9th Protocol: Evapotranspiration

Cheyenne Rueda and Simone Massaro

05 July 2021

Contents

1 Motivation

In this chapter we will focus on evapotranspiration. This term is well known in many environmental study fields, but, what is the most precise definition given to this term?

Evapotranspiration (ET) is referred according to the relationship of two other factors, such as the soil water evaporation and plant transpiration. The actual evapotranspiration is the one that is measured from these two factors mentioned before, considering changes and not ideal conditions in the meteorological conditions and the type of soil (Labedzki, 2011). In contrast, potential evapotranspiration is that one that occurs with ideal conditions such as abundance of water storage and meteorological conditions. Some differences between potential and actual evapotranspiration are the infiltration capacity of soil, possible diseases, pH of soil and its fertility.

Labedzki, L. (Ed.). (2011). Evapotranspiration. BoD–Books on Demand.

2 Background

3 Sensors and measuring principle

4 Analysis

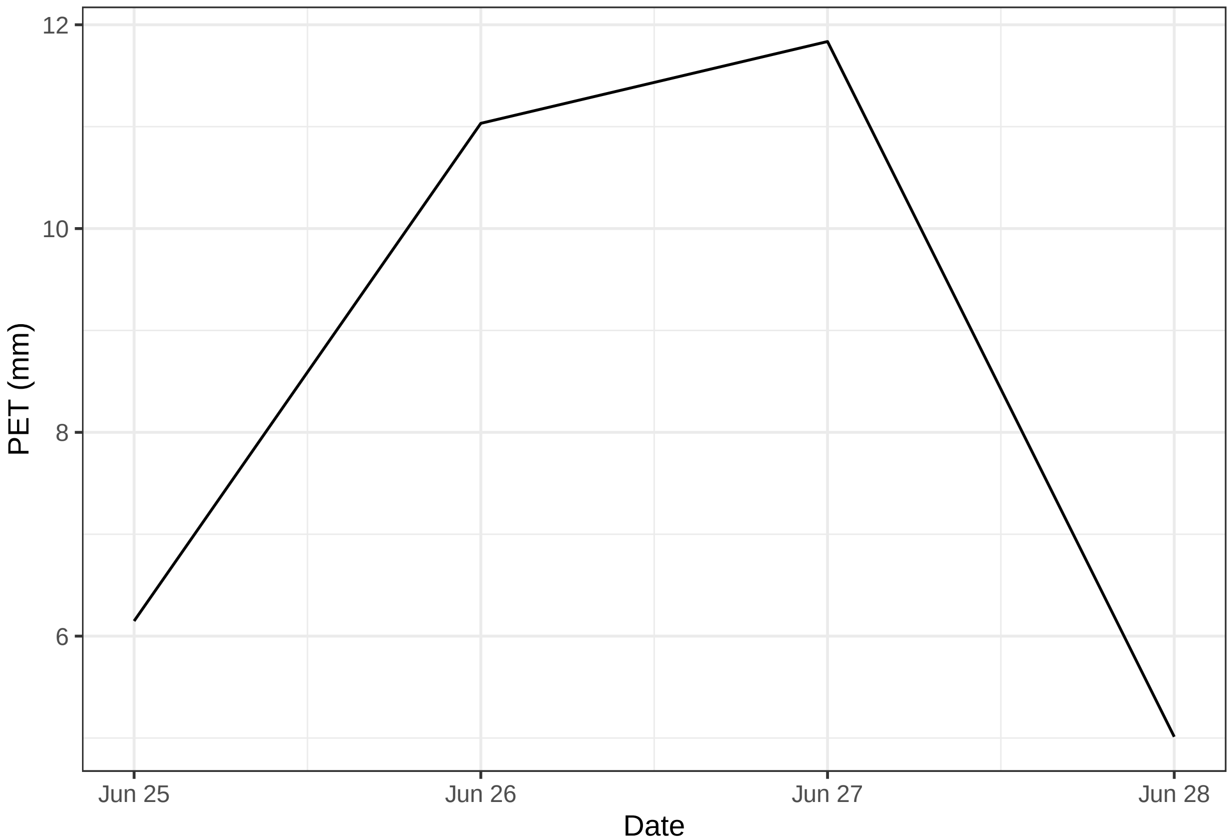
4.1 Potential evapotranspiration

```
library(tidyverse)
library(lubridate)
theme_set(theme_bw())
et <- read_csv(
  "../Data_lectures/09_Turbulent_fluxes_I_ET/ET_data_forst_botanical_garden.csv",
  locale = locale(decimal_mark = ",")) %>%
  rename(loc_id = replicates)
meteo <- read_csv("../Data_lectures/09_Turbulent_fluxes_I_ET/MeteoData_BotanicalGarden.csv")
```

```
#' Potential evapotranspiration using Priestley-Taylor equation
#' input data is W/s internally converted to MJ/d
calc_pet <- function(T_air, Rn, G){
  g <- 0.067 # kPa K -1
  Rn <- Rn * 0.0864 # convert to MJ/day
  G <- G * 0.0864 # convert to MJ/day
  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)
  return(pet)
}
```

```
#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))
```

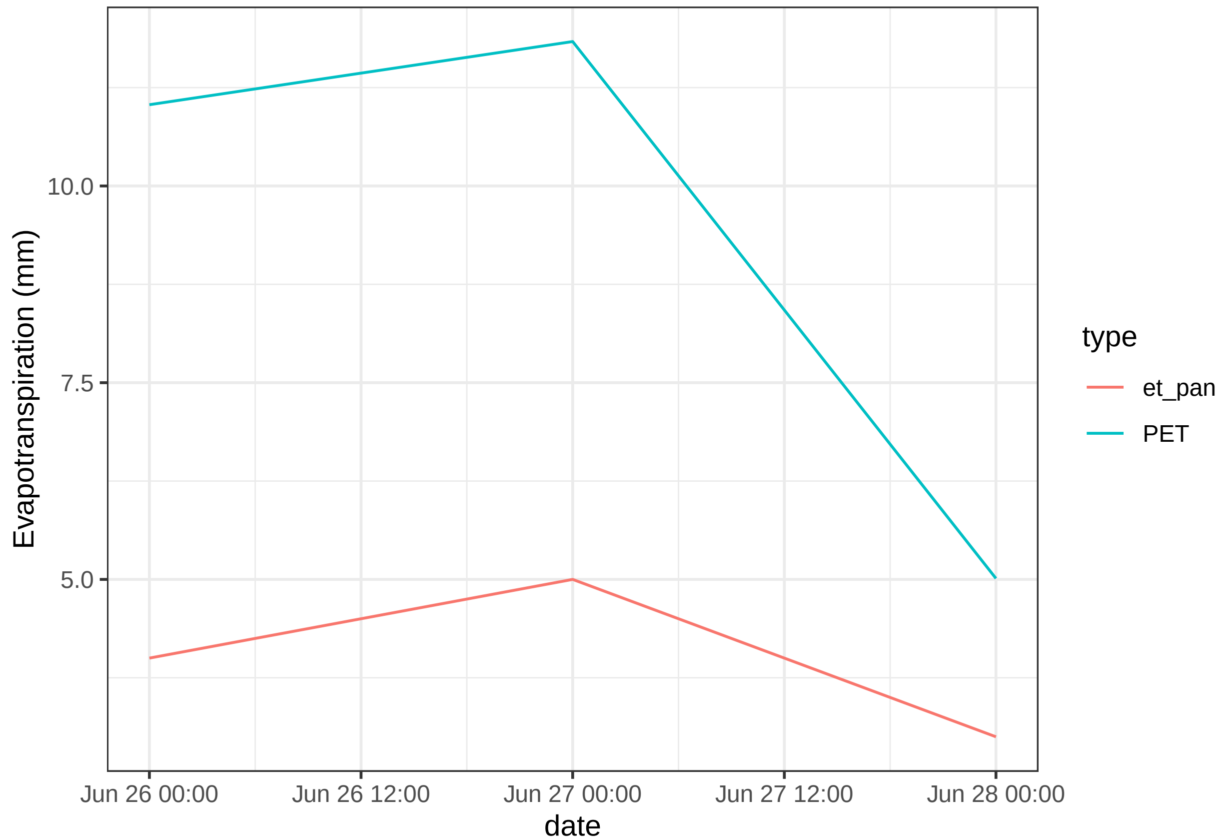
```
ggplot(meteo_d, aes(Date, PET)) +
  geom_line() +
  labs(y="PET (mm)")
```



During the four days of measurement the potential ET has been estimated using the Priestley-Taylor. 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.

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

```
et %>%
  left_join(meteo_d, by = c("date"= "Date")) %>%
  drop_na() %>%
  gather("type", "et", et_pan, PET) %>%
  ggplot(aes(date, et, col=type)) +
  geom_line() +
  labs(y= "Evapotranspiration (mm)")
```



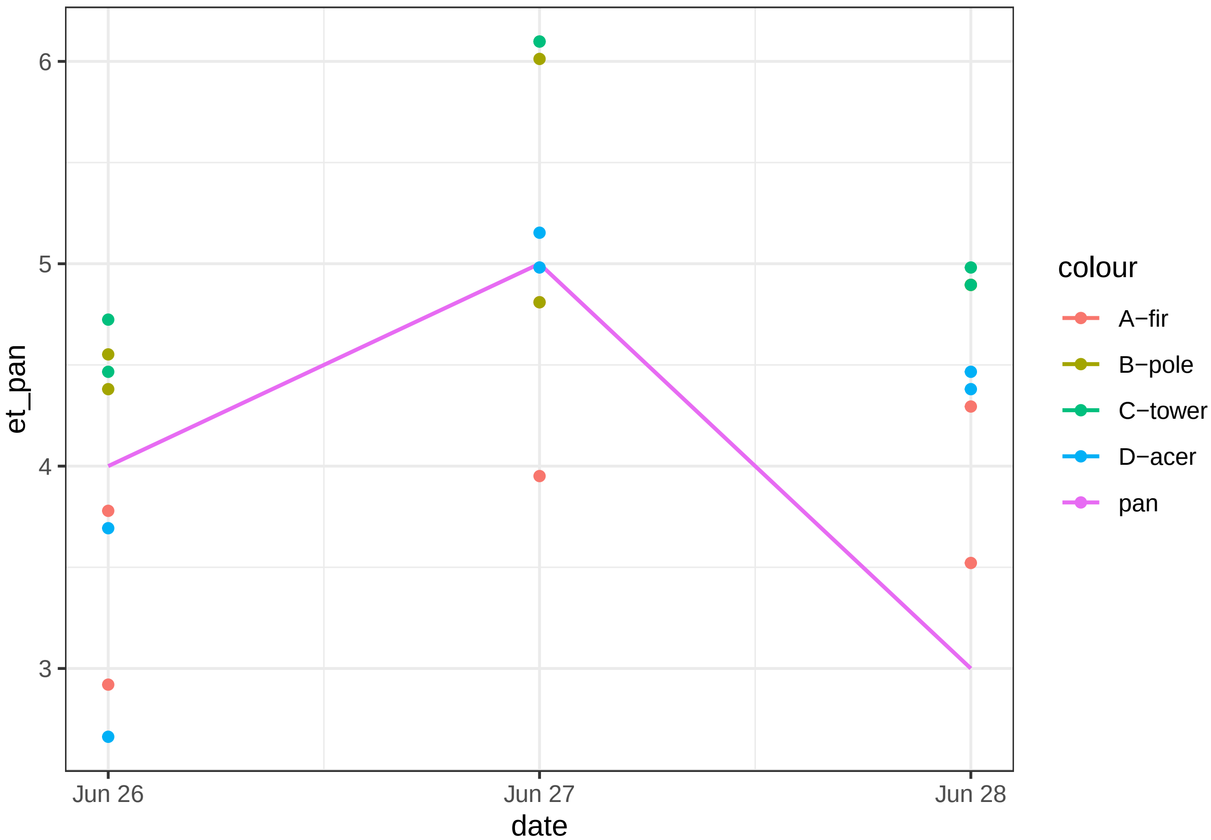
TODO

4.2 Piché evaporimeters

```
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
pich_int_area <- (pi/4 * pich_inn_d ^2 ) # cm^2
```

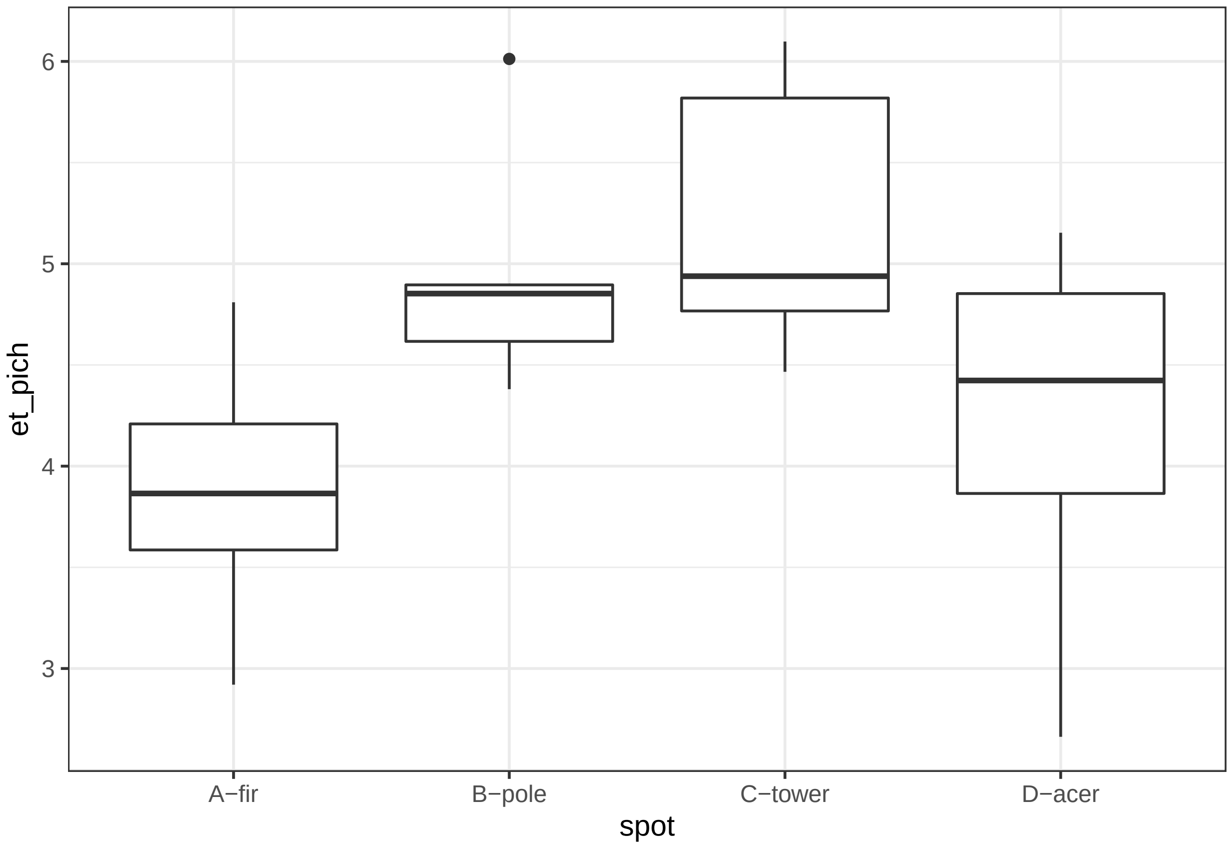
```
et <- et %>%
  group_by(loc_id) %>%
  mutate(
    # here the scale is the opposite, the lower the number the more the water
    diff_pich = pich_height_cm - lag(pich_height_cm),
    et_pan = lag(pan_height_mm) - pan_height_mm,
    # need to convert to right unit
    et_pich = diff_pich * pich_dish_area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

```
et %>%
  drop_na() %>% #removing first empty day
ggplot(aes(date)) +
  geom_line(aes(y=et_pan, col="pan"), size=.7) +
  geom_point(aes(y = et_pich, col=spot))
```




```
ggplot(et, aes(spot, et_pich)) +
  geom_boxplot()
```

```
## Warning: Removed 8 rows containing non-finite values (stat_boxplot).
```



PET WIP

```
# this is like floor_date(x, "day"), but instead of changing the day at midnight
# it changes to the next day after the given hour
floor_time_day <- function(date, hour){
  date <- case_when(
    hour(date) <= hour ~ date, #keeping the same day
    hour(date) > hour ~ date + days(1) # going to next day
  )
  floor_date(date, "day")
}
```

```
meteo %>%
  group_by(Date = floor_date(Date, "day")) %>%
  summarise(R_n_d = mean(`NetRadiation_Wm-2`) , G_d = mean(`GroundHeatflux_Wm-2`), T_air = mean(TA_degC
  mutate(
    PET = calc_pet(T_air, R_n_d, G_d)
  )
```

```
## # A tibble: 4 x 5
##   Date                R_n_d   G_d T_air   PET
##   <dttm>              <dbl> <dbl> <dbl> <dbl>
## 1 2021-06-25 00:00:00  86.4  5.36  17.1  5.72
## 2 2021-06-26 00:00:00 145.   6.49  17.6  9.85
## 3 2021-06-27 00:00:00 154.   9.30  19.7 10.7
## 4 2021-06-28 00:00:00  65.7  4.69  21.1  4.62
```

```
#adding R_n and G to the
meteo %>%
```

```r
  # 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
  group_by(Date = floor_date(Date, "hour")) %>%
  summarise(across(everything(), mean)) %>%
  mutate(PET = calc_pet(TA_degC, `NetRadiation_Wm-2`, `GroundHeatflux_Wm-2`)) %>%
  group_by(Date = floor_date(Date, "day")) %>%
  summarise(PET = mean(PET))
```

```
## # A tibble: 4 x 2
##   Date                  PET
##   <dttm>              <dbl>
## 1 2021-06-25 00:00:00  6.08
## 2 2021-06-26 00:00:00 11.0
## 3 2021-06-27 00:00:00 11.8
## 4 2021-06-28 00:00:00  5.00
```

```r
#adding R_n and G to the
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
  group_by(Date = floor_date(Date, "30 min")) %>%
  summarise(across(everything(), mean)) %>%
  mutate(PET = calc_pet(TA_degC, `NetRadiation_Wm-2`, `GroundHeatflux_Wm-2`)) %>%
  group_by(Date = floor_date(Date, "day")) %>%
  summarise(PET = mean(PET))
```

```
## # A tibble: 4 x 2
##   Date                  PET
##   <dttm>              <dbl>
## 1 2021-06-25 00:00:00  6.14
## 2 2021-06-26 00:00:00 11.0
## 3 2021-06-27 00:00:00 11.8
## 4 2021-06-28 00:00:00  5.01
```

```r
#adding R_n and G to the
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
  group_by(Date = floor_date(Date, "day")) %>%
  summarise(across(everything(), mean)) %>%
  mutate(PET = calc_pet(TA_degC, `NetRadiation_Wm-2`, `GroundHeatflux_Wm-2`)) %>%
  group_by(Date = floor_date(Date, "day")) %>%
  summarise(PET = mean(PET))
```

```
## # A tibble: 4 x 2
##   Date                  PET
##   <dttm>              <dbl>
## 1 2021-06-25 00:00:00  5.72
## 2 2021-06-26 00:00:00  9.85
## 3 2021-06-27 00:00:00 10.7
## 4 2021-06-28 00:00:00  4.62
```

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

```
ggplot(aes(date, et, fill=type)) +
geom_bar(stat="identity", position="dodge")
```