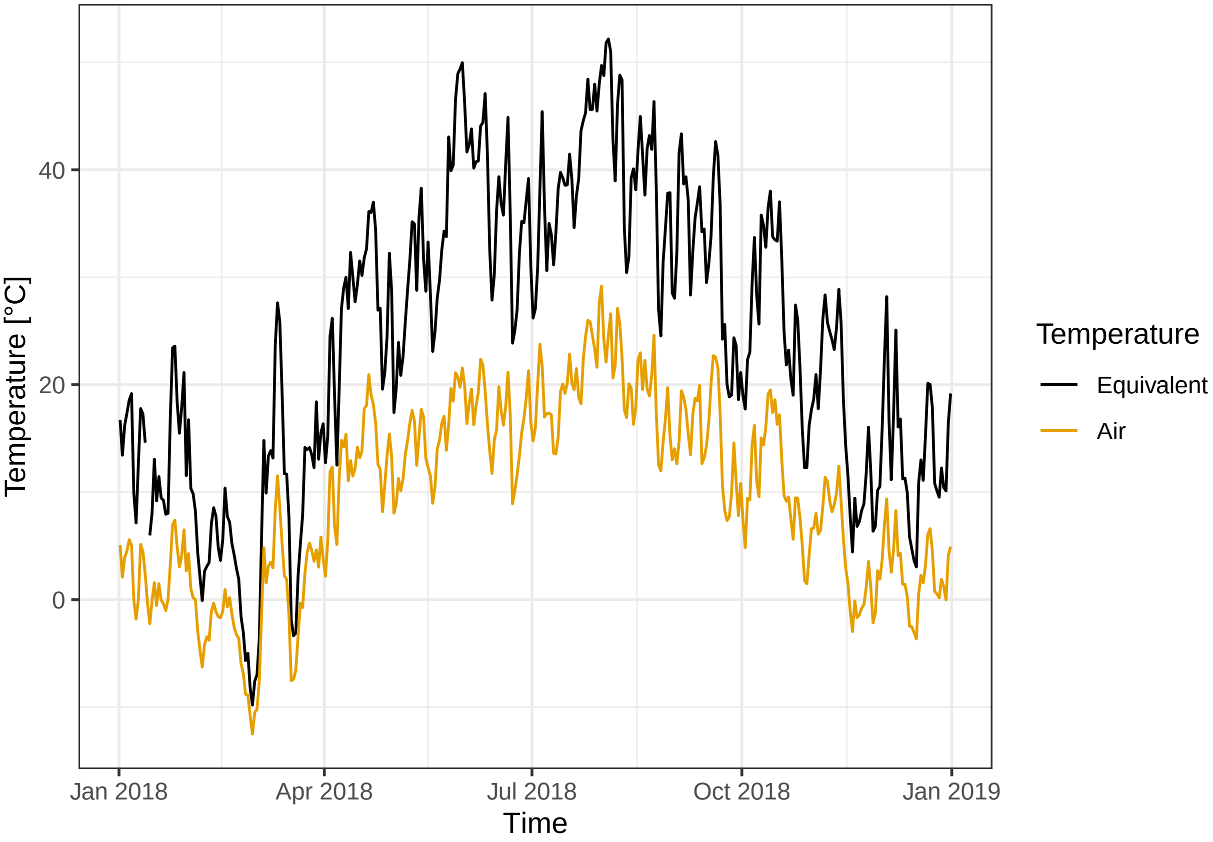

Figure 26: Equivalent temperature compared to air temperature. Data averaged over one day. Data from Hainich national park 2018.

4.4.7 Field humidity measurements

The humidity has been measured in the field using two different psychrometer, an assmann and an spinning one. The measures of the the instruments are compared in table 2. The two measures are comparable but show a significant difference (~20 % in relative humidity). This is incongruence is probably due to inaccuracies in the measurement and possibly different conditions between the measures. Moreover the thermometers of the spinning psychrometer are not protected from the direct sunlight which can compromise the reading, in fact the dry temperature is higher and the wet temperature is lower in the spinning psychrometer.

```
spin <- tibble(
  t_dry = 16,
  t_wet = 10.8,
  p = 977.2
)

spin <- spin %>%
  mutate(
    es_wet = get_es(t_wet),
    es = get_es(t_dry),
    ea = get_ea_dry(es_wet, t_dry, t_wet, p),
    td = get_td(ea),
    rh = get_rh(ea, es),
    abs_hum = get_abs_hum(ea, t_dry),
    spec_hum = get_spec_hum(ea, p),
    mix_ratio = get_mix_ratio(ea, p),
    p_def = get_p_def(ea, es),
    t_eq = get_t_eq(t_dry, mix_ratio)

  )
```

```
assman1 <- tibble(
  t_dry = 15.4,
  t_wet = 11.8,
  p = 977.2
) %>%
  mutate(
    es_wet = get_es(t_wet),
    es = get_es(t_dry),
    ea = get_ea_dry(es_wet, t_dry, t_wet, p),
    td = get_td(ea),
    rh = get_rh(ea, es),
    abs_hum = get_abs_hum(ea, t_dry),
    spec_hum = get_spec_hum(ea, p),
    mix_ratio = get_mix_ratio(ea, p),
    p_def = get_p_def(ea, es),
    t_eq = get_t_eq(t_dry, mix_ratio)

  )
```

```
spin %>%
  bind_rows(assman1) %>%
  t() %>%
  kableExtra::kbl(col.names = c("Spinning", "Assmann"), digits = 1, caption = "Comparioson of measuremen
  kableExtra::kable_styling(latex_options = "hold_position")
```

Table 2: Comparioson of measurement between spinning and assman psychrometer

	Spinning	Assmann
t_dry	16.0	15.4
t_wet	10.8	11.8
p	977.2	977.2
es_wet	13.0	13.9
es	18.2	17.5
ea	9.7	11.6
td	6.4	9.0
rh	53.2	66.2
abs_hum	7.3	8.7
spec_hum	6.2	7.4
mix_ratio	6.2	7.5
p_def	8.5	5.9
t_eq	31.6	34.1

5 Precipitation

5.1 Motivation

Precipitation is a very important process in climatology or weather meteorology. Precipitation in relation with water, occurs when atmospheric conditions become saturated enhancing 100% of relative humidity and thus, water condensates and later, it precipitates. Besides, precipitation is a key process part of the water cycle. Its function is to provide with water the natural ecosystem in the planet. Water is the first component needed by plants, to allow the processes of photosynthesis and transpiration.

5.2 Background

The precipitation is usually in form of liquid water, but can also be solid in case of snow or hail. In the previous section, we referred to water condensation. Although this is important part, first needs to be a cloud formation. This takes place by the presence of a warm air flow, thus its cooler surrounding will rise due to a lower density. Together with an increase in height, the air parcel expands and cools down with the dry adiabatic lapse rate. Then, condensation occurs when air temperature and dew point temperature. During the condensation latent heat is released.

It is measured as a volume of water per unit area (m^3/m^2). The standard unit is the millimeters (mm) of rain, which corresponds to a liter per squared meter. The precipitation intensity is measured in millimeter per hour (mm/h).

5.3 Sensors and measuring principle

The precipitation sensors, pluviometers, measure the amount of rainfall on a know area.

The most common ones uses a **tipping bucket** that when gets filled with water it switch position and empty itself. By counting the number of swings of the bucket it is possible to calculate the amount of precipitation. This sensors are the most common one, as they are simple and relatively reliable, however they can produce inaccurate measurements if the amount of precipitation is very low that the buckets doesn't fill or when there is an high rain intensity that the bucket cannot fill fast enough.

To overcome this limitation there are **weighting pluviometers**, which have a very sensitive scale under a bucket that collects all the precipitations.

A completely different type of sensors are the **laser beam** pluviometers, that can estimate the number, size and speed of water droplet and therefore produce an accurate measurement of rain.

Precipitation measurements present additional challenges, in presence of strong winds there is turbulence around the instrument that reduce the amount of rain collected. To mitigate this there are wind shield that can be installed around pluviometers. Moreover precipitation can be solid, hence heating may be required to properly measure it. Finally precipitation has an high spatial heterogeneity, which requires the use of several sensors.

5.4 Analysis

```
prec <- read_csv(here("Data_lectures/5_Precipitation/P_4sites.csv"))
et <- read_csv(here("Data_lectures/5_Precipitation/ET_4sites.csv"))
sites <- read_csv(here("5_precipitation/station_data.csv")) %>%
  rename(site=`Site-abb`, full_name = Site)
```

```
prec_avg <- prec %>%
  select(-Date) %>%
  gather("site", "prec") %>%
    group_by(site) %>%
  summarise(prec=mean(prec))

et_avg <- et %>%
  select(-Date) %>%
  gather("site", "et") %>%
    group_by(site) %>%
  summarise(et=mean(et))

sites_avg <- sites %>%
  inner_join(et_avg, by="site") %>%
  inner_join(prec_avg, by="site")
```

5.4.1 ET and precipitation at different latitudes

The latitude has an clear influence on precipitation, the closer to the equator the bigger the precipitation (Figure 27). This relation depends on the local conditions, for example is not true if deserts are included, but still show an important global pattern.

```
prec_t <- prec %>%
  gather("site", "prec", -Date)

et_t <- et %>%
  gather("site", "et", -Date)

prec_et <- left_join(prec_t, et_t, by=c("Date", "site"))

sites <- sites %>%
  inner_join(prec_et, by="site")
```

```
ggplot(sites, aes(Lat, prec, colour=site)) +
  geom_boxplot() +
   labs(y="Total yearly precipitation [mm]",
       x="Latitude [deg]") +
  scale_color_colorblind()
```

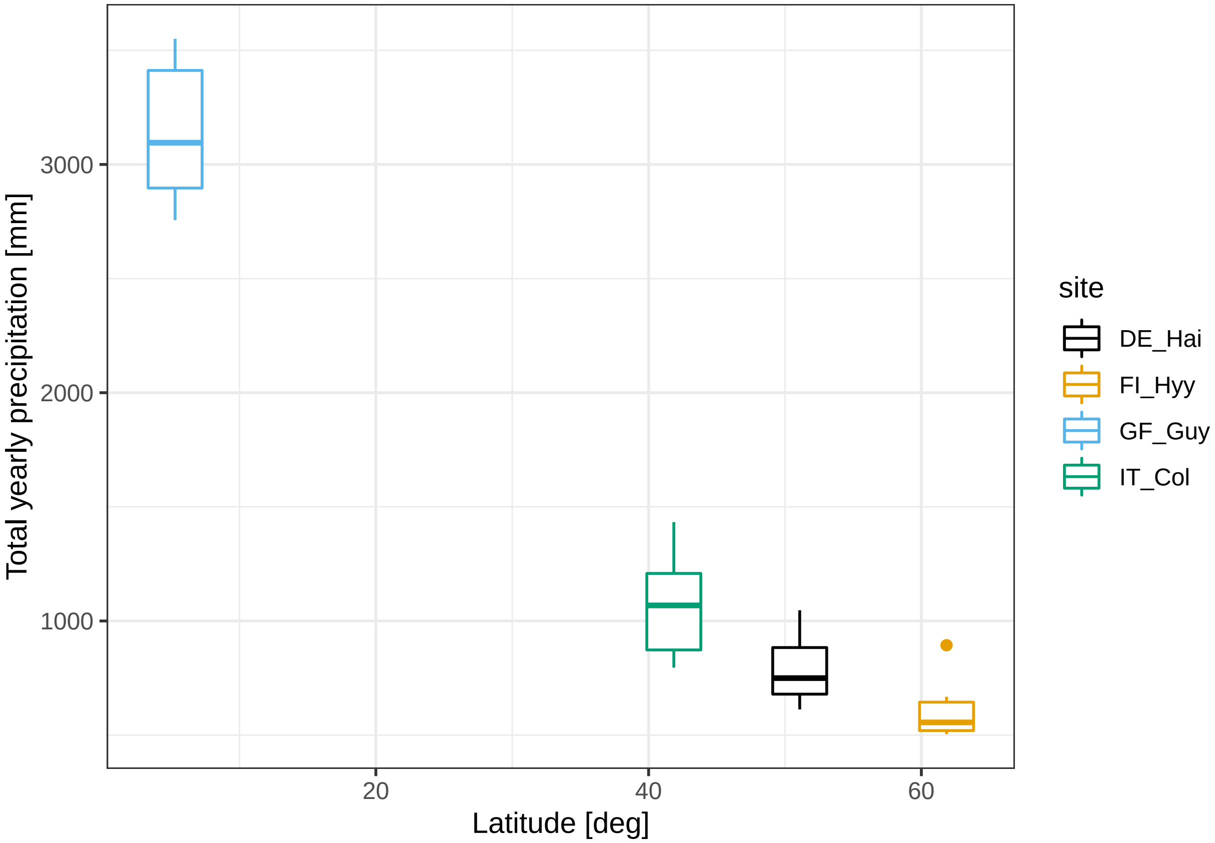

Figure 27: Cumulative yearly precipitation in relation to the latitide 2004-2009.

The evapotranspiration (ET) also shows a similar pattern (Figure 28). In the tropical forest the ET is much higher than the rest. The latitude pattern is not completely followed by the site in Italy, is likely due to the summer drought in the mediterran ecoesystem that limits the total ET as there is a lack of water.

```
(et_box <- ggplot(sites, aes(Lat, et, colour=site)) +
  geom_boxplot() +
    labs(y="Total evapotranspiration [mm]",
        x="Latitude [deg]")) +
  scale_color_colorblind()
```

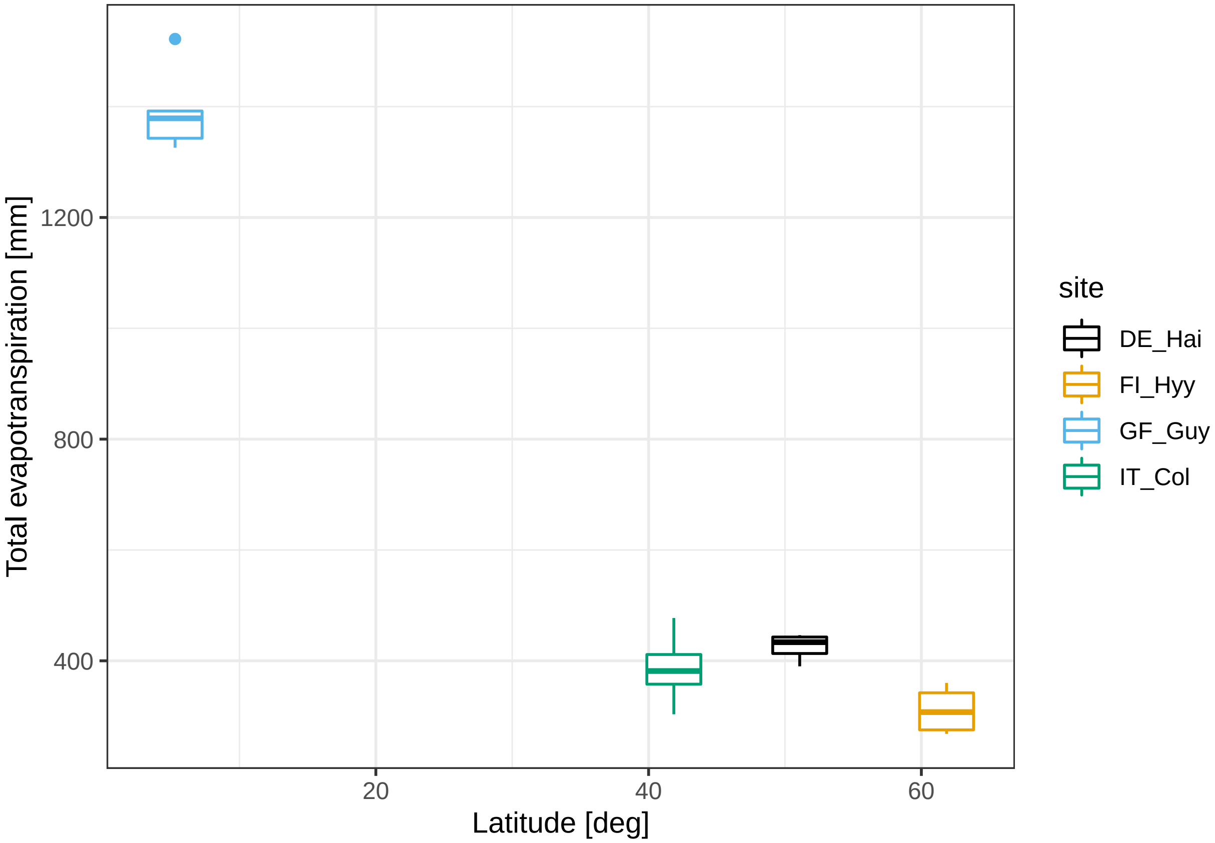

Figure 28: Cumlative evapotranspiration at different latitudes. Data from Fluxnet database 2004-2009.

5.4.2 Evapotranspiration index

The ratio between evapotranspiration and precipitation (evapotranspiration index) is a very useful indicator how precipitation is used by plants during photosynthesis. Evapotranspiration index is the fraction between precipitation and what is coming back to atmosphere through evapotranspiration.

The evapotranspiration index shows have a more interesting pattern (Figure 29). The sites in Germany and and Finland have similar values of ET_idx and and overall good efficiency in using water. For the site in French Guyana that ET_idx is lower and this probably due to the fact that the ecosystem is energy limited, the amount of radiation coming from the sun is not enough to evaporate all the available water.

The lowest ET_idx is Italy and its value is probably connected to the water stress of plants, that close stomata and reduce the transpiration even if there may be some water available. Moreover the precipitations are concentrated in the winter when the potential ET is lower and the plant cannot effectively use all the water.

```
sites$et_idx <- sites$et/sites$prec

sites %>%
ggplot(aes(site, et_idx, colour = site))+
         geom_boxplot()+
         labs( x="Site", y="Evapotranspiration index [mm/mm]") +
  scale_color_colorblind()
```

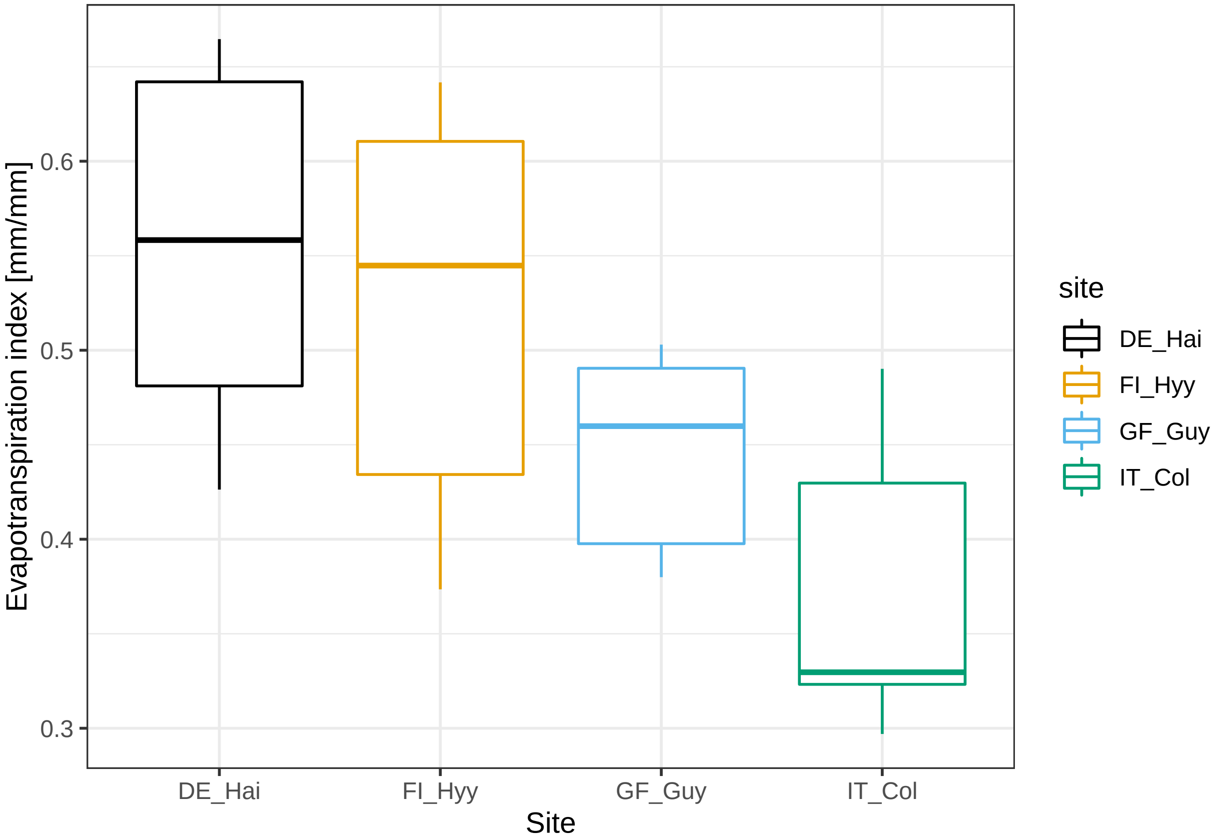

Figure 29: Evapotranspiration index for different sites. Data from Fluxnet database 2004-2009.

6 Air Pressure

6.1 Motivation

Air pressure is constantly regulating the characteristics of the atmosphere and impacting the ecosystems. Winds and cloud formation all depends on the differences in air pressure.

The air pressure depends on the weight of the air column above and on the air temperature. In fact the air molecules move faster when air is hot and this increases the pressure (Bonan 2019).

6.2 Background

There are different processes that affects air pressure. The main one is the reduction of air pressure with the increase of elevation. The pressure depends on the weight of the air column above, at a high elevation there is less air above and thus, the pressure is lower.

This dependency of pressure with elevation is commonly used in altimeters to estimate the elevation. However, in case of weather measurements the height effects needs to be removed.

The air pressure has an exponential decay with height. The following formula can be used to estimate the pressure using the measured pressure at a know elevation.

$$p(0) = p(z) exp(\frac{g \Delta z}{R_d T})$$

The air temperature is an important component of the formula, as warm air is less dense and therefore the air column weights less. However, air temperature also depends on altitude, hence, the mean temperature over the column is used considering an estimate decrease in temperature of $-0.65K/100m$. The following equation can be used

$$T = T_{station} + 0.00325 * z$$

Another correction can be made for air humidity, as wet air is less dense than dry air. This is done by estimating the virtual temperature, which corresponds to the temperature where dry air would have the same density of the wet air. The virtual temperature is always higher than the real one. The following equation can be used for the correction

$$T_v = T(1 + 0.608q)$$

where q is the specific humidity in (Kg/Kg)

The inverse of the previous formula can be used to calculate the difference of height between 2 points with known pressures.

$$\Delta z = -\frac{R_d}{gT} log(\frac{p_z}{p_0})$$

One peculiarity of air pressure is the wide range of units used around the globe to measure it. The SI defines the Pa (N/m^2) as the unit for pressure. However, this is a small value so hPa (100 Pa) is commonly used as a reference amount of this unit. Another unit that commonly used in barometers is the torr or mmHg, that originates from the millimeters of mercury used in the first barometers. Those are the values to change between units: $760mmHg = 760torr = 1013.25hPa$.

Pressure influences the boiling point temperature of water. The following equation can be used to estimate the relationship:

$$T_{boil} = 100 + 2.804 \times 10^{-2}(p - 1013.25hPa) - 1.384 \times 10^{-5}(p - 1013.25hPa)^2$$

6.3 Sensors and measuring principle

There are several sensors to measure the air pressure and each of these use different measurement principles.

- **Mercury barometer**. This is the oldest barometer and works by having a column of mercury in a tube with vacuum on one side and air in the other. On the mercury there is the gravitational force that make it going down, while the air pressure pushes the column up. This two forces reaches and equilibrium and therefore it is possible to read the pressure using the height of the mercury column. This sensor is not commonly used nowadays anymore. First of all, because mercury is dangerous and then, it also requires error corrections for both: temperature (mercury expands with higher temperatures) and gravity acceleration constant, which changes depending on altitude and latitude.
- **Aneroid barometer**. They have an aneroid capsule with vacuum (or low pressure) inside, air pressure tends to reduce the collapse the capsule while a spring keeps in open. By measuring the width of the capsule is possible to estimate the air pressure. The width of the capsule can be measured both in analog instruments or digital one, using a capacitor. Those are the most widely used pressure sensors as they are compact, reliable and require no error correction.
- **Boiling barometer**. First, it measures the boiling temperature of water, and then, uses this information to estimate the air pressure. There is a heater to make water boil and then an accurate thermometer measure the temperature of the water vapour. The main disadvantage is their reduced convenience due to the procedure to boil water at each sample, but they can have a high accuracy, up to 0.5 hPa (Richner, Joss, and Ruppert 1996).

6.4 Analysis

```
pres <- read_csv(here("Data_lectures/6_Air_pressure/TA_RH_PA_Leinefelde.csv"))
```

```
#utility funcs from air humidity notebook

# temp is in degrees celcius
get_es <-  function(ta)  6.1078 * exp((17.08085 * ta) / (234.175 + ta))

rh2ea <- function(rh, e_s) rh/100 * e_s
get_spec_hum <- function(e_a, p) 0.622 * e_a / (p - 0.378 * e_a) # note Kg Kg-1

c2k <- function(c) c + 273.15
k2c <- function(k) k - 273.15

# get virtual temperature. Ta is the air temp and q the specific humidity
get_tv <- function(ta, q) ta * (1 + 0.608 * q)

Rd <- 287.05 # J Kg-1 K-1 gas constant of dry air
get_press_sea_level <- function(pz, tv, Dz, g = 9.81) {
  pz * exp((g * Dz)/(Rd * c2k(tv)))
}
```

6.4.1 Air pressure sea level

The air pressure at sea level is always higher than at Leinefelde (Figure 30) The difference is around 53 hPa, and it is quite constant during the year. In the plot it is also possible to see that there are no clear seasonal patterns during the year as the pressure oscillate roughly +/- 25 hPa around the mean. Moreover, it is also interesting to notice the stability of the air pressure in a short time frame, as the data plotted has a 30 min frequency but no high frequency patterns can be observed.

```
heigth_diff <- 451 + 44 # elevation + tower height

pres <- pres %>%
  mutate(
    es = get_es(TA_degC),
    ea = rh2ea(RH_Perc, TA_degC),
    q = get_spec_hum(ea, PA_hPa),
    tv = get_tv(TA_degC, q),
    p0 = get_press_sea_level(PA_hPa, tv, heigth_diff)
  )
```

```
pres %>%
        gather("location", "pressure", p0, PA_hPa, factor_key = T) %>%
ggplot(aes(Date, pressure, color=location)) +
  geom_line() +
labs(y="Pressure (hPa)") +
scale_colour_colorblind(name="Location", labels = c("Sea level", "Leinefelde"))
```

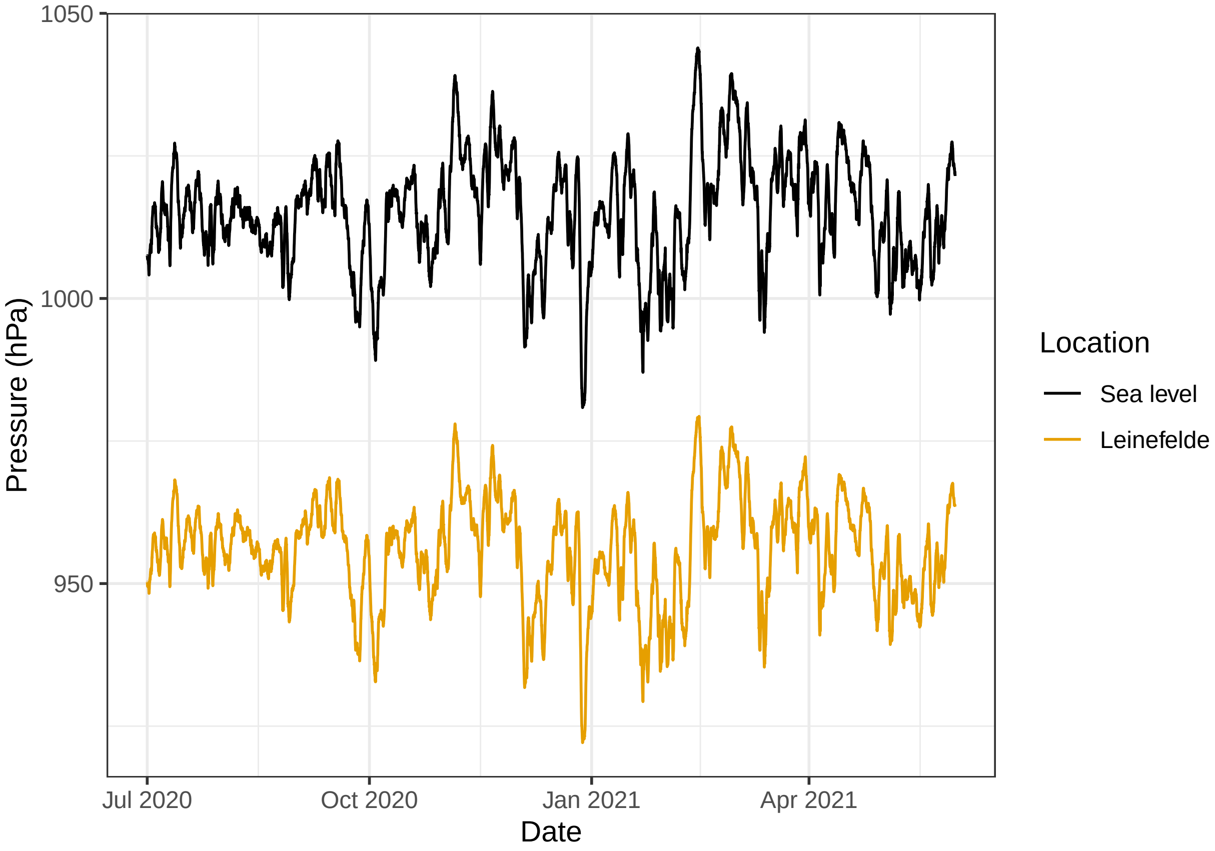

Figure 30: Air pressure at Leinefelde and estimated air pressure at sea level. Pressure has been corrected for air humidity. Data from Leinefelde flux tower (451 m + 41m tower) July 2020 - May 2021, 30 min frequency.

6.4.2 Air pressure Brocken and water boling temperature

Calculate the air pressure at top of the Brocken mountain (1141 m) for a pressure of 991.3 hPa and an air temperature of 15°C at the North campus (185 m). Assume a mean air temperature decrease of 0.5 K/100 m. At which temperature would water boil at the North Campus and on the Brocken?

```r
pa_nc <- 991.3  # Air pressure north campus
Rd <- 287.05 # J Kg-1 K-1 gas constant of dry air
g <- 9.81# m/s^2
z1 <- 1141 #m Brocken mountain Height
z0 <- 185 #m North campus Height
Z <- z0-z1
ta <- 15 # C measuered at the north campus
t_column <- ta + 0.00325 * Z #correction of temperature for elevation
#hPa would be the pressure at the Brocken mountain
pa_brocken<-  (991.3 * exp((g * Z)/(Rd * c2k(t_column)))) %>% round(2)
```

On the top of the Brocken the air pressure would be 883.95 hPa

```r
t_boil_nc <- (100 + 2.804e-2 * (pa_nc-1013.25) - 1.384e-5 *(pa_nc-1013.25)^2 ) %>%
                round(2)
t_boil_brocken <- (100 + 2.804e-2 * (pa_brocken-1013.25) - 1.384e-5 *
                     (pa_brocken-1013.25)^2 ) %>% round(2)
```

The water would boil at 99.38 °C at the North Campus and 96.14 °C on the Brocken. As expected, the boiling temperature is lower at a higher elevation.

The pressure value of the Brocken mountain was calculated using the formula mentioned in the theory background, adding correction for temperature decrease.

$$p(0) = p(z)exp(\frac{g\Delta z}{R_dT})$$

7 Wind

7.1 Motivation

Wind is a very important factor of Earth climate. It is the main phenomenon that transport matter and energy between different location on the globe and layers of the atmosphere. Wind properties, like speed and direction, can vary a lot. For example, it can be ranged from a smooth breeze to a storm or a hurricane. In case of hurricanes, it is normally a devastating disturbance that can reach a very destructive power on earth ecosystems. Moreover wind is also the main way many plant disperse seeds and pollen. Animals as well use wind for movement. Finally winds also shapes the earth surface trough erosion of rocks and sediments movement.

7.2 Background

Wind can be represented as a vector in 3 dimensions, two on the horizontal plan (u and v) and one vertical (w). However, the vertical wind component is small and its means is by definition 0 so it is often not included in wind measurements. In this scenario the wind is commonly expressed in angular format, with the total intensity on the horizontal plan and the direction.

The fact that the wind is a 3d vector and has this two formats leads to complexities in handling wind data. In particular the direction cannot be averaged directly, as it's a circular variable, but can needs to be transformed to the vector components, averaged and then transformed back to an angle.

The average of the wind speed can lead to problems as there two different, but both correct ways to do it (Grange 2014). The first one is doing the averages of the vector components, the second one is doing the average of the absolute values of the wind speed. The first method will always result in smaller values, as wind from opposite direction can be averaged to zero. In the following protocol the average of the absolute values will be used.

Lastly, the are several conflicting ways to define the wind direction, when using vectors the direction is the where the wind is going, but in weather forecast the wind direction is where the wind is coming from. Finally also the definition of the orientation of the u and v components can change between different instruments and software.

If there is a neutrally stratified atmosphere, which means there are no important turbulent fluxes, the wind speed above the canopy can be modeled. Due to the friction with the surface the closest the wind is to the canopy the lower the speed, reaching zero at the boundary. The wind profile can be estimated with the following formula:

$$u(z) = \frac{u_*}{k} ln(\frac{z-d}{z_0})$$

where:

- z in (m) is the height
- $u(z)$ in (m/s) is the speed of the wind at height z
- u_* in (m/s) is the friction velocity. This is independent from the height and indicates the mechanical turbulence. It can be calculated using this equation $\sqrt{\frac{1}{\rho}u'w'}$
- k is the Von Karman constant (0.4)
- z_0 in (m) is the height where the wind speed is theoretically zero. It can be estimated as 0.1 the canopy height.
- d in (m) is the displacement height, which accounts for the shift of the wind profile to the presence of a canopy. It can be estimated as 2/3 of the canopy height.

7.3 Sensors and measuring principle

There are many types of instruments used for measuring wind speed. Here some of them will be described.

Cup Anemometer : it consists of a set of three cups, crossing a vertical basement stick. This cross shape allows to measure the horizontal wind velocity at a specific height. The wind speed is derived from number of cycles/time or turning velocity. For measuring the wind direction, a wind vane is used. It points to the direction where the wind is coming. This is through a potentiometer to detect the right direction.

Propeller anemometer : the way this this instrument works is very similar to the cup anemometer. It points to the mean wind direction at that moment. With the use of three propeller anemometer pointing different direction, three dimensional wind can be measured.

Ultrasonic anemometer thermometer : This uses the speed of sound to measure the wind. Normally, it will be displayed in three directions to measured all directions and get a more accurate measurement value. One of the advantages of using the ultrasonic anemometer is the small fluctuations detected on the measures. The speed of sounds depends on temperature and air humidity. Thus, the following equations allow the calculation of speed of sound and the temperature at high frequencies;

$$C_l = \frac{D}{2}(\frac{1}{time_A - A} + \frac{1}{time_B - A})$$

$$C_l = \sqrt{K_a * R_a * T_{av}}$$

Where:

- $K_a = 1.4$
- $R_a = 287.05 J/Kg * K$
- $T_{av} = T(1 + 0.513 * q)$

Hot wire anemometer : When a current flow is introduce within a wire, there is a release of heat. Then, the air flow goes throguh the wire and cools down removing the released energy. It can be apply in two different ways;

At a constant current, the change of temperature is measure with a thin thermocouple. This can be hard at a high speed wind.

At a constant temperature, with a temperature change the current is regulated, such that the temperature is held constant and thus, with a high wind there will be a high current as well.

Each type of anemometer has its more limitations. For example the starting speed of cup and propeller anemometer is that it starts to rotate when speed is 0.5 m/s. When wind flow stops, but the cup anemometer keeps rotating a bit longer until it fully stops. In case of low wind speed, sonic anemometer are the best instruments to use but in case or rain, it cannot do measurements instead.

During installation the anemometers there are some tips to take into account. Better to set them far above ground, this way the roughness of the lower layer above soil's surface will not be affecting the measures. The same with any other object around in the area. In case of sonic anemometer, is important to protect it against birds or any type of insect that make small variation when measuring.

7.4 Analysis

```
library(tidyverse)
library(lubridate)
library(clifro) # for windrose
library(patchwork)
library(ggthemes)
theme_set(theme_bw()) # ggplot theme
```

```
wind <- read_csv(here::here("Data_lectures/7_Wind/Winddata_Botanical_garden.csv")) %>%
  drop_na() %>%
  rename(WS_0.5m = WS_05m, wd=WD_deg)
```

```
deg2rad <- function(deg) deg * pi / 180
rad2deg <- function(rad) rad * 180 / pi

# calculates the wind angular average over the provide input.
# intend to be used together with group_by and summarize
wind_dir_average <- function(wd){
  dir <- deg2rad(wd)
  # calc the vector components and then make the mean
  u <- cos(dir) %>% mean
  v <- sin(dir) %>% mean
  # convert back to a direction. Note atan2 uses y,x
  avg_dir <- atan2(v, u)
  # need to convert in 0 - 360 range
  avg_dir <- avg_dir %% (2*pi)
  return(rad2deg(avg_dir))
}
```

```
# wind gathered
wind_g <- wind %>%
  gather("height", "windspeed", WS_0.5m, WS_1m, WS_2m, WS_5m, WS_10m) %>%
  # converts the height into a numeric value
  mutate(height = as.numeric(gsub(".*?([0-9]+).*", "\\1", height)))

wind_1d<- wind %>%
  mutate(Date = floor_date(Date, unit = "1 day")) %>%
  group_by(Date) %>%
  summarise(across(c(-wd), mean), wd = wind_dir_average(wd))

wind_g_1d <- wind_1d %>%
  gather("height", "windspeed", WS_0.5m, WS_1m, WS_2m, WS_5m, WS_10m) %>%
  # converts the height into a numeric value
  mutate(height = as.numeric(gsub(".*?([0-9\\.]+).*", "\\1", height)))
```

7.4.1 Wind averages

The wind speed and direction have been averaged at 1 hour. Vectorial average has been used for wind direction. In figure 31 the average direction is compared with the original data. Between the 15th and the 16th of January there are some data points with a wind direction close to 0 °N, but the average is around 350 °N.

```
wind_1h <- wind %>%
  group_by(round_date(Date, unit = "1 hour")) %>%
  summarise(across(-wd, mean), wd = wind_dir_average(wd))
```

```
wind %>%
  filter(between(Date, as_datetime("2021-01-15") , as_datetime("2021-01-17"))) %>%
ggplot()+
  geom_point(aes(Date, wd, colour="10 mins"), size=.8) +
  geom_line(aes(Date, wd, colour="1 hour"),
            data=filter(wind_1h, between(Date, as_datetime("2021-01-15"),
                                          as_datetime("2021-01-17")))) +
  labs(y="Wind direction [°N]", colour="Frequency") +
  scale_y_continuous(breaks = c(0, 90, 180, 270, 360),
                     labels = c('N (0°)', 'E (90°)', 'S(180°)',
                                'W(270°)', 'N (360°)'), limits = c(-10, 370)) +
  scale_color_colorblind()
```

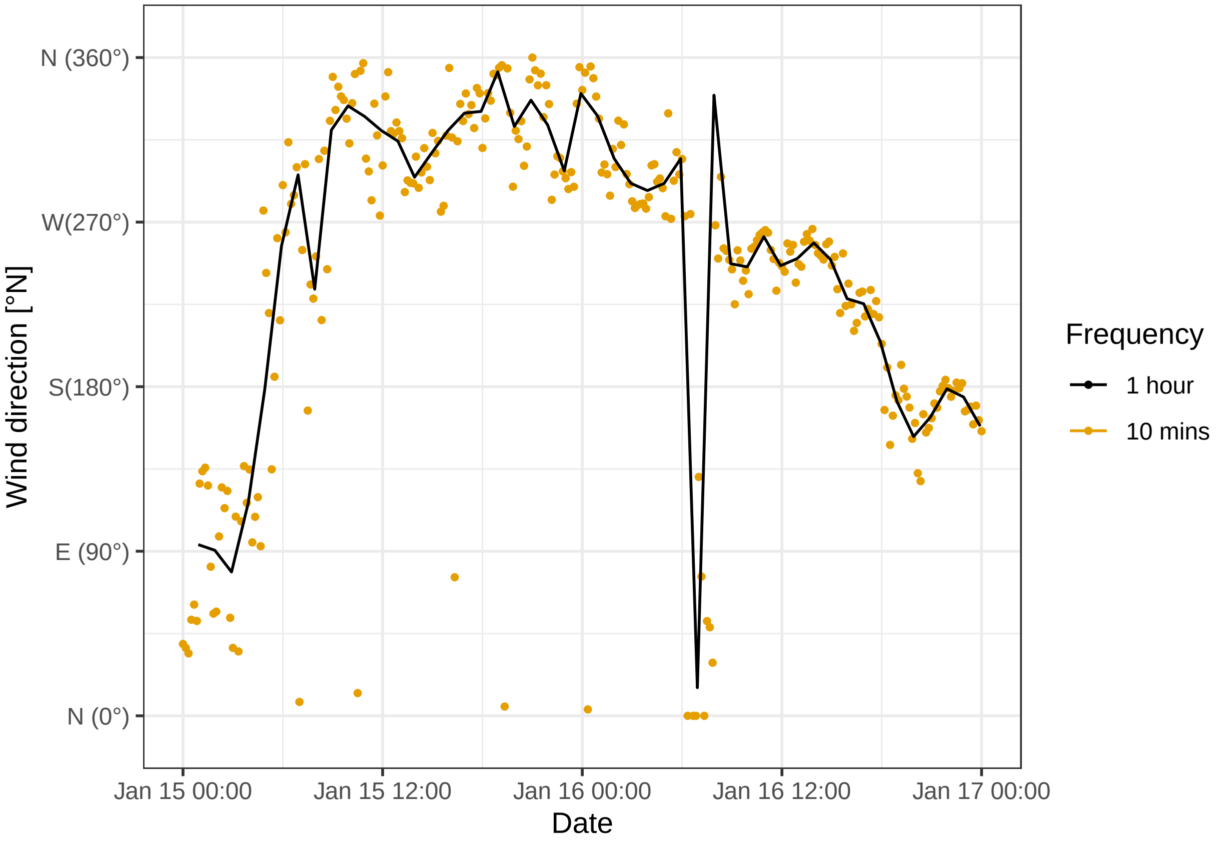

Figure 31: Comparison of wind direction original data (10 mins) and hourly average. Data from botanical garden 15th-17th January 2021.

7.4.2 Wind speed and height

As it is possible to appreciate in Figure 32 , wind speed gets faster at 10m height. This makes sense when having in mind the vertical wind profile graph that increases with height. The pattern is also consistent across different seasons.

Wind speed depends on height, increasing with height following a logarithmic profile (Figure 33). The plot was made using daily averages instead of high frequency data, to reduce the variation in the dataset.

```
(wind_g_1d %>%
  #just one month otherwise the plot is too compressed
  filter(between(Date, as_datetime("2020-01-15") , as_datetime("2020-02-15") )) %>%
  mutate(height = fct_reorder(as_factor(height), sort(height, decreasing = T))) %>%
  ggplot(aes(Date, windspeed, col=height))+
  geom_line() +
  scale_color_colorblind() +
  labs(y="Windspeed [m/s]", colour="Height [m]", title="(a) Winter month")) /
(wind_g_1d %>%
  #just one month otherwise the plot is too compressed
  filter(between(Date, as_datetime("2020-06-15") , as_datetime("2020-07-15") )) %>%
  mutate(height = fct_reorder(as_factor(height), sort(height, decreasing = T))) %>%
  ggplot(aes(Date, windspeed, col=height))+
  geom_line() +
  scale_color_colorblind() +
  labs(y="Windspeed [m/s]", colour="Height [m]", title="(b) Summer month")) +
plot_layout(guide="collect")
```

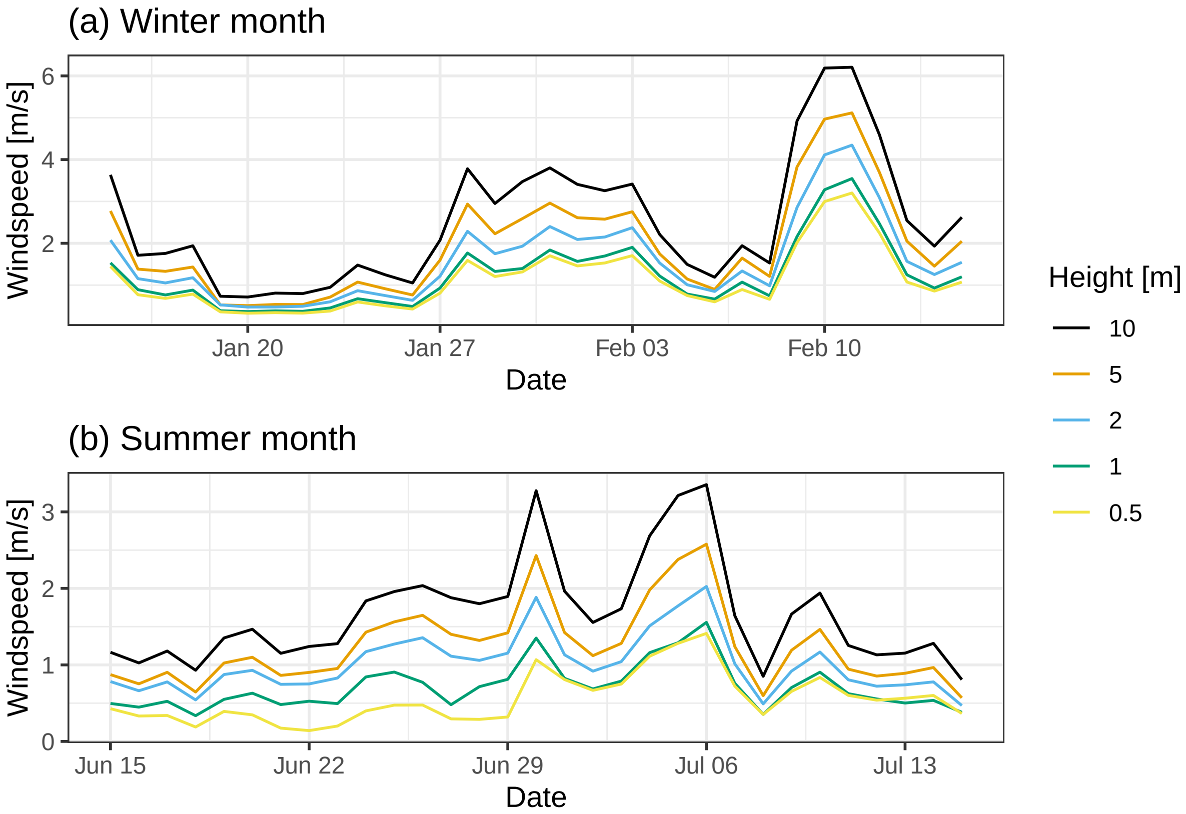

Figure 32: Time series of wind speed at different heigth. (a) is a summer month (15th Jan 2020 - 15th Feb 2020). (b) is a winter month(15th Jun 2020 - 15th Jul 2020). Data from botanical garden.

```
wind_prof <- wind_g_1d %>%
  group_by(height) %>%
  summarize(windspeed=mean(windspeed))
#### fit logarithmic wind profile to data and estimate the parameters u*, z0 and d
# initial values
u_star_start <- 0.1
d_start <- 0.3
z0_start <- 0.05

log_prof_model <- nls(windspeed ~u_star/0.4*(log((height - d)) - log(z0)),
                 start = list(u_star=u_star_start,
                              d=d_start,
                              z0=z0_start),
                 na.action = na.exclude, data=wind_prof)
wind_prof <- mutate( wind_prof,
  pred_ws = predict(log_prof_model))
```

```
wind_g_1d %>%
  ggplot() +
  geom_boxplot(aes(windspeed, height, group=height)) +
  geom_line(aes(x=pred_ws, y=height, colour="Estimated\nlog profile"),
            data = wind_prof, col=colorblind_pal()(2)[2] ) +
  geom_point(aes(x=pred_ws, y=height, colour="Estimated\nlog profile"),
             data = wind_prof, col=colorblind_pal()(2)[2] ) +
  labs(x="Windspeed [m/s]", y="Height [m]", colour="")
```

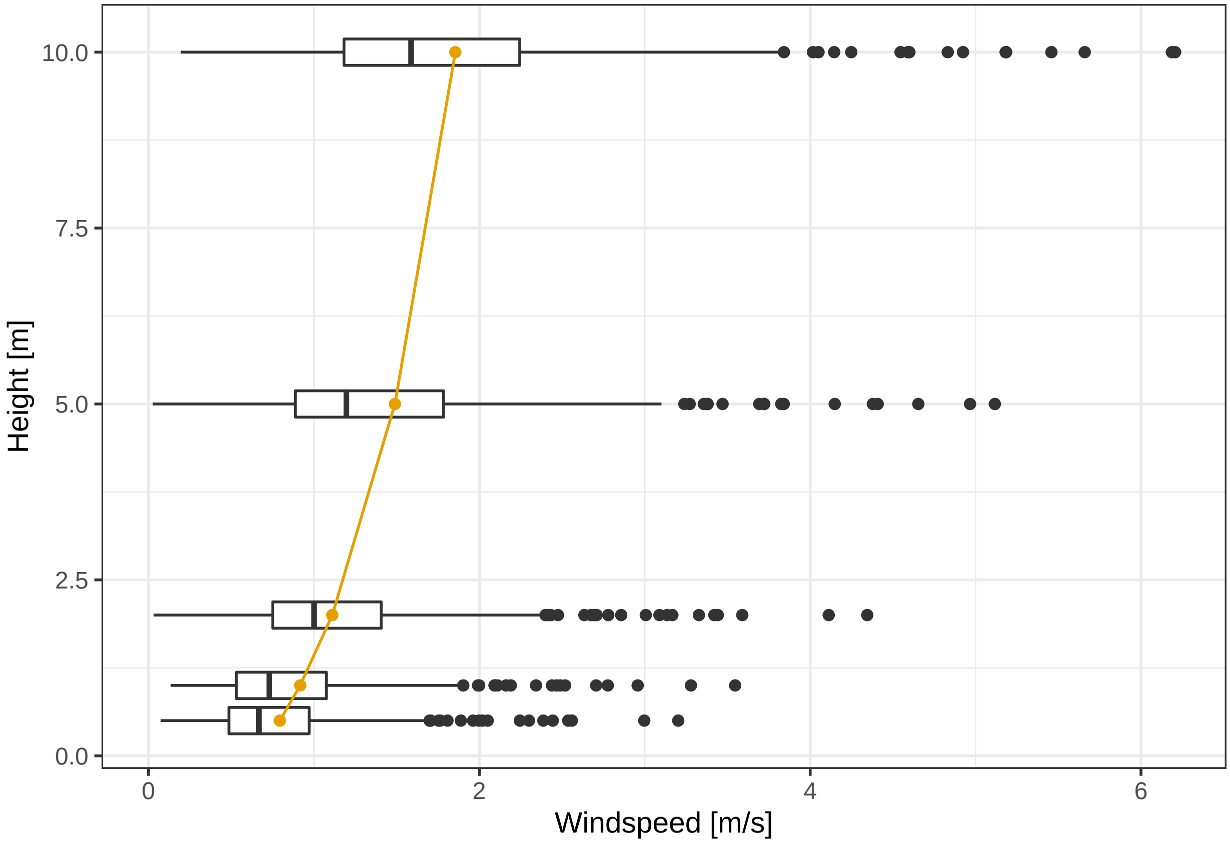

Figure 33: Distribution of daily means of wind speed at the different heights. The yellow line was obtaining by fitting a wind log profile to the mean of daily means. Data from botanical garden Jan 2020 - Feb 2021.

7.4.3 Windspeed over year

The wind rose for different quarters of the year are plotted (Figure 34) and then yearly pattern of wind speed and wind direction analyzed separately.

During the spring and winter the wind is stronger. In summer daily average oscillates around 1.5 m/s (Figure 35).

The graphs displays the variation of the wind speed along year. It is faster from end of December to beginning of April. During summer the mean wind speed is lower but some days it gets faster than others. This variability is originated depending on when the wind is coming from.

The wind direction is usually coming from overall the south (Figure 36) for the majority of th year, with the exception of the early summer where the north direction is also common. The reason why the prevalent winds are from the south is probably connected with the direction of the slope of the hill where the station is located, which is in the south direction.

```
wind_q <- wind %>%
  mutate(quarter = quarter(Date),
         quarter = case_when(
           quarter == 1 ~ "Jan-Mar",
           quarter == 2 ~ "Apr-Jun",
           quarter == 3 ~ "Jul-Sep",
           quarter == 4 ~ "Oct-Dec",
         ))
windrose(wind_q$WS_10m, wind_q$wd, wind_q$quarter, n_col= 2, col_pal="YlGnBu",
         ggtheme = "bw")
```

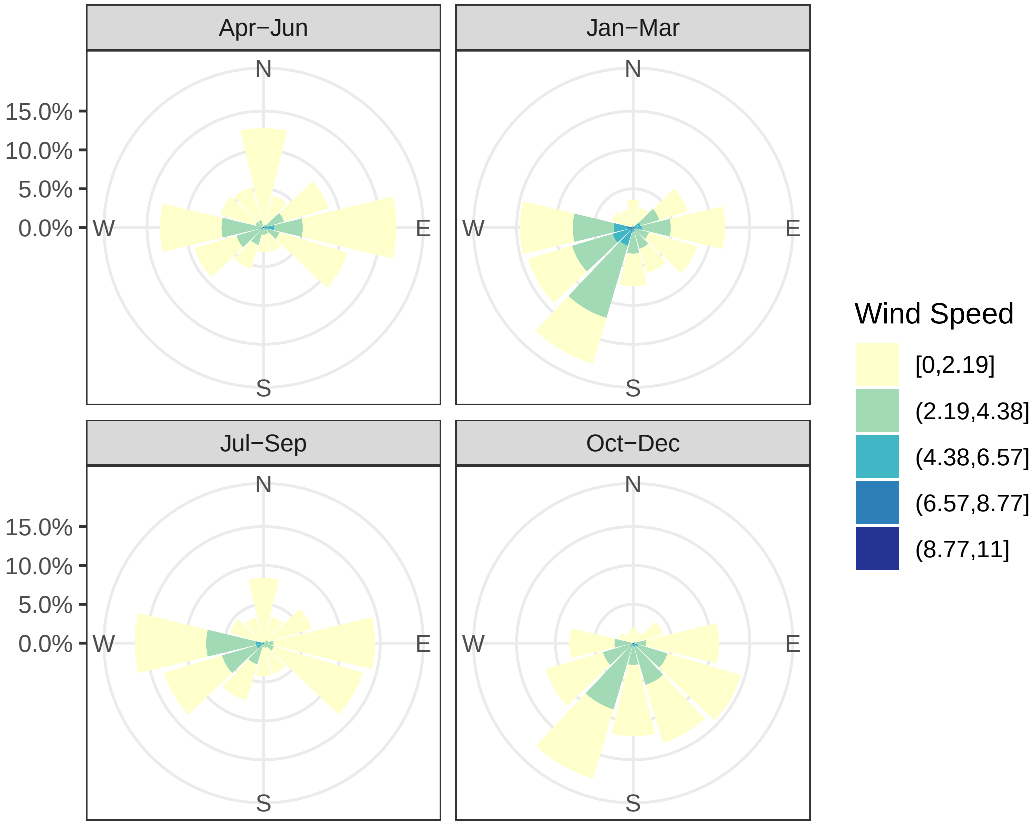

Figure 34: Wind rose for different quarters of the year. Data from botanical garden Jan 2020 - Feb 2021.

```r
ggplot(wind_1d, aes(Date, WS_10m)) +
  geom_line() +
  labs(y="Wind speed (m/s)")
```

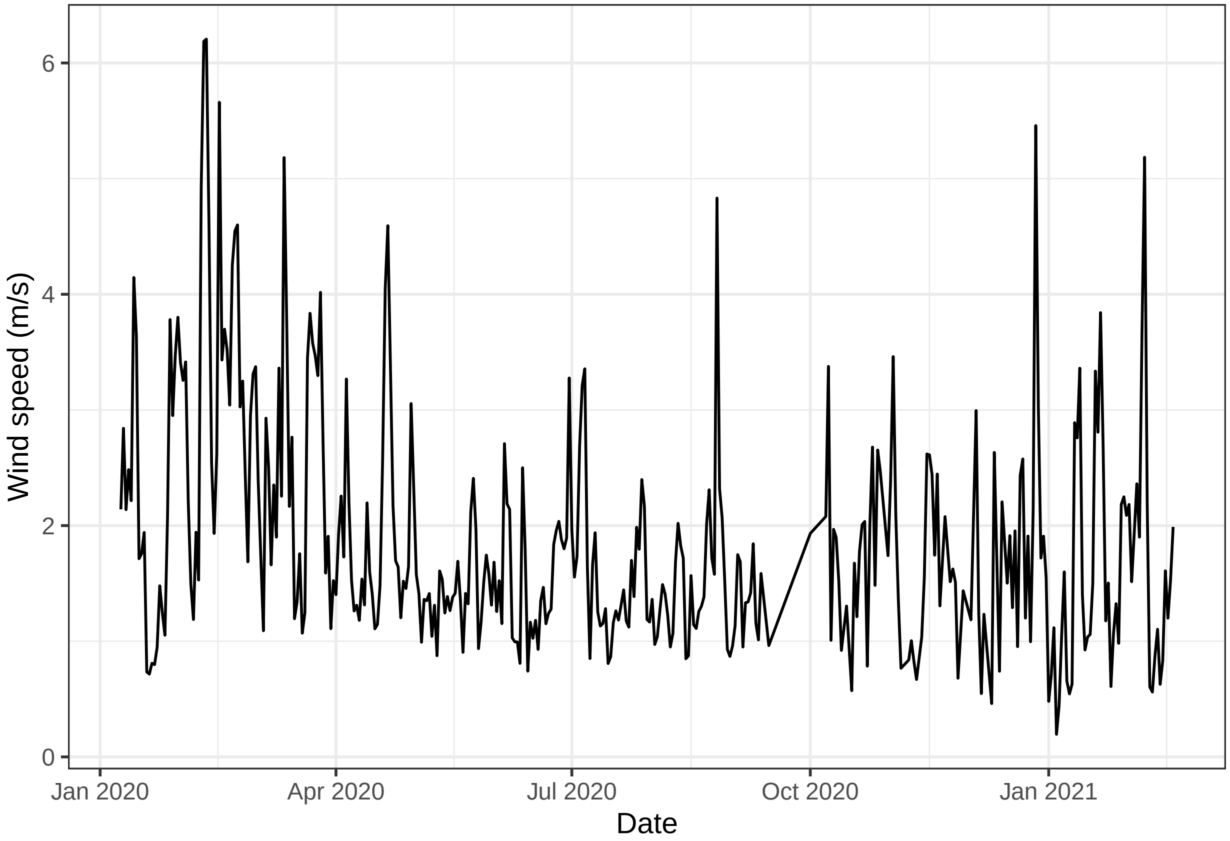

Figure 35: Daily averages of wind speed at 10 meters. Data from forest botanical garden January 2020 - February 2021.

```r
# data frame with months start and end to draw background
months <- map_df(1:14, function(n_mon){
  start <- as_datetime("2020-01-01")
  # offset to the correct month start
  month(start) <- month(start) + n_mon - 1
  end <- start
  # adding one month to get to the end and removing one day
  month(end) <- month(end) + 1
  day(end) <- day(end) - 1
  tibble(start = start, end = end,
         month= month(start, label = T), quarter= quarter(start))
} )
```

```r
wind %>%
  group_by(round_date(Date, unit = "1 weeks")) %>%
  summarise(across(c(matches("WS"), Date), mean), wd = wind_dir_average(wd)) %>%
  ggplot() +
  geom_rect( #add months in the background to be able to read the figure
    aes(xmin = start, xmax = end, fill = month),
    ymin = -Inf, ymax = Inf, alpha = 0.6,
    data = months
  ) +
  scale_fill_brewer(palette = "Set3") +
  geom_point(aes(Date, wd)) +
  coord_polar(theta="y") +
  labs(y="Wind direction", fill="Month") +
    scale_y_continuous(breaks = c(90, 180, 270, 360),
                       labels = c('E', 'S', 'W', 'N' ), limits=c(0, 360))
```

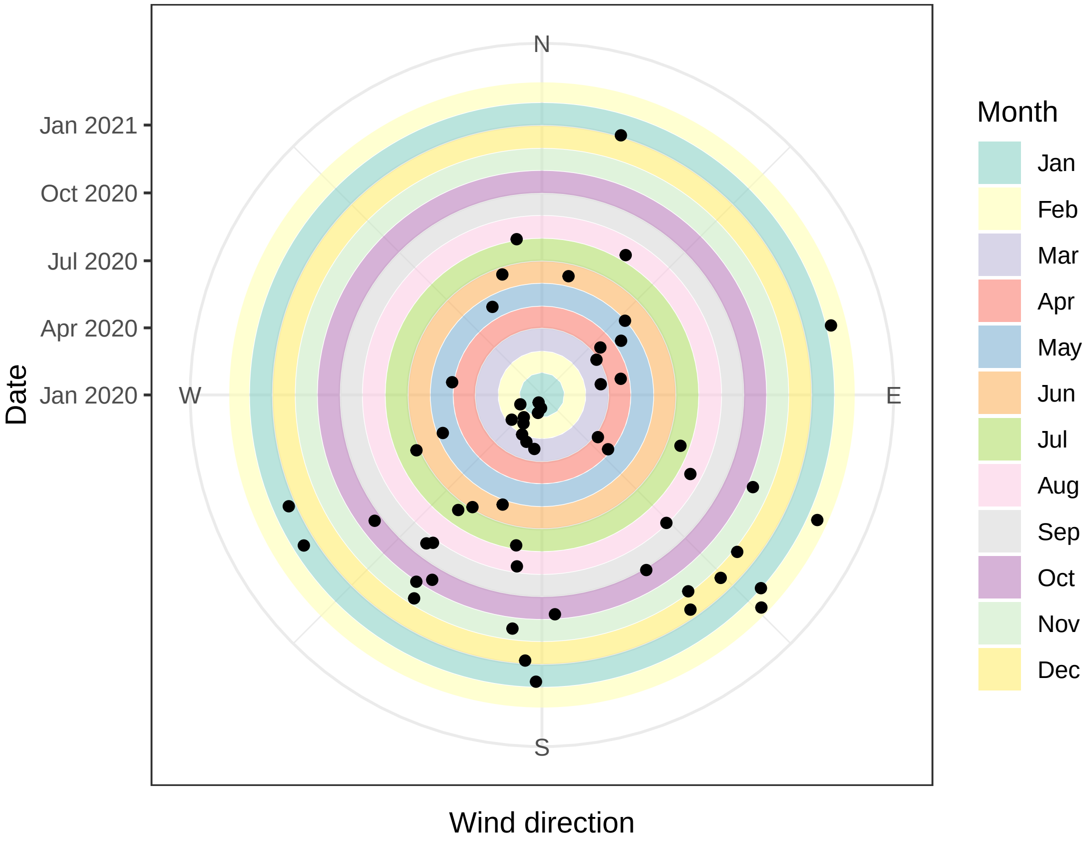

Figure 36: Weekly average of wind directions for the year (black dots). The distance from the center and the different background indicates the date, while the position in the circle the wind direction. Data from forest botanical garden January 2020 - February 2021.

8 Soil physics

8.1 Motivation

Soil is the physical foundation of terrestrial ecosystems (Lal and Shukla 2004). The soil is composed by different layers and is the interface between the lithosphere and the biosphere. Several processes fundamental like decomposition and nutrient transport take place into the soil, hence studying soil physical, chemical and biological properties is crucial to understand ecosystems.

Soil temperature and water content have a direct impact on two key process in the ecosystem: photosynthesis and respiration, which are basis of carbon dynamics. The former takes place in the canopy, but it requires the transpiration of water that comes from the soil. Moreover the soil temperature influences the leaves energy balance. The majority of ecosystem respiration takes places in the soil (Yuste et al. 2005) and the soil temperature and humidity are the main variables that control the soil respiration. The respiration increases exponentially with temperature (Lloyd and Taylor 1994), however at high temperature it is often limited by water availability (Orchard and Cook 1983).

8.2 Background

Soil temperature and soil water content are both factors that alters soil respiration rate. Soil respiration is a process measured in $\frac{\mu mol}{m^2 s}$ of CO_2 (Courtois et al. 2018). Soil is an important environmental variable partly in charge of an efficient ecosystem activity. Soil moisture drives changes in stomata conductance, photosynthesis rate and energy partitioning. Water potential is the main reason of the movement of water through osmosis, gravity and air pressure.

Soil water potential is constitute by its matrix potential (adhesive force water-soil particles), osmotic potential (gradient of solute's [C]), pressure potential(pressure of air) and gravitational potential (mass of water).

The matric potential is mainly influenced by the physical proprieties of the soil water potential, other factors from the soil are very important: particles size, pore volume or pore size. This varies depending on the type of soil, being sandy the soil with the thickest particles size with bigger pores and thus, less water retention capacity.

The soil can store energy as heat, which results in different temperature in the different layers of the soil. The heat flux in the soil depends both on the time and and temperature gradient.

This dependece on the temperature gradient is described by the Fourier's law which illustrates how the heat transfer in soil with the conduction of its components, from higher to lower temperatures.

$$G = -k\frac{\partial T}{\partial z}$$

where:

- k: Thermal conductivity in (W/mK)
- T: Temperature in K
- z: depth in m

If the time component is also included in the equation the following relation is true:

$$\frac{\partial T}{\partial t} = \frac{1}{C_v}[\frac{\partial}{\partial z}(k\frac{\partial T}{\partial z})]$$

where C_v is heat capacity in $J\ m^{-3}K{-}1$

The temperature along soil profile varies between day and night. During the day, the temperature of soil is higher than the temperature of air and thus starts to get more equally until night is set. After this occurs, the temperature of air is higher than the soil temperature, with this peak being more distinguish during midnight.

8.3 Sensors and measuring principle

8.3.1 Soil hydrology

For soil hydrology the main variable that are measured are the water content and the water potential.

To measure water content **time domain reflectrometry** it is used. The principle behind this sensor is that the dielectric constant of the soil depends on the amount of water present. The dielectric constant is estimated by measuring the travel time of a electromagnetic impulse between two rods inserted into the soil.

Water potential is measured using **tensionmeters**, that consists in a porous ceramic cap connect with a water reservoir under vacuum. The has a low water potential therefore attracts the water from the tensiometer, until an equilibrium point is reached. By measuring the, negative, pressure in the water filled tube the water potential of the soil can be estimated.

8.3.2 Soil temperature

Soil temperature is measured by **temperature sensors** placed at different depth. Usually a resistance sensor, like the *Pt100*, is used as long term stability is more important than high frequency response rate. The presence of multiple temperature sensor along the soil profile allows also to estimate the soil heat fluxes.

Soil heat flux can also be directly measured using **heat flux plates**, which using a series of thermocouple estimated the amount of heat transferred between the hot and the cold side.

8.4 Analysis

```
library(tidyverse)
library(lubridate)
library(FME)
library(kableExtra)
```

```
soil <- read_csv(here::here("Data_lectures/8_Soil_physics/Soil_temperature_bot_garden.csv"))
```

8.4.1 Soil Heat flux

Calculate the soil heat flux in 5cm depth and the soil storage flux for the soil layer above from the soil temperature profile. Assume a thermal conductivity $k = 3.5 W/mK$ *and a heat capacity* $c_p = 3.5 \times 10^6 J/m^3 K$*. Compare your calculations with direct measurements of the soil heat flux. How do thermal conductivity and heat capacity affect your results?*

We proceed with the calculation of soil heat flux at 5cm depth, suing the temperature difference between 5 and 10 cm in soil. Also, soil heat storage will be calculated.

```
c2k <- function(c) c + 273.15
k2c <- function(k) k - 273.15

#using as a reference the soil at 5 cm
d_T_space <- soil$Tsoil_5cm_degC - soil$Tsoil_10cm_degC
d_z_space <- - 0.05 # m


d_z_time <- - 0.03 # m
d_T_time <- c2k(lag(soil$Tsoil_5cm_degC)) - c2k(soil$Tsoil_5cm_degC)
d_t <- 600 # seconds manually calculated from the dataset (10 min)

Cv <- 3.5e6 # J m-3 K-1
k <- 3.5 # W m-1 K-1
```

```
soil <- soil %>%
  mutate(
    flux_time = d_z_time * d_T_time * Cv / d_t, # W m-2
    flux_space = - k * (d_T_space / d_z_space), # W m-2
    soil_flux = flux_time + flux_space)
```

```
soil %>%
  filter(week(Date)==15) %>%
  ggplot(aes(Date, soil_flux)) +
  geom_line(aes(col="Calculated"))+
  geom_line(aes(y=SoilHeatFlux_Wm2, col="Measured")) +
  scale_colour_colorblind() +
  labs(y="Soil heat flux (W/m2)", col="Type heat flux")
```

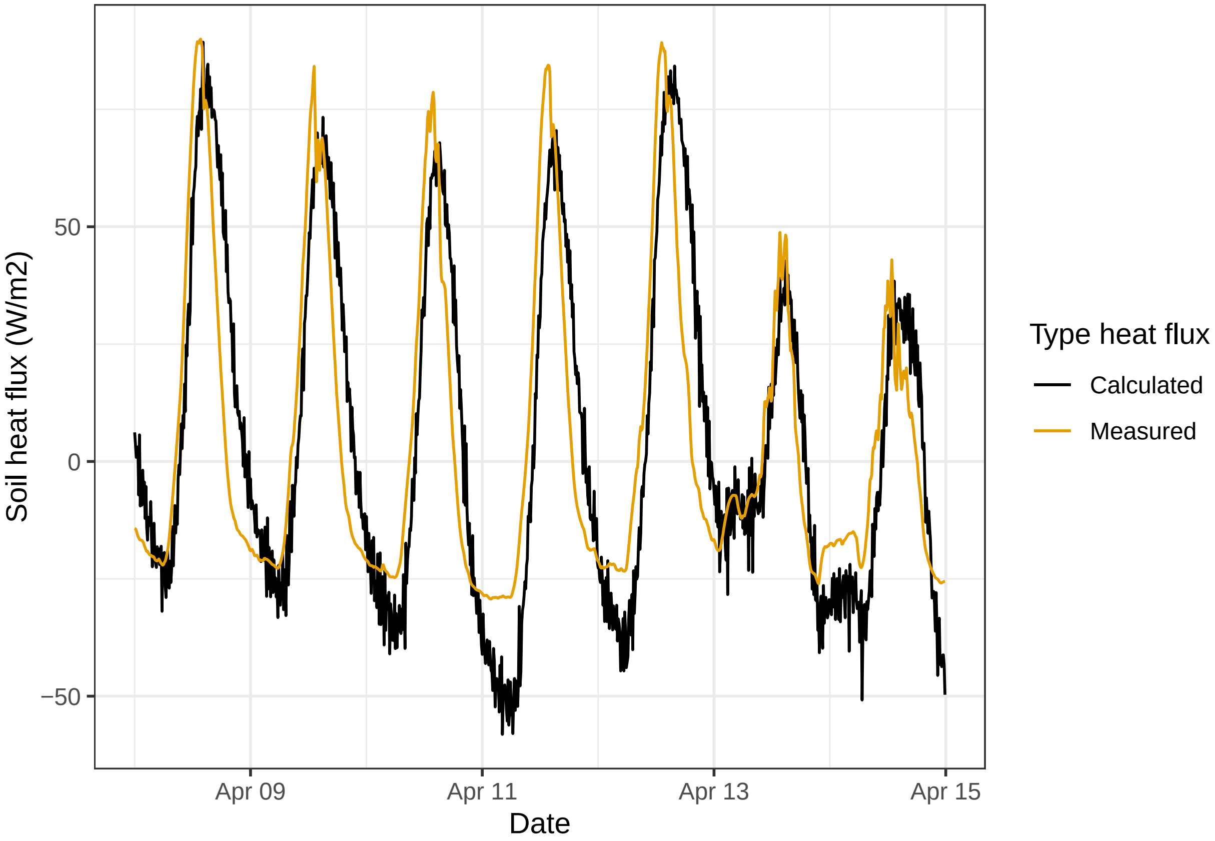

Figure 37: Comparison between heat flux measured (red) and calculated (black) over 1 week. Data collected at the botanical garden on the 8th-15th April 2021 at frequency of 10 minutes.

The heat flux measured at 5cm depth is compared with the one calculated in Figure 37. The two fluxes are overall comparable, with the exception of the early morning, when the measured flux stays constant at around -25 W/m^2 while the calculated fluxes drops further.

```r
calc_soil_flux <- function(pars){
  d_T_space <- soil$Tsoil_5cm_degC - soil$Tsoil_10cm_degC
  d_z_space <- - 0.05#m
  d_z_time <- - 0.03 # m
  d_T_time <- c2k(lag(soil$Tsoil_5cm_degC)) - c2k(soil$Tsoil_5cm_degC)
  flux_time = d_z_time * d_T_time * pars$Cv / d_t # W m-2
  flux_space = - pars$k * (d_T_space / d_z_space) # W m-2
  soil_flux <-  flux_time + flux_space
  tibble(soil_flux=soil_flux[-1]) #remove first row that is NA
}
pars <- list(Cv = 3.6e6, k=3.5)
sens <- sensFun(calc_soil_flux, pars, map=NULL)
```

```r
summary(sens) %>%
  knitr::kable(
    booktabs = TRUE,
    digits = 2,
    caption = 'Flux sensitivity'
  ) %>%
  kable_styling(latex_options = "HOLD_position")
```

Table 3: Flux sensitivity

	value	scale	L1	L2	Mean	Min	Max	N
Cv	3.6e+06	3.6e+06	0.72	4.44	0.16	-216.0	86.4	5500
k	3.5e+00	3.5e+00	1.25	4.51	0.84	-85.4	217.0	5500

8.4.1.1 Influence of Cv and K A local sensitivity analysis (Soetaert and Petzoldt 2010) was made for the two parameters used in the flux calculation: thermal conductivity (k) and heat capacity (C_v). The thermal conductivity has the biggest influence on the the overall flux (Table 3).

8.4.2 Soil temperature profile

The Figure 38 shows the soil temperature for one day of may. It is possible to clearly see the difference of how at 0.5m depth the temperature of soil remains more stable regarding the other depths. During the changes of one day between day and night, at this depth the soil does not have enough time to change the temperature so fast. The opposite is happening at a depth of 2cm or 5cm in soil, where the variations of temperature are stronger. Just before the dawn in the morning, the soil reaches the lowest temperature values, between 8-9 °C at this depths. Then during the day, the highest temperature is reached (12-13 °C) around 6pm where the sun starts going down starting the sunset.

```
soil_g <- soil %>%
  gather("depth", "T_soil", starts_with("Tsoil")) %>%
  mutate(depth = str_extract(depth, "\\d+"))
```

```
soil_g %>%
  # last day available to bet as close as possible to summer.
  # Number 126 was got as `max(yday(soil$Date))`
  filter(yday(Date)==126) %>%
  ggplot(aes(Date, T_soil, col=as_factor(depth))) +
  geom_line() +
  labs(y="Temperature soil (°C)", colour="Depth (cm)") +
  scale_color_colorblind()
```

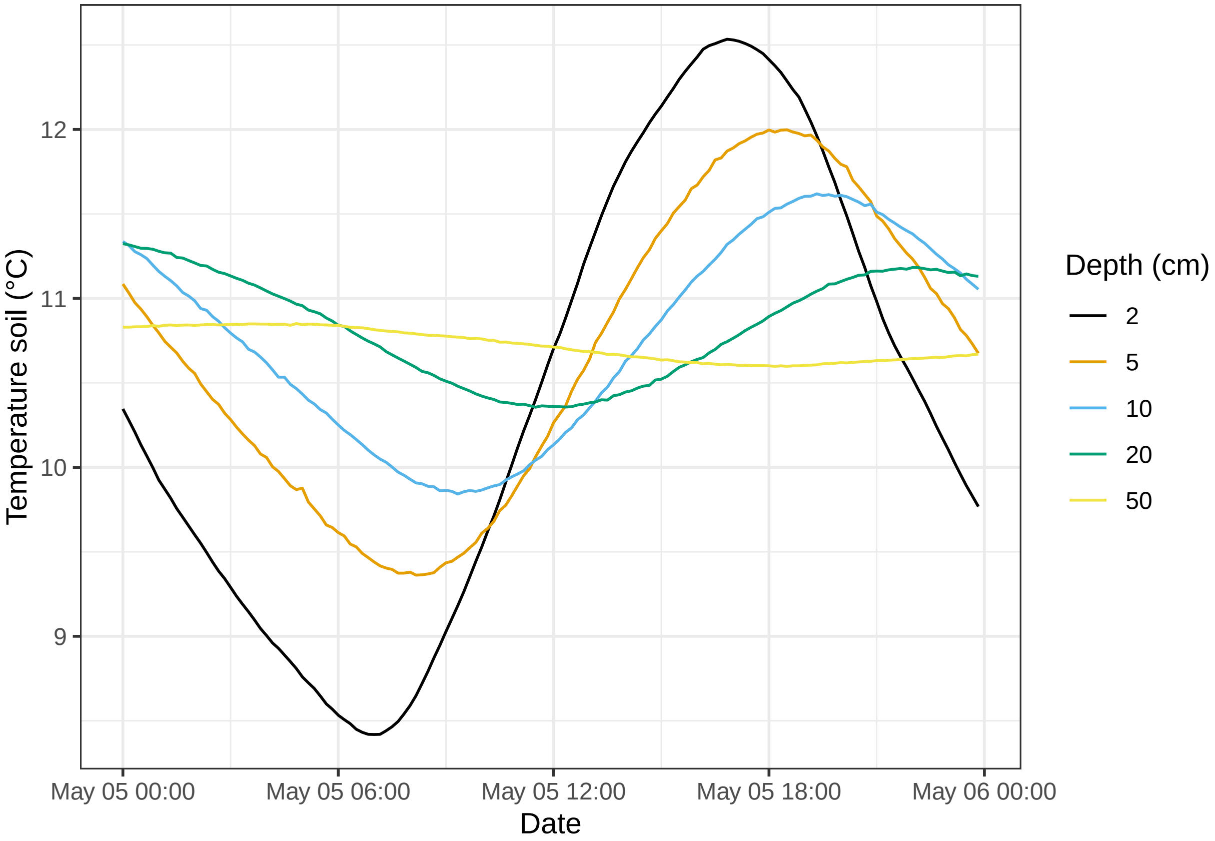

Figure 38: Soil temperature at different depths for one day. Data from 5th of May 2020, forest botanical garden.

9 Evapotranspiration

9.1 Motivation

The evapotranspiration (ET) is an important variable for ecosystems, as it represents the amount of water that is evaporating. This is a good indication of the amount of photosyntetic activity and by comparing it the potential ET water stress of ecosystems can be estimated.

Evaporanspiration is important to hold the global and continental scale hydrologic cycle. Evapotranspiration is influenced by a combination of factors; the increase of temperature, decrease of bulk canopy due to higher CO2 emissions and large scale land use change. (Likens and 1935- 2009)

9.2 Background

The ET is the actual amount of water that evaporates. In contrast the potential ET is the amount of water that would evaporate if there would be unlimited water available in the soil (Labedzki 2011).

The difference between the ET and the potential ET indicates the water stress of an ecosystem.

The measurement unit for ET is mm. The ET is often the ET on a daily basis.

The ET is measured in the field using different instruments, while the potential ET can be either measured or estimated using other enviromental variables.

In particular the Penman-Monteith is the most precise way to estimate the ET. It takes into account the energy available from the sun and the speed at which the water can evaporate, that depends on temperature, air humidity and wind.

$$LE = \frac{s(R_n - G) + \rho_a c_p \left(e_s - e_a\right)/r_a}{s + \gamma\left(1 + r_a/r_s\right)}$$

where:

- S Slope of saturation vapor pressure (kPa / K)
- R N Net radiation (W m -2)
- G Ground heat flux (W m -2)
- ρ Air density (kg m -3)
- C_p Specific heat capacity at constant pressure (J kg -1 K -1)
- e_s Saturation vapor pressure (kPa)
- e_a Actual vapor pressure (kPa)
- γ Psychrometric constant (kPa K -1)
- r_s Stomata resistance (s m -1)
- r_a Aerodynamic resistance (s m -1)

A useful aproximation of the Penman-Monteith equation is the Priestley-Taylor formula which requires less meteorolical data

$$PET \approx 1.26 \frac{s}{s + \gamma} \left(R_N - G\right)$$

where:

- PET is the daily evapotranspiration
- s is the saturation vapour pressure (kPa/K)
- R_N is the net radiation in MJ/m^2d
- G is the soil heat flux in MJ/m^2d

9.3 Sensors and measuring principle

There are many different ways to measure evapotranspiration,some are more accessible than others and depending the scenario some of them might be more suitable than others. The measurement of evapotranspiration claims quantitative data, and this data can be measure by water evaporation and the energy flux between soil and atmosphere. Rana and Kater (2000), describes the different ways to measure evapotranspiration based on hydrological, micrometeorological, plant physiology and analytic approaches. The first one includes: soil water balance and weighing lysimeters, the second one, energy balance and Bowen ration, aerodynamic method and eddy covariance. Plant physiology approach is based on sap flow method or chambers system. Last one, analytical approach are based on Penman-Montheith model. After this one, empirical approach can also be taken into account, such as process based on crop coefficient approach and soil water balance modeling (Rana and Katerji 2000).

Some of the most frequently measurements applied to evapotranspiration and that we have seeing in class are:

- **Evaporation pan**: a circle panel where precipitation is accumulated and then, with the help of a measuring bucket and based on a scale, water loss can be measured by the difference between the potential evapotranspiration of 2 days. Some of the errors are related to the expansion of water, wrong reading, limited recording for a volume of water and the possibility of the oasis effect to occur.
- **Piché evaporimeter**: is a type of atometer applied in the measurement rate of evaporation from a wet disc with absorbent paper. Results are dependent on wind speed that goes through the disc, as well as the wet bulb saturation deficit. The rate of evaporation is usually expressed as the volume of water evaporated per unit area in unit time.
- **Weighing lysimeter**: they allow the mass or volumetric soil water content variation to be measured by weighing the lysimeter and determining its change of mass over time. They can measure the net infiltration from precipitation or irrigation systems and the quantity of net evaporation between different wetting events (Meissner, Rupp, and Haselow 2020).

Bowen ratio energy balance method: also known as BREB method has been widely used to quantify water balance. It estimates the latent heat flux from a surface using measurements of air temperature and humidity gradients, net radiation and soil heat flux. In comparison with other methods such as eddy covariance or weighting lysimeters, is an indirect method (Todd, Evett, and Howell 2000). Some advantages are that it does not require information about aerodynamic characteristics of zone of interest, it integrates latent hear fluxes over a wide areas and gives an estimation of these flux in short time period intervals (e.g. less than half hour).

9.4 Analysis

9.4.1 Potential evapotranspiration

During the four days of measurement the potential ET has been estimated using the Priestley-Taylor (Figure 39). The daily potential ET varies a significantly during the 4 days, ranging from 6mm to almost 12 mm. This reflects the change in the weather conditions, as the day 1 and 4 were cloudy and colder.

```r
library(tidyverse)
library(lubridate)
library(scales)
theme_set(theme_bw())

et <- "Data_lectures/09_Turbulent_fluxes_I_ET/ET_data_forst_botanical_garden.csv" %>%
  here::here() %>%
  read_csv(locale = locale(decimal_mark = ",")) %>%
  rename(loc_id = replicates)
meteo <-
  "Data_lectures/09_Turbulent_fluxes_I_ET/MeteoData_BotanicalGarden.csv" %>%
  here::here() %>%
  read_csv()
```

```
#' Potential evapotranspiration using Priestley-Taylor equation
calc_pet <- function(T_air, Rn, G){
  g <- 0.067 # kPa K -1
  s <- ( 4098 * (0.6108 * exp((17.27 * T_air ) / (T_air +237.3) ) ) ) / ( T_air + 237.3 )^2
  pet <- 1.26 * s * (Rn - G)/ (s + g)
}
```

```
#adding R_n and G to the
meteo_d <- meteo %>%
  # Need to convert from W (J/s) to MJ/d, using a factor 0.0864
  # calculating with high frequency data and then averaging over the day
  mutate(PET = calc_pet(TA_degC, `NetRadiation_Wm-2`, `GroundHeatflux_Wm-2`)) %>%
  group_by(Date = floor_date(Date, "day")) %>%
  summarise(PET = mean(PET))
```

```
meteo_d <- meteo %>%
  group_by(Date = floor_date(Date, "day")) %>%
  # Need to convert from W (J/s) to MJ/d, using a factor 0.0864
  summarise(R_n_d = mean(`NetRadiation_Wm-2`) * 0.0864,
            G_d = mean(`GroundHeatflux_Wm-2`) * 0.0864,
            T_air = mean(TA_degC)) %>%
  mutate(
    PET = calc_pet(T_air, R_n_d, G_d)
  )
```

```
et <- et %>%
  group_by(loc_id) %>%
  mutate(
    et_pan = lag(pan_height_mm) - pan_height_mm,
  )
```

```
ggplot(meteo_d, aes(Date, PET)) +
  geom_line() +
  labs(y="PET [mm]")
```

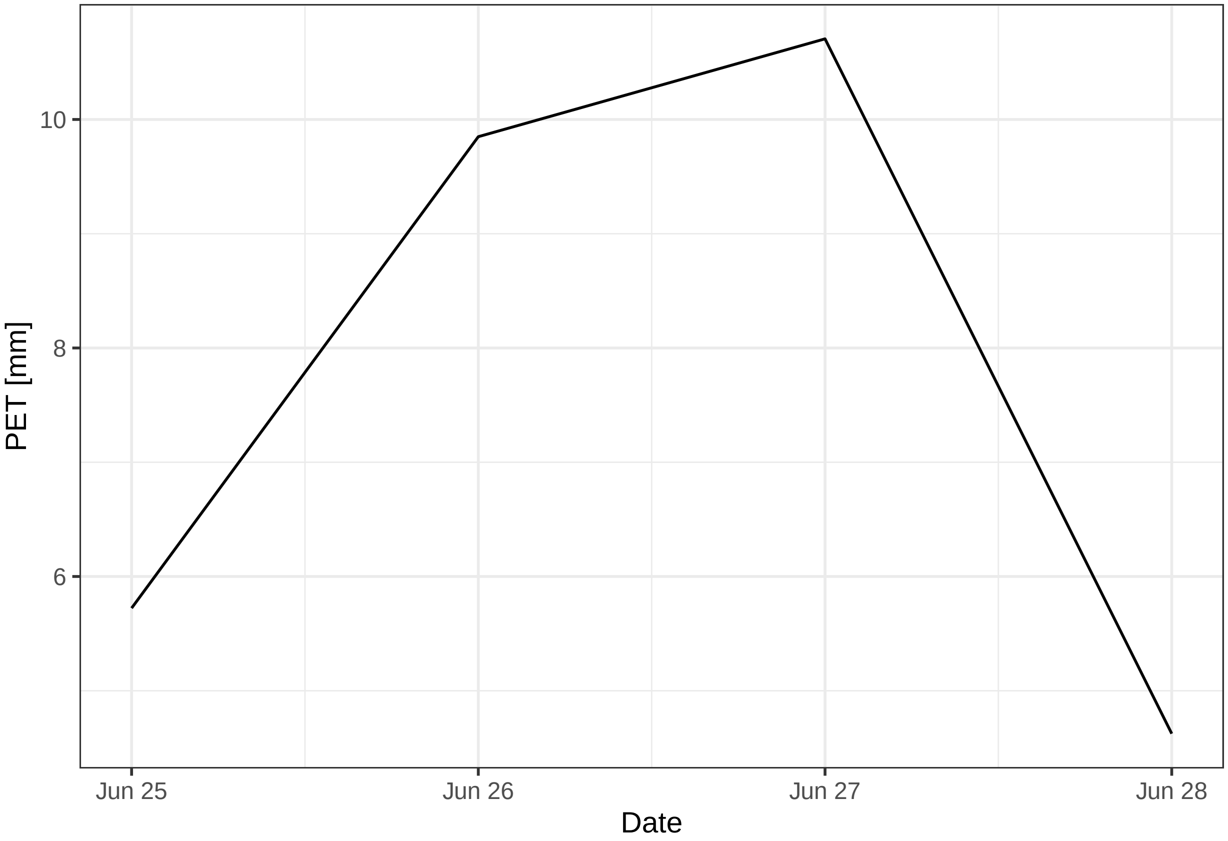

Figure 39: Potential evapotranspiration estimated using the Priestley-Taylor equation. Data from the botanical garden for the 25-29th of June 2021.

9.4.2 Field Measurements

The potential ET (PET) has been measured using 8 Piché evaporimeters in 4 different locations around the botanical garden. Moreover a evaporation pan has aslo been used to measure the PET.

The piché evaporimeters were put in different location, some in the shade, some under the sunlight. Therefore a variation in the measured PET is expected (Figure 40), but overall the measures are comparable and is cleat that the ET on the 27th of June was higher.
The piché evaporimeters between each other (Figure 41), it possible to see a reasonable pattern, for example the PET in the tower (exposed to the sun) is much higher than in the canopy of the fir. Finally the two evaporimeters in the same location report similar values for all location, but the fir. This is a confirmation that the field measurements are correct. The different measures may be related to problem in the piché evaporimeters sealing.

Finally the PET of the pan is compared with the PET estimated with the Priestley-Taylor equation (Figure 42). However, the values are very different with the PET from Priestley-Taylor being more than the double of the PET from the Pan. All the deta processing steps have been double checked but no explanation has been found for this difference.

```
pich_d <- 3 #cm
pich_inn_d <- 0.9 # cm
# calc area exposed to air:
# 2 times the area of the pare dish (two sides) - the area of glass
pich_dish_area <- 2 * (pi / 4 * pich_d ^ 2) - (pi / 4 * pich_inn_d ^ 2) # cm^2
# area inside the tube
pich_int_area <- (pi/4 * pich_inn_d ^2 ) # cm^2
```

```
et <- et %>%
  group_by(loc_id) %>%
  mutate(
    # here the scale is the opposite, the lower the number the more the water
    diff_pich = pich_height_cm - lag(pich_height_cm),
    # need to convert to mm of ET
    et_pich = diff_pich * pich_dish_area * pich_int_area / 10
  )
```

```
spot_cols <- hue_pal()(4)
names(spot_cols) <- unique(et$spot)
et %>%
  drop_na() %>% #removing first empty day
ggplot(aes(date)) +
  geom_line(aes(y=et_pan, col="Pan"), size=.7) +
  geom_jitter(aes(y = et_pich, col=spot), width = .06, height = 0) +
  scale_color_manual(values=c(spot_cols, "Pan" = "black" )) +
  labs(y="Potential Evapotranspiration [mm]", colour="Location") +
  scale_color_colorblind()
```

```
et %>%
  drop_na() %>%
ggplot(aes(spot, et_pich, col=spot)) +
  geom_boxplot() +
  scale_color_colorblind() +
  labs(x='Location', y = "PET [mm]", col="Location")
```

```
et %>%
  left_join(meteo_d, by = c("date"= "Date")) %>%
  gather("type", "et", PET, et_pan, factor_key = T) %>%
```

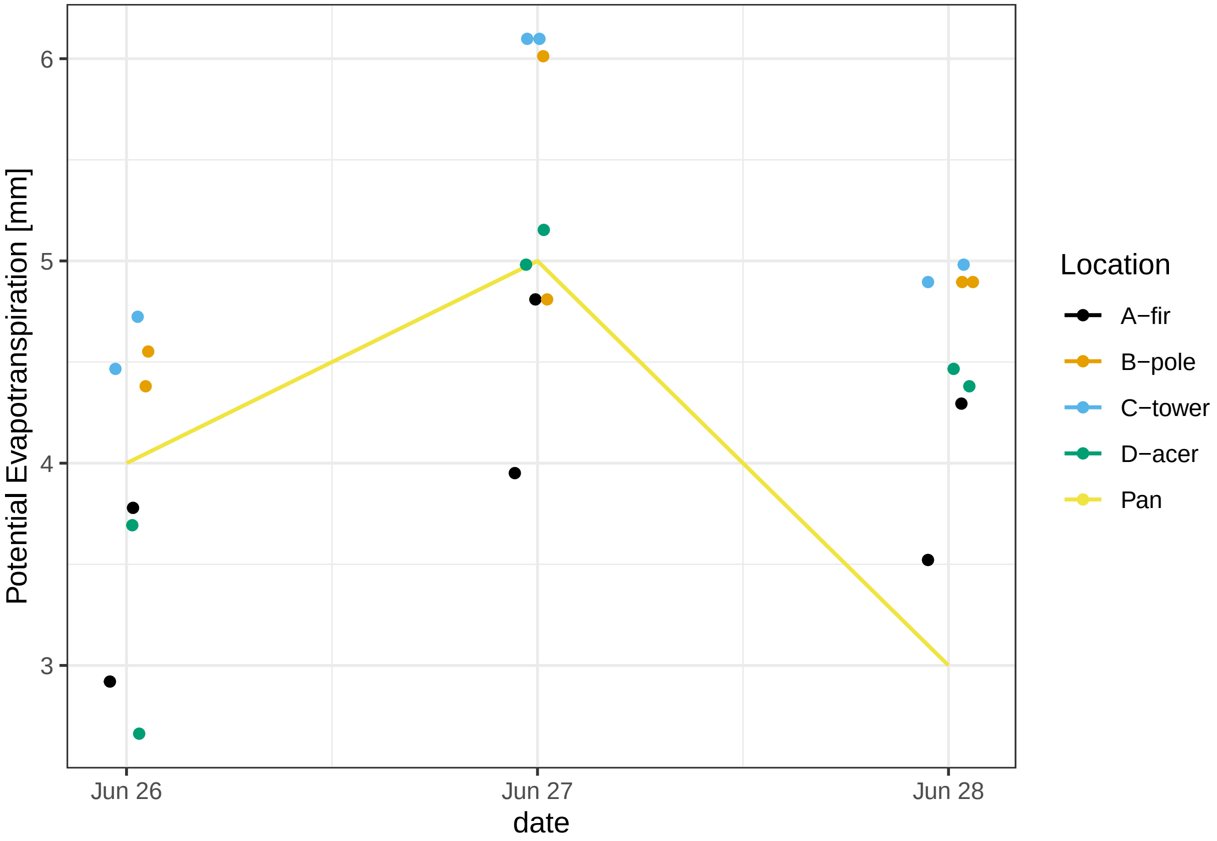

Figure 40: Comparison of Potential Evapotranspiration between the evaporatin pan and 8 Piché evaporimeters in 4 different locations (fir, pole, tower, acer). Data from field measurement 26-28 June 2021.

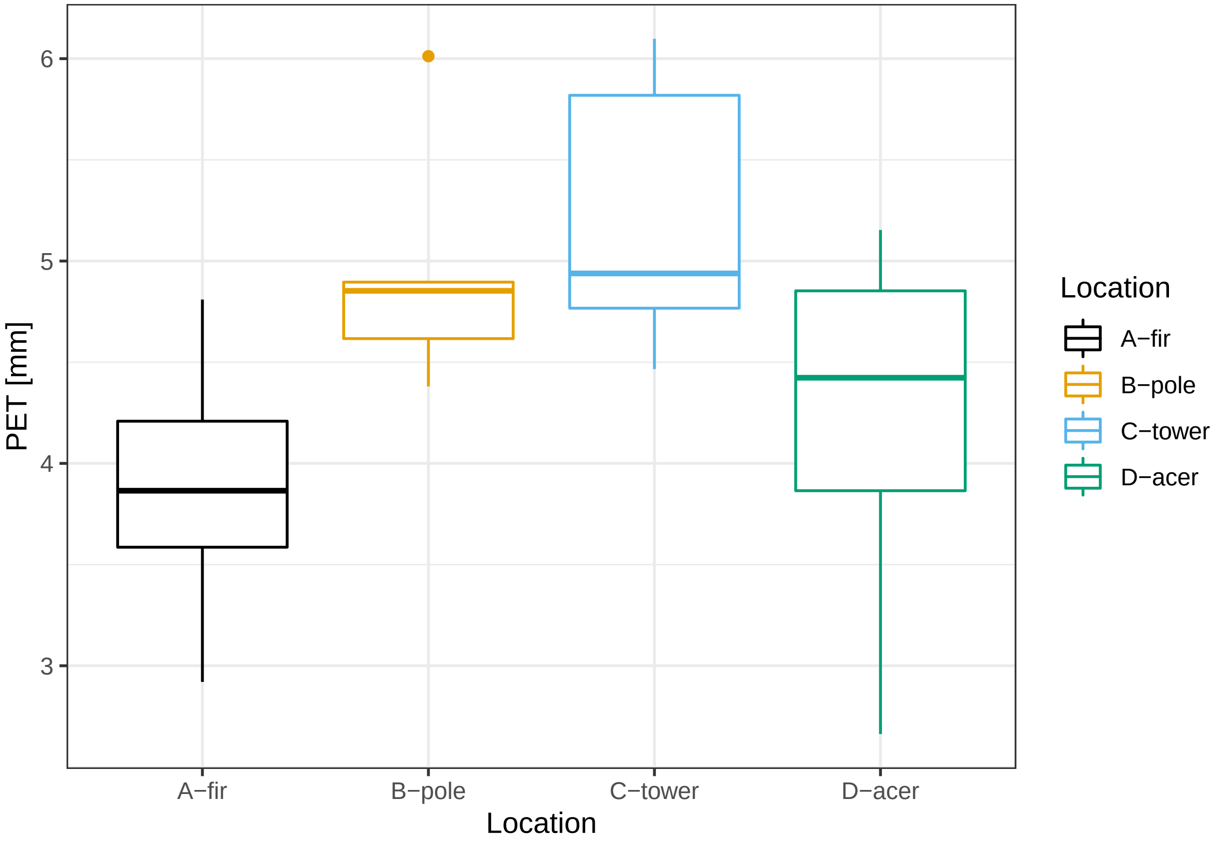

Figure 41: PET at the different location. Data from field measurements 26-28 June 2021.

```
ggplot(aes(date, et, col=type)) +
geom_line() +
labs(y= "Potential Evapotranspiration [mm]", x="Date", col="PET") +
scale_color_colorblind(labels = c("Priestley-Taylor", "Evap. pan"))
```

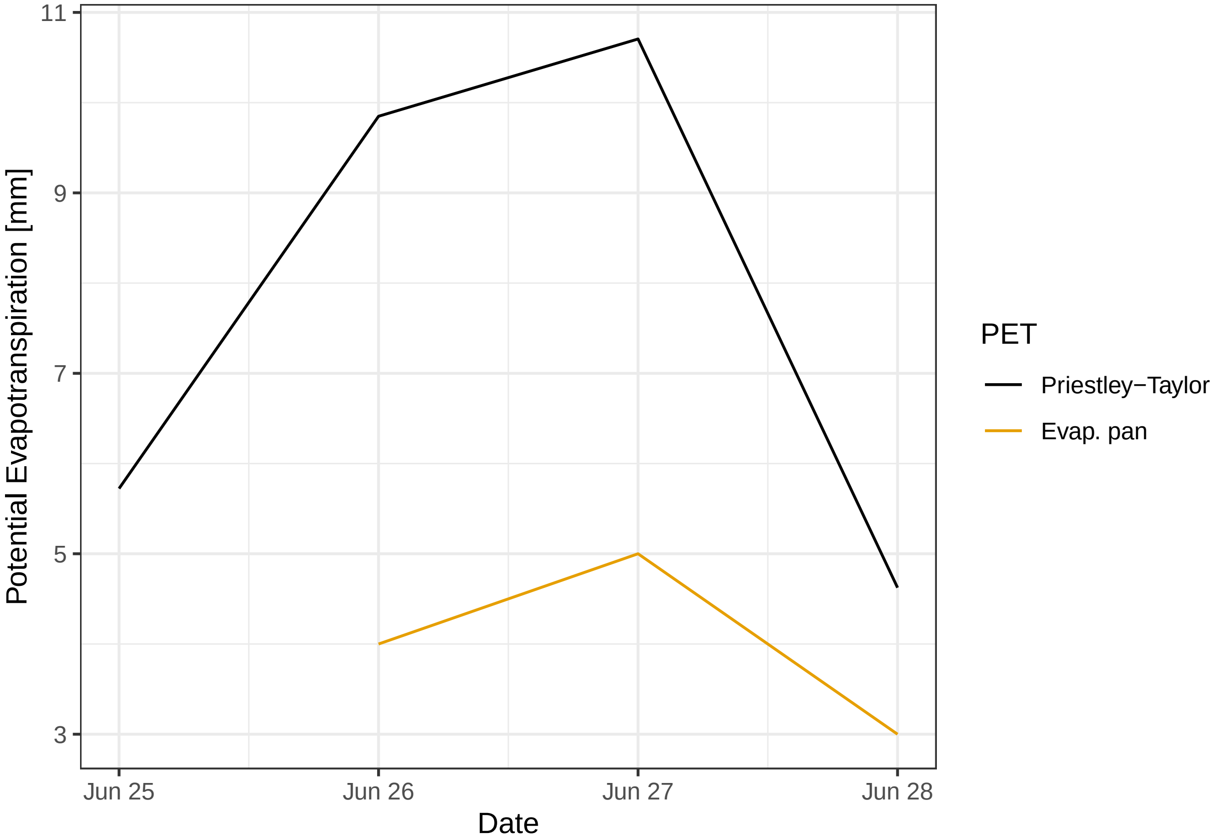

Figure 42: Comparison of the PET between the field measure with the evaporation pan and the estimation using the Priestley-Taylor equation. Data from field measurements and botanical garden 25-28 June 2021.

10 Turbulent fluxes CO_2

10.1 Motivation

Part of the turbulent fluxes regulating the natural systems on earth is CO_2 flux. In this chapter, we learned some ways to trace the flux of this important gas, also well known as green house gas. First, a small introduction about the carbon cycle in land ecosystems. Carbon approaches vegetation through the photosynthesis. Afterwards, carbon is allocated in different parts of plants; leafs, stem and root, while part of it is also lost by the leaching of soil and erosion. During the process of plant respiration, Carbon dioxide is also released in small amounts, together with the respiration of roots and other microorganisms present in soil. Although seems as a process easy to quantify, it also has its adversities, such as the turbulent movement of gas fluxes in forests.

10.2 Background

The measurement of the net CO_2 exchange can be achieved by direct or continuous way, this way ecosystem is not disturbed. Also it can be measured by the integration over the entire ecosystem. Some researches have been going on measuring the carbon cycle at a global scale. This is known as the integrated carbon observation system (ICOS). The ICOS consists on the long term assessment of European carbon balance, with sample plots at different levels: atmosphere, terrestrial ecosystem and ocean. This programme is not the only one, also Fluxnet; measuring CO_2, water vapor and energy flux all along the world. Thanks to these type of initiatives, other natural factors can be analyzed such as temperature, precipitation, temperature and soil moisture. As an example, the drought experienced in Europe in 2018, where a big decreasing in the CO_2 uptake in central Europe was experienced, while and increase was observed in southern and northern countries of Europe. Besides the consequences of this drought period, the European average had a smaller effect than in 2003.

A simple definition of flux is the exchange of any quantity per area and time. The flux of CO_2 and H_2O is measured with $\mu mol\ m^{-2}s{-}1$ and $mmol\ m^{-2}s{-}1$ respectively. Leaf evapotranspiration and Heat is measured by Wm^{-2}.

10.3 Sensors and measuring principle

The data measurement of gas fluxes can be done with the use of different ways, by data driven or with the use of instruments. Later, this data is required in ecosystem modelling. According to what flux is going to be measured, some instruments can be required as an additional equipment. The instruments can be: chambers, lysimeter and sap flow. The following formula describes the principle on which **gas chambers** are based on to measured CO_2 flux in soil.

The CO_2 flux assumption is based on its linear increase, using the following formula;

$$F_c = \frac{dc}{dt}\frac{MV}{A}$$

where:

- dc: change of CO_2 concentration ($\mu mol\ mol^{-1}$)
- dt: change in time (s)
- M = $p/R\ T$ molar volume ($mol\ m^{-3}$)
- R = 8.314 $J\ mol^{-1}K^{-1}$
- V: chamber volume (m^3)
- A: chamber surface area (m^2)

10.4 Analysis

1. Calculate the half hourly latent heat flux, the net ecosystem exchange of CO_2 and the sensible heat flux from the high frequency turbulence data.

```
library(tidyverse)
library(cowplot) #multiple plots
library(naniar) # to replace missing values
library(ggthemes)
theme_set(theme_bw())
```

10.5 Respiration chambers

Field data were collected with respiration chambers. Four chambers were installed and for each of them the CO_2 concentration measured roughly every 10 minutes.

The CO_2 concentration increase linearly over time (Figure 43) in all chamber, the slope of this increase is a bit different for each chamber and varies slightly with time, but is overall constant (Figure 44).

The Co_2 flux is the computed (Figure 45). The Different chambers have different respiration rates, higher in 1 and 2, lower in 3 and 4 and intermediate in 5. The respiration between different chambers is overall comparable.

In Figure 46 the time flux over time is plotted. This is expected to be constant, but this is only the case for the chamber 4, that is very stable. For the other chambers there is a much higher variation and in particular in the first two chambers the flux increases over time. This is probably due to measurement errors as the field it was hard to estimate when the CO_2 concentration in the chamber was stabilizing.

```
co2 <- read_csv2(here::here("Data_lectures/soilCO2flux.csv"),
                 col_names=c("time", "co2", "chamber"), skip=1)
# loading some data directly from in the R code for simplicity
p_a <- 989.5 #hPa - air pressure
T_a <- 16 + 273.15 # °C - air temperature
diam <- 0.152 # m - radius of chamber
top_height <- 0.138 # m - height of top part of the chamber (same for everything)

# area
area <- pi / 4 *diam^2

# chamber specific data
chambers <- tribble(
  ~"Tsoil",~"chamber",~"height",
  17,     1, 0.16225,
  17,     2, 0.17,
  16.8, 3, 0.1495,
  16.7, 4, 0.154,
  17.2, 5, 0.159
) %>%
  mutate(
    Volume= pi / 4 * diam^ 2 * (height+top_height))
# Molar Gas Constant
R <- 8.314 # J/mol K
```

```
# Molar volume
M <- p_a / (R*T_a)

co2flux <- co2 %>%
  # Adding chamber information
  left_join(chambers) %>%
  group_by(chamber) %>%
  mutate(
    dt = time - lag(time),
    dc = co2 - lag(co2),
    Fc = (dc/dt)*(M*Volume/area),
    # for plotting
    chamber = as.factor(chamber)
  )
```

```
ggplot(co2flux, aes(time, co2, col=chamber)) +
  geom_line() +
  scale_color_colorblind() +
  labs(x="Time [s]", y="CO2 [umol mol-1]")
```

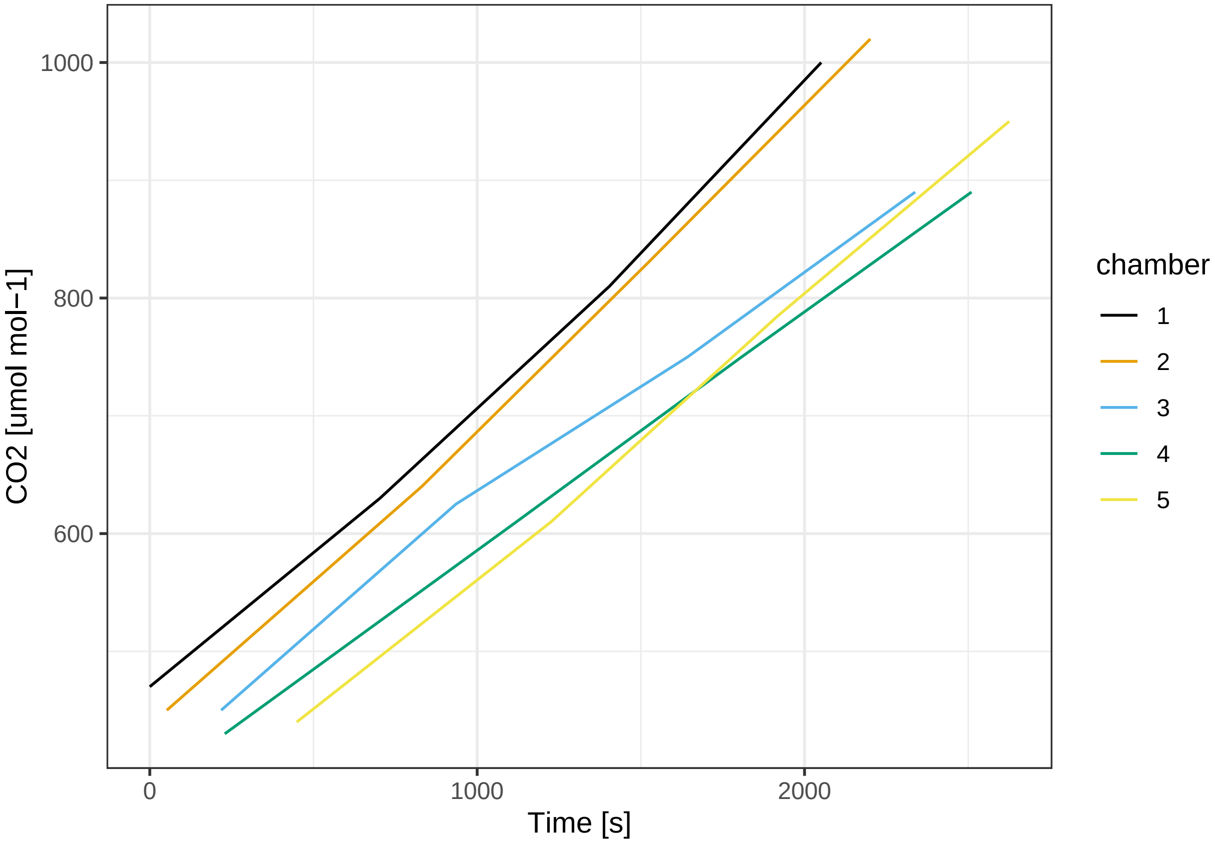

Figure 43: Increase of CO_2 concetration over time in each of the chamber. Data from field measurements.

```
ggplot(co2flux, aes(time, dc/dt, col=chamber)) +
  geom_line() +
  scale_color_colorblind() +
  labs(x="Time [s]", y="Slope CO2 increase [umol mol-1 s-1]")
```

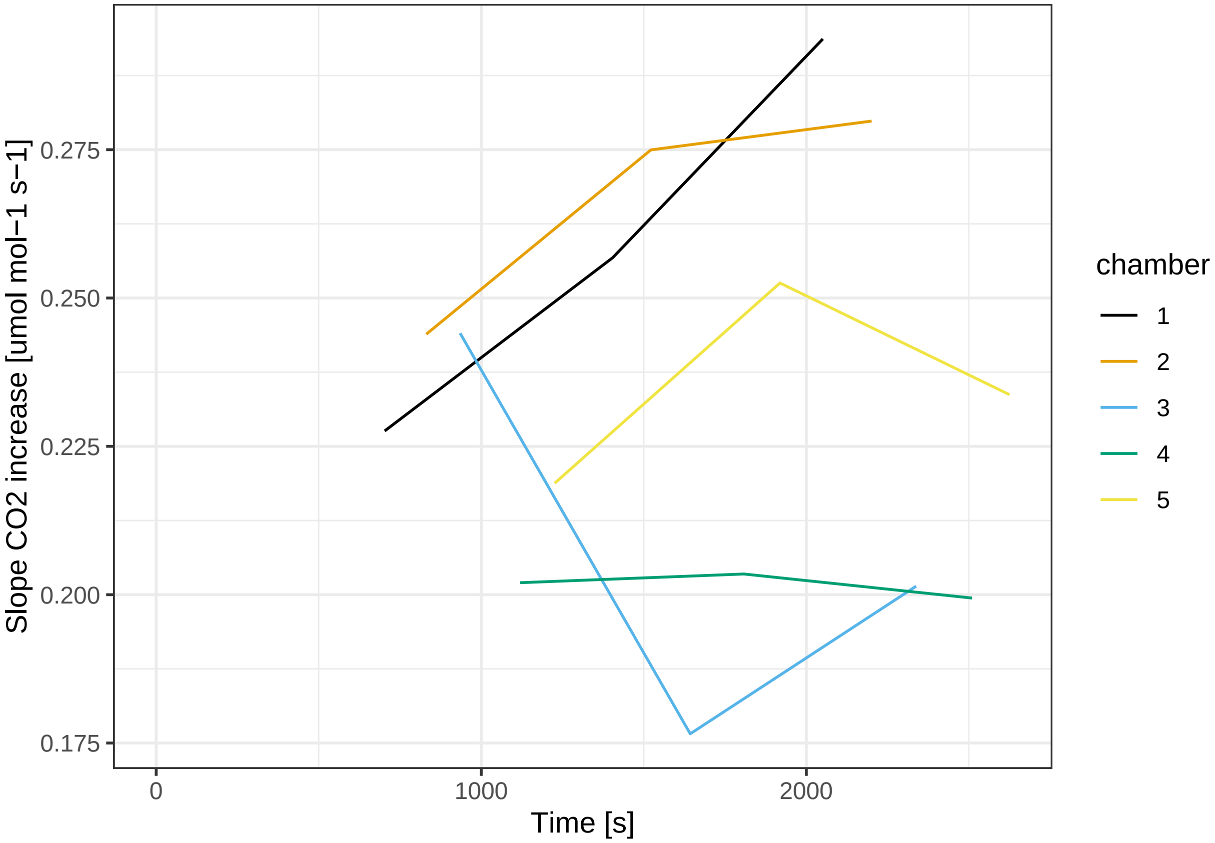

Figure 44: Slope of the increase in CO_2 concetration over time in each of the chamber. Data from field measurements.

```
ggplot(co2flux, aes(chamber, Fc, col=chamber)) +
  geom_boxplot() +
  scale_color_colorblind() +
  labs(x="Chamber", y="CO2 flux [umol s-1 m-2]")
```

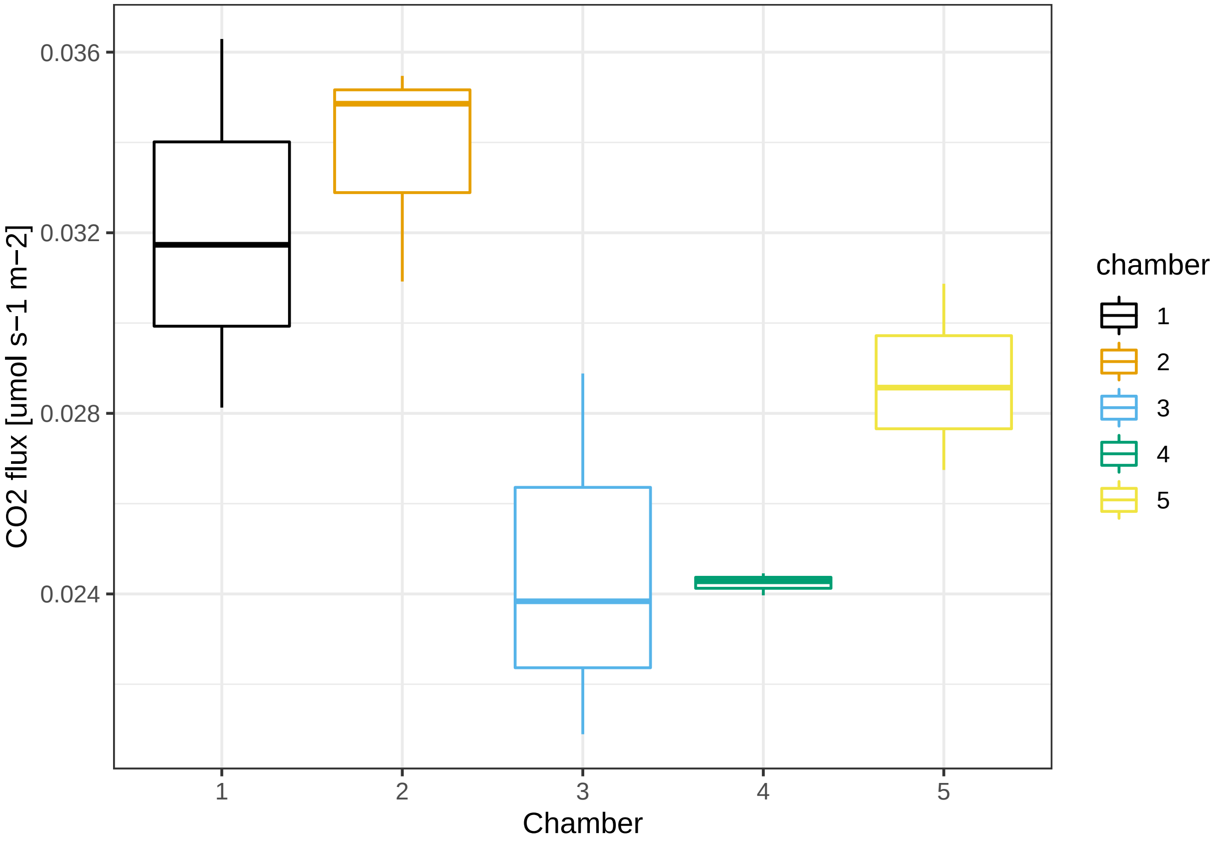

Figure 45: CO_2 flux measured in the different chambers. Data from field measurements.

```
ggplot(co2flux, aes(time, Fc, col=chamber)) +
  geom_line() +
  scale_color_colorblind() +
  labs(x="Time [s]", y="CO2 flux [umol s-1 m-2]")
```

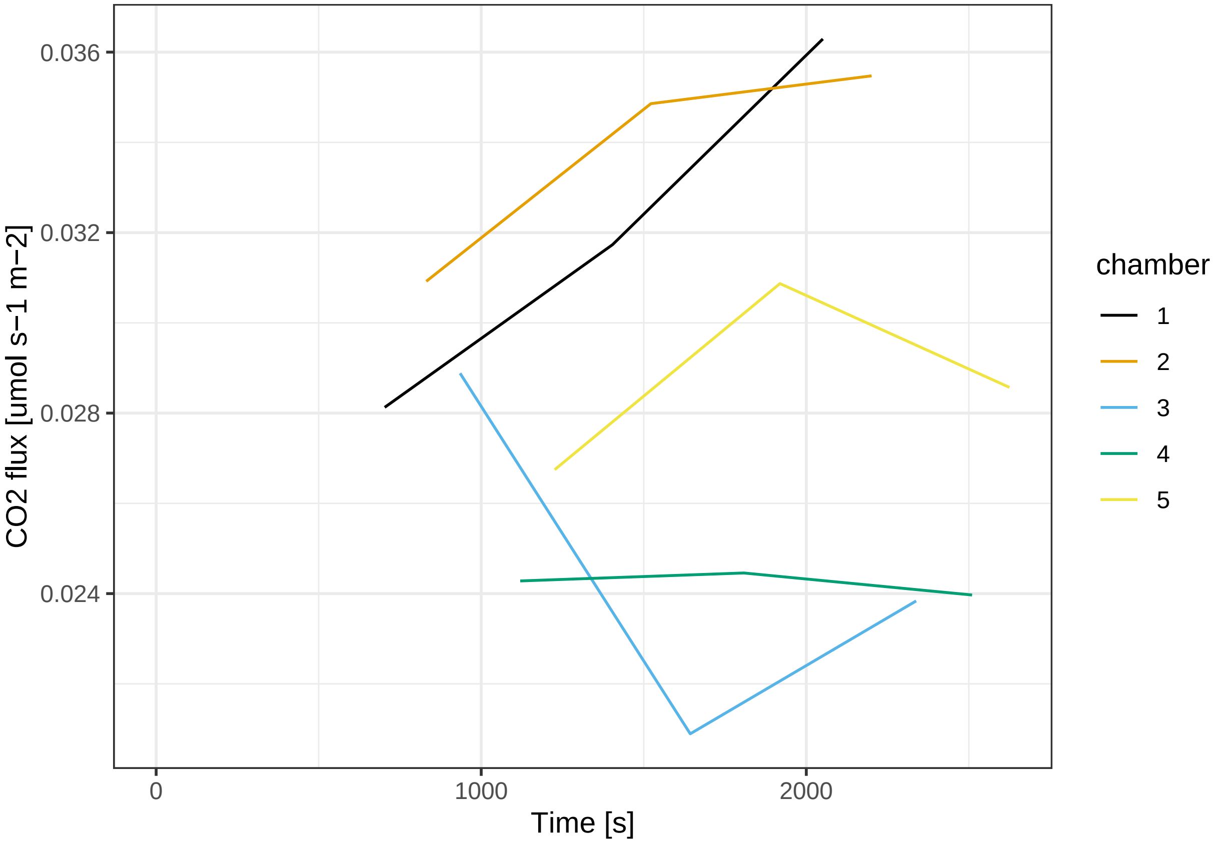

Figure 46: CO_2 flux measured in the different chambers over time. Data from field measurements.

11 Turbulent Fluxes Eddy Covariance

11.1 Motivation

The atmosphere and the ecosystem exchange energy and gases through turbulent flows. The main gas transported are CO_2 and water vapour, while energy is transferred as sensible heat fluxes and momentum fluxes.

Eddy Covariance (EC) is a technique that measures the entity of the turbulent fluxes. It can measure a wide range of variables, automatically, continuously, with high accuracy and with minimal impact on the ecosystems. It is primary method used to monitor the carbon balance of ecosystems.

11.2 Background

The air flow next to the surface becomes turbulent, with eddies that transport mass and energy between the atmosphere and the ecosystems. It is possible to measure the properties of the eddies and use them to estimate the different fluxes. Therefore in an Eddy Covariance setup the first instrument needed is an anemometer that measures the 3 components of the wind, this can then be paired with a gas analyzer to measure the change of concentration. The fluxes can then be calculated as the covariance between the vertical wind and the property of interest.

In particular those are the most common fluxes that are measured:

- $\boldsymbol{CO_2}$. Carbon dioxide is one of the main variable measured in EC setups, as it is the main outcome of photosynthesis and respiration. The flux can be calculated as:

$$F_{co2} = \rho_m \overline{w`c_{co2}`}$$

where ρ_m is the air molar density and $\overline{w`c`}$ the covariance between the vertical wind speed (w) and the CO_2 concentration. The measurement unit is $\mu mol/m^2 s$.

- $\boldsymbol{H_2O}$. The amount of water vapour exchanged can be The flux can be calculated as:

$$E_{h2o} = \rho_m \overline{w`c_{h2o}`}$$

where ρ_m is the air molar density and $\overline{w`c`}$ the covariance between the vertical wind speed (w) and the H2O concentration. The measurement unit is $\mu mol/m^2 s$.

- **Latent heat**. Latent heat flux is the amount of energy that is transferred due the evaporation of water vapour. In fact it is directly connected to the H_2O flux and can be calculated as:

$$LE = \lambda E$$

where λ is the latent heat of vaporization of water and E is the H_2O flux.

The unit of measurement is W/m^2

- **Sensible heat**. The sensible heat flux is the amount of energy exchanged as heat. The heat energy can be transferred through the movement of air masses. The flux can be calculated as:

$$H = \rho_a c_p \overline{w`T`}$$

where ρ_a is the air density, c_p is the air heat capacity and $\overline{w`T`}$ the covariance between the vertical wind speed (w) and T the air temperature. The measurement unit is W/m^2.

- **Momentum**. Momentum is the amount of mechanical energy that is transferred from the wind to the ecosystem. It is by definition negative, as the the ecosystem cannot transfer mechanical energy to the wind.

The flux can be calculated using only measures from a 3D anemometer as:

$$\tau = \rho_a \sqrt{(\overline{w`u`})^2 + (\overline{w`v`})^2}$$

where ρ_a is the air density and u and v the two horizontal wind components. The measurement unit is Kg/ms^2.

The Eddy Covariance technique is very powerful, however, flux calculation are based on several assumptions and require complex post processing to reduce the errors.

First of all Eddy covariance can be applied only when there is a well developed turbulent layer, which is often not the case at night. Then measurements in heterogenous enviroments or on slopes can affect the turbulence and hence the flxes measurers.

The EC data needs to be preprocessed to remove instruments errors (despiking and calibration) and compensate for non ideal instrument setup (time lag correction, coordinate rotations). Flux can then be calculated for a time period and then additional correction applied due to the loss of high frequency information (spectral corrections) and change in gas density (Webb-Pearmen-Leuning correction).

Each instrument and variable need to have each of the steps mentioned above tailored for its specific characteristics.

11.3 Sensors and Measuring principles

Eddies have different sizes and speed but many of them are small and very fast (fraction of seconds), therefore instruments with high response rate are needed.

A basic EC setup consist of two instruments, an **anemometer** and a **gas analyzer**. The anemometer is a 3D sonic anemometer, that can therefore also measure the vertical component of the wind and have high frequency reading. The gas analyzer measure the concentrations using the absorption of specific infrared wavelenghts. The most commonly measured gas are CO_2, H_2O and methane. Gas analyzer can be open path, where the measurement chamber is directly exposed to the air or closed path where the measure take place in a separate chamber and the air is pumped through a pipe.

The EC instruments are mounted on a tower as they need to be above the ecosystem where the turbulence layer is well developed.

11.4 Analisys

```
ec_col_names <-  c("TIMESTAMP","TIMESTAMPS","u","v","w","T_sonic",
                   "SA_DIAG_VALUE","CO2_ABS","H2O_ABS","CO2_CONC",
                   "H2O_CONC","CO2_POW_SAM","H2O_POW_SAM","CO2_POW_REF",
                   "H2O_POW_REF","co2","h2o","T_CELL","PRESS_CELL",
                   "GA_DIAG_CODE","T_DEW","CO2_STR")
```

```
ec_test_path <-
  "Data_lectures/10_Turbulent_fluxes_II/10_Turb_fluxes_CO2/Reinshof_flux_HF_202105300001.dat" %>%
  here::here()
ec <-read_csv(ec_test_path, skip=4, col_names = ec_col_names, na=c("", "NaN"))

files <- dir_ls(here::here("Data_lectures/10_Turbulent_fluxes_II/10_Turb_fluxes_CO2/"))

# taking 4 sample data for plots at 4 different moment of the day
# need to convert integer and remove the last element
idx <- seq(1, 48, length.out=10) %>% as.integer() %>% head(-1)
ec_samples_paths <- files[idx]
```

```
ec_samples <- map(ec_samples_paths,
                  ~read_csv(.x, skip=4, col_names = ec_col_names, na=c("", "NaN")))
```

11.4.1 Raw flux calculation

The fluxes are first calculated without applying any correction.

```
# the emp
process_ec_file <- function(file, p=function(){}) {
  ec <-read_csv(file, skip=4, col_names = ec_col_names, col_types = cols(), na=c("", "NaN"))
  time <- str_extract(file, "\\d+.dat$") %>%
    parse_date_time("YmdHM")
  flux <- calc_fluxes(ec) %>%
    mutate(time = time)
  p() # step progress bar
  return(flux)
}
```

```
# Molar Gas Constant
R <- 8.314 # J/mol K

#' Calcuates all fluxes
calc_fluxes <- function(ec){
  # celsius to kelvin
  T_a <- ec$T_CELL + 273.15
  # need to convert pressure to Pa from kPa
  p_a <- ec$PRESS_CELL * 1e3

  # how to calc this?
  rho_a <- 1
  # How to calc lambda
  # how to calc Cp

  tibble(
    rho_m = calc_rho_m(p_a, T_a),
    co2 = calc_co2_flux(ec$w, ec$co2, rho_m),
    h2o = calc_h2o_flux(ec$w, ec$h2o, rho_m),
    sens_heat = calc_sens_heat_flux(ec$w, ec$T_sonic, rho_a),
    lat_heat = calc_lat_heat_flux(h2o),
    mom = calc_mom_flux(ec$u, ec$v, ec$w, rho_a)
  )
}

calc_co2_flux <- function(w, co2, rho_m){
  cov(w, co2, use="complete.obs") * rho_m
}

calc_h2o_flux <- function(w, h2o, rho_m){
  cov(w, h2o, use="complete.obs") * rho_m
}

calc_mom_flux <-  function(u, v, w, rho_a){
  sqrt(
    cov(w, u, use="complete.obs")^2 * cov(w, v, use="complete.obs")^2
  ) * rho_a
}

calc_sens_heat_flux <- function(w, T_sonic, rho_a, Cp= 1000){
  cov(w, T_sonic, use="complete.obs") * rho_a * Cp
}

calc_lat_heat_flux <- function(h2o_flux, lambda = 2.256 ) {
  h2o_flux * lambda
}

# air molar density
calc_rho_m <- function(p, T_a){
  mean( p / (R * T_a), na.rm=T)
}
```

```
files <- dir_ls(here::here("Data_lectures/10_Turbulent_fluxes_II/10_Turb_fluxes_CO2/"))
with_progress({
  # this is to have a progress bar
  p <- progressor(along=files)
  ec_flux <- cache_rds(map_dfr(files, process_ec_file, p))
})
```

11.4.2 Flux corrections

In order to properly measure the fluxes several corrections are necessary. In the subsequent section the main correction are analyzed individually and tested then the fluxes with the corrections are calculated.

11.4.2.1 Remove implausible values The first step is not a proper correction, but a plausibility check. All the readings outside sensible ranges are discarded. To test this function some artificial data has been generated and as seen in Table 4 the out of range values were correctly removed.

```
remove_implausible <- function(ec){
  ec %>%
    replace_with_na_at(c("u", "v", "w"), ~abs(.x) > 10) %>%
    replace_with_na_at("co2", ~!between(.x, 350, 700)) %>%
    replace_with_na_at("T_sonic", ~!between(.x, -15, 50)) %>%
    replace_with_na_at("h2o", ~!between(.x, 2, 30))
}
```

```
# manually checking with some fake data that the function is working as intended
tibble(
  u = seq (-15, 15, length.out=20),
  v = u,
  w = u,
  co2 = seq (30, 800, length.out=20),
  T_sonic = seq (-20, 80, length.out=20),
  h2o = seq (0, 40, length.out=20),
) %>%
  remove_implausible() %>%
  kbl(booktabs=T, caption="Example dataset after removal of implausible values") %>%
  kable_styling(latex_options = "hold_position")
```

Table 4: Example dataset after removal of implausible values

u	v	w	co2	T_sonic	h2o
NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	-14.736842	2.105263
NA	NA	NA	NA	-9.473684	4.210526
NA	NA	NA	NA	-4.210526	6.315790
-8.6842105	-8.6842105	-8.6842105	NA	1.052632	8.421053
-7.1052632	-7.1052632	-7.1052632	NA	6.315790	10.526316
-5.5263158	-5.5263158	-5.5263158	NA	11.578947	12.631579
-3.9473684	-3.9473684	-3.9473684	NA	16.842105	14.736842
-2.3684211	-2.3684211	-2.3684211	354.2105	22.105263	16.842105
-0.7894737	-0.7894737	-0.7894737	394.7368	27.368421	18.947368
0.7894737	0.7894737	0.7894737	435.2632	32.631579	21.052632
2.3684211	2.3684211	2.3684211	475.7895	37.894737	23.157895
3.9473684	3.9473684	3.9473684	516.3158	43.157895	25.263158
5.5263158	5.5263158	5.5263158	556.8421	48.421053	27.368421
7.1052632	7.1052632	7.1052632	597.3684	NA	29.473684
8.6842105	8.6842105	8.6842105	637.8947	NA	NA
NA	NA	NA	678.4211	NA	NA
NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA

11.4.2.2 Despike Despiking consists in removing spikes in the data, which are moment where suddenly very different. Here the spikes are considered as the values where the different from the mean is bigger than a certain number of times (default 5) the standard deviation. This is a simple despiking method and the number should be properly tuned for each measured variable according to their characteristics and sensor behavior.

Figure 47 shows how this despiking filter works. The limit of 1.8 standard deviations is artificially low and used in the plot only as an example of the deskipe. All the values that are outside the range mean +/- 1.8 standard deviation are removed and replaced with NAs.

```r
despike <- function(data, times_sd = 5) {
  mean_data <- mean(data, na.rm=T)
  sd_data <- sd(data, na.rm=T)
  spikes <- abs(data - mean_data) > (times_sd*sd_data)
  # spikes will be NAs
  data[spikes] <- NA
  return(data)
}
```

```r
plots_despike <- map(ec_samples, function(ec){
  co2_mean <- mean(ec$co2, na.rm=T)
  co2_sd <- sd(ec$co2, na.rm=T) * 1.8
  ggplot(ec, aes(TIMESTAMP, despike(co2, times_sd = 1.8))) +
    geom_hline(yintercept = co2_mean) +
    geom_hline(yintercept = c(co2_mean + co2_sd, co2_mean - co2_sd), linetype=2) +
    geom_line(aes(y = co2), color="grey60") +
    geom_line(colour = colorblind_pal()(2)[2]) +
    labs(x="Time", y="CO2 [umol mol-1]")
})

do.call(plot_grid, c(plots_despike[c(1, 3, 6, 9)], nrow=2))
```

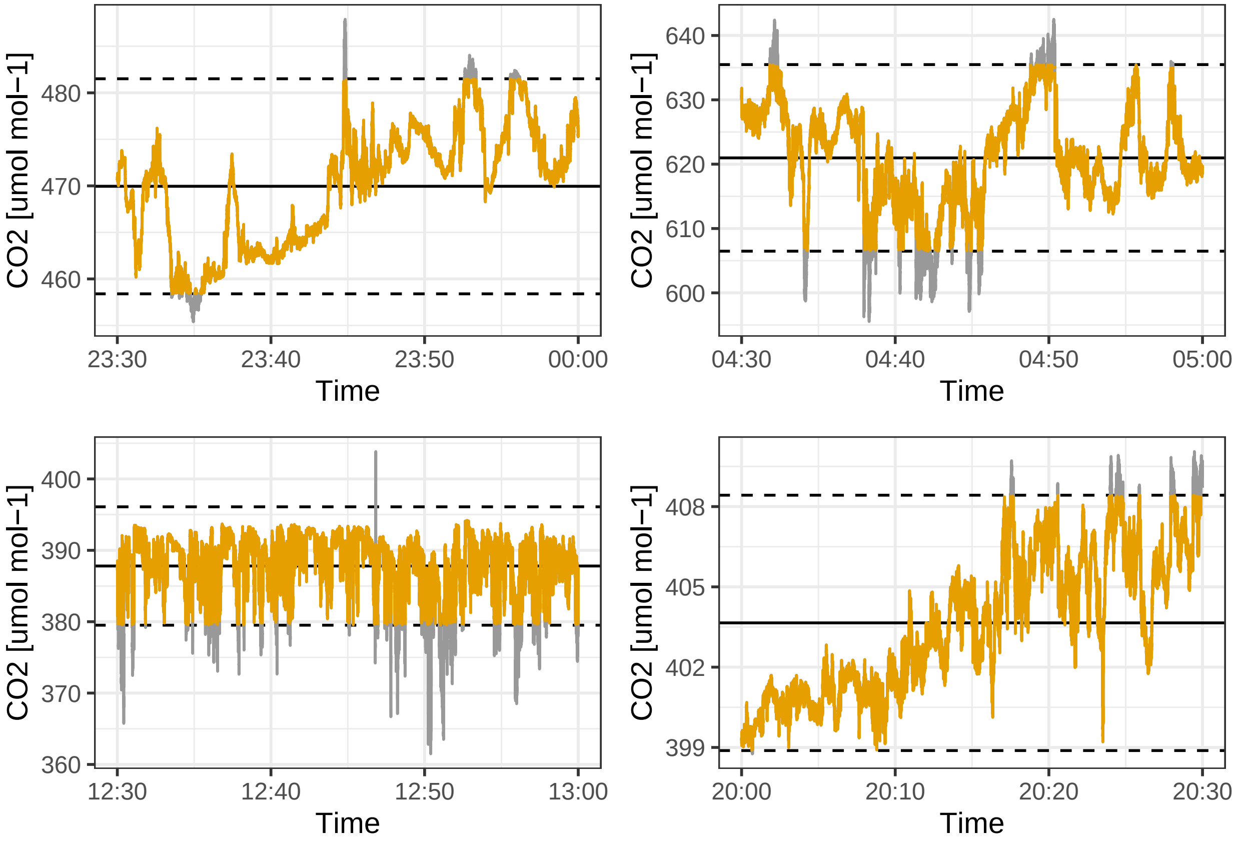

Figure 47: Example of how despiking work with articially low factor of +/- 1.8 standard deviations. Grey sections is the data that was removed. Dashed lines are the mean +/- 1.8 standard deviation. Each subplot is a different time of the day. Data from Hainich national park 31st May 2021.

11.4.2.3 Rotations The reference system of the wind is rotated in order to have a zero mean in the vertical wind component and a zero mean wind direction.

Table 5, shows the values of the wind means before and after the double rotation, which works correctly.

```r
# wind should have 3 columns u, v, and w
double_rotation <- function(wind){

  wind %>%
    # 1st rotation around z into the mean wind direction
    mutate(
      # need to use atan2 otherwise the angle may have the wrong sign
      theta = atan2(mean(v,na.rm = T), mean(u,na.rm = T)),
      u_1 = u*cos(theta) + v*sin(theta),
      v_1 = -u*sin(theta) + v*cos(theta),
      w_1 = w
    ) %>%
    # 2nd rotation around new y-axis to nullify the vertical wind speed
    # to be used for further analysis
    mutate(
      phi = atan2(mean(w_1,na.rm = T), mean(u_1,na.rm = T)),
      u_2 = u_1*cos(phi) + w_1*sin(phi),
      v_2 = v_1,
      w_2 = -u_1*sin(phi) + w_1*cos(phi)
    ) %>%
    # rename variables
    mutate(
      u = u_2,
      v = v_2,
      w = w_2) %>%
    # remove temp variables
    select(
      -u_1, -v_1, -w_1, -theta, -phi,
    )
}
```

```r
rot_wind <- double_rotation(ec)

tribble(~"Variable", ~"Before correction", ~"After correction",
  "Mean vertical wind", mean(ec$w), mean(rot_wind$w),
  "Mean wind direction", mean(atan2(ec$v, ec$u)), mean(atan2(rot_wind$v, rot_wind$u))) %>%
  kbl(booktabs = T, caption="Mean wind speed and direction before and after double rotation") %>%
  kable_styling(latex_options = "hold_position")
```

Table 5: Mean wind speed and direction before and after double rotation

Variable	Before correction	After correction
Mean vertical wind	-0.0136382	0.0000000
Mean wind direction	1.9793592	0.0005655

11.4.3 Time lag correction

Due to spatial separation between the gas analyzer and the anemometer each Eddy is measured at slightly different moments in time, this difference even if it often very small can lead to serious underestimation of fluxes. Therefore a time lag correction has been developed to compensate this. For each variable and half an hour a time lag is calculated in order to maximize the correlation between the vertical wind speed and the gas concentration. This is done by trying different time lags and then finding the one where the absolute correlation is maximized.

```r
# maybe find a better name, lag that supports negative values
lag2 <- function(x, n){
  if (n >= 0){
    lag(x, n)
  }
  else{
    lead(x, -n)
  }
}

#lags the second argument
lagged_cor <-  function(x, y, n){
  cor(x, lag2(y, n), use="complete.obs")
}

max_n_cor <- function(w, gas){
  # maximise the absolute value of the covariance
  opt <- optimize(function(n){
    # the optimize function use doubles, leg uses ints
    abs(lagged_cor(w, gas, as.integer(n)))
  }, interval = c(-700, 700), maximum = T, tol=1)
  return(opt$maximum)
}

time_lag_correction <- function(w, gas){
  lag2(gas, max_n_cor(w, gas) %>% as.integer )
}
```

Figure 48 show the time log correction for the CO_2 sensor. As it can been seen during the day (7h - 17h) there is a clear peak in the correlation plot and the time lag found by the optimization algorithm is in a sensible range. However during night the is no peak and the found time lag is outside a sensible range. This is probably due to the fact that at night the turbulent fluxes are not well developed and therefore it is difficult to correctly apply EC. In order to better test this hypothesis in figure 49 the time lag correction has been applied to the the sonic temperature, where it should be 0 as it is measured together with the wind speed. The time lag correction is correctly found to be 0 during the day, but not for the night. This confirms that the time lag correction is working properly and the problem is related to the fluxes themselves.

In real life conditions more complex algorithm and quality assurance system are used to avoid this kind of errors.

```r
plots_lag <- map(ec_samples, function(ec){
  co2_lagged_cor <- tibble( n = seq(-700, 700, 5),
                         lagged_cor = map_dbl(n,~ lagged_cor(ec$w, ec$co2, .x)))
  max_cor <- max_n_cor(ec$w, ec$co2) %>% round(0)
  ggplot(co2_lagged_cor, aes( x= n , y = lagged_cor)) +
    geom_line() +
    geom_vline(xintercept = max_cor, col=colorblind_pal()(2)[2]) +
    labs(title=ec$TIMESTAMP[1] %>%
           format("%Hh%M") %>%
           paste(" lag:", max_cor), y="correl", x="time lag")
})

do.call(plot_grid, c(plots_lag))
```

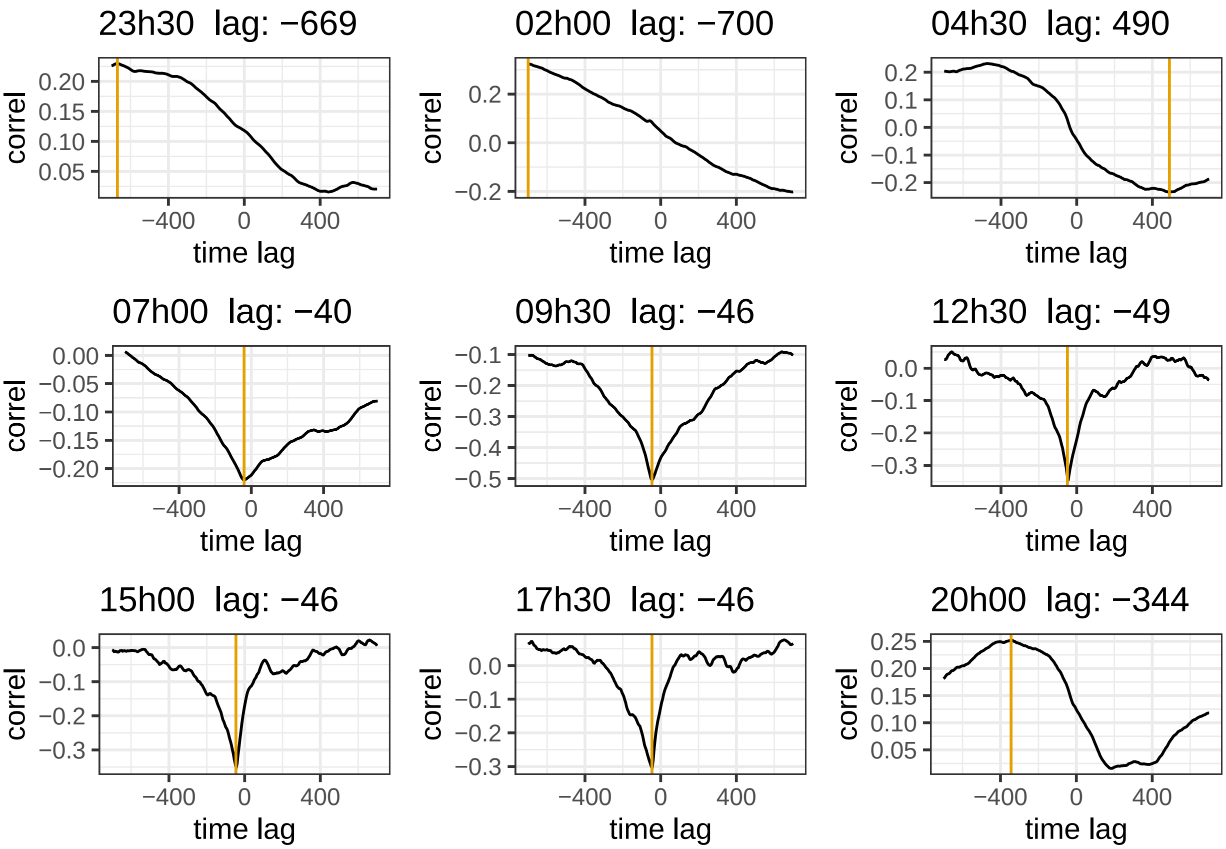

Figure 48: Time lag correction for CO_2. The plots are the correlation coefficient with different time lags, the yellow line is at the max value found by the optimization algorithm. The subplot are for different time of the day. Data from Hainich national park first week June 2021.

```
plots_lag <- map(ec_samples, function(ec){
  co2_lagged_cor <- tibble( n = seq(-700, 700, 5),
                         lagged_cor = map_dbl(n,~ lagged_cor(ec$w, ec$T_sonic, .x)))
  max_cor <- max_n_cor(ec$w, ec$T_sonic) %>% round(0)
  ggplot(co2_lagged_cor, aes( x= n , y = lagged_cor)) +
    geom_line() +
    geom_vline(xintercept = max_cor, col=colorblind_pal()(2)[2]) +
    labs(title=ec$TIMESTAMP[1] %>%
           format("%Hh%M") %>%
           paste(" lag:", max_cor), y="correl", x="time lag")
})

do.call(plot_grid, c(plots_lag))
```

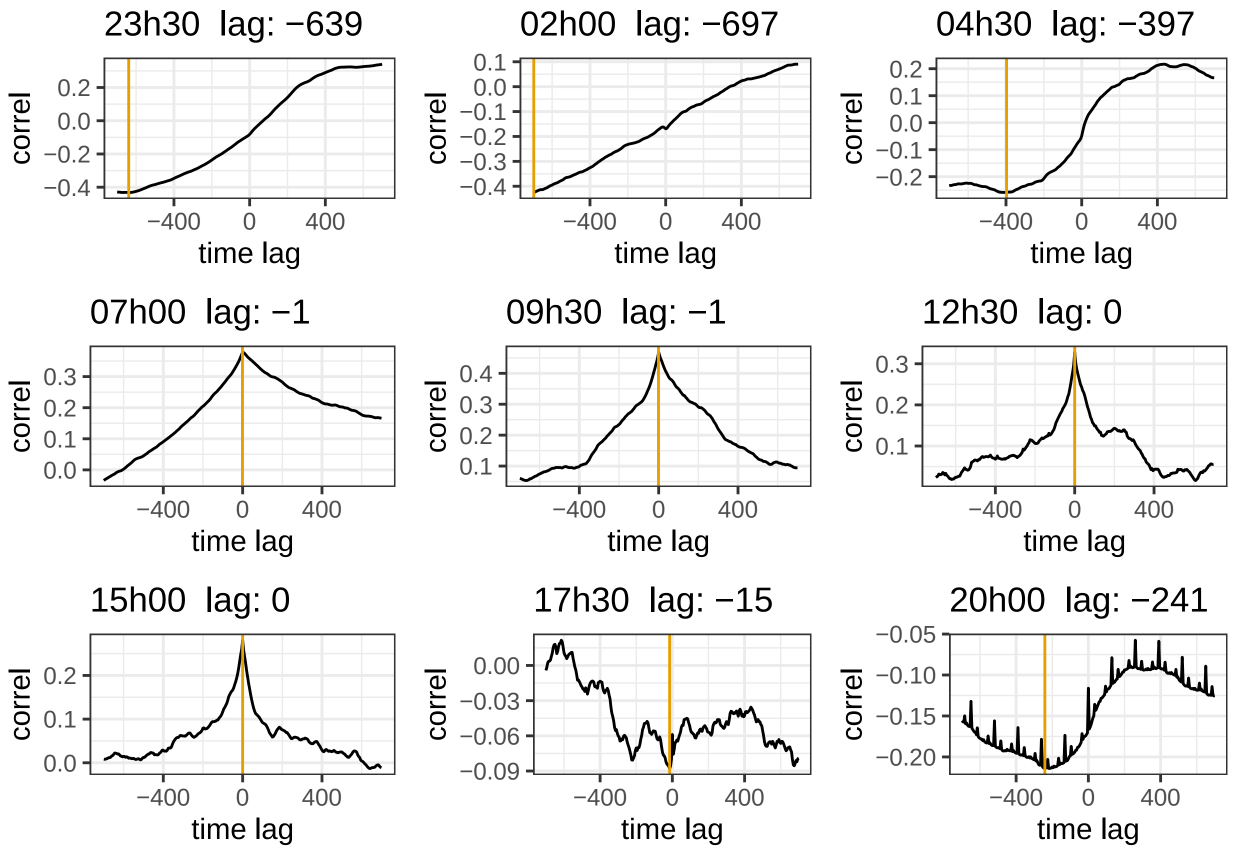

Figure 49: Time lag optimization for sonic temperature. The expected time lag optimization is 0 as the sonic temperature is measured by the anemometer itself. The plots are the correlation coefficient with different time lags, the yellow line is at the max value found by the optimization algorithm. The subplot are for different time of the day. Data from Hainich national park first week June 2021.

11.4.4 Corrected fluxes

All the correction are applied to the fluxes calculations and each variable is analyzed.

```r
apply_corr <- function(ec){
  ec %>%
    #remove_implausible() %>% # this step is very slow and not really needed.
    mutate(across(c(u, v, w, co2, h2o, T_sonic), despike)) %>%
    double_rotation %>%
    mutate(across(c(co2, h2o), ~time_lag_correction(w, .x)))
}

process_ec_file_corr <- function(file, p=function(){}) {
  ec <-read_csv(file, skip=4, col_names = ec_col_names, col_types = cols(), na=c("", "NaN"))
  time <- str_extract(file, "\\d+.dat$") %>%
    parse_date_time("YmdHM")
  flux <- ec %>%
    apply_corr %>%
    calc_fluxes %>%
    mutate(time = time)
  p() # step progress bar
  return(flux)
}
```

```r
files <- dir_ls(here::here("Data_lectures/10_Turbulent_fluxes_II/10_Turb_fluxes_CO2/"))

with_progress({
  # this is to have a progress bar
  p <- progressor(along=files)
  ec_flux_corr <- cache_rds(map_dfr(files, process_ec_file_corr, p))
})
```

11.4.4.1 CO_2 fluxes The CO_2 flux (Figure 50) has a clear daily pattern, negative during the day and positive during the night. This is expected as during the day there is photosynthesis and CO_2 is absorbed by plants resulting in an overall negative flux, while during the night there is only respiration that produces positive CO_2 fluxes.

The importance of correction can also be seen with the raw fluxes underestimating the CO_2 flux. This is striking on the 30 of May and is mainly due to the missing time lag correction.

```
ggplot() +
  geom_hline(yintercept = 0, linetype=2) +
  geom_line(aes(time, co2, col="raw"), data = ec_flux) +
  geom_line(aes(time, co2, col="corrected"), data= ec_flux_corr) +
  scale_color_colorblind() +
  labs(x="Time", y="CO2 flux [umol m-2 s-1]", col="Flux")
```

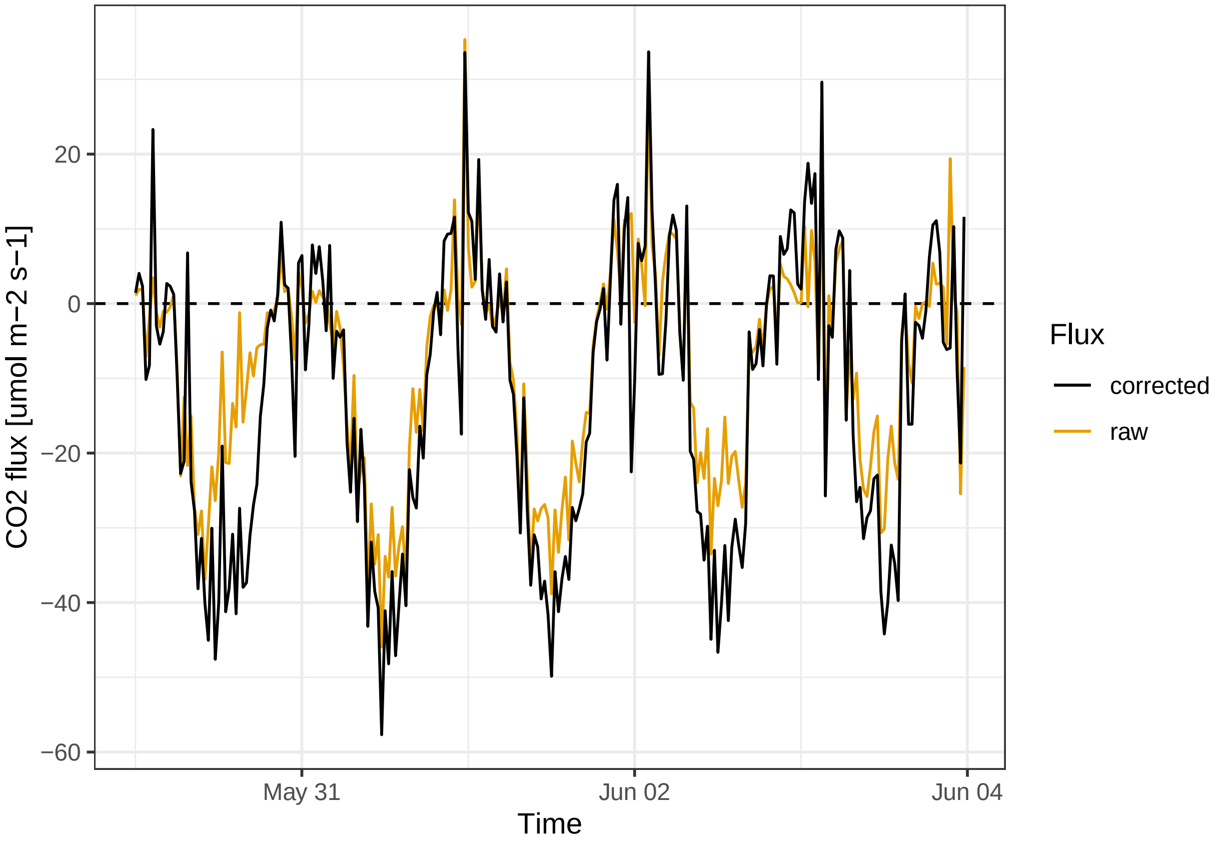

Figure 50: CO_2 fluxes. Comparison between corrected and raw fluxes calculations. Fluxes calculated for every half an hour. Data from Hainich national park first week June 2021.

11.4.4.2 ***H_2O* fluxes** The water fluxes have also a a daily pattern (Figure 51), with the majority of the water exchanged during the day from the ecosystem to the atmosphere. The evapotraspiration of water during the day is the process that drives the fluxes.

In this figure 51 the importance of water flux correction is evident.

```
ggplot() +
  geom_hline(yintercept = 0, linetype=2) +
  geom_line(aes(time, h2o, col="raw"), data = ec_flux) +
  geom_line(aes(time, h2o, col="corrected"), data= ec_flux_corr) +
  scale_color_colorblind() +
  labs(x="Time", y="H2O flux [mmol m-2 s-1]", col="Flux")
```

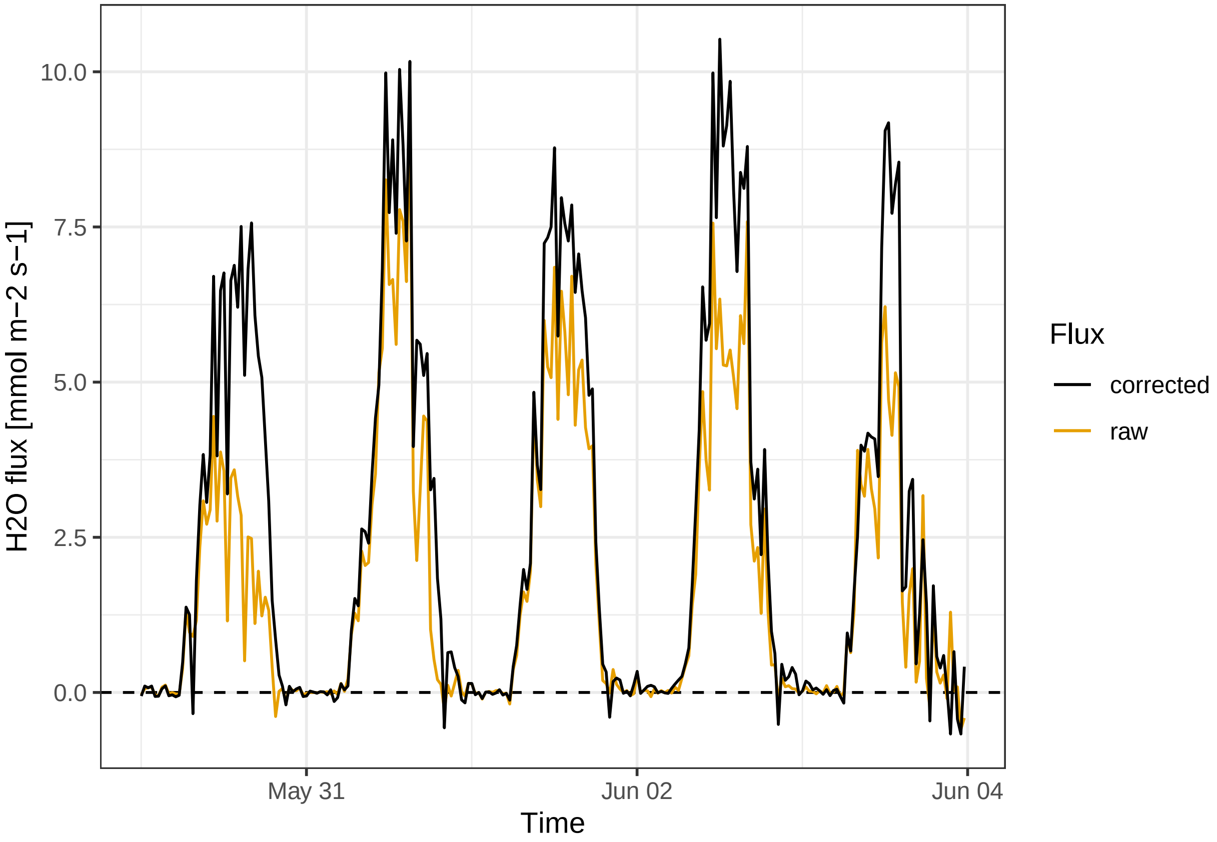

Figure 51: *H_2O* fluxes. Comparison between corrected and raw fluxes calculations. Fluxes calculated for every half an hour. Data from Hainich national park first week June 2021.

11.4.4.3 Latent heat fluxes The latent heat flux (Figure 52) is by definition has the same pattern of water vapour fluxes.

```
ggplot() +
  geom_hline(yintercept = 0, linetype=2) +
  geom_line(aes(time, lat_heat, col="raw"), data = ec_flux) +
  geom_line(aes(time, lat_heat, col="corrected"), data= ec_flux_corr) +
  scale_color_colorblind() +
  labs(x="Time", y="Latent heat flux [W m-2]", col="Flux")
```

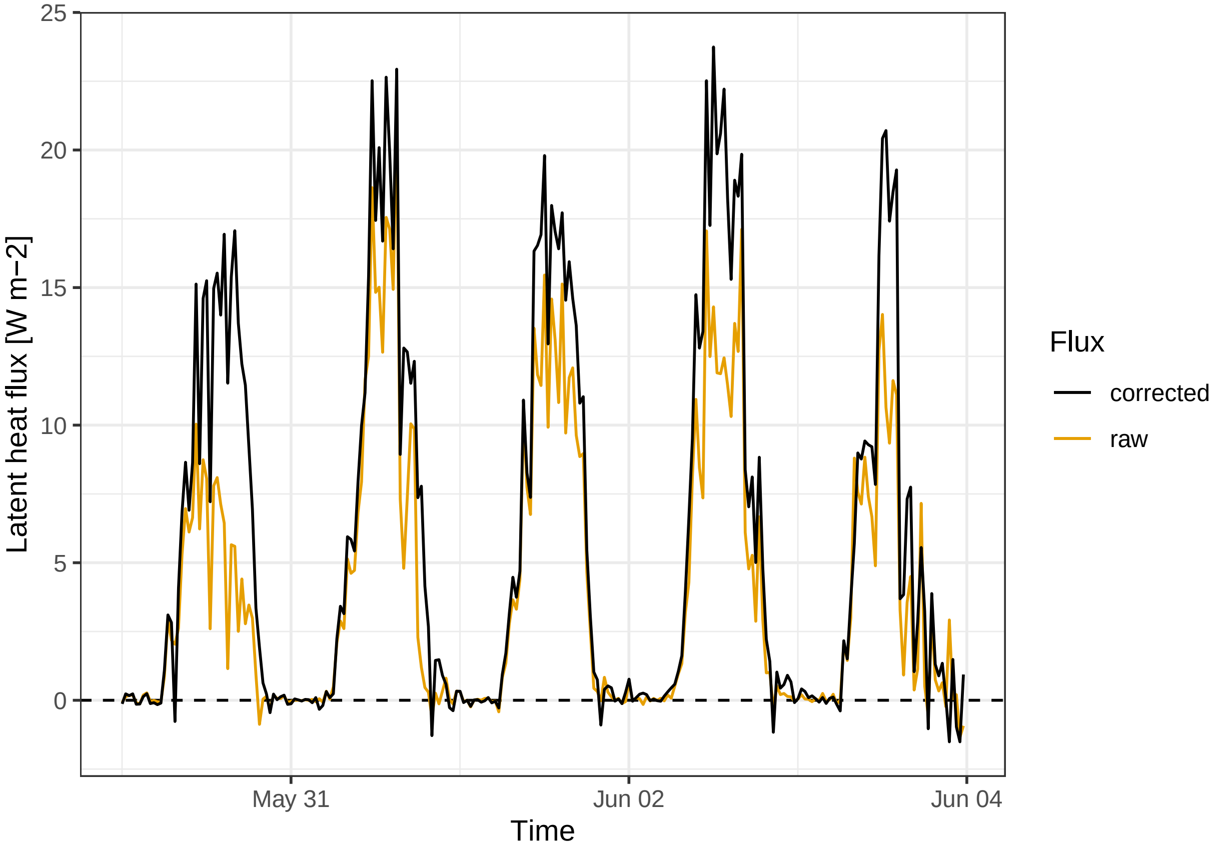

Figure 52: Latent heat fluxes. Comparison between corrected and raw fluxes calculations. Fluxes calculated for every half an hour. Data from Hainich national park first week June 2021.

11.4.4.4 Sensible heat fluxes The sensible heat flux (Figure 53) is driven the the different temperature between the surface and the atmosphere. During the day the surface heats up due to the incoming solar radiation and transfers the energy to the atmosphere. During the night the situation is often the inverse, but the amount of the flux is dependend on the weather conditions of each day.

The calculation of the sensible heat flux doesn't require a gas a analyzer as the air temperature is measured by the sonic anemometer, hence there is no time lag and the entity of the flux correction is limited.

```
ggplot() +
  geom_hline(yintercept = 0, linetype=2) +
  geom_line(aes(time, sens_heat, col="raw"), data = ec_flux) +
  geom_line(aes(time, sens_heat, col="corrected"), data= ec_flux_corr) +
  scale_color_colorblind() +
  labs(x="Time", y="Sensible heat flux [W m-2]", col="Flux")
```

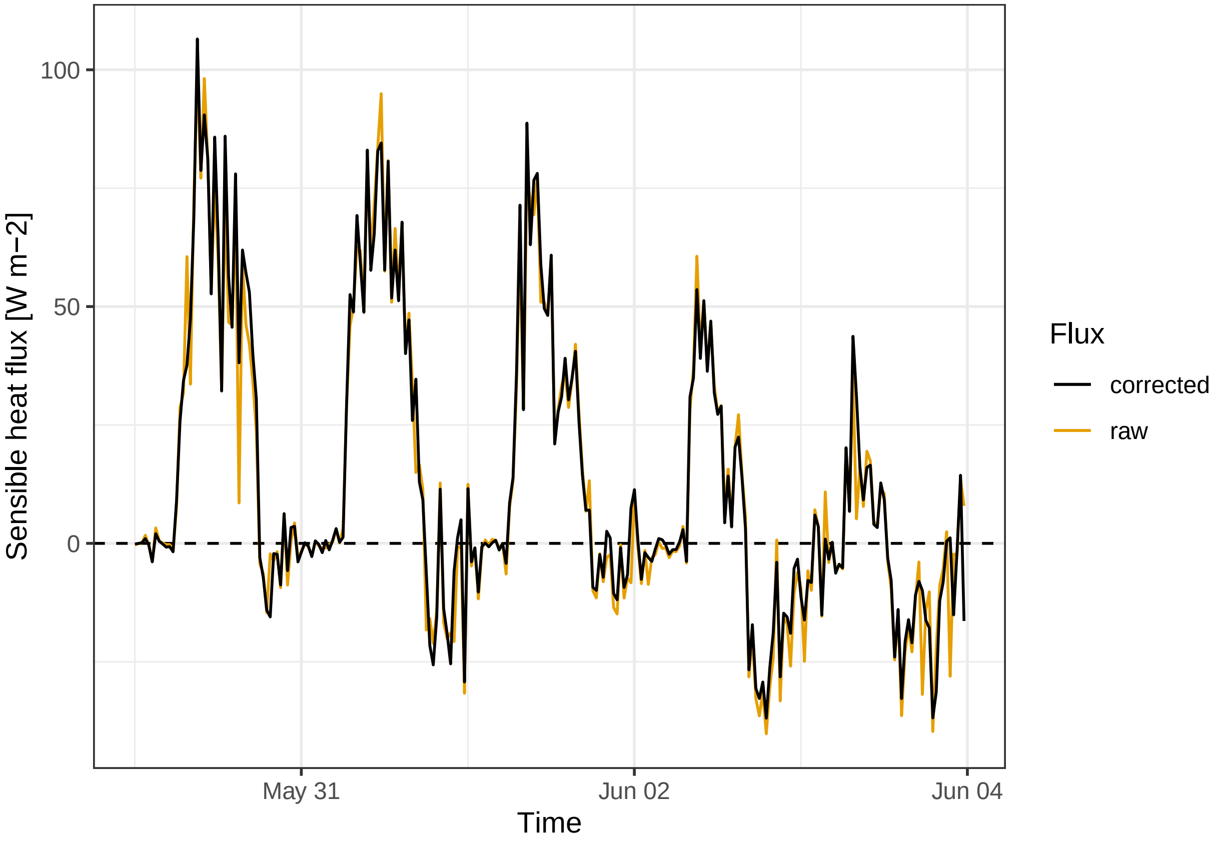

Figure 53: Sensible heat fluxes. Comparison between corrected and raw fluxes calculations. Fluxes calculated for every half an hour. Data from Hainich national park first week June 2021.

11.4.4.5 Momentum fluxes The momentum flux (Figure 54) is dependent on the wind speed, that is often higher during the afternoons and very low during the night. Moreover the flux is always positive, as expected.

The main flux correction relevant to the momentum is the coordinate rotation, as there is not time lag. The raw fluxes are often significantly different from the corrected ones.

```r
ggplot() +
  geom_line(aes(time, mom, col="raw"), data = ec_flux) +
  geom_line(aes(time, mom, col="corrected"), data= ec_flux_corr) +
  scale_color_colorblind() +
  labs(x="Time", y="Momentum flux [kg m-1 s-2]", col="Flux")
```

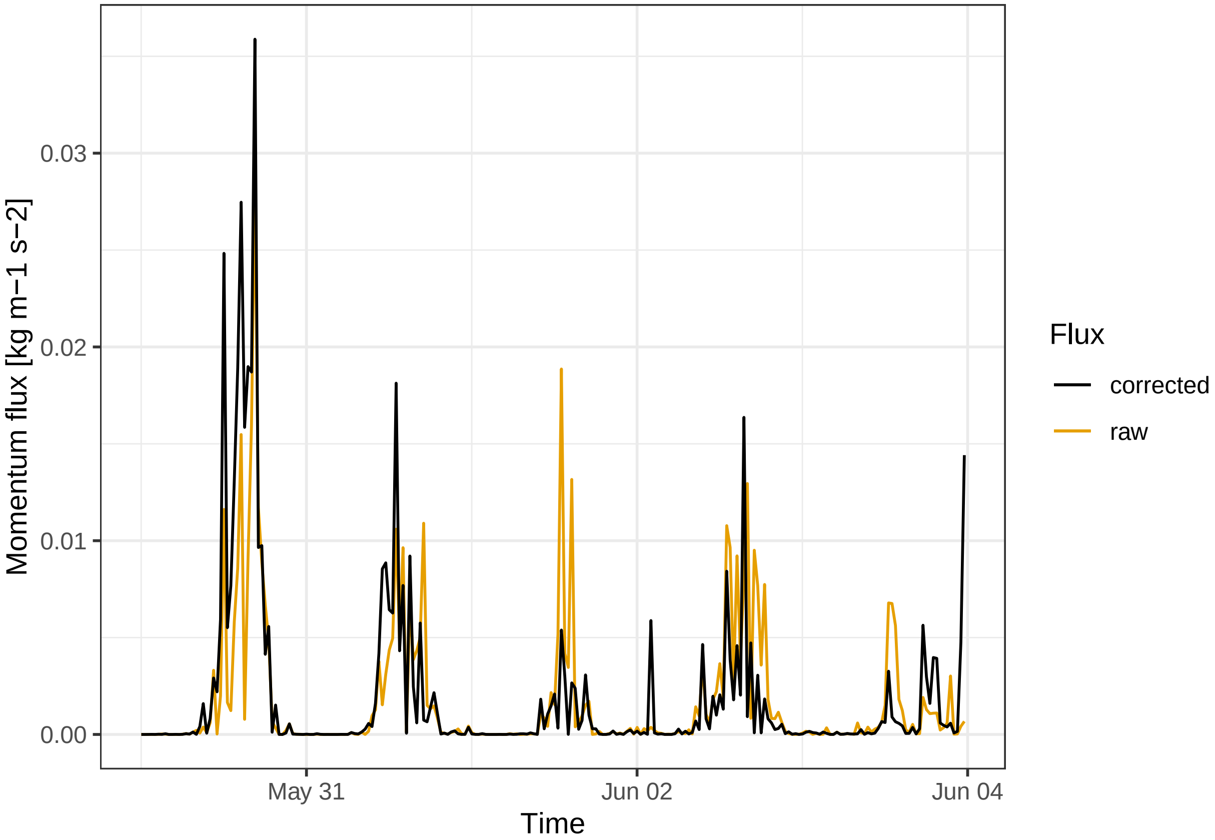

Figure 54: Momentum fluxes. Comparison between corrected and raw fluxes calculations. Fluxes calculated for every half an hour. Data from Hainich national park first week June 2021.

References

Bonan, Gordon. 2019. *Climate Change and Terrestrial Ecosystem Modeling.* Cambridge: Cambridge University Press. https://doi.org/10.1017/9781107339217.

Courtois, Elodie A., Clément Stahl, Joke Van den Berge, Laëtitia Bréchet, Leandro Van Langenhove, Andreas Richter, Ifigenia Urbina, Jennifer L. Soong, Josep Peñuelas, and Ivan A. Janssens. 2018. "Spatial Variation of Soil CO2, CH4 and N2o Fluxes Across Topographical Positions in Tropical Forests of the Guiana Shield." *Ecosystems* 21 (7): 1445–58. https://doi.org/10.1007/s10021-018-0232-6.

Grange, Stuart. 2014. "Technical Note: Averaging Wind Speeds and Directions," June. https://doi.org/10.13140/RG.2.1.3349.2006.

Harries, J. E. 1996. "The Greenhouse Earth: A View from Space." *Quarterly Journal of the Royal Meteorological Society* 122 (532): 799–818. https://doi.org/10.1002/qj.49712253202.

Labedzki, Leszek. 2011. *Evapotranspiration.* BoD – Books on Demand.

Lal, Rattan, and Manoj K. Shukla. 2004. *Principles of Soil Physics.* CRC Press.

Likens, Gene E., and 1935-. 2009. "Encyclopedia of Inland Waters." Elsevier. https://scholar.google.com/scholar_lookup?title=Encyclopedia+of+inland+waters&author=Likens%2C+Gene+E.&publication_year=2009.

Lloyd, J., and J. A. Taylor. 1994. "On the Temperature Dependence of Soil Respiration." *Functional Ecology* 8 (3): 315–23. https://doi.org/10.2307/2389824.

Meissner, Ralph, Holger Rupp, and Lisa Haselow. 2020. "Chapter 7 - Use of Lysimeters for Monitoring Soil Water Balance Parameters and Nutrient Leaching." In *Climate Change and Soil Interactions*, edited by Majeti Narasimha Vara Prasad and Marcin Pietrzykowski, 171–205. Elsevier. https://doi.org/10.1016/B978-0-12-818032-7.00007-2.

Orchard, Valerie A., and F. J. Cook. 1983. "Relationship Between Soil Respiration and Soil Moisture." *Soil Biology and Biochemistry* 15 (4): 447–53. https://doi.org/10.1016/0038-0717(83)90010-X.

Perkins, Sid. 2019. "Core Concept: Albedo Is a Simple Concept That Plays Complicated Roles in Climate and Astronomy." *Proceedings of the National Academy of Sciences* 116 (51): 25369–71. https://doi.org/10.1073/pnas.1918770116.

Rana, G, and N Katerji. 2000. "Measurement and Estimation of Actual Evapotranspiration in the Field Under Mediterranean Climate: A Review." *European Journal of Agronomy* 13 (2): 125–53. https://doi.org/10.1016/S1161-0301(00)00070-8.

Richner, Hans, Jürg Joss, and Paul Ruppert. 1996. "A Water Hypsometer Utilizing High-Precision Thermocouples." *Journal of Atmospheric and Oceanic Technology* 13 (1): 175–82. https://doi.org/10.1175/1520-0426(1996)013%3C0175:AWHUHP%3E2.0.CO;2.

Soetaert, Karline, and Thomas Petzoldt. 2010. "Inverse Modelling, Sensitivity and Monte Carlo Analysis in r Using Package FME." *Journal of Statistical Software* 33 (1): 1–28. https://doi.org/10.18637/jss.v033.i03.

Stephens, Graeme L., Martin Wild, Paul W. Stackhouse, Tristan L'Ecuyer, Seiji Kato, and David S. Henderson. 2012. "The Global Character of the Flux of Downward Longwave Radiation." *Journal of Climate* 25 (7): 2329–40. https://doi.org/10.1175/JCLI-D-11-00262.1.

Todd, Richard W, Steven R Evett, and Terry A Howell. 2000. "The Bowen Ratio-Energy Balance Method for Estimating Latent Heat Flux of Irrigated Alfalfa Evaluated in a Semi-Arid, Advective Environment." *Agricultural and Forest Meteorology* 103 (4): 335–48. https://doi.org/10.1016/S0168-1923(00)00139-8.

Yuste, J. Curiel, M. Nagy, I. A. Janssens, A. Carrara, and R. Ceulemans. 2005. "Soil Respiration in a Mixed Temperate Forest and Its Contribution to Total Ecosystem Respiration." *Tree Physiology* 25 (5): 609–19. https://doi.org/10.1093/treephys/25.5.609.