Bioclimatology report

Cheyenne Rueda and Simone Massaro

11 September 2021

1 Shortwave radiation 5
1.1 Motivation 5
1.2 Background 5
1.3 Sensors and measuring principle 7
1.4 Analysis 9
1.4.1 Albedo 11
1.4.2 Absorption coefficient 13
1.4.3 Transmission coefficient 15
1.4.4 Diffuse and direct radiation 20
2 Longwave radiation 25
2.1 Motivation 25
2.2 Background 25
2.3 Sensors and measuring principle 27
2.4 Analysis 28
2.4.1 Surface and Sky temperature 31
2.4.2 Net radiation 39
2.4.3 Change emissity in the sensor 42
3 Air temperature 45
3.1 Motivation 45
3.2 Background 45
3.3 Sensors and measuring principle 46
3.4 Analysis 47
3.4.1 Hainich time series 47
3.4.2 Temperature sensors response time 53
4 Air humidity 59
4.1 Motivation 59
4.2 Background 59
4.3 Sensors and measuring principle 59
4.4 Analysis 60
4.4.1 How and when do actual and saturation vapour pressure differ? 63

Bioclimatology report

1 Shortwave radiation

1.1 Motivation

The sun is the energy source of the radiation incoming to Earth. This radiation is classified as short-wave or long-wave radiation. Besides,short-wave radiation is considered to be more intensive than long-wave due to its higher wave frequency. Albedo is defined as the fraction of radiation being returned from the material of the surface, for example, it can be land, canopy or water. Around 30% of the light coming from the sun is reflected back to space when arriving to the atmosphere. This article describes the albedo of earth in relation with the variations in temperature of planet. Normally albedo will stay constant, but natural processes such as the active eruption of a volcano can liberate some particles such as sulfur dioxide, this will be reflected at some point on the partial cooling of earth due to more particles scattering the sun radiation (Perkins 2019). With same effect, the canopy on earth helps to maintain a good temperature enough for the welfare on ecosystems.

1.2 Background

Electromagnetic radiation is measured in joules (J). The radiant energy measured per time is a flux and its units are $J/s = W$ (watts). The energy flux liberated by an object differs with the temperature and is defined by Planck's law. Therefore, there is a direct relationship between a high temperature and a greater radiation rate.

The sun liberates short-wave radiation $(0.2-4)\mu m$. Solar radiation is composed by ultraviolet radiation with a lower value 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 broadband emissivity (ε);

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

This law was developed for a blackbody. A blackbody is described as the body that absorbs radiation at a wavelength frequency and emits the highest amount of energy capable at any wavelength as well (Bonan, 2019). Bonan (p42, 2019), defines emissivity as "*the ratio of the actual emittance to the blackbody emittance".*

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

$\sigma(\lambda)$ Is the process of acquiring a photon energy and transforming it into internal energy

$r(\lambda)$

Reflectivity is the change in direction of the electromagnetic wave.

$t(\lambda)$

Transmissivity is the fraction of energy going through a medium without experimenting any change.

In addition, shortwave is formed by 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 developed specifically for this. Following, some of them will be introduced and some of them will be used in the practicals of the course. These instruments are known as:

- **Pyrheliometer**: allows the direct measurement of shorwave 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_Ha
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_ca
  #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

Derive the albedo from the shortwave radiation components. How and why do the albedo varies over the year?

```
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)
```

```
ggplot(sw_b_d, aes(x=Date))+
  geom_line(aes(y=albedo, colour =
        'Albedo')) +
  labs(y = "Albedo", x = "Time period",
        color = "",
       caption="daily average",
        title="Albedo over year")
```

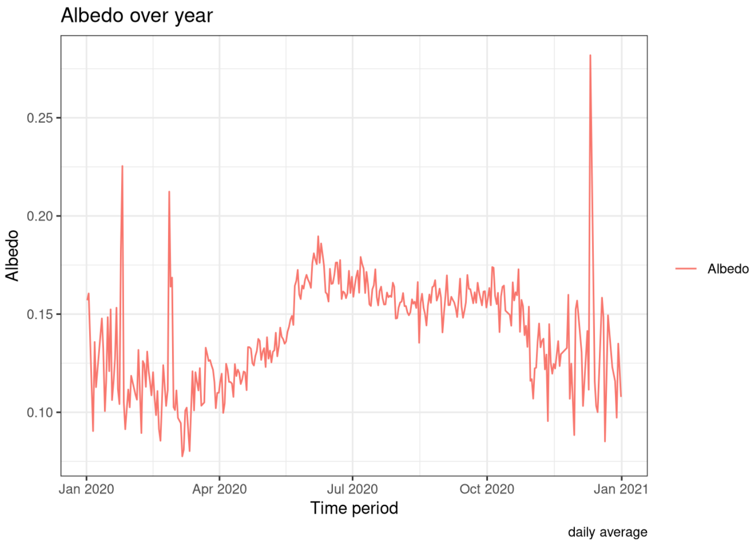


Figure 1.1: The albedo is relatively constant across the year, in the range 10-17%. During the summer the albedo is higher, due to the higher reflecantance of the green canopy compared to the dark ground. During the winter there are some peaks in the albedo and can be explained by measurement errors due to the low amount of radiation.

1.4.2 Absorption coefficient

How large is the absorption coefficient in the shortwave radiation range? How does the absorption coefficient varies with time and what could a high and low absorption coefficient indicate for a plant canopy?

```
ggplot(sw_b_d, aes(x=Date))+
  geom_line(aes(y=absorbed, colour =
        'Absorbed'))+
  labs(y = "Fraction absorbed", x = "Time
        period", color = "",
      caption="daily average",
        title="Absorption coefficient") +
  ylim(0.5,1)
```

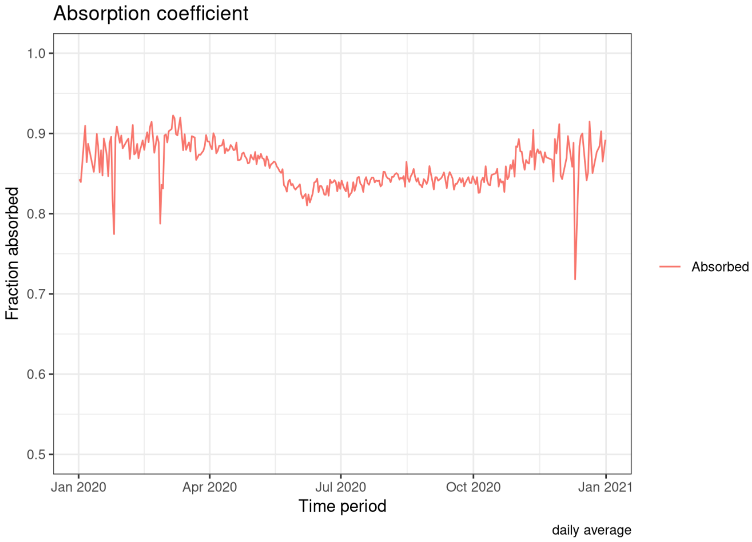


Figure 1.2: The absorption is quite constant, but during winter is higher (up to 90%) to then decrease when there are leaves (83%). The peaks during the winter can be connected to measurement errors due to the low amount of radiation.

1.4.3 Transmission coefficient

Derive the transmission coefficient of the forest. How does the transmission coefficient vary throughout the day and the year? Discuss potential reasons.

```
ggplot(sw_b_d, aes(x=Date))+
  geom_line(aes(y=transmitted, colour =
        'Trasmission coef'))+
  labs(y = "Trasmission coef", x = "Time
        period", color = "",
       caption="daily average",
        title="Tramission coefficient over
        year") +
  ylim(0,.5)
```

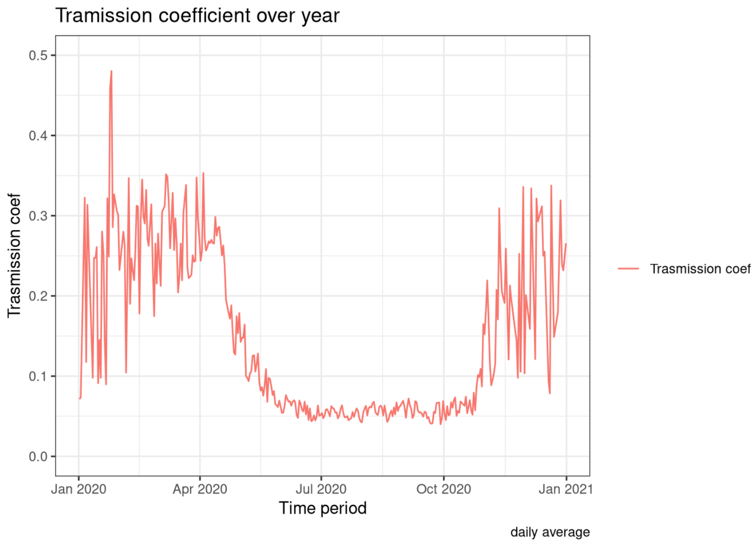


Figure 1.3: During the year there is a clear difference between summer and winter. During winter the transmission is around 30% while during summer it is around 5%. This shows how effective are the trees in capturing available light.

```
ggplot(sw_b_d, aes(x=Date))+
  geom_line(aes(y=belowcanopy))+
  labs(y = "Transmitted radiation
        (W/m2)", x = "Time period",
       title="Radiation transmitted below
        the canopy")
```

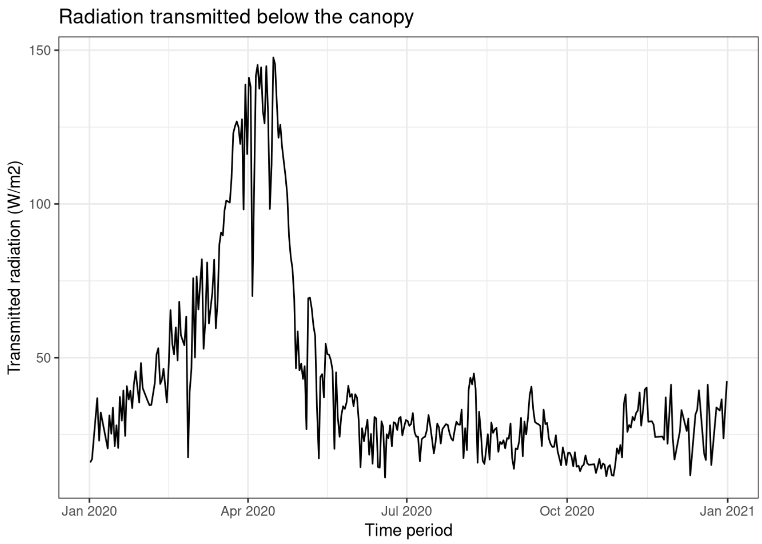


Figure 1.4: The absolute value of the radiation below the canopy is relatively constant between summer and winter with the notable exception of spring. There are no leaves in the canopy yet, but the solar radiation is getting stronger. This can explain the why in beech forest there is a some undergrowth only during spring.

```
sw_b %>%
  filter(between(Date,
         as_datetime("2020-07-1"),
         as_datetime("2020-07-2"))) %>%
ggplot(aes(x=Date))+
  geom_line(aes(y=transmitted, colour =
         'Trasmission coef'))+
  labs(y = "Trasmission coef", x = "Time
         period", color = "",
       title="Tramission coefficient over a
         summer day")
```

Figure 1.5: During summer days there is a daily cycle of the transmission coefficient. 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.

1.4.4 Diffuse and direct radiation

Calculate the direct solar radiation and plot global, diffuse and direct solar radiation. Compare a clear sky and a cloudy day. What's the difference and why?

```
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)
        %>%
  ggplot(aes(x=Date, y=rad, color=type))
        +
    geom_line() +
  scale_color_discrete(labels=c("direct",
        "diffuse", "total")) +
  labs(y="Radiation (W/m2)", x="Time
        period (weekly)",
       caption="daily average",
        title="Shortwave radiation
        components")
```

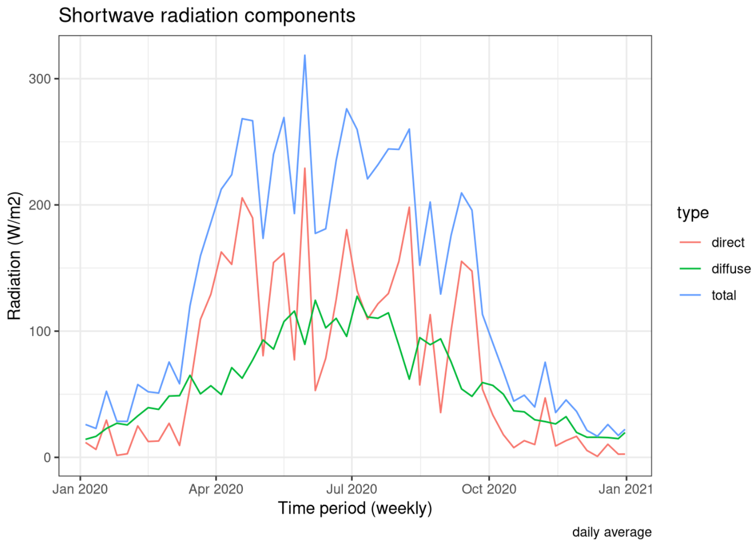


Figure 1.6: The total radiation is sum of the diffuse and direct. The amount of direct radiation changes a lot during the year. You can see that April and September were relatively sunny months (direct and total radiation are similar), while July and August were cloudy (big difference between direct and total) The diffuse radiation has a its own pattern across the year.

```
shortwavedata%>%
  filter(week(Date)==28) %>%
  gather("type", "rad",
         Shortwave_incoming_W_m.2 ,
         direct)  %>%
  ggplot(aes(x=Date, y=rad, color=type))
         +
    geom_line() +
  scale_color_discrete(labels=c("direct",
         "total")) +
  labs(y="Radiation (W/m2)", x="Time
         period (weekly)",
       title="Diffuse and direction
         radiation in a week")
```

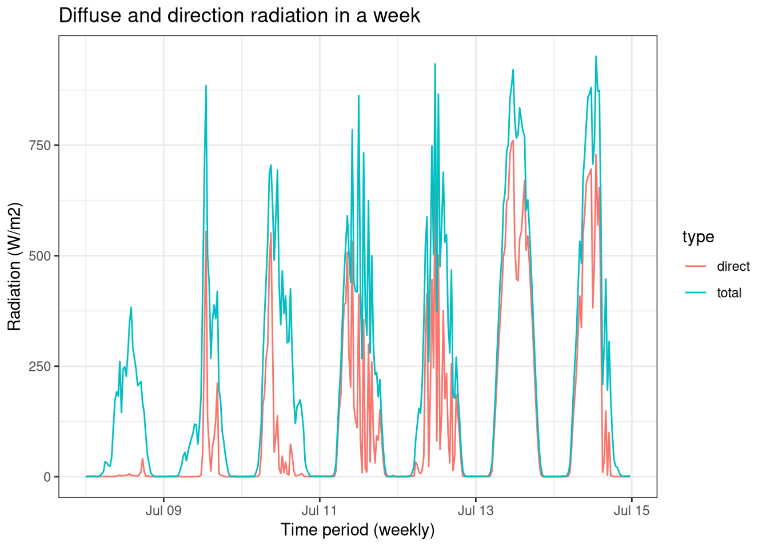


Figure 1.7: This is a summer week from July, where it is possible to clearly see the difference between cloudy and sunny day. The 8th of July (first day) was cloudy with virtually no direct radiation, while the 13th of July was very sunny with almost all radiation direct.

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

The incoming longwave radiation depends on the temperature of the sky and it is partly abosrbed by the ecosystems and partly reflected 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 ecosystem emits longwave radiation depending on the 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.

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

[1].

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

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

```
# 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)
```

```r
# 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

Derive the sky and surface temperature from the longwave radiation components. How do they differ and why? Discuss! During which periods of the year sky and surface temperature differ the most and the less?

The surface and the sky temperature are plotted for one year using a weekly (Figure 2.1) and daily aggregation (Figure 2.2).

```
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")
```

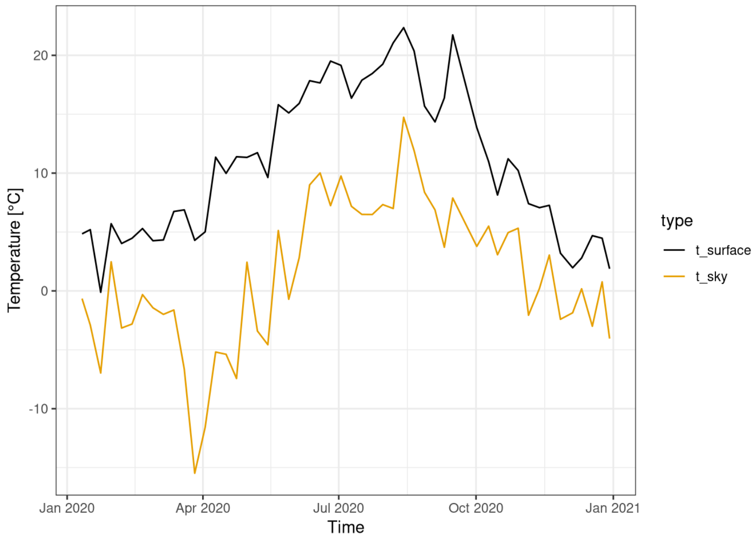


Figure 2.1: Weekly average of sky and surface temperatures over one year. Data from botanical garden 2020.

```
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")
```

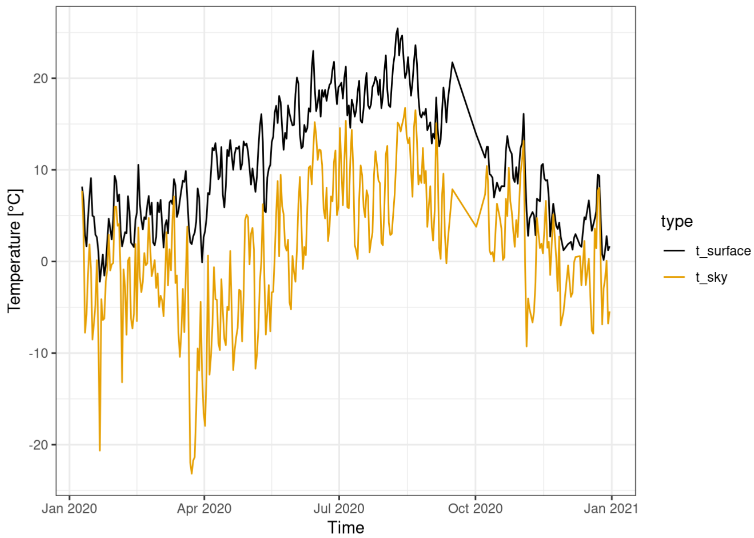


Figure 2.2: Daily average of sky and surface temperatures over one year. Data from botanical garden 2020.

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.

The difference between the sky at the surface temperature is also analyzed using high frequency data (10 minutes) for a month (Figure 2.3) and a week (Figure 2.4).

```
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")
```

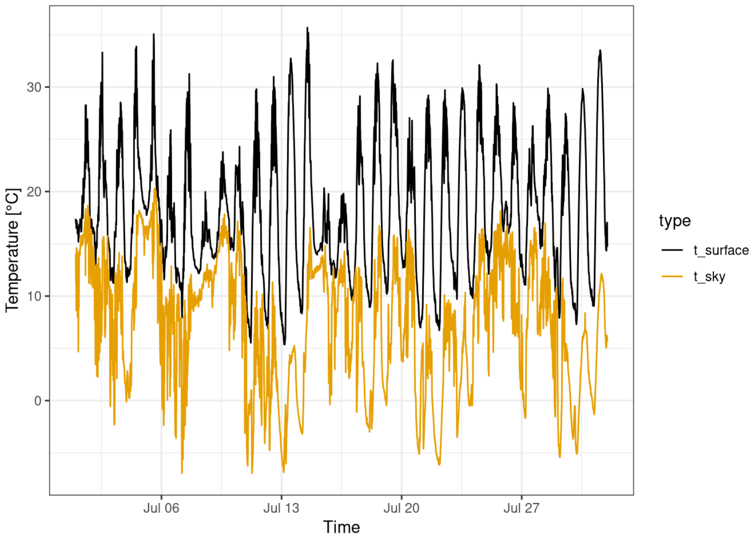


Figure 2.3: Sky and surface temperatures during July 2020. Data frequency 10 minutes. Data from botanical garden.

```
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")
```

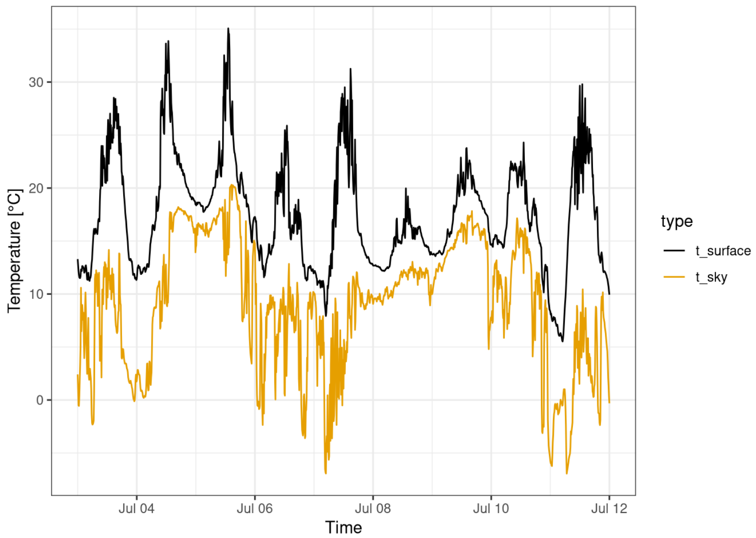


Figure 2.4: Sky and surface temperatures during first weeek of July 2020. Data frequency 10 minutes. Data from botanical garden.

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.

2.4.2 Net radiation

Calculate the net radiation over the meadow in the forest botanical garden and plot the four components. How do the four components change over the season and why? Which unexpected results you found? Discuss!

The longwave and shortwave components are merged to calculate the overall net radiation for one year in figure 2.5.

```
rad_w %>%
  gather(key="type", value="radiation",
         lw_in, lw_out, 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")
```

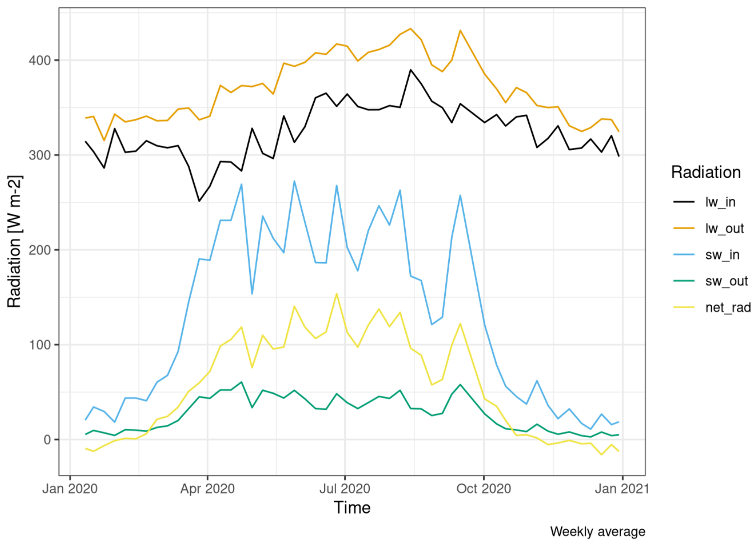


Figure 2.5: Net radiation over the year. The four components (shortwave incoming, shortwave outgoing, longwave incoming, longwave outgoing) are showed. Data averaged over a week. Data from botanical garden 2020.

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.

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 $19^{\circ}C$ 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]")
```

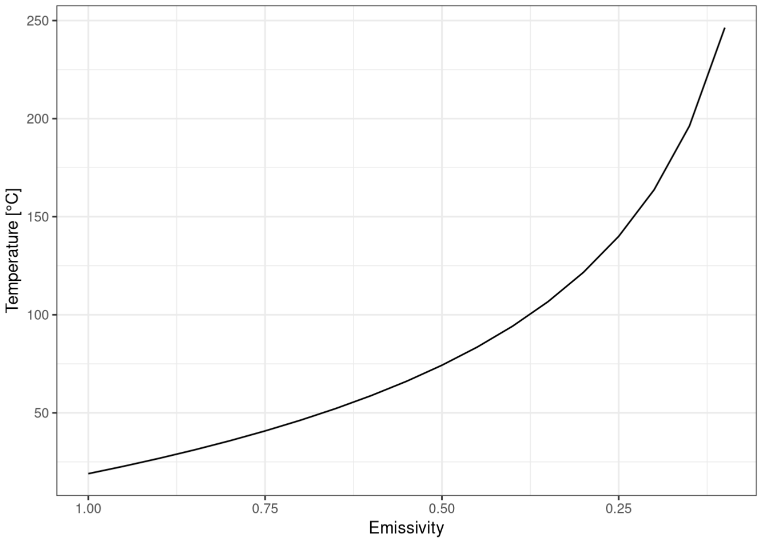


Figure 2.6: Estimanted temperature measured by infrared themometer for different emissivity. Temperature at emissivity 1 is 19 °C

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 mantain 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 ($^\circ$C) and Fahrenheit ($^\circ$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 expasion of materials. The increase in temperature results in an increase of volume or lenght 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 luquid) 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

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

between_dates <- function(x, start, end){
  between(x,
          parse_date_time(start, c("dm",
         "dmy")),
          parse_date_time(end, c("dm",
         "dmy")))
}
```

3.4.1 Hainich time series

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 3.1 and 3.2). 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 3.1). Moreover also the yearly mean are similar.

In general the air and the soil temperature follow a similar pattern, but the daily oscillation in the air temperature are bigger (Figure 3.3).

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

```
##
## ── Column specification
```

```
## cols(
##   Date = col_datetime(format = ""),
##   TA_44m = col_double(),
##   TA_40m = col_double(),
##   TA_32m = col_double(),
##   TA_20m = col_double(),
##   TA_10m = col_double(),
##   TA_2m = col_double(),
##   Tsoil_002m_degC = col_double()
## )
```

```
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")
```

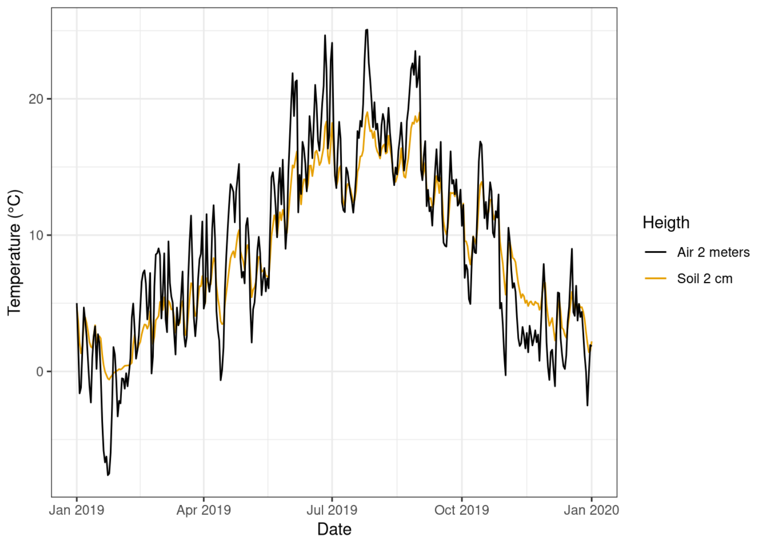


Figure 3.1: Comparison time series temperature at 2 m and 2 cm in the soil for 2019. Measurement averaged over 1 day. Data from Hainich national park

```
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")
```

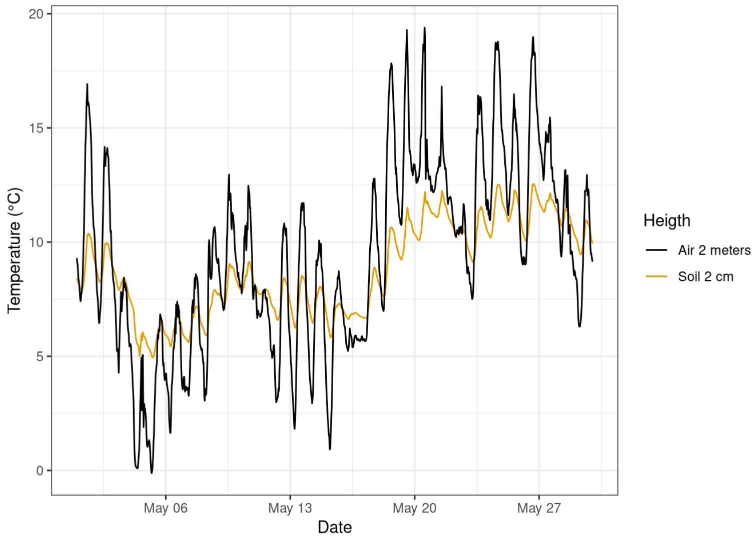


Figure 3.2: Comparison time series temperature at 2 m and 2 cm in the soil for the month of May 2019. measurement frequency 30 min. Data from Hainich national park

```
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)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

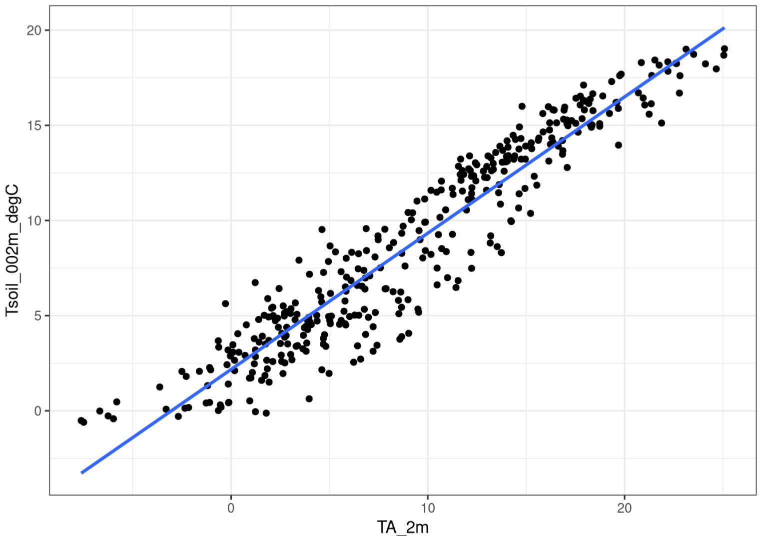


Figure 3.3: Scatter plot with regression line between temperature at 2 m and 2 cm in the soil for 2019. measurement averaged over 1 day. Data from Hainich national park

```
cor(temp$TA_2m, temp$Tsoil_002m_degC)
```

```
## [1] 0.9287618
```

```
cat(" Mean Air: ", mean(temp$TA_2m),
          "\n",
     "Mean soil: ",
          mean(temp$Tsoil_002m_degC))
```

```
##  Mean Air:  9.132301
```

```
## Mean soil: 8.718412
```

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.

3.4.2.1 Resistance sensor

```
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_se
        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) )
```

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

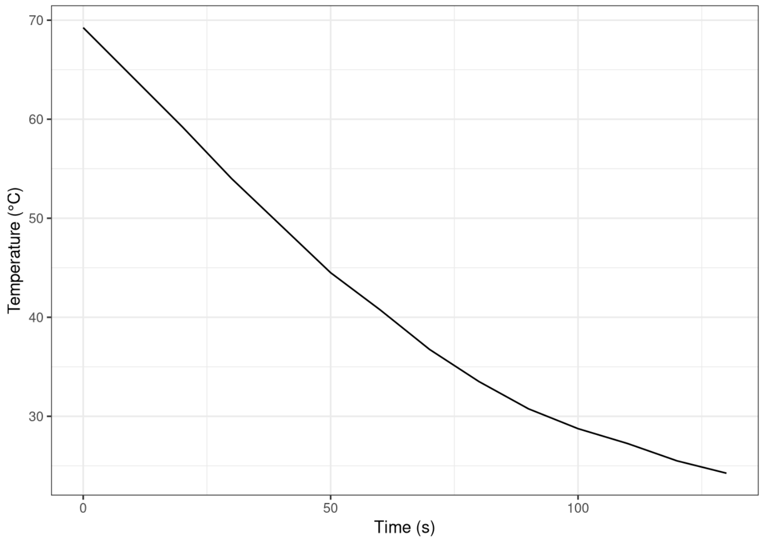


Figure 3.4: Temperature decrease over time for resistance thermometer

```
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))
```

```
##
## Call:
## lm(formula = time ~ log_t, data =
```

```
temp_res)
##
## Coefficients:
## (Intercept)        log_t
##       3.644       77.279
```

The time response rate for the resistance sensor is 4 seconds.

3.4.2.2 Mercury termometer

```
temp_merc <-
        read.csv(here("3_air_temperature/mercury_termc
        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)) )
```

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

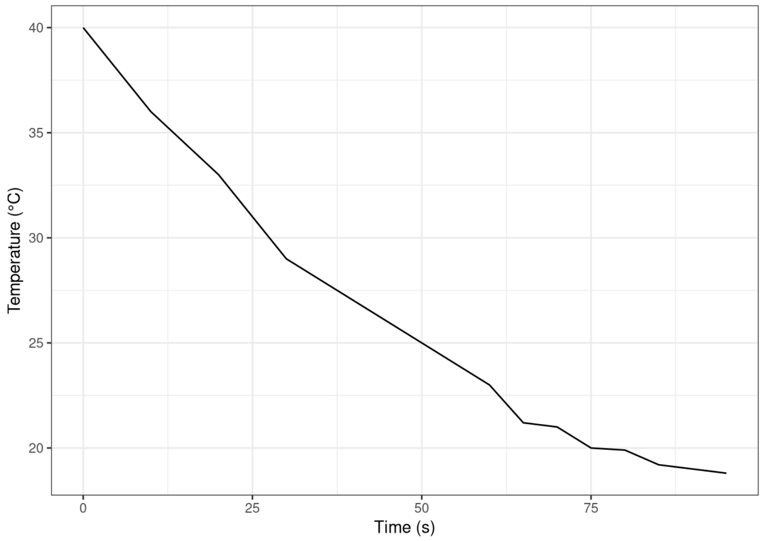


Figure 3.5: Temperature decrease over time for mercury thermometer

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

```
##
## Call:
## lm(formula = time ~ log_t, data =
temp_merc)
##
## Coefficients:
## (Intercept)        log_t
##      -2.791       87.550
```

The time response rate for the mercury sensor is 4 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

4.2 Background

Actual vapor pressure

Saturation vapor pressure

Dewpoint temperature

Absolute humidity

Specific humidity

Mixing ratio Vapor pressure deficit Equivalent temperature

- Saturation vapour pressure [hPa]

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

- Dewpoint temperature [ºC]

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

4.3 Sensors and measuring

principle

4.4 Analysis

The following

1. Actual vapor pressure
2. Saturation vapor pressure
3. Dewpoint temperature
4. Absolute humidity
5. Specific humidity
6. Mixing ratio
7. Vapor 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 humidty
        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)
```

```
head(hum)
```

```
## # A tibble: 6 x 12
##   datetime               ta    rh
pa    es    ea    td abs_hum spec_hum
mix_ratio p_def  t_eq
##   <dttm>              <dbl> <dbl>
<dbl> <dbl> <dbl> <dbl>   <dbl>    <dbl>
<dbl> <dbl> <dbl>
## 1 2018-01-01 00:00:00  9.26  68.5  943
```

```
11.7  8.01  3.64    6.15     5.30
5.33  3.68  22.6
## 2 2018-01-01 00:30:00  9.04  70.4  943
11.5  8.11  3.81    6.23     5.37
5.40  3.42  22.5
## 3 2018-01-01 01:00:00  8.55  77.3  943
11.1  8.61  4.66    6.62     5.70
5.73  2.54  22.9
## 4 2018-01-01 01:30:00  9.01  74.0  943
11.5  8.50  4.48    6.53     5.63
5.66  2.99  23.2
## 5 2018-01-01 02:00:00  8.74  76.5  943
11.3  8.64  4.71    6.64     5.72
5.75  2.66  23.1
## 6 2018-01-01 02:30:00  7.98  84.8
944.  10.7  9.09  5.44    7.01     6.02
6.05  1.63  23.1
```

4.4.1 How and when do actual and saturation vapour pressure differ?

```
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"))
```

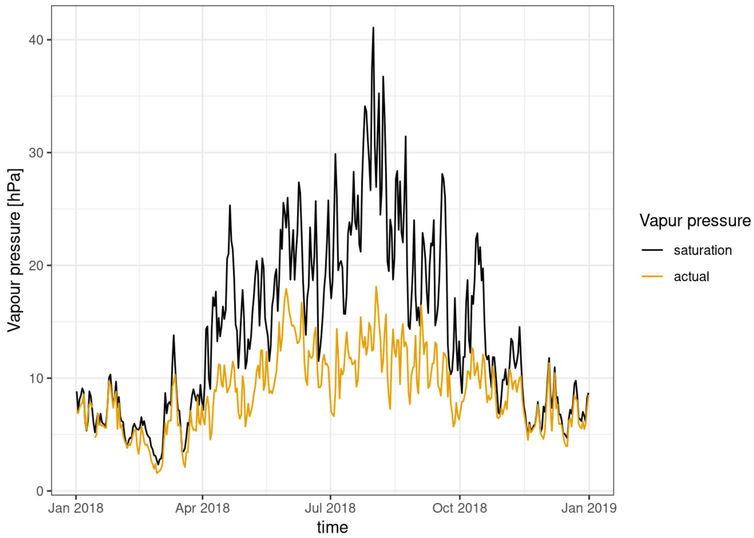


Figure 4.1: Comparison between saturation vapour pressure and actual vapour pressure for one year. Data averaged on a day. Data from Hainich national park 2018.

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

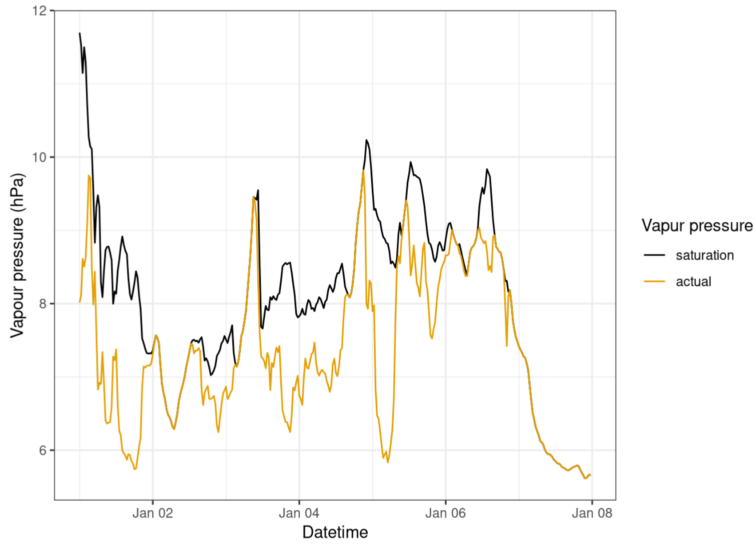


Figure 4.2: Comparison between saturation vapour pressure and actual vapour pressure for the first week of teh year. Data averaged over 30 minutes. Data from Hainich national park 2018.

During the year, the biggest differences between actual and saturation vapour pressure can be found after summer period, normally due to the first incoming rains with the beginning of autumn.

4.4.2 Compare air and dew point temperature, how do they differ and how are they related to relative humidity?

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

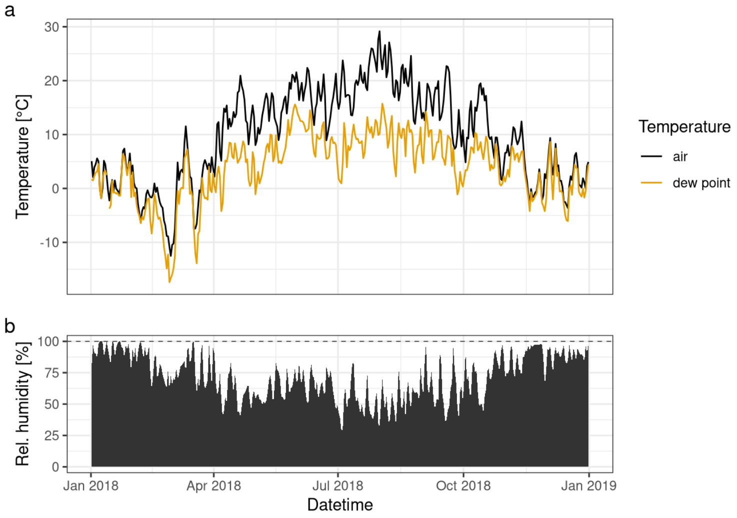


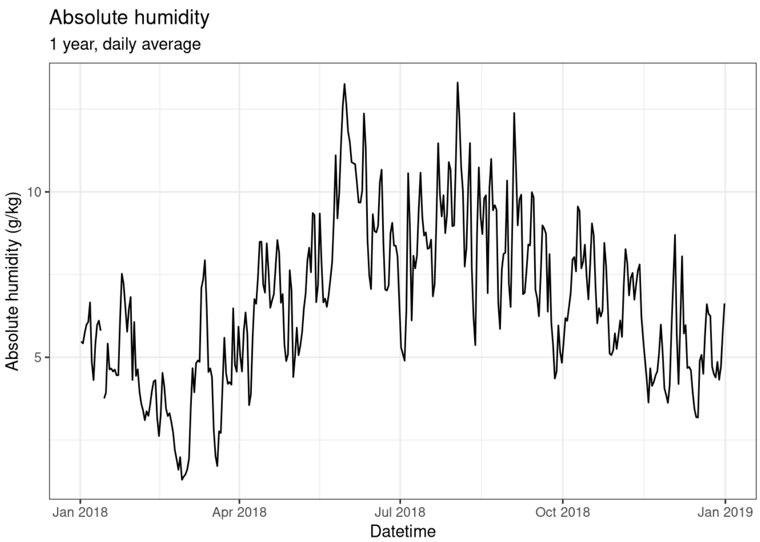
Figure 4.3: Comparison of air temperature with dew point temperature (fig. a), showed in relation with the relative humidity (fig. b). Data averaged over one day. Data from Hainich national park 2018.

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 beggining and end of the year more similarities can be seen.

4.4.3 When is absolute humidity largest and why?

```
hum_d %>%
```

```
  ggplot(aes(datetime, abs_hum))+
    geom_line() +
  labs(title="Absolute humidity",
         x="Datetime", y="Absolute
         humidity (g/kg)", subtitle = "1
         year, daily average")
```

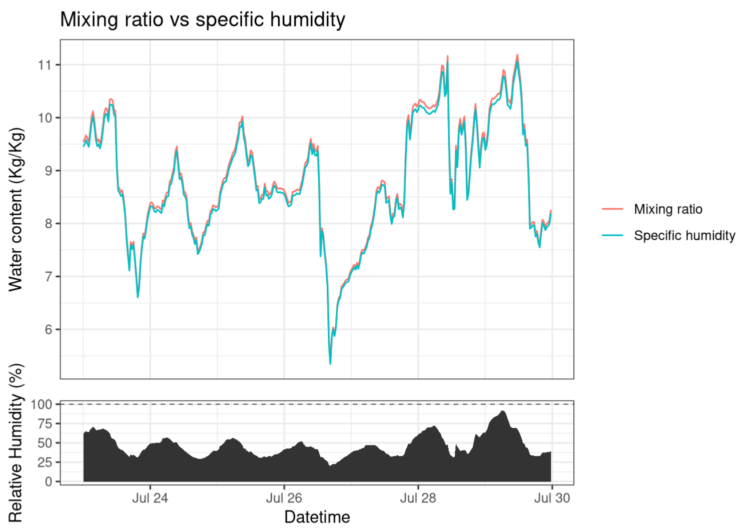


(#fig:abs_hum)Absolute humidity over the year. Data averaged over one day. Data from Hainich national park 2018.

4.4.4 Compare mixing ratio and specific humidity, how and why do they differ?

```
( 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="", title="Mixing ratio vs
       specific humidity") +
  scale_color_discrete(labels=c("Mixing
       ratio", "Specific humidity")) +
  remove_x_axis()) +          (hum %>%
filter(week(datetime) ==30) %>%
ggplot(aes(datetime, rh))+
  geom_area() +
geom_hline(yintercept = 100,
       linetype="dashed", size=.2) +
labs(y="Relative Humidity (%)",
       x="Datetime") ) +
         plot_layout(heights = c(4, 1))
```



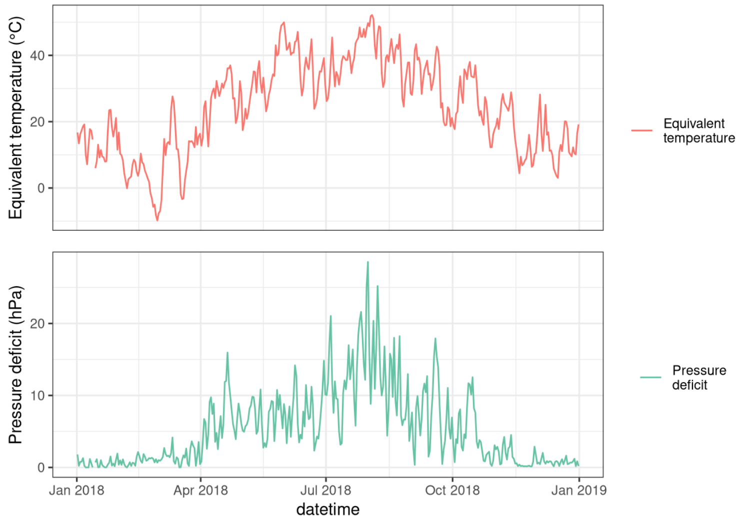
```r
cor(hum$mix_ratio-hum$spec_hum, hum$rh,
        "pairwise.complete.obs")
```

```
## [1] -0.1639707
```

In this case they do not differ to much?

4.4.5 When is the vapour pressure deficit and equivalent temperature largest?

```r
(hum_d %>%
  ggplot(aes(datetime, t_eq,
        colour="Equivalent\ntemperature"))+
    geom_line() +
    labs(y="Equivalent temperature (°C)",
        colour="")  +
    remove_x_axis()
) /
(hum_d %>%
   ggplot(aes(datetime, p_def,
        colour="Pressure\ndeficit")) +
   geom_line() +
   scale_colour_brewer(palette = "Set2")
        +
   labs(y="Pressure deficit (hPa)",
        colour="")
 )
```



During summer is the largest vapour pressure deficit and equivalent temperature values.

4.4.6 Field data

```
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)

                 )        spin
```

```
## # A tibble: 1 x 13
##   t_dry t_wet     p es_wet    es    ea
td    rh abs_hum spec_hum mix_ratio p_def
t_eq
##   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl>
<dbl> <dbl>   <dbl>    <dbl>     <dbl>
<dbl> <dbl>
## 1    16  10.8  977.   13.0  18.2  9.69
6.37  53.2    7.27     6.19      6.23
8.52  31.6
```

```
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)

                  )assman1
```

```
## # A tibble: 1 x 13
##   t_dry t_wet     p es_wet    es    ea
td    rh abs_hum spec_hum mix_ratio p_def
t_eq
##   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl>
<dbl> <dbl>   <dbl>    <dbl>     <dbl>
<dbl> <dbl>
## 1  15.4  11.8  977.   13.9  17.5  11.6
8.98  66.2    8.71     7.41      7.47
5.93  34.1
```

```
assman1 <- tibble(
  t_dry = 16.6,
  t_wet = 11.0,
  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)

                  )assman1
```

```
## # A tibble: 1 x 13
##   t_dry t_wet     p es_wet    es    ea
td    rh abs_hum spec_hum mix_ratio p_def
t_eq
##   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl>
<dbl> <dbl>   <dbl>    <dbl>     <dbl>
<dbl> <dbl>
## 1  16.6    11  977.   13.1  18.9  9.62
6.25  50.8    7.19     6.14      6.18
9.30  32.1
```

5 Precipitation

5.1 Motivation

Water is needed to plants for photosynthesis and transpiration.

5.2 Background

The precipitation is measured in millimeter (mm)

5.3 Sensors and measuring principle

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.cs
        %>%
  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")
```

5.5 ET and precipitation at different latitudes

Visualise the annual sums of evapotranspiration and precipitation according to its geographic latitude (boxplot?). How and why does precipitation and evapotranspiration varies? Which impact has the underlying ecosystem?

```
# this t makes no sense as name TODO
        chanve this
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"))
```

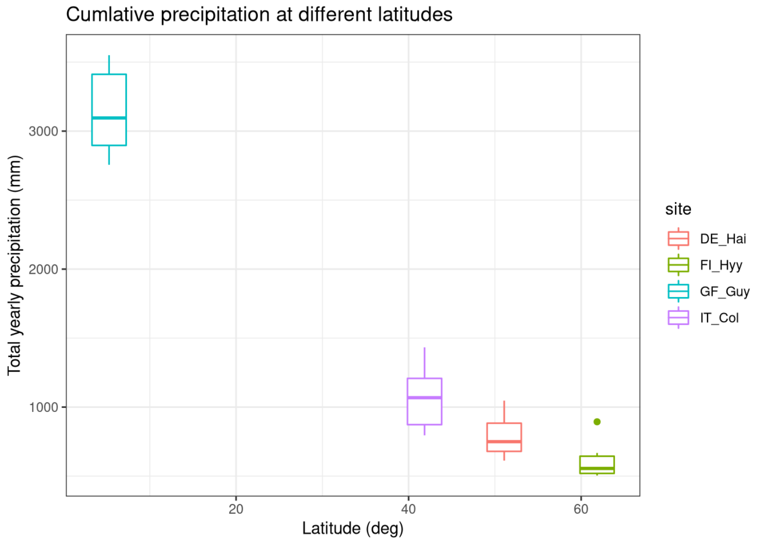
```
sites <- sites %>%
  inner_join(prec_et, by="site")
```

```
sites
```

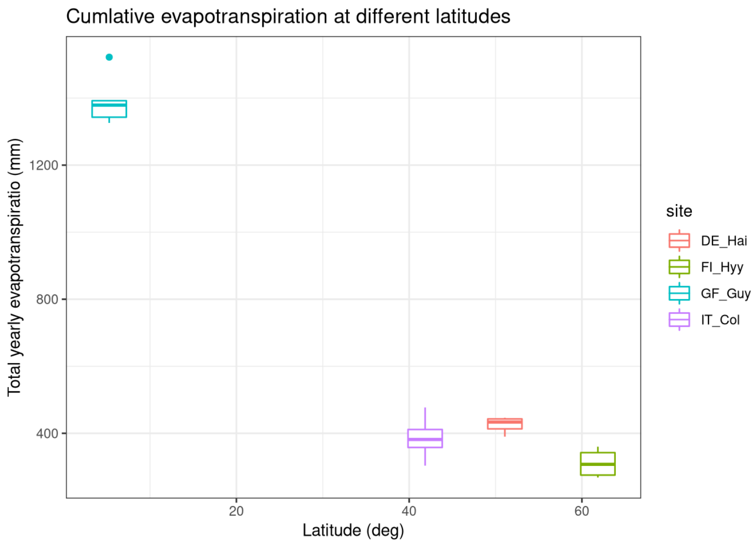
```
## # A tibble: 24 x 11
##    site   full_name   Lat  Long Height
Veg_type Mean_airT Mean_precipitation
Date         prec    et
##    <chr>  <chr>     <dbl> <dbl>  <dbl>
<chr>        <dbl>              <dbl>
<date>     <dbl> <dbl>
##  1 DE_Hai Hainich    51.1  10.5    430
DBF              8.3                720
2004-01-01  763   390.
##  2 DE_Hai Hainich    51.1  10.5    430
DBF              8.3                720
2005-01-01  736   446.
##  3 DE_Hai Hainich    51.1  10.5    430
DBF              8.3                720
2006-01-01  660.  432.
##  4 DE_Hai Hainich    51.1  10.5    430
DBF              8.3                720
2007-01-01 1047.  446.
##  5 DE_Hai Hainich    51.1  10.5    430
DBF              8.3                720
2008-01-01  612.  407.
##  6 DE_Hai Hainich    51.1  10.5    430
DBF              8.3                720
2009-01-01  924.  435.
##  7 FI_Hyy Hyytiala   61.8  24.3    181
ENF              3.8                709
```

```
2004-01-01  514.  273.
##  8 FI_Hyy Hyytiala   61.8  24.3    181
ENF             3.8                709
2005-01-01  668.  268.
##  9 FI_Hyy Hyytiala   61.8  24.3    181
ENF             3.8                709
2006-01-01  504.  281.
## 10 FI_Hyy Hyytiala   61.8  24.3    181
ENF             3.8                709
2007-01-01  538.  345.
## # … with 14 more rows
```

```
(prec_box <-  ggplot(sites, aes(Lat,
        prec, colour=site)) +
  geom_boxplot() +
   labs(title="Cumlative precipitation at
        different latitudes",
        y="Total yearly precipitation
        (mm)",
        x="Latitude (deg)"))
```




```
(et_box <- ggplot(sites, aes(Lat, et,
         colour=site)) +
  geom_boxplot() +
    labs(title="Cumlative
         evapotranspiration at different
         latitudes",
         y="Total yearly evapotranspiratio
         (mm)",
         x="Latitude (deg)"))
```



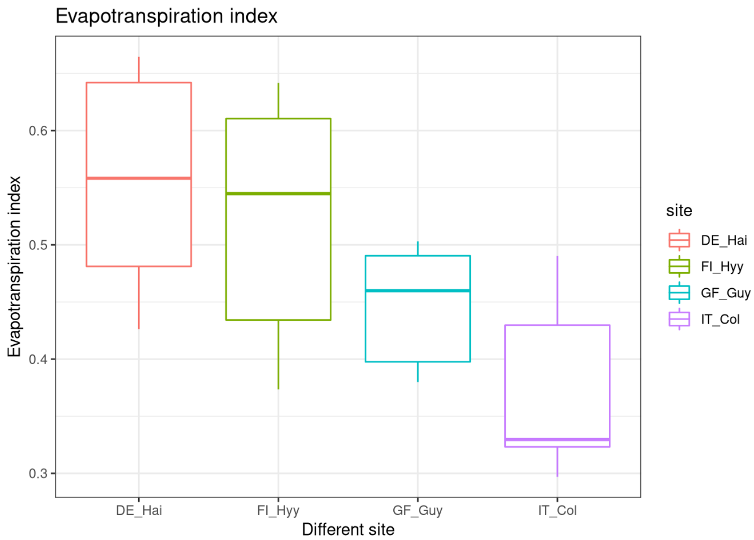
Evapotranspiration index

Precipitation is used as a driver/source for evapotranspiration. The ratio between evapotranspiration and precipitation (evapotranspiration index = ET_idx) is a very useful indicator how precipitation is used by plants during photosynthesis. Calculate the ET_idx for all sites and visualise it. What surprised you and how do the ET_idx differs across ecosystems? Discuss.

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

ggplot(sites,
       aes(site, et_idx, colour = site))+
         geom_boxplot()+
```

```
          labs(title="Evapotranspiration
         index", x="Different site",
         y="Evapotranspiration index")
```



As we can see,evapotranspiration index is the fraction between precipitation and what is coming back to atmosphere through evapotranspiration.

what is intercepted by cannopy and what is being lose by soil hydrology processes. Other variables such as wind speed, surface humidity and solar radiation are not being taken into account in the calculation of this index, which probably are the ones causing differences on what it could have been expected at the beginning for some of the places.

we can see a higher evapotranspiration index in places where there is plenty precipitation and evapotranspiration rate by canopy. Hainich and Hyytiala are represented as the higher, this can be due to their higher latitudes providing this ecosystems with more precipitation rate. Guyaflux is at lower latitude and thus more close to the equator. This means a more tropical weather where high amount of rain and sun is normally occurring. Collelongo is place in a drier climate with high radiation but less annual precipitation rates.

6 Air Pressure

6.1 Motivation

Air pressure is commonly known by experts as a factor that is constantly regulating the characteristics of natural systems. Changes in cloud densities, clouds movement and humid conditions of air. All these are examples of how air dynamic alters natural components of earth. But, what is the reason that explains the movement of air pressure? The molecules floating in the air have different directional movements or for instance, changes in its momentum. Following the same process as water particles when heated, air molecules move faster when air is hot. Air pressure is as well strongly related with the weather humidity conditions. Air is fluid. The molecules can easily move in all directions (Bonan 2019).

Theory explains the different layers the atmosphere is composed by. Each of these layers presents air in different amounts. The air of the atmosphere exerts pressure through all of them and this is known as *air pressure* or *atmospheric pressure*, which makes force towards the center of the planet from all directions, to all the earth component. More concretely, the troposphere (the coolest layer to earth surface) has the higher amount of air.

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 a 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: 760 mmHg = 760 torr = 1013.25 hPa.

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

$T_{boil} = 100 + 2.804 \times 10^{-2}(p - 1013.25hPa) - 1.384 \times 10^{-5}(p - 101$

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

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

```r
#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)))
}
```

6.5 Air pressure sea level

Calculate the air pressure at sea level for the Leinefelde site.

```
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_discrete(name="Location",
        labels = c("Sea level",
        "Leinefelde"))
```

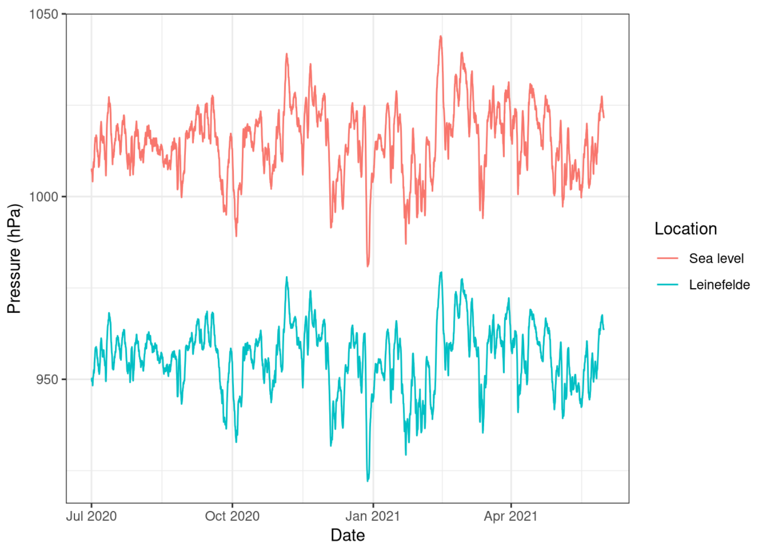


Figure 6.1: Air pressure at Leinefelde and estimated air pressure at sea level. Pressure has been corrected for air humidity. Data from Leinefelde flux tower (451 m + 41m tower) July 2020 - May 2021, 30 min frequency.

The air pressure at sea level is always higher than at Leinefelde (Figure 6.1) 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.

6.6 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?

```
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

```
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 $99.38\ ^{\circ}\mathrm{C}$ at the North Campus and $96.14\ ^{\circ}\mathrm{C}$ 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.

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

7 Wind

7.1 Motivation

Wind is a very important factor of the bioclimatology on earth. It allows a high number of plants in the dispersal of seeds and pollen, meaning an important role on plants reproduction. It also helps animals and insects to travel for longer distance around the globe. Some interesting aspects to learn about wind is its speed, energy and direction at a concrete point. Regarding wind speed, it can be described using different terms that variate upon the strenght it has. For example, it can be a smooth breeze, a strom or a hurricane in special cases. In case of hurricanes, it is normally a devastating disturbance that can reach a very destructive power on earth ecosystems. In terms of temperature, wind can be distinguish by its closeness to the poles or the equator. Wind also cause erosion of rocks, mountains and soils.

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 horizonal plan and the direction.

The fact that the wind is a 3d vector and has this two formats ledas 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 measuread 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;

$$C_l = D/2 * ((1/timeA - A) + (1/timeB - A))$$

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

* When

$$K_a = 1.4$$

,

$$R_a = 287.05 J/Kg * K$$

,

$$T_{av} = T(1 + 0.513 * q)$$

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.

Common errors with anemometer. 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.

When installation the anemometers there are some tips to have 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 instect 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

Calculate 1 hour averages of 10 minute mean wind speed and direction data.

```
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", colour="") +
  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))
```

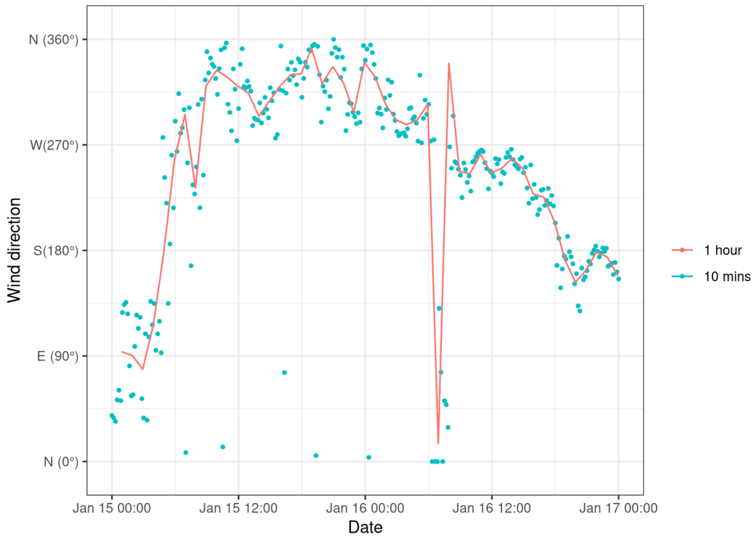


Figure 7.1: Comparison of wind direction original data (10 mins) and hourly average. Data from botanical garden 15th-17th January 2021.

The wind speed and direction have been averaged at 1 hour. Vectorial average has been used for wind direction. In figure 7.1 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, but the average is around 350.

7.4.2 Wind speed and height

How does the wind speed change with height? Characterize the wind pattern in different heights at the botanical garden

```
## plotting the same thing using gather
(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")
```

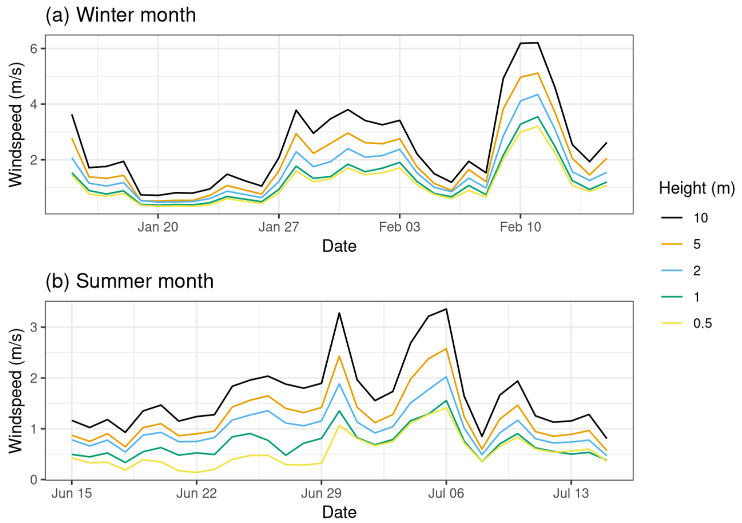


Figure 7.2: Time series of wind speed at different heigth. (a) is a summer month (15th Jan 2020 - 15th Feb 2020). (b) is a winter month(15th Jun 2020 - 15th Jul 2020). Data from botanical garden.

As it is possible to appreciate in Figure 7.2 , wind speed gets faster at 10m height. This makes sense when having in mind the vertical wind profile graph that increases with height.

```
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) +
  geom_point(aes(x=pred_ws, y=height,
        colour="Estimated\nlog profile"),
        data = wind_prof) +
  labs(x="Windspeed (m/s)", y="Height
        (m)", colour="")
```

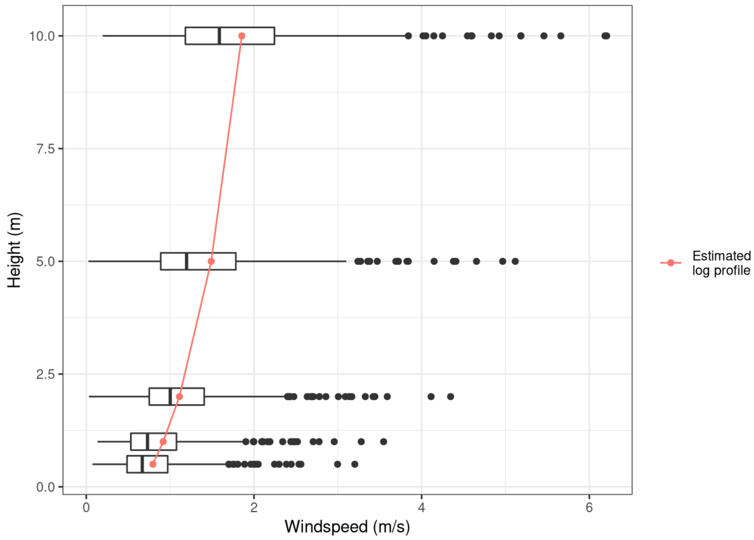


Figure 7.3: Distribution of daily means of wind speed at the different heights. The red line was obtaining by fitting a wind log profile to the mean of daily means. Data from botanical garden Jan 2020 - Feb 2021.

Wind speed depends on height, increasing with height following a logarithmic profile (Figure 7.3). The plot was made using daily averages instead of high frequency data, to reduce the variation in the dataset.

7.4.3 Windspeed over year

How does the wind speed varies over the year and what can explain the variability?

```
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")
```

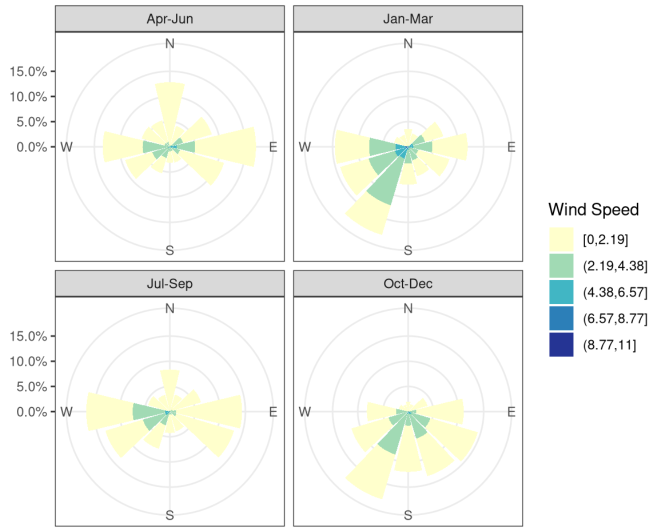


Figure 7.4: Wind rose for different quarters of the year. Data from botanical garden Jan 2020 - Feb 2021.

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

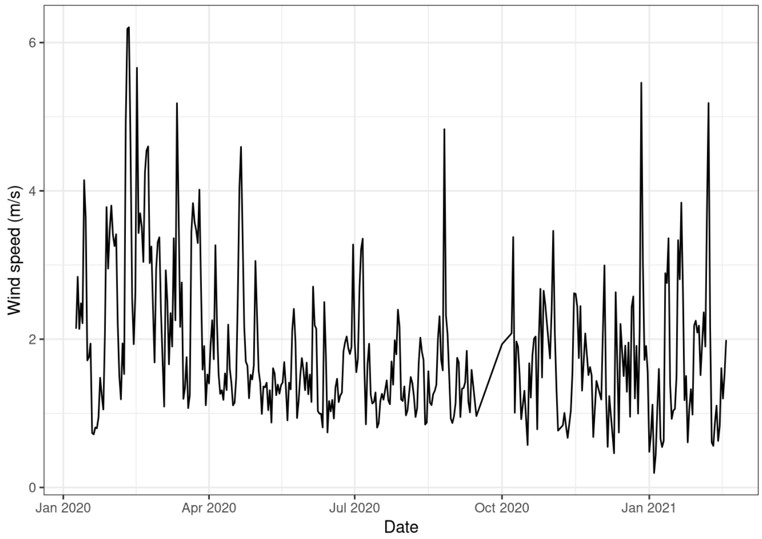


Figure 7.5: Daily averages of wind speed at 10 meters. Data from forest botanical garden January 2020 - February 2021.

During the spring and winter the wind is stronger. In summer daily average oscillates around 1.5 m/s (Figure 7.5)

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.

```
# 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))
} )
```

```
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))
```

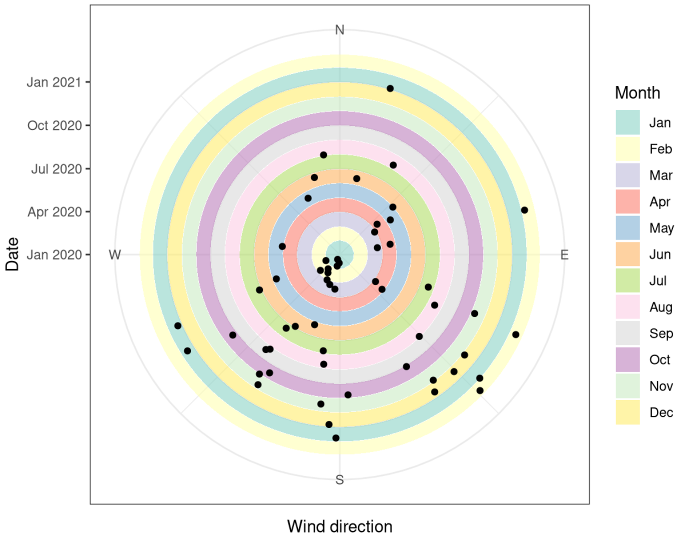


Figure 7.6: Weekly average of wind directions for the year. The distance from the center and the different background indicates the date, while the position in the circle the wind direction. Data from forest botanical garden January 2020 - February 2021.

8 Soil physics

8.1 Motivation

During the previous protocols, it has been described that earth is formed by different atmospheric layers. Well, is not only about these but also about the ones that are found more at the land surface. Soil is the first layer found when land surface is reached. It consists on the middle face between the lithosphere and the atmosphere. Thus, soil is related with other factors, such as biology and hydrology. Ecosystems are established on the soil, and soil is shaping them as the main source of resources for the natural habitats to expand (Lal and Shukla 2004).

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 mail 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 moisture (Orchard and Cook 1983).