Compound soil and atmospheric drought events and CO_2 fluxes of a mixed deciduous forest: Occurrence, impact, and temporal contribution of main drivers

Scripts of the data analysis

Liliana Scapucci and Ankit Shekhar

2024-01-10

In this R project there are all the analysis and the scripts of the plots used for the manuscript, the order is dependent on the figure that will be also a section of the project.

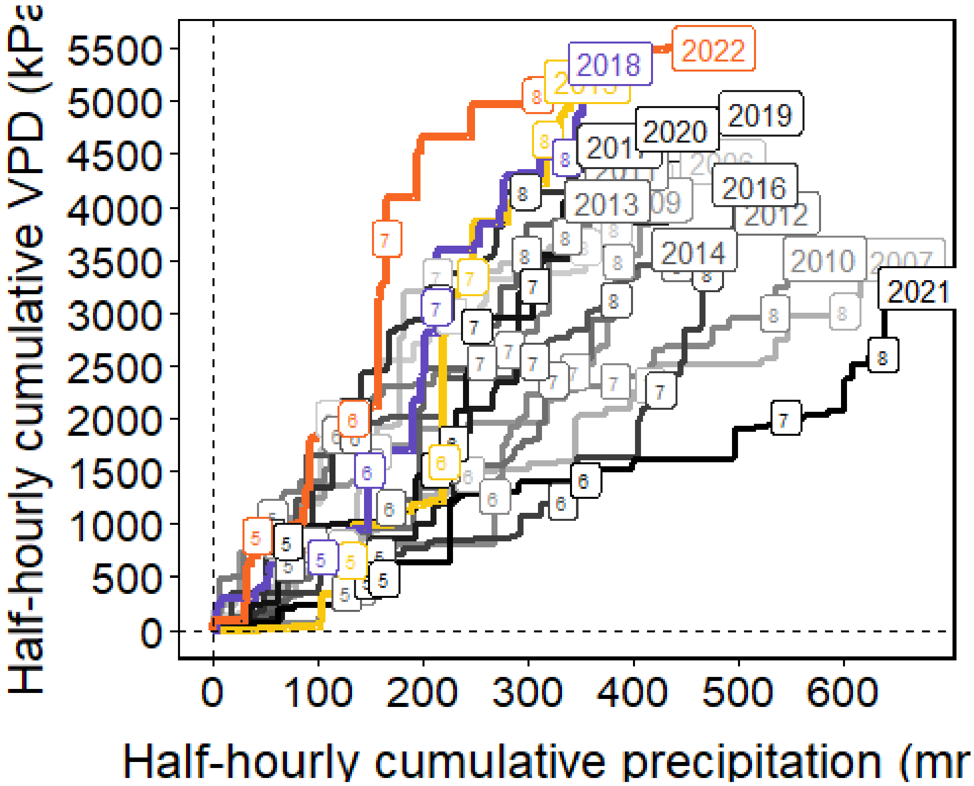
Index

0. Required packages and colors

```
.libPaths("")
library(dplyr)

library(lubridate)

library(qpcR)

library(ggplot2)
library(stringr)
library(ggpubr)
library(cowplot)

library(scales)
library(gridExtra)

library(plotrix)
library(tidyr)

library(ggpmisc)

library(zoo)

library(randomForest)

library(party)

library(SHAPforxgboost)
library(xgboost)

library(lme4)

library(patchwork)

library(lattice)
library(caret)

library(bigleaf)
library(tibble)

# Theme:
th1 <- theme(legend.title = element_blank(), legend.background = element_blank(),
legend.direction="vertical",
             legend.text = element_text(size = 14), legend.key = element_rect(fill = "white"),
             legend.box.background = element_rect(colour = "NA"),  legend.spacing.y = unit(0,
"mm"),
             axis.title.y =element_text(size=18, color = "black"), axis.title.x
=element_text(size=18, color = "black"),
             axis.text.y =element_text(size=16, color = "black"),axis.text.x =
element_text(size = 16, color = "black"),
             strip.background = element_rect(fill = "white"), strip.text = element_text(face =
"bold", size = 16),
             panel.background=element_rect(color="black", fill = "white", linewidth = 1),
             axis.ticks.x = element_line(color = "black"), axis.ticks.y = element_line(color =
"black"),
             panel.grid.major = element_blank(),
             panel.grid.minor = element_blank())
```

1. Figure 1: Cumulative VPD and precipitation

```
## Figure 1 ----
load("lae_30MIN.RData") #eddy covariance data from above canopy (30 min res.)
Cummulative_Met_plot_data = lae_30MIN %>% mutate(Month = month(Timestamp)) %>%
  filter(year %in% c(2005:2022)) %>% filter(Month %in% c(5:9)) %>%
  group_by(year) %>% mutate(CumVPD = cumsum(VPD_f/10), CumPrec = cumsum(Precip), CumNEP =
cumsum(-NEE_f), CumET = cumsum(ET_f), CumTair = cumsum(Tair_f))

p = Cummulative_Met_plot_data %>%   filter(!year %in% c(2008)) %>%
  ggplot(., aes(x = CumPrec, y = CumVPD, color = year, group = year)) +
  geom_line(data = . %>% filter(year != 2015) %>% filter(year != 2018) %>% filter(year !=
2022), aes(x = CumPrec, y = CumVPD), linewidth = 1.2) +  # geom_line(alpha = 0.3, size = 1) +
  geom_line(data = . %>% filter(year == 2015), aes(x = CumPrec, y = CumVPD), color = "#F9C80E",
linewidth = 1.5) +
  geom_line(data = . %>% filter(year == 2018), aes(x = CumPrec, y = CumVPD), color = "#6248BF",
linewidth = 1.5) +
  geom_line(data = . %>% filter(year == 2022), aes(x = CumPrec, y = CumVPD), color = "#F86624",
linewidth = 1.5) +
  geom_label(data = . %>% filter(year != 2015) %>% filter(year != 2018) %>% filter(year !=
2022) %>% group_by(year, Month) %>% summarise_if(is.numeric, last),
             aes(label = Month), size = 2.5) +
  geom_label(data = . %>% filter(year== 2015) %>% group_by(year, Month) %>%
summarise_if(is.numeric, last),
             aes(label = Month), size = 2.5, col = "#F9C80E") +
  geom_label(data = . %>% filter(year== 2018) %>% group_by(year, Month) %>%
summarise_if(is.numeric, last),
             aes(label = Month), size = 2.5, col = "#6248BF") +
  geom_label(data = . %>% filter(year== 2022) %>% group_by(year, Month) %>%
summarise_if(is.numeric, last),
             aes(label = Month), size = 2.5, col = "#F86624") +

  geom_label(data = . %>% filter(year != 2015) %>% filter(year != 2018) %>% filter(year !=
2022) %>% group_by(year) %>%
               summarise_if(is.numeric, last), aes(label = year), size = 4) +
  geom_label(data = . %>% filter(year == 2015)%>% group_by(year) %>%
               summarise_if(is.numeric, last), aes(label = year), size = 4, col = "#F9C80E") +
  geom_label(data = . %>% filter(year == 2018)%>% group_by(year) %>%
               summarise_if(is.numeric, last), aes(label = year), size = 4, col = "#6248BF") +
  geom_label(data = . %>% filter(year == 2022)%>% group_by(year) %>%
               summarise_if(is.numeric, last), aes(label = year), size = 4, col = "#F86624") +
  th1 + theme(axis.title.y = element_text(vjust = +2), axis.title.x = element_text(vjust = -
1))+
  labs(x = 'Half-hourly cumulative precipitation (mm)', y = 'Half-hourly cumulative VPD (kPa)')
+
  guides(color = 'none') +   scale_y_continuous(breaks = seq(0,5500,500)) +
  scale_x_continuous(breaks = seq(0,650,100)) +
  scale_colour_gradient(low = "grey80", high = "black") +
  geom_vline(xintercept = 0, lty = 'dashed') + geom_hline(yintercept = 0, lty = 'dashed')
p
```



2. Figure 2: Tair, VPD and SWC 2015-2018-2022 vs mean 2005-2022

In this figure we take the 5 days moving average of Tair, VPD and SWC of 2015, 2018, 2022 and we plot it against the mean and standard error of 2004-2022 (like in the previous section).

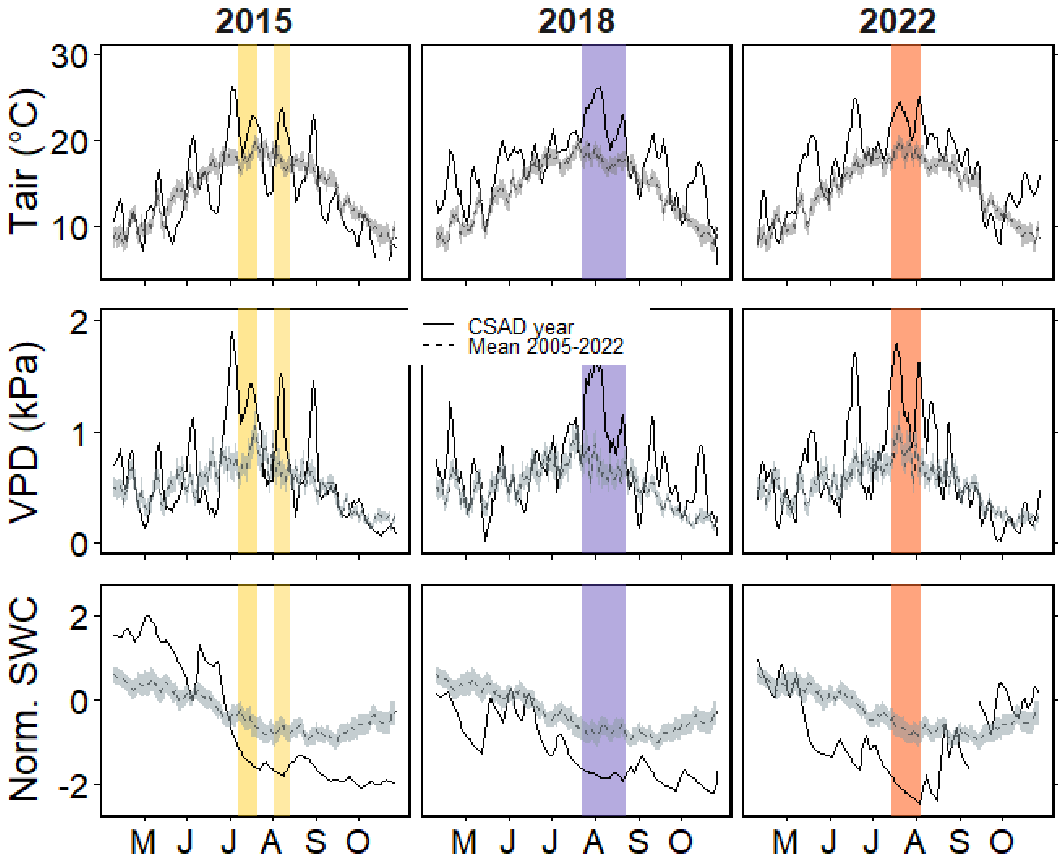
First we calculate the 5 days moving average for the three years with a CDH using the R-package *zoo*:

```r
load("CH_LAE_meteo_flux_2004_2022_selected_variables_daily.RData")
meteo <- lae %>% filter(year >2004) %>% dplyr::select(doy, mean_Tair_f, mean_VPD_f,
mean_SWC_stnd)
colnames(meteo) <- sub("mean_", "", colnames(meteo))
meteo_avg <- meteo %>% group_by(doy) %>% summarise(mean_Tair = mean(Tair_f, na.rm = TRUE),
                                                   mean_VPD = mean(VPD_f, na.rm = TRUE),
                                                   mean_SWC = mean(SWC_stnd, na.rm = TRUE),
                                                   se_Tair = std.error(Tair_f,na.rm = TRUE),
                                                   se_VPD = std.error(VPD_f, na.rm = TRUE),
                                                   se_SWC = std.error(SWC_stnd, na.rm = TRUE),
                                                   sd_Tair = sd(Tair_f,na.rm = TRUE),
                                                   sd_VPD = sd(VPD_f, na.rm = TRUE),
                                                   sd_SWC = sd(SWC_stnd, na.rm = TRUE)) %>%
ungroup()

met_22 <- lae %>% filter(year == 2022)
meteo_avg$year <- "mean"
met_15 <- lae %>% filter(year == 2015)
met_18 <- lae %>% filter(year == 2018)

met_15$period <- 2015
met_18$period <- 2018
met_22$period <- 2022

# 5 days moving average of the meteo variables
met_15 <- met_15 %>% arrange(doy)
met_18 <- met_18 %>% arrange(doy)
met_22 <- met_22 %>% arrange(doy)

met_15$Tair_5d <- rollapply(met_15$mean_Tair_f, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
met_18$Tair_5d <- rollapply(met_18$mean_Tair_f, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
met_22$Tair_5d <- rollapply(met_22$mean_Tair_f, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")

met_15$VPD_5d <- rollapply(met_15$mean_VPD_f, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
met_18$VPD_5d <- rollapply(met_18$mean_VPD_f, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
met_22$VPD_5d <- rollapply(met_22$mean_VPD_f, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")

met_15$SWC_5d <- rollapply(met_15$mean_SWC_stnd, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
met_18$SWC_5d <- rollapply(met_18$mean_SWC_stnd, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
met_22$SWC_5d <- rollapply(met_22$mean_SWC_stnd, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")

## Calculation of the duration, max VPD, min SWC and Max Air T during the CSAD events
# calculation of duation, max and min TAIR VPD AND SWC during the csad events


lae %>%  filter(year == 2015) %>% filter(doy %in% c(188:202, 214:225)) %>% summarise(max_Tair =
max(mean_Tair_f, na.rm = T),
                                                              sd_Tair= sd(mean_Tair_f, na.rm =
T),
                                                              max_VPD = max(mean_VPD_f, na.rm =
T),
                                                              sd_VPD = sd(mean_VPD_f, na.rm =
T),
                                                               min_SWC = min(mean_SWC_stnd,
na.rm = T),
                                                               sd_SWC = sd(mean_SWC_stnd, na.rm
= T))

## # A tibble: 1 × 6
##   max_Tair sd_Tair max_VPD sd_VPD min_SWC sd_SWC
##      <dbl>   <dbl>   <dbl>  <dbl>   <dbl>  <dbl>
## 1     26.9    3.03    22.4   4.87   -1.83  0.197

lae %>% filter(year == 2018) %>% filter(doy %in% c(204:235)) %>% summarise(max_Tair =
max(mean_Tair_f, na.rm = T),
                                                               sd_Tair= sd(mean_Tair_f, na.rm =
T),
                                                               max_VPD = max(mean_VPD_f, na.rm =
T),
                                                               sd_VPD = sd(mean_VPD_f, na.rm =
T),
                                                               min_SWC = min(mean_SWC_stnd,
na.rm = T),
                                                               sd_SWC = sd(mean_SWC_stnd, na.rm
= T))

## # A tibble: 1 × 6
##   max_Tair sd_Tair max_VPD sd_VPD min_SWC sd_SWC
##      <dbl>   <dbl>   <dbl>  <dbl>   <dbl>  <dbl>
## 1     27.7    2.88    21.9   4.84   -1.93 0.0972

lae %>% filter(year == 2022) %>% filter(doy %in% c(195:216)) %>% summarise(max_Tair =
max(mean_Tair_f, na.rm = T),
                                                               sd_Tair= sd(mean_Tair_f, na.rm =
T),
                                                               max_VPD = max(mean_VPD_f, na.rm =
T),
                                                               sd_VPD = sd(mean_VPD_f, na.rm =
T),
                                                               min_SWC = min(mean_SWC_stnd,
na.rm = T),
                                                               sd_SWC = sd(mean_SWC_stnd, na.rm
= T))

## # A tibble: 1 × 6
##   max_Tair sd_Tair max_VPD sd_VPD min_SWC sd_SWC
##      <dbl>   <dbl>   <dbl>  <dbl>   <dbl>  <dbl>
## 1     28.3    2.64    24.3   5.20   -2.51  0.200
```

Then we plot the data together:

```
ta <- ggplot() +
  geom_rect(data = met_15%>% filter(doy %in% c(188:202)), aes(xmin = 188, xmax = 202, ymin = -
Inf, ymax = +Inf), alpha = 0.04, , fill = color3[1]) +
  geom_rect(data = met_15%>% filter(doy %in% c(214:225)), aes(xmin = 214, xmax = 225, ymin = -
Inf, ymax = +Inf), alpha =  0.04, , fill = color3[1]) +
  geom_rect(data = met_18%>% filter(doy %in% c(204:235)), aes(xmin = 204, xmax = 235, ymin = -
Inf, ymax = +Inf), alpha =  0.02, fill = color3[2]) +
  geom_rect(data = met_22%>% filter(doy %in% c(195:216)), aes(xmin = 195, xmax = 216, ymin = -
Inf, ymax = +Inf), alpha =  0.04, fill = color3[3]) +# 2015
  geom_line(data = met_15, aes(x = doy, y= Tair_5d, col = "CSAD year", lty = "CSAD year")) +
  # 2018
  geom_line(data = met_18, aes(x = doy, y= Tair_5d, col = "CSAD year", lty = "CSAD year")) +
  # 2022
  geom_line(data = met_22, aes(x = doy, y= Tair_5d, col = "CSAD year", lty = "CSAD year")) +
  # rest of the years
```

```r
  geom_line(data = meteo_avg, aes(x = doy, y= mean_Tair, col = "Mean", lty = "Mean"), alpha =
0.8) +
  geom_ribbon(data = meteo_avg, aes(x = doy, ymin = mean_Tair - se_Tair, ymax = mean_Tair +
se_Tair, fill = "Mean 2005-2022"), alpha = 0.5) +
  scale_color_manual(values = c("grey2", "black"), name = "Years", labels = c("CSAD year",
"Mean 2005-2022")) +
  scale_linetype_manual(values = c(1, 2), name = "Years", labels = c("CSAD year", "Mean 2005-
2022")) +
  scale_fill_manual(values = c("grey50"), labels = c("Mean 2005-2022"), guide = "none")  +
  scale_y_continuous(name = "Tair (°C)", limits = c(5, 30), sec.axis = sec_axis(trans=~(.*1),
name= "", labels = c())) +
  scale_x_continuous(breaks= c(121, 152, 182, 213, 244, 274), labels = c(), limits = c(100,
300)) +
  facet_grid(cols = vars(period)) +
  xlab("") + th1 + theme(legend.position = "none", axis.text.x = element_blank(), axis.title.x
= element_blank())


vpd <- ggplot() +
  geom_rect(data = met_15%>% filter(doy %in% c(188:202)), aes(xmin = 188, xmax = 202, ymin = -
Inf, ymax = +Inf, fill = as.character(year)), alpha = 0.04) +
  geom_rect(data = met_15%>% filter(doy %in% c(214:225)), aes(xmin = 214, xmax = 225, ymin = -
Inf, ymax = +Inf, fill = as.character(year)), alpha =  0.04) +
  geom_rect(data = met_18%>% filter(doy %in% c(204:235)), aes(xmin = 204, xmax = 235, ymin = -
Inf, ymax = +Inf, fill = as.character(year)), alpha =  0.02) +
  geom_rect(data = met_22%>% filter(doy %in% c(195:216)), aes(xmin = 195, xmax = 216, ymin = -
Inf, ymax = +Inf, fill = as.character(year)), alpha =  0.04) +
  # 2015
  geom_line(data = met_15, aes(x = doy, y= VPD_5d/10, col = "CSAD year", lty = "CSAD year")) +
  # 2018
  geom_line(data = met_18, aes(x = doy, y= VPD_5d/10, col = "CSAD year", lty = "CSAD year")) +
  # 2022
  geom_line(data = met_22, aes(x = doy, y= VPD_5d/10, col = "CSAD year", lty = "CSAD year")) +
  # rest of the years
  geom_line(data = meteo_avg, aes(x = doy, y= mean_VPD/10, col = "Mean", lty = "Mean"), alpha =
0.8) +
  geom_ribbon(data = meteo_avg, aes(x = doy, ymin = (mean_VPD - se_VPD)/10, ymax = (mean_VPD +
se_VPD)/10), fill ="#899BA1", alpha = 0.5) +
  scale_color_manual(values = c("grey2", "black"), name = "Years", labels = c("CSAD year",
"Mean 2005-2022")) +
  scale_linetype_manual(values = c(1, 2), name = "Years", labels = c("CSAD year", "Mean 2005-
2022")) +
  scale_fill_manual(values = color3, guide = "none") +
  scale_y_continuous(name = "VPD (kPa)", limits = c(0, 2.0), breaks = seq(0, 2.0, by = 1),
                     sec.axis = sec_axis(trans=~(.*1), breaks = seq(0, 2.5, by = 1), name= "",
labels = c())) +
  scale_x_continuous(breaks= c(121, 152, 182, 213, 244, 274), labels = c(), limits = c(100,
300)) +
  facet_grid(cols = vars(period)) +
  xlab("") + th1 +theme(legend.position = c(0.45, 0.9), legend.direction = "vertical",
                        legend.key.height = unit(1.5, "mm"), axis.text.x = element_blank(),
                        axis.title.x = element_blank(), legend.text = element_text(size = 10),
                        strip.text = element_blank())

swc <- ggplot() + th1 +
  geom_rect(data = met_15%>% filter(doy %in% c(188:202)), aes(xmin = 188, xmax = 202, ymin = -
Inf, ymax = +Inf, fill = as.character(year)), alpha = 0.04) +
  geom_rect(data = met_15%>% filter(doy %in% c(214:225)), aes(xmin = 214, xmax = 225, ymin = -
Inf, ymax = +Inf, fill = as.character(year)), alpha =  0.04) +
  geom_rect(data = met_18%>% filter(doy %in% c(204:235)), aes(xmin = 204, xmax = 235, ymin = -
Inf, ymax = +Inf, fill = as.character(year)), alpha =  0.02) +
  geom_rect(data = met_22%>% filter(doy %in% c(195:216)), aes(xmin = 195, xmax = 216, ymin = -
Inf, ymax = +Inf, fill = as.character(year)), alpha =  0.04) +
  # 2015
  geom_line(data = met_15, aes(x = doy, y= SWC_5d, col = "CSAD year", lty = "CSAD year")) +
  # 2018
  geom_line(data = met_18, aes(x = doy, y= SWC_5d, col = "CSAD year", lty = "CSAD year")) +
  # 2022
  geom_line(data = met_22, aes(x = doy, y= SWC_5d, col = "CSAD year", lty = "CSAD year")) +
  # rest of the years
  geom_line(data = meteo_avg, aes(x = doy, y= mean_SWC, col = "Mean", lty = "Mean"), alpha =
0.8) +
  geom_ribbon(data = meteo_avg, aes(x = doy, ymin = mean_SWC - se_SWC, ymax = mean_SWC +
se_SWC), fill ="#899BA1", alpha = 0.5) +
  scale_color_manual(values = c("grey2", "black"), name = "Years", labels = c("CSAD year",
"Mean")) +
  scale_linetype_manual(values = c(1,2), name = "Years", labels = c( "CSAD year", "Mean")) +
  scale_fill_manual(values = color3, guide = "none") +
  scale_y_continuous(name = "Norm. SWC", limits = c(-2.5, 2.5),breaks = c(-2, 0,2),
                     sec.axis = sec_axis(trans=~(.*1), breaks = c(-2, 0, 2), name= "", labels =
c()))  +
  scale_x_continuous(breaks= c(121, 152, 182, 213, 244, 274), labels = c("M", "J", "J", "A",
"S", "O"), limits = c(100, 300)) +
  facet_grid(cols = vars(period)) +
  xlab("")  +theme(legend.position = "none",  strip.text = element_blank())


m <- plot_grid(ta, vpd, swc, ncol = 1, align = "v", rel_heights = c(1.25,1.08,1.35))

## Warning: Removed 164 rows containing missing values (`geom_line()`).
## Removed 164 rows containing missing values (`geom_line()`).
## Removed 164 rows containing missing values (`geom_line()`).

## Warning: Removed 165 rows containing missing values (`geom_line()`).

## Warning: Removed 164 rows containing missing values (`geom_line()`).
## Removed 164 rows containing missing values (`geom_line()`).
## Removed 164 rows containing missing values (`geom_line()`).

## Warning: Removed 165 rows containing missing values (`geom_line()`).

## Warning: Removed 164 rows containing missing values (`geom_line()`).
## Removed 164 rows containing missing values (`geom_line()`).
## Removed 164 rows containing missing values (`geom_line()`).

## Warning: Removed 165 rows containing missing values (`geom_line()`).

m
```



3. Figure 3: Fluxes 2015-2018-2022 vs mean 2005-2022

The first plots of this section are the ones comparing the 5 days moving avg of the fluxes against the mean 2005-2022. So the first step is to calculate the five days moving average of the fluxes above canopy. Then we need to load a new dataset for the below-canopy fluxes to investigate the Rff from CDH years against the mean 2019-2021.

```
# selecting the variables we need
lae_mean <- lae %>% dplyr::select(mean_NEP, mean_GPP_DT, mean_Reco_DT, year, doy) %>%
group_by(doy) %>%
  filter(year > 2004) %>%
  summarise(NEP = mean(mean_NEP, na.rm = TRUE),
            GPP_DT = mean(mean_GPP_DT, na.rm = TRUE),
            Reco_DT = mean(mean_Reco_DT, na.rm = TRUE),
            NEP_se = std.error(mean_NEP, na.rm = TRUE),
            GPP_se = std.error(mean_GPP_DT, na.rm = TRUE),
            Reco_se = std.error(mean_Reco_DT, na.rm = TRUE))
# filtering for different years
lae_15 <-lae %>% filter(year == 2015) %>% arrange(doy)
lae_18 <-lae %>% filter(year == 2018) %>% arrange(doy)
lae_22 <-lae %>% filter(year == 2022) %>% arrange(doy)
# calculating 5 days moving average:
## 2015##
lae_15$Reco_5d <- rollapply(lae_15$mean_Reco_DT, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
lae_15$NEP_5d <- rollapply(lae_15$mean_NEP, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
lae_15$GPP_5d <- rollapply(lae_15$mean_GPP_DT, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
## 2018 ##
lae_18 <- lae_18%>% arrange(doy)
lae_18$Reco_5d <- rollapply(lae_18$mean_Reco_DT, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
lae_18$NEP_5d <- rollapply(lae_18$mean_NEP, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
lae_18$GPP_5d <- rollapply(lae_18$mean_GPP_DT, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
## 2022 ##
lae_22$Reco_5d <- rollapply(lae_22$mean_Reco_DT, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
lae_22$NEP_5d <- rollapply(lae_22$mean_NEP, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
lae_22$GPP_5d <- rollapply(lae_22$mean_GPP_DT, width=5, FUN=function(x) mean(x, na.rm=TRUE),
partial=TRUE, align="center")
```

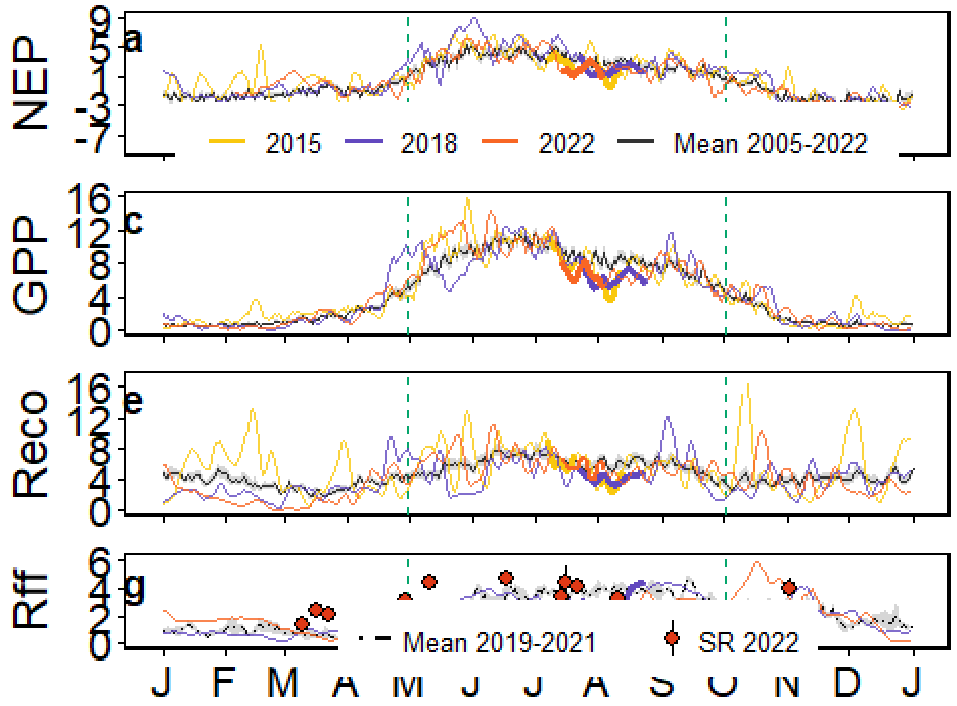
Loading the below-canopy fluxes 2014-2022 and extract the selected intervals:

```
load("CH_LAS_meteo_flux_2014_2022_selected_vars_norm_DAILY.RData") #named las
las1 <- las %>% mutate(year =year(Date)) %>% mutate(doy =yday(Date)) %>% filter(year >= 2019 &
year <= 2021) %>%  dplyr::select(doy, mean_Reco_DT_U50)
colnames(las1) <- sub("mean_", "", colnames(las1))
las_avg <- las1 %>% group_by(doy) %>% dplyr::summarise(mean_Rff = mean(Reco_DT_U50)) %>%
ungroup()
las_se <- las1  %>% group_by(doy) %>% dplyr::summarise(se_Rff = std.error(Reco_DT_U50))  %>%
ungroup()
las_mean <- merge(las_avg, las_se, by = "doy")
las_mean <- las_mean%>% arrange(doy)
las_mean$year = "mean"
# 5 days moving average for 2018 and 2022:
las_18 <- las %>% mutate(year =year(Date)) %>% mutate(doy =yday(Date)) %>% filter(year == 2018)
%>% arrange(doy)
las_18$Rff_5d <- rollapply(las_18$mean_Reco_DT_U50, width=5, FUN=function(x) mean(x,
na.rm=TRUE), partial=TRUE, align="center")

## 2022 ##
las_22 <- las %>% mutate(year =year(Date)) %>% mutate(doy =yday(Date)) %>% filter(year == 2022)
%>% arrange(doy)
las_22$Rff_5d <- rollapply(las_22$mean_Reco_DT_U50, width=5, FUN=function(x) mean(x,
na.rm=TRUE), partial=TRUE, align="center")

## Soil Respiration (SR) fluxes
load("CH_LAE_SR_2022.RData")
sr.day <- sr.22 %>% mutate(doy = yday(timestamp)) %>% group_by(doy) %>% summarise(mean_flux =
mean(flux, na.rm = T),
                                                                                    se_flux =
std.error(flux, na.rm = T))
```

Now the data are ready to be plotted:

```
#NEP
s <- ggplot() +
  geom_vline(xintercept = 120, col = "#05A66C", lty = "dashed") +
  geom_vline(xintercept = 275, col = "#05A66C", lty = "dashed") +
  # rest of the years
  geom_line(data = lae_mean, aes(x = doy, y= NEP, col = "Mean 2004-2022", lty = "Mean 2005-
2022")) +
  geom_ribbon(data = lae_mean, aes(x = doy, ymin = NEP - NEP_se, ymax = NEP + NEP_se), fill =
color3[4], alpha = 0.2) +
  #2015
  geom_line(data = lae_15, aes(x = doy, y = NEP_5d, col = "2015"), alpha = .8) +
```

```r
  geom_line(data = lae_15 %>% filter(doy %in% c(188:202)), aes(x = doy, y = NEP_5d, col =
"2015", lty = "2015"), linewidth = 1.2) +
  geom_line(data = lae_15 %>% filter(doy %in% c(214:225)), aes(x = doy, y = NEP_5d, col =
"2015", lty = "2015"), linewidth = 1.2) +
  # 2018
  geom_line(data = lae_18, aes(x = doy, y = NEP_5d, col = "2018", lty = "2018"), alpha = .8) +
  geom_line(data = lae_18 %>% filter(doy %in% c(204:235)), aes(x = doy, y = NEP_5d, col =
"2018", lty = "2018"), linewidth = 1.2) +
  # 2022
  geom_line(data = lae_22, aes(x = doy, y= NEP_5d, col = "2022", lty = "2022"), alpha = .8) +
  geom_line(data = lae_22%>% filter(doy %in% c(195:216)), aes(x = doy, y = NEP_5d, col =
"2022", lty = "2022"),
            linewidth = 1.2) +



  scale_color_manual(values = color3, name = "Years", labels = c("2015", "2018","2022","Mean
2005-2022")) +
  scale_linetype_manual(values = c(1,1,1,1), name = "Years", labels = c("2015",
"2018","2022","Mean 2005-2022"),
                        guide=guide_legend(override.aes=list(linetype=c(1,1,1,1),
lwd=c(1,1,1,1)))) +
  scale_y_continuous(name = "NEP", breaks = seq(-7, 12, by = 4)) +
  scale_x_continuous(breaks= c(1, 32, 60, 91, 121, 152, 182, 213, 244, 274, 305, 335, 366),
                     labels = c()) +
  xlab("") +
  th1 +  theme(legend.position = c(0.5, 0.1), legend.direction = "horizontal",
               axis.text.x = element_blank(), axis.title.x = element_blank(), legend.text =
element_text(size = 10))

#GPP
g <- ggplot() +
  geom_vline(xintercept = 120, col = "#05A66C", lty = "dashed") +
  geom_vline(xintercept = 275, col = "#05A66C", lty = "dashed") +
  # rest of the years
  geom_line(data = lae_mean, aes(x = doy, y = GPP_DT, col = "Mean", lty = "Mean")) +
  geom_ribbon(data = lae_mean, aes(x = doy, ymin = GPP_DT - GPP_se, ymax = GPP_DT + GPP_se),
fill = color3[4], alpha = 0.2) +
  #2015
  geom_line(data = lae_15, aes(x = doy, y = GPP_5d, col = "2015", lty = "2015"), alpha = .8) +
  geom_line(data = lae_15 %>% filter(doy %in% c(188:202)), aes(x = doy, y = GPP_5d, col =
"2015", lty = "2015"), linewidth = 1.2) +
  geom_line(data = lae_15 %>% filter(doy %in% c(214:225)), aes(x = doy, y = GPP_5d, col =
"2015", lty = "2015"), linewidth = 1.2) +
  # 2018
  geom_line(data = lae_18, aes(x = doy, y = GPP_5d, col = "2018", lty = "2018"), alpha = .8) +
  geom_line(data = lae_18 %>% filter(doy %in% c(204:235)), aes(x = doy, y = GPP_5d, col =
"2018", lty = "2018"), linewidth = 1.2) +
  # 2022
  geom_line(data = lae_22, aes(x = doy, y= GPP_5d, col ="2022", lty = "2022"), alpha = .8) +
  geom_line(data = lae_22%>% filter(doy %in% c(195:216)), aes(x = doy, y = GPP_5d, col =
"2022",  lty = "2022"), linewidth = 1.2) +

  scale_color_manual(values = color3, name = "Years", labels = c("2015", "2018","2022","Mean"))
+
  scale_linetype_manual(values = c(1,1,1,1), name = "Years", labels = c("2015",
"2018","2022","Mean")) +

  scale_y_continuous(name = "GPP", breaks = seq(0,16, by = 4)) +
  scale_x_continuous(breaks= c(1, 32, 60, 91, 121, 152, 182, 213, 244, 274, 305, 335, 366),
                     labels = c()) +

  xlab("") +
  th1 + theme(legend.position = "none", axis.text.x = element_blank(), axis.title.x = 
element_blank())

#Reco
r <- ggplot() +
  geom_vline(xintercept = 120, col = "#05A66C", lty = "dashed") +
  geom_vline(xintercept = 275, col = "#05A66C", lty = "dashed") +
  # rest of the years
  geom_line(data = lae_mean, aes(x = doy, y= Reco_DT, col = "Mean", lty = "Mean")) +
  geom_ribbon(data = lae_mean, aes(x = doy, ymin = Reco_DT - Reco_se, ymax = Reco_DT + 
Reco_se), fill = color3[4], alpha = 0.2) +
  #2015
  geom_line(data = lae_15, aes(x = doy, y = Reco_5d, col = "2015", lty = "2015"), alpha = .8) +
  geom_line(data = lae_15 %>% filter(doy %in% c(188:202)), aes(x = doy, y = Reco_5d, col = 
"2015", lty = "2015"), linewidth = 1.2) +
  geom_line(data = lae_15 %>% filter(doy %in% c(214:225)), aes(x = doy, y = Reco_5d, col = 
"2015", lty = "2015"), linewidth = 1.2) +
  # 2018
  geom_line(data = lae_18, aes(x = doy, y = Reco_5d, col = "2018", lty = "2018"), alpha = .8) +
  geom_line(data = lae_18 %>% filter(doy %in% c(204:235)), aes(x = doy, y = Reco_5d, col = 
"2018", lty = "2018"), linewidth = 1.2) +
  # 2022
  geom_line(data = lae_22, aes(x = doy, y= Reco_5d, col = "2022", lty = "2022"), alpha = .8) +
  geom_line(data = lae_22%>% filter(doy %in% c(195:216)), aes(x = doy, y = Reco_5d, col 
="2022", lty = "2022"), linewidth = 1) +

  scale_color_manual(values = color3, name = "Years", labels = c("2015", "2018","2022","Mean")) 
+
  scale_linetype_manual(values = c(1,1,1,1), name = "Years", labels = c("2015", 
"2018","2022","Mean")) +


  scale_y_continuous(name = "Reco", breaks = seq(0, 16, by = 4), limits = c(0, 17)) +
  scale_x_continuous(breaks= c(1, 32, 60, 91, 121, 152, 182, 213, 244, 274, 305, 335, 366),,
                     labels = c()) +

  xlab("") + th1  + theme(legend.position = "none", axis.text.x = element_blank(), axis.title.x 
= element_blank())


#Rff
f <- ggplot() +
  geom_vline(xintercept = 120, col = "#05A66C", lty = "dashed") +
  geom_vline(xintercept = 275, col = "#05A66C", lty = "dashed") +
  # rest of the years
  geom_line(data = las_mean, aes(x = doy, y= mean_Rff, col = "Mean 2019-2021", linetype = "Mean 
2019-2021")) +
  geom_ribbon(data = las_mean, aes(x = doy, ymin = mean_Rff - se_Rff, ymax = mean_Rff + 
se_Rff),
              fill = color3[4], alpha = 0.2) +
  # 2018
  geom_line(data = las_18, aes(x = doy, y = Rff_5d), alpha = .8, col = "#6248BF") +
  geom_line(data = las_18 %>% filter(doy %in% c(204:235)),
            aes(x = doy, y = Rff_5d),
            linewidth = 1.2, col = "#6248BF") +
  # 2022
  geom_line(data = las_22, aes(x = doy, y= Rff_5d), alpha = .8, col = "#F86624") +
  geom_line(data = las_22 %>% filter(doy %in% c(195:216)), aes(x = doy, y = Rff_5d),
            linewidth = 1.2, col = "#F86624") +
  # SR 2022
  geom_pointrange(data = sr.day, aes(x = doy, y = mean_flux, ymin = mean_flux - se_flux, ymax = 
mean_flux+se_flux,
                                     fill = "SR 2022"), pch = 21) +

  scale_linetype_manual(values = c("Mean 2019-2021" = 4),
                        labels = c("Mean 2019-2021"), name  = "",
                         guide=guide_legend(override.aes=list(linetype=c(4), lwd=c(1)))) +
  scale_color_manual(values = c("Mean 2019-2021" = "grey2"),
                     labels = c("Mean 2019-2021"), name = "") +
  scale_fill_manual(values = "#E03916", labels = "SR 2022", name = "") +

  scale_y_continuous(name = "Rff", breaks = seq(0, 8, by = 2), limits = c(-0.1,6.2)) +
  scale_x_continuous(breaks= c(1, 32, 60, 91, 121, 152, 182, 213, 244, 274, 305, 335, 366),
                     labels = c("J", "F", "M", "A", "M", "J", "J", "A", "S", "O", "N", "D",
"J")) +

  xlab("") + th1  + theme(legend.position = c(0.55, 0.1), legend.box = "horizontal",
                          legend.text = element_text(size = 10), legend.spacing.x = unit(0.05,
'cm')) +
  guides()

t<-ggarrange(s,g,r,f, ncol = 1, nrow = 4,  labels = c("a", "c", "e", "g"), heights =
c(1,1,1,1.3),
               label.x = 0.12, label.y = 0.95, common.legend = F,
               align = "v")

## Warning: Removed 36 rows containing missing values (`geom_line()`).

t
```



We want to calculate the mean CO2 flux and sd during the CSAD events.

```
mean.2015.co2 <- lae_15 %>% mutate(DOY = yday(Date)) %>% filter(DOY %in% c(188:202, 214:225))
%>%
  summarise(avg.NEP = mean(mean_NEP, na.rm = T), sd.NEP = sd(mean_NEP, na.rm = T),
            avg.GPP = mean(mean_GPP_DT, na.rm = T), sd.GPP = sd(mean_GPP_DT, na.rm = T),
            avg.Reco = mean(mean_Reco_DT, na.rm = T), sd.Reco = sd(mean_Reco_DT, na.rm= T),
            )


mean.2018.co2 <- lae_18 %>% mutate(DOY = yday(Date)) %>% filter(DOY %in% c(204:235)) %>%
  summarise(avg.NEP = mean(mean_NEP, na.rm = T), sd.NEP = sd(mean_NEP, na.rm = T),
            avg.GPP = mean(mean_GPP_DT, na.rm = T), sd.GPP = sd(mean_GPP_DT, na.rm = T),
            avg.Reco = mean(mean_Reco_DT, na.rm = T), sd.Reco = sd(mean_Reco_DT, na.rm= T),
            )

mean.2022.co2 <- lae_22 %>% mutate(DOY = yday(Date)) %>% filter(DOY %in% c(195:216)) %>%
  summarise(avg.NEP = mean(mean_NEP, na.rm = T), sd.NEP = sd(mean_NEP, na.rm = T),
            avg.GPP = mean(mean_GPP_DT, na.rm = T), sd.GPP = sd(mean_GPP_DT, na.rm = T),
            avg.Reco = mean(mean_Reco_DT, na.rm = T), sd.Reco = sd(mean_Reco_DT, na.rm= T),
            )


mean.co2 <- lae_mean %>% filter(doy %in% c(188:235)) %>%
  summarise(avg.NEP = mean(NEP, na.rm = T), sd.NEP = sd(NEP, na.rm = T),
            avg.GPP = mean(GPP_DT, na.rm = T), sd.GPP = sd(GPP_DT, na.rm = T),
            avg.Reco = mean(Reco_DT, na.rm = T), sd.Reco = sd(Reco_DT, na.rm= T),
            )

mean.2018.co2.las <- las_18 %>% filter(doy %in% c(204:235)) %>%
  summarise(avg.Rff = mean(mean_Reco_DT_U50, na.rm = T), sd.Rff= sd(mean_Reco_DT_U50, na.rm =
T))

mean.2022.co2.las <- las_22 %>% filter(doy %in% c(195:216)) %>%
  summarise(avg.Rff = mean(mean_Reco_DT_U50, na.rm = T), sd.Rff= sd(mean_Reco_DT_U50, na.rm =
T))


mean.co2.las <- las_mean %>% filter(doy %in% c(195:235)) %>%
  summarise(avg.Rff = mean(mean_Rff, na.rm = T), sd.Rff= sd(mean_Rff, na.rm = T))
```

The second part of figure 3 is the bar charts with the difference between the mean 2004-2022 (or 2019-2021 for Rff) and the fluxes, calculated per each day during the CSAD and then summed. The standard error of the mean values was also cumolated for the CSAD period to obtain the standard error on the final flux.

Sum of the difference:

```
diff_mean = function(x) {x - xmean}

# 2015
xmean = lae_mean %>% filter(doy %in% c(188:202, 214:225))%>% arrange(doy) %>%
dplyr::select(NEP)
dNEP15 <- sum(lae_15 %>% filter(doy %in% c(188:202, 214:225)) %>% arrange(Date) %>%
mutate(delta_NEP = diff_mean(mean_NEP)) %>%
  dplyr::select(delta_NEP))

xmean = lae_mean %>% filter(doy %in% c(188:202, 214:225))%>% arrange(doy)  %>%
dplyr::select(GPP_DT)
dGPP15 <- sum(lae_15 %>% filter(doy %in% c(188:202, 214:225)) %>% arrange(Date) %>%
mutate(delta_GPP = diff_mean(mean_GPP_DT)) %>%
    dplyr::select(delta_GPP))

xmean = lae_mean %>% filter(doy %in% c(188:202, 214:225)) %>% arrange(doy) %>%
dplyr::select(Reco_DT)
dReco15 <- sum(lae_15 %>% filter(doy %in% c(188:202, 214:225)) %>% arrange(Date) %>%
mutate(delta_Reco = diff_mean(mean_Reco_DT)) %>%
                dplyr::select(delta_Reco))
```

```
# 2018

xmean = lae_mean %>% filter(doy %in% c(204:235)) %>% arrange(doy) %>% dplyr::select(NEP)
dNEP18 <- sum(lae_18%>% filter(doy %in% c(204:235)) %>% arrange(Date) %>% mutate(delta_NEP =
diff_mean(mean_NEP)) %>%
                dplyr::select(delta_NEP))

xmean = lae_mean %>% filter(doy %in% c(204:235)) %>% arrange(doy) %>% dplyr::select(GPP_DT)
dGPP18 <- sum(lae_18 %>% filter(doy %in% c(204:235)) %>% arrange(Date) %>% mutate(delta_GPP =
diff_mean(mean_GPP_DT)) %>%
                dplyr::select(delta_GPP))

xmean = lae_mean %>% filter(doy %in% c(204:235)) %>% arrange(doy) %>% dplyr::select(Reco_DT)
dReco18 <- sum(lae_18 %>% filter(doy %in% c(204:235)) %>% arrange(Date) %>% mutate(delta_Reco =
diff_mean(mean_Reco_DT)) %>%
                 dplyr::select(delta_Reco))

# 2022
xmean = lae_mean %>% filter(doy %in% c(195:216)) %>% arrange(doy) %>% dplyr::select(NEP)
dNEP22 <- sum(lae_22%>% filter(doy %in% c(195:216)) %>% arrange(Date) %>% mutate(delta_NEP =
diff_mean(mean_NEP)) %>%
                dplyr::select(delta_NEP))

xmean = lae_mean %>% filter(doy %in% c(195:216)) %>% arrange(doy) %>% dplyr::select(GPP_DT)
dGPP22 <- sum(lae_22 %>% filter(doy %in% c(195:216)) %>% arrange(Date) %>% mutate(delta_GPP =
diff_mean(mean_GPP_DT)) %>%
                dplyr::select(delta_GPP))

xmean = lae_mean %>% filter(doy %in% c(195:216)) %>% arrange(doy) %>%  dplyr::select(Reco_DT)
dReco22 <- sum(lae_22 %>% filter(doy %in% c(195:216)) %>% arrange(Date) %>% mutate(delta_Reco =
diff_mean(mean_Reco_DT)) %>%
                 dplyr::select(delta_Reco))

# Rff
#2018
xmean = las_mean %>% filter(doy %in% c(204:235)) %>%arrange(doy) %>%  dplyr::select(mean_Rff)
cdh_ff_18<- las_18 %>% filter(doy %in% c(204:235)) %>% arrange(doy) %>% mutate(delta_Rff =
diff_mean(mean_Reco_DT_U50))
# 2022
xmean = las_mean %>% filter(doy %in% c(195:216)) %>%arrange(doy) %>%  dplyr::select(mean_Rff)
cdh_ff_22 <- las_22 %>% filter(doy %in% c(195:216)) %>% arrange(doy) %>% mutate(delta_Rff =
diff_mean(mean_Reco_DT_U50))
```

Sum of the standard error:

```
se_NEP = c(sum(lae_mean %>% filter(doy %in% c(188:202, 214:225)) %>% dplyr::select(NEP_se)),
           sum(lae_mean %>% filter(doy %in% c(204:235)) %>% dplyr::select(NEP_se)),
           sum(lae_mean %>% filter(doy %in% c(195:216)) %>% dplyr::select(NEP_se)))

se_GPP = c(sum(lae_mean %>% filter(doy %in% c(188:202, 214:225)) %>% dplyr::select(GPP_se)),
           sum(lae_mean %>% filter(doy %in% c(204:235)) %>% dplyr::select(GPP_se)),
           sum(lae_mean %>% filter(doy %in% c(195:216)) %>% dplyr::select(GPP_se)))

se_Reco = c(sum(lae_mean %>% filter(doy %in% c(188:202, 214:225)) %>% dplyr::select(Reco_se)),
            sum(lae_mean %>% filter(doy %in% c(204:235)) %>% dplyr::select(Reco_se)),
            sum(lae_mean %>% filter(doy %in% c(195:216)) %>% dplyr::select(Reco_se)))

se_Rff = c(NA,
           sum(las_mean %>% filter(doy %in% c(204:235)) %>% dplyr::select(se_Rff)),
           sum(las_mean %>% filter(doy %in% c(195:216)) %>% dplyr::select(se_Rff)))
```

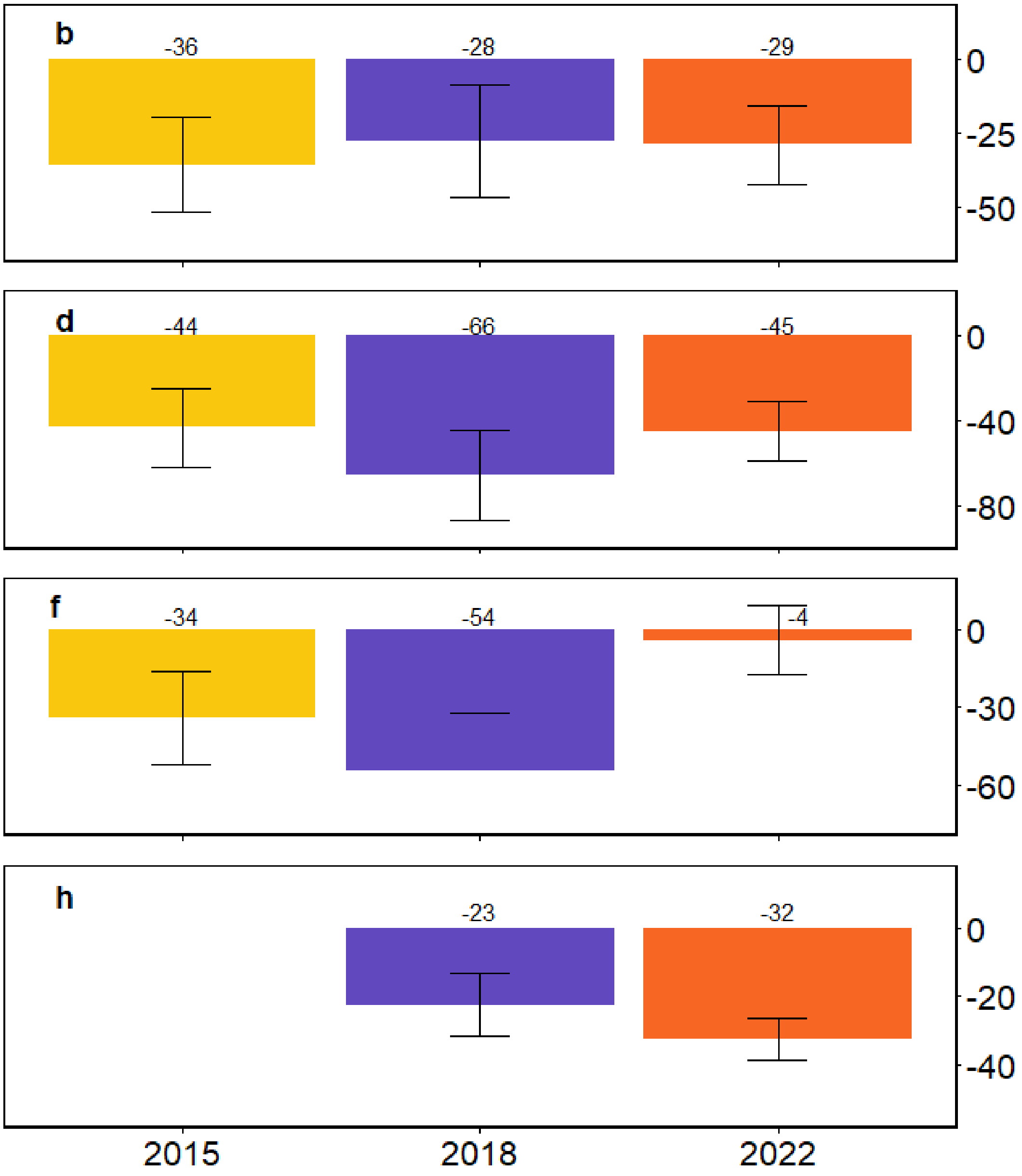
Creation of the the data set that we used for plotting:

```
year = c("2015", "2018", "2022")
dNEP = c(dNEP15, dNEP18, dNEP22)
dGPP = c(dGPP15, dGPP18, dGPP22)
dReco = c(dReco15, dReco18, dReco22)
dRff = c(NA, sum(cdh_ff_18$delta_Rff),  sum(cdh_ff_22$delta_Rff))

diff_cdh <- data.frame(year, dNEP, dGPP, dReco, dRff, se_NEP, se_GPP, se_Reco, se_Rff)
```

Data are ready to be plotted:

```
dn <- ggplot(diff_cdh) +
  geom_col(aes(x = year, y = dNEP, fill = year)) +
  geom_errorbar(aes(x = year, ymin = dNEP - se_NEP, ymax = dNEP + se_NEP), width =0.2) +
  geom_text(aes(x = year, y = 5, label=round(dNEP)), size = 4) +
  scale_fill_manual(values = color3) +
  scale_y_continuous(breaks = c(0, -25, -50), position = "right", limits = c(-65, 15)) +
  ylab("") +
  xlab("") +
  th1 + theme(legend.position = "none", axis.text.x = element_blank(), axis.title.x =
element_blank())

dg <- ggplot(diff_cdh) +
  geom_col(aes(x = year, y = dGPP, fill = year)) +
  geom_errorbar(aes(x = year, ymin = dGPP - se_GPP, ymax = dGPP + se_GPP), width =0.2) +
  geom_text(aes(x = year, y = 5,  label=round(dGPP)), size = 4) +
  scale_fill_manual(values = color3) +
  scale_y_continuous(breaks = c(0, -40, -80), position = "right", limits = c(-95, 15)) +
  ylab("") +
  xlab("") +
  th1 + theme(legend.position = "none", axis.text.x = element_blank(), axis.title.x =
element_blank())


dr <- ggplot(diff_cdh) +
  geom_col(aes(x = year, y = dReco, fill = year)) +
  geom_errorbar(aes(x = year, ymin = dReco - se_Reco, ymax = dReco + se_Reco), width =0.2) +
  geom_text(aes(x = year, y = 5, hjust = c(0.5, 0.5, -0.5),label=round(dReco)), size = 4) +
  scale_fill_manual(values = color3) +
  scale_y_continuous(breaks = c(0, -30, -60), position = "right", limits = c(-75,15)) +
  ylab("") +
  xlab("") +
  th1 + theme(legend.position = "none", axis.text.x = element_blank(), axis.title.x =
element_blank())

drf <- ggplot(diff_cdh) +
  geom_col(aes(x = year, y = dRff, fill = year)) +
  geom_errorbar(aes(x = year, ymin = dRff - se_Rff, ymax = dRff + se_Rff), width =0.2) +
  geom_text(aes(x = year, y = 5, label=round(dRff)), size = 4) +
  scale_fill_manual(values = color3[c(2,3)]) +
  scale_y_continuous(breaks = c(0, -20, -40), position = "right", limits = c(-55, 15)) +
  ylab("") +
  xlab("") +
  th1 + theme(legend.position = "none")
ggarrange(dn,dg,dr, drf, ncol = 1, nrow = 4,  labels = c("b", "d", "f", "h"),
          align = "v", heights = c(1,1,1,1.3), label.x = 0.05, label.y = 0.95)
```

```
## Warning: Removed 1 rows containing missing values (`position_stack()`).

## Warning: Removed 1 rows containing missing values (`geom_text()`).
```



4. Figure 4: CVI with Random Forest

We used a random forest model with conditional variable importance for the analysis of the drivers of NEP and Rff. For NEP we only considered the growing season while for the Rff we used the whole year. We used the R-packages *randomForest* and *party*.

Conditional variable importnace (CVI) for NEP:

```
# lae.daily <- lae %>% mutate(year == year(Date)) %>% filter(year > 2004) %>%
dplyr::select(year, doy, mean_NEP, mean_VPD_f, mean_Rg_f, mean_Tair_f, mean_SWC_stnd) %>%
#   filter(doy %in% 121:274)
#
#
# colnames(lae.daily) <- c("Year", "DOY", "NEP", "VPD_f", "Rg_f", "Tair_f", "SWC_stnd")
# for(i in c(2005:2022))
# {
```

```
#    Lae_SHAP_train = lae.daily %>% dplyr::select(Year, DOY, NEP, Tair_f, VPD_f, Rg_f,
SWC_stnd) %>% filter(Year ==i) %>%
#      na.omit()
#  for(j in c(1:10))
#   {
#    RF_cforest = cforest(NEP ~  DOY + VPD_f  + Rg_f + Tair_f + SWC_stnd, controls =
cforest_unbiased(), data = Lae_SHAP_train)
#    RF_varImp = varimp(RF_cforest, conditional = TRUE, nperm = 1)
#    RF_varImp = as.data.frame(RF_varImp)
#    RF_varImp$Var = rownames(RF_varImp)
#    RF_varImp = RF_varImp %>% mutate(Year = i, rep = j)
#     if(j == 1)
#     {
#       RF_varImp1 = RF_varImp
#     }
#     if(j>1)
#     {
#       RF_varImp1 = rbind(RF_varImp,RF_varImp1)
#     }
#     print(j)
#   }
#
#   if(i == 2005)
#   {
#     RF_varImp_All = RF_varImp1
#   }
#   if(i > 2005)
#   {
#     RF_varImp_All = rbind(RF_varImp1,RF_varImp_All)
#   }
#   print(i)
# }
#
# Cond_VarImp_daylight = RF_varImp_All %>%
#   group_by(Var) %>% summarise(Imp_LTm = mean(RF_varImp), ImpLTsd = 2*sd(RF_varImp)/sqrt(19))
%>%
#   right_join(., RF_varImp_All %>% filter(Year > 2004) %>% group_by(Year, Var) %>%
#                 summarise(Impm = mean(RF_varImp), Impsd = 3*sd(RF_varImp)), by = 'Var')
#
# save(Cond_VarImp, file="LAE_Cond_VarImp_RF.RData")
```

Same model but for Rff:

```
# MF.day <- las %>% mutate(year = lubridate::year(Date)) %>% mutate(doy =
lubridate::yday(Date))
# MF.day$mean_Rg <- PPFD.to.Rg(MF.day$mean_PPFD_IN_FF, J_to_mol = 4.6, frac_PAR = 0.5)
# LAS_subset <- MF.day %>% dplyr::select(year, doy, mean_Reco_U50, mean_SWC_stnd_20,
mean_Ta_FF, mean_Rg,  mean_TS_5cm) %>% filter(doy %in% 121:274)
# colnames(LAS_subset) <- c("Year", "DOY", "Rff", "SWC", "Tair_ff", "Rg", "TS")
#
# #takes up to 40 min for the whole year
# for(i in c(2018, 2019, 2020, 2021, 2022))
# {
#  LAS_train = LAS_subset %>% filter(Year == i) %>% na.omit()
#  for(j in c(1:10))
#  {
#    RF_cforest.las = cforest(Rff ~  DOY  + Rg+ Tair_ff + SWC +TS, controls =
cforest_unbiased(), data = LAS_train)
#    RF_varImp.las = varimp(RF_cforest.las, conditional = TRUE, nperm = 1)
#    RF_varImp.las = as.data.frame(RF_varImp.las)
#    RF_varImp.las$Var = rownames(RF_varImp.las)
#    RF_varImp.las = RF_varImp.las %>% mutate(Year = i, rep = j)
#    if(j == 1)
#     {
#       RF_varImp1.las = RF_varImp.las
#     }
#     if(j>1)
#     {
#       RF_varImp1.las = rbind(RF_varImp.las,RF_varImp1.las)
#     }
#     print(j)
#   }
#
#   if(i == 2018)
#   {
#     RF_varImp_All.las = RF_varImp1.las
#   }
#   if(i > 2018)
#   {
#     RF_varImp_All.las = rbind(RF_varImp1.las,RF_varImp_All.las)
#   }
#   print(i)
# }
#
# Cond_VarImp.las = RF_varImp_All.las %>%
# group_by(Var) %>% summarise(Imp_LTm = mean(RF_varImp.las), ImpLTsd =
2*sd(RF_varImp.las)/sqrt(19)) %>%
# right_join(., RF_varImp_All.las %>% filter(Year > 2004) %>% group_by(Year, Var) %>%
# summarise(Impm = mean(RF_varImp.las), Impsd = 3*sd(RF_varImp.las)), by = 'Var')#%>%
filter(Var %in% c('VPD_f','SWC_stnd')) %>% table(Cond_VarImp.las$Var)
#
# LAS_Cond_VarImp_RF <- Cond_VarImp.las
# save(Cond_VarImp.las, file = "CVI_LAS.RData")
```

We saved the results from the analysis in a file that we now import here to plot the results.

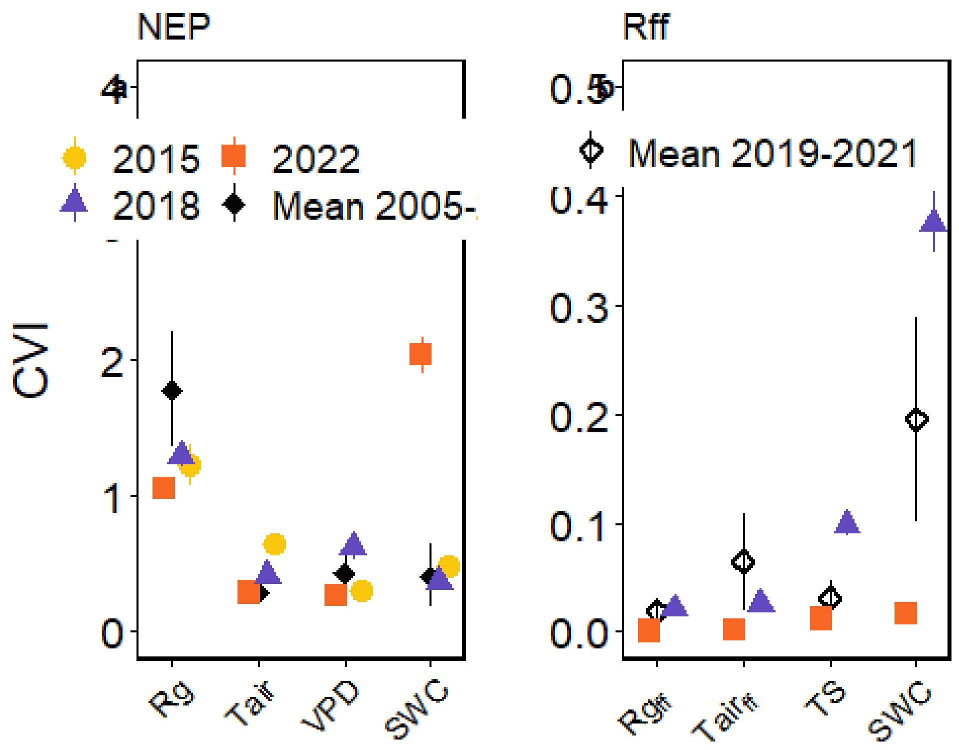
```
# NEP
load("LAE_Cond_VarImp_RF.RData")
nep <- Cond_VarImp%>% filter(Year %in% c(2015, 2018, 2022)) %>%
  ggplot(.) +
  geom_pointrange(data = . %>% filter(Year == 2022),
                  mapping = aes(x = Var, y = Imp_LTm, ymin = Imp_LTm - ImpLTsd, ymax = Imp_LTm
+ ImpLTsd, color = 'Mean 2005-2022', shape = 'Mean 2005-2022'),
                  position = position_nudge(x = 0), size = 0.8, stroke = 1.1) +
  geom_pointrange(data = . %>% filter(Year == 2022),
                  mapping = aes(x = Var, y = Impm, ymin = Impm - Impsd, ymax = Impm + Impsd,
color = '2022',
                                 shape = '2022'), position = position_nudge(x = -0.1), size =
0.8,stroke = 1.1) +
  geom_pointrange(data = . %>% filter(Year == 2015),
                  mapping = aes(x = Var, y = Impm, ymin = Impm - Impsd, ymax = Impm + Impsd,
color = '2015',
                                 shape = '2015'), position = position_nudge(x = 0.2), size =
0.8,stroke = 1.1) +
  geom_pointrange(data = . %>% filter(Year == 2018),
                  mapping = aes(x = Var, y = Impm, ymin = Impm - Impsd, ymax = Impm + Impsd,
color = '2018',
                        shape = '2018'), position = position_nudge(x = 0.1), size = 0.8, stroke
= 1.1) +
  labs(x = 'Variable', y = 'CVI') +
  ylim(0, 5) +
  ggtitle("NEP") +
  scale_y_continuous(limits = c(0, 4))+
  scale_x_discrete(limits = c("Rg_f", "Tair_f", "VPD_f", "SWC_stnd"), labels =
c('Rg','Tair','VPD', "SWC"), expand = c(0.1, 0.1)) + th1 +
  scale_color_manual('Period', values = c('#F9C80E','#6248BF','#F86624','grey2')) +
  scale_shape_manual('Period', values = c(19,17,15,18)) +th1 + guides(col = guide_legend(ncol =
2), shape = guide_legend(ncol = 2)) +
  theme(axis.text.x = element_text(angle = 45,hjust = 1, size = 12), legend.direction =
"horizontal",
        legend.position = c(0.5, 0.8),axis.title.x = element_blank())
```

```
## Scale for y is already present.
## Adding another scale for y, which will replace the existing scale.
```

```
# RFF

load("CVI_LAS.RData")

rff <- Cond_VarImp.las %>% filter(Year %in% c(2018:2022)) %>%
  ggplot(.) +
  geom_pointrange(data = . %>% filter(Year %in% c(2019:2021)), mapping = aes(x = Var, y =
Imp_LTm, ymin = Imp_LTm - ImpLTsd, ymax = Imp_LTm + ImpLTsd,
                                                                             color = 'Mean
2019-2021', shape = 'Mean 2019-2021'),
                  position = position_nudge(x = 0), size = 0.5, stroke = 1.1) +
  geom_pointrange(data = . %>% filter(Year == 2022),
                  mapping = aes(x = Var, y = Impm, ymin = Impm - Impsd, ymax = Impm + Impsd
                                ), position = position_nudge(x = -0.1), size = 0.8,stroke =
1.1, col = "#F86624", shape = 15) +
  geom_pointrange(data = . %>% filter(Year == 2018),
                  mapping = aes(x = Var, y = Impm, ymin = Impm - Impsd, ymax = Impm + Impsd
                              ), position = position_nudge(x = 0.2), size = 0.8, stroke = 1.,
col = '#6248BF', shape = 17) +
  labs(x = 'Variable', y = 'CVI') +
 scale_x_discrete(limits = c("Rg", "Tair_ff", "TS","SWC"),
                  labels = c("Rg"["ff"]~"","Tair"["ff"]~"", "TS","SWC"), expand = c(0.1, 0.1))
+
  ylim(0, 0.5)+
  ggtitle("Rff") +
  theme(legend.position = 'right') +
  scale_color_manual('Period', values = c('grey2')) +
  scale_shape_manual('Period', values = c(5)) +th1 +
  theme(axis.text.x = element_text(angle = 45,hjust = 1, size = 12),
        legend.direction = "horizontal", legend.position = c(0.35, 0.85),
        axis.title.y = element_blank(),axis.title.x = element_blank())

cvi.plot <- plot_grid(nep, rff, ncol = 2, align = "hv", labels = "auto", label_x = 0.22,
label_y = 0.92, label_size = 12)
```

```
## Warning: Removed 1 rows containing missing values (`geom_pointrange()`).
## Removed 1 rows containing missing values (`geom_pointrange()`).
## Removed 1 rows containing missing values (`geom_pointrange()`).
## Removed 1 rows containing missing values (`geom_pointrange()`).

## Warning: Removed 3 rows containing missing values (`geom_pointrange()`).

## Warning: Removed 1 rows containing missing values (`geom_pointrange()`).

## Warning: Removed 1 rows containing missing values (`geom_segment()`).

## Warning: Removed 1 rows containing missing values (`geom_pointrange()`).
```

```
cvi.plot
```



5. Figure 5: SHAP for NEP

First we use the daily mean of NEP during the growing season to run the model with the *xgboost* package and then we use the *SHAPforxgboost* package to obtain the SHAP values.

```
load("Level_4_PIFluxMeteo_allVars_LAE_2004_2022.RData") # to load the 30 min fluxes and
calculate the daylight mean
lae_30MIN <- Level4_Fluxes_LAE_2004_2022_All_Variables
lae_30MIN = lae_30MIN %>% mutate(Date = date(Timestamp)) %>% mutate(doy = yday(Timestamp)) %>%
mutate(year = year(Timestamp)) %>%
  mutate(SWC_type = if_else(Date < as.Date('2020-03-30'),'S1','S2'))
lae_30MIN = lae_30MIN %>% group_by(SWC_type) %>% dplyr::mutate(SWC_stnd =
as.numeric(scale(SWC_20cm))) %>% ungroup()
lae_30MIN = lae_30MIN %>% mutate(NEP = -NEE_f)

for(yr in c(2005:2022))
{
  y_var <-  "NEP"
  dataXY = lae_30MIN %>% filter(year == yr) %>% filter(Rg_f >10) %>% filter(doy %in% 120:274)
%>%
    dplyr::select(doy, Date, year, NEP, Tair_f, VPD_f, Rg_f, SWC_stnd) %>%
    group_by(Date) %>% summarise_if(is.numeric, mean, na.rm = TRUE) %>%
    dplyr::select(-year, -Date) %>% na.omit()

  dataX <- as.matrix(dataXY %>% dplyr::select(-NEP))

  params <- list(objective = "reg:squarederror",  # For regression
                 eta = 0.02,
                 max_depth = 10,
                 gamma = 0.01,
                 subsample = 0.98,
                 colsample_bytree = 0.86)
```

```r
  mod <- xgboost::xgboost(data = dataX,
                          label =  as.matrix(dataXY$NEP),
                          params = params, nrounds = 200,
                          verbose = FALSE,
                          early_stopping_rounds = 8)

  shap_values <- shap.values(xgb_model = mod, X_train = dataX)
  str(shap_values)
  # remember that it only works with matrix, so add here o before a matrix conversion
  # The ranked features by mean |SHAP|

  # To prepare the long-format data:
  shap_long <- shap.prep(xgb_model = mod, X_train = dataX)
  shap.plot.summary(shap_long)
  shap_long_cast = inner_join(reshape2::dcast(shap_long, ID ~ variable, value.var =
c('value')),
                              shap_long %>% filter(variable == 'doy') %>% mutate(DOY_day =
rfvalue) %>% dplyr::select(ID, DOY_day), by = 'ID')

  shap_long_cast$Year = yr
  shap_long$Year = yr
  if(yr == 2005)
  {
    SHAPs_cast_CDH_LAE = shap_long_cast
    SHAPs_CDH_LAE = shap_long
  }

  if(yr > 2005)
  {
    SHAPs_cast_CDH_LAE = rbind(SHAPs_cast_CDH_LAE, shap_long_cast)
    SHAPs_CDH_LAE = rbind(SHAPs_CDH_LAE, shap_long)
  }
  print(yr)
}
```

```
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -3.96 -3.54 -2.79 -2.54 -2.7 ...
##   ..$ Tair_f  : num [1:155] -0.8606 -0.6991 -0.583 0.0129 -0.355 ...
##   ..$ VPD_f   : num [1:155] -1.8 -1.72 -1.59 -1.32 -1.06 ...
##   ..$ Rg_f    : num [1:155] -0.0372 0.5533 0.5684 -1.6969 0.0273 ...
##   ..$ SWC_stnd: num [1:155] -0.0351 -0.1311 -0.0815 -0.2714 -0.085 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.177 0.884 0.795 0.385 0.229
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "VPD_f" "doy" "Tair_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 5.91
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2005
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    149 obs. of  5 variables:
##   ..$ doy     : num [1:149] -5.06 -5.16 -4.88 -3.56 -4 ...
##   ..$ Tair_f  : num [1:149] 0.40595 0.18693 -0.00716 0.05642 -0.11614 ...
##   ..$ VPD_f   : num [1:149] -0.1832 0.111 -0.1261 -0.0745 -0.163 ...
##   ..$ Rg_f    : num [1:149] -0.00151 0.22828 0.8131 0.88769 0.53711 ...
##   ..$ SWC_stnd: num [1:149] 0.06881 0.09525 0.0351 0.03348 -0.00932 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.316 0.602 0.541 0.374 0.285
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "Tair_f" "VPD_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 5.74
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2006
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] 0.379 0.957 0.405 0.991 0.406 ...
##   ..$ Tair_f  : num [1:155] 0.0132 0.9233 0.5074 0.9378 0.8015 ...
##   ..$ VPD_f   : num [1:155] 0.235 0.352 0.235 0.471 0.239 ...
##   ..$ Rg_f    : num [1:155] 2.069 2.135 1.103 1.625 -0.441 ...
##   ..$ SWC_stnd: num [1:155] -0.14603 0.00105 -0.16345 0.00768 -0.29701 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.396 0.826 0.515 0.486 0.476
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "VPD_f" "SWC_stnd" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 7.35
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2007
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -4.59 -5.35 -4.86 -5.71 -4.18 ...
##   ..$ Tair_f  : num [1:155] 0.42 -1.261 0.429 0.128 0.209 ...
##   ..$ VPD_f   : num [1:155] 0.2426 -0.3019 0.4514 -0.0703 0.2669 ...
##   ..$ Rg_f    : num [1:155] 0.316 -0.486 0.452 0.286 0.388 ...
##   ..$ SWC_stnd: num [1:155] -1.74 -1.99 -1.81 -1.84 -1.62 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.137 0.851 0.815 0.311 0.292
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "VPD_f" "SWC_stnd" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 7.86
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2008
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    113 obs. of  5 variables:
##   ..$ doy     : num [1:113] 0.589 0.796 0.69 0.22 0.104 ...
##   ..$ Tair_f  : num [1:113] 0.7798 0.8626 0.0988 -0.5747 -0.0528 ...
##   ..$ VPD_f   : num [1:113] 0.471 0.296 0.583 -0.19 -1.783 ...
##   ..$ Rg_f    : num [1:113] 1.17 1.23 1.28 1.2 -5.37 ...
##   ..$ SWC_stnd: num [1:113] 0.4211 0.4967 0.4282 0.337 0.0797 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.567 0.621 0.608 0.473 0.437
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "Tair_f" "SWC_stnd" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 7.54
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2009
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    127 obs. of  5 variables:
##   ..$ doy     : num [1:127] -3.388 -3.658 -1.24 -0.467 -0.784 ...
##   ..$ Tair_f  : num [1:127] 0.1139 -0.0345 -0.0289 0.2328 0.022 ...
##   ..$ VPD_f   : num [1:127] 0.8194 0.5888 0.0189 1.4704 0.5418 ...
##   ..$ Rg_f    : num [1:127] 0.966 0.373 -1.342 1.496 -0.658 ...
##   ..$ SWC_stnd: num [1:127] 0.249 -0.127 -0.504 0.269 0.221 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.619 0.795 0.673 0.272 0.211
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "VPD_f" "doy" "Tair_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 8.52
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2010
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    153 obs. of  5 variables:
##   ..$ doy     : num [1:153] -1.826 0.172 0.918 2.078 1.854 ...
##   ..$ Tair_f  : num [1:153] 0.339 0.535 0.593 1.337 1.258 ...
##   ..$ VPD_f   : num [1:153] 0.23 0.466 0.396 0.717 0.766 ...
##   ..$ Rg_f    : num [1:153] 0.714 1.171 1.147 1.428 1.36 ...
##   ..$ SWC_stnd: num [1:153] 0.51 0.625 0.599 0.797 0.696 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.561 1.099 0.774 0.719 0.548
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "SWC_stnd" "Tair_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 8.88
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2011
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    153 obs. of  5 variables:
##   ..$ doy     : num [1:153] -7.13 -6.21 -4.98 -5.08 -3.98 ...
##   ..$ Tair_f  : num [1:153] -0.152 0.136 0.218 0.193 0.323 ...
##   ..$ VPD_f   : num [1:153] -0.80955 -0.29624 -0.01531 -0.52168 0.00132 ...
##   ..$ Rg_f    : num [1:153] -0.384 -0.207 -0.302 0.218 1.278 ...
##   ..$ SWC_stnd: num [1:153] -0.379 -0.338 -0.31 -0.292 -0.286 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.618 1.598 0.472 0.352 0.23
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "VPD_f" "Tair_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 9.06
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2012
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -1.143 -2.557 -3.139 -0.894 -1.375 ...
##   ..$ Tair_f  : num [1:155] -0.112 0.455 -0.559 0.472 0.559 ...
##   ..$ VPD_f   : num [1:155] -1.392 0.227 0.118 -1.138 -1.213 ...
##   ..$ Rg_f    : num [1:155] -7.5027 1.3544 0.0901 -6.9661 -3.3843 ...
##   ..$ SWC_stnd: num [1:155] -0.324 -0.498 -1.206 -0.377 -0.384 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.973 0.804 0.775 0.597 0.463
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "Tair_f" "SWC_stnd" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 8.79
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2013
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -0.636 -0.594 -0.234 0.317 -1.077 ...
##   ..$ Tair_f  : num [1:155] 1.422 0.453 1.203 1.885 0.475 ...
##   ..$ VPD_f   : num [1:155] -1.104 -0.424 -1.13 -0.846 0.648 ...
##   ..$ Rg_f    : num [1:155] -4.047 -0.321 -4.84 -2.832 1.246 ...
##   ..$ SWC_stnd: num [1:155] 0.628 0.159 0.506 0.644 -0.567 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.929 0.8 0.589 0.448 0.441
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "VPD_f" "doy" "SWC_stnd" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 7.9
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2014
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -3.03 -1.21 -1.99 -1.26 -1.48 ...
##   ..$ Tair_f  : num [1:155] 0.4012 -0.0987 0.773 0.7622 -0.2883 ...
##   ..$ VPD_f   : num [1:155] 0.295 -1.547 -0.169 -1.317 -0.115 ...
##   ..$ Rg_f    : num [1:155] 0.826 -7.673 0.331 -6.727 -0.84 ...
##   ..$ SWC_stnd: num [1:155] -0.208 -0.066 -0.2258 -0.0718 -0.2242 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.673 0.97 0.718 0.637 0.62
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "Tair_f" "VPD_f" "SWC_stnd" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 7.93
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2015
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -3.92 -2.99 -2.22 -6.65 -6.28 ...
##   ..$ Tair_f  : num [1:155] -0.4299 0.0356 -0.8629 -1.5347 -1.6489 ...
##   ..$ VPD_f   : num [1:155] 1.079 0.044 -2.032 -0.473 -0.506 ...
##   ..$ Rg_f    : num [1:155] 1.046 1.052 -7.392 0.521 0.619 ...
##   ..$ SWC_stnd: num [1:155] -0.0203 -0.1559 -0.4415 0.2642 -0.1243 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.991 1.377 0.964 0.43 0.318
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "VPD_f" "Tair_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 9.22
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2016
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -4.43 -2.42 -4.58 -2.73 -3.86 ...
##   ..$ Tair_f  : num [1:155] 0.28 -0.033 -0.345 0.307 -0.417 ...
##   ..$ VPD_f   : num [1:155] 0.0682 -1.0571 0.3411 0.6436 -1.047 ...
##   ..$ Rg_f    : num [1:155] 0.232 -4.385 0.521 0.77 -0.512 ...
##   ..$ SWC_stnd: num [1:155] -0.92 -0.361 -0.766 -0.607 -0.832 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.169 1.108 0.753 0.674 0.413
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "Tair_f" "VPD_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 7.95
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2017
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] 0.3145 0.2422 0.2831 -0.0567 0.371 ...
##   ..$ Tair_f  : num [1:155] 0.306 0.378 0.452 0.187 0.525 ...
##   ..$ VPD_f   : num [1:155] 0.442 -0.99 -0.689 -0.94 -0.509 ...
##   ..$ Rg_f    : num [1:155] 0.3061 -2.7165 -0.2219 -0.0787 -0.2489 ...
##   ..$ SWC_stnd: num [1:155] 0.992 0.674 1.187 0.578 1.269 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.203 1.125 0.734 0.563 0.396
##   ..- attr(*, "names")= chr [1:5] "SWC_stnd" "Rg_f" "VPD_f" "doy" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 8.84
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2018
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -2.89 -4 -3.59 -2.99 -2.31 ...
##   ..$ Tair_f  : num [1:155] -0.121 0.318 0.516 -0.73 -0.686 ...
##   ..$ VPD_f   : num [1:155] -0.181 0.338 0.511 -0.874 -0.643 ...
##   ..$ Rg_f    : num [1:155] -1.628 0.359 1.027 -2.446 -1.902 ...
##   ..$ SWC_stnd: num [1:155] 0.384 0.164 0.221 -0.171 0.289 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.535 1.012 0.774 0.576 0.278
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "Tair_f" "VPD_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 10.2
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2019
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] 0.531 0.188 0.194 0.21 0.826 ...
##   ..$ Tair_f  : num [1:155] 0.673 0.91 0.759 0.693 1.2 ...
##   ..$ VPD_f   : num [1:155] -0.0119 -1.0947 -0.4062 -0.1595 1.0541 ...
##   ..$ Rg_f    : num [1:155] 0.43 -3.92 -3.49 -3.09 1.93 ...
##   ..$ SWC_stnd: num [1:155] -0.193 0.203 0.064 0.19 0.595 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.881 0.713 0.551 0.502 0.267
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "Tair_f" "doy" "VPD_f" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 9.37
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2020
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    155 obs. of  5 variables:
##   ..$ doy     : num [1:155] -3.8 -2.19 -2.96 -3.11 -4.92 ...
##   ..$ Tair_f  : num [1:155] -1.349 -0.811 -1.499 -0.699 -0.722 ...
##   ..$ VPD_f   : num [1:155] -0.157 -0.722 -0.281 0.203 -0.284 ...
##   ..$ Rg_f    : num [1:155] -0.519 -5.782 -1.225 1.422 0.491 ...
##   ..$ SWC_stnd: num [1:155] -0.087 0.3398 0.1577 0.1207 0.0272 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 2.099 0.992 0.487 0.408 0.302
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "doy" "Tair_f" "SWC_stnd" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 8.73
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2021
## List of 3
##  $ shap_score     :Classes 'data.table' and 'data.frame':    146 obs. of  5 variables:
##   ..$ doy     : num [1:146] -1.56 -2.17 -2.79 -1.19 -1.93 ...
##   ..$ Tair_f  : num [1:146] -0.162 -0.111 0.172 0.509 0.117 ...
##   ..$ VPD_f   : num [1:146] 0.442 0.806 0.401 0.805 0.551 ...
##   ..$ Rg_f    : num [1:146] -0.0638 0.6328 0.3661 0.8313 0.5982 ...
##   ..$ SWC_stnd: num [1:146] -0.1731 -0.1403 -0.1849 -0.0384 -0.2564 ...
##   ..- attr(*, ".internal.selfref")=<externalptr>
##  $ mean_shap_score: Named num [1:5] 1.009 0.932 0.897 0.661 0.551
##   ..- attr(*, "names")= chr [1:5] "Rg_f" "VPD_f" "doy" "SWC_stnd" ...
##  $ BIAS0          :Classes 'data.table' and 'data.frame':    1 obs. of  1 variable:
##   ..$ BIAS: num 8.55
##   ..- attr(*, ".internal.selfref")=<externalptr>
## [1] 2022

save(SHAPs_cast_CDH_LAE, file = 'SHAP_values_casted_NEP_2005_2022.RData')
save(SHAPs_CDH_LAE, file = 'SHAP_values_long_NEP_2005_2022.RData')
```

Datasets with the results from the SHAP analysis are created so we can use them for plotting:

```
load('SHAP_values_casted_NEP_2005_2022.RData')
col.fun <- c("#F25A38", "#6CB7DB", "#546C80", "#785216")
SHAPs_CDH_LAE$DOY_day <- SHAPs_CDH_LAE$ID + 119

## 2015
mean.cast.15 <- SHAPs_cast_CDH_LAE%>% filter(Year == 2015) %>% dplyr::select(Tair_f, Rg_f,
VPD_f, SWC_stnd, DOY_day) %>%
  filter(DOY_day %in% c(188:202, 214:225)) %>% summarise_if(is.numeric, mean, na.rm = T)
row.names(mean.cast.15) <- "mean_value"
mean.15 <- as.data.frame(t(mean.cast.15))
mean.15 <- rownames_to_column(mean.15, "vars") %>% filter(vars != "DOY_day")
err.cast.15 <- SHAPs_cast_CDH_LAE%>% filter(Year == 2015) %>% dplyr::select(Tair_f, Rg_f,
VPD_f, SWC_stnd, DOY_day) %>%
  filter(DOY_day %in% c(188:202, 214:225)) %>% summarise_if(is.numeric, std.error, na.rm = T)
row.names(err.cast.15) <- "se_value"
std.15 <- as.data.frame(t(err.cast.15))
std.15 <- rownames_to_column(std.15, "vars") %>% filter(vars != "DOY_day")
sum15.lae.csad <- merge(mean.15, std.15, by = "vars")
```

```
## 2018
mean.cast.18 <- SHAPs_cast_CDH_LAE%>% filter(Year == 2018) %>% dplyr::select(Tair_f, Rg_f,
VPD_f, SWC_stnd, DOY_day) %>%
  filter(DOY_day %in% 204:235) %>% summarise_if(is.numeric, mean, na.rm = T)
row.names(mean.cast.18) <- "mean_value"
mean.18 <- as.data.frame(t(mean.cast.18))
mean.18 <- rownames_to_column(mean.18, "vars") %>% filter(vars != "DOY_day")
err.cast.18 <- SHAPs_cast_CDH_LAE%>% filter(Year == 2018) %>% dplyr::select(Tair_f, Rg_f,
VPD_f, SWC_stnd, DOY_day) %>%
  filter(DOY_day %in% c(204:235)) %>% summarise_if(is.numeric, std.error, na.rm = T)
row.names(err.cast.18) <- "se_value"
std.18 <- as.data.frame(t(err.cast.18))
std.18 <- rownames_to_column(std.18, "vars") %>% filter(vars != "DOY_day")
sum18.lae.csad <- merge(mean.18, std.18, by = "vars")


## 2022
mean.cast.22 <- SHAPs_cast_CDH_LAE%>% filter(Year == 2022) %>% dplyr::select(Tair_f, Rg_f,
VPD_f, SWC_stnd, DOY_day) %>%
  filter(DOY_day %in% c(195:216)) %>% summarise_if(is.numeric, mean, na.rm = T)
row.names(mean.cast.22) <- "mean_value"
mean.22 <- as.data.frame(t(mean.cast.22))
mean.22 <- rownames_to_column(mean.22, "vars") %>% filter(vars != "DOY_day")
err.cast.22 <- SHAPs_cast_CDH_LAE%>% filter(Year == 2022) %>% dplyr::select(Tair_f, Rg_f,
VPD_f, SWC_stnd, DOY_day) %>%
  filter(DOY_day %in% c(195:216)) %>% summarise_if(is.numeric, std.error, na.rm = T)
row.names(err.cast.22) <- "se_value"
std.22 <- as.data.frame(t(err.cast.22))
std.22 <- rownames_to_column(std.22, "vars") %>% filter(vars != "DOY_day")
sum22.lae.csad <- merge(mean.22, std.22, by = "vars")


## MEAN 2005-2022
mean.cast <- SHAPs_cast_CDH_LAE %>% dplyr::select(Tair_f, Rg_f, VPD_f, SWC_stnd, DOY_day) %>%
  filter(DOY_day %in% c(188:235)) %>% summarise_if(is.numeric, mean, na.rm = T)
row.names(mean.cast) <- "mean_value"
mean.tot <- as.data.frame(t(mean.cast))
mean.tot <- rownames_to_column(mean.tot, "vars") %>% filter(vars != "DOY_day")
err.cast <- SHAPs_cast_CDH_LAE  %>% dplyr::select(Tair_f, Rg_f, VPD_f, SWC_stnd, DOY_day) %>%
  filter(DOY_day %in% c(188:235)) %>% summarise_if(is.numeric, std.error, na.rm = T)
row.names(err.cast) <- "se_value"
std.tot <- as.data.frame(t(err.cast))
std.tot <- rownames_to_column(std.tot, "vars") %>% filter(vars != "DOY_day")
sum.lae.csad <- merge(mean.tot, std.tot, by = "vars")

shap.15 <- SHAPs_cast_CDH_LAE %>% filter(Year == 2015) %>%
  ggplot() +
  annotate("rect", xmin = 188, xmax = 202, ymin = -Inf, ymax = +Inf, fill = "#F9C80E", alpha =
0.4) +
  annotate("rect", xmin = 214, xmax = 225, ymin = -Inf, ymax = +Inf, fill = "#F9C80E", alpha =
0.4) +

  geom_area(aes(x = DOY_day, y = Tair_f, fill= "Ta"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Tair_f, col= "Ta"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = Rg_f, fill = "Rg"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Rg_f, col = "Rg"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = VPD_f, fill = "VPD"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = VPD_f, col = "VPD"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = SWC_stnd, fill="SWC"), alpha = 0.4) +
  geom_line(aes(x = DOY_day, y = SWC_stnd, col="SWC"), linewidth = 0.5) +
  scale_y_continuous(breaks = seq(-5, 7.5, by = 2.5), limits = c(-5, 7.5)) +
  scale_x_continuous(breaks= c(121, 152, 182, 213, 244, 274),
                     labels = c("M", "J", "J", "A", "S", "O")) +
  scale_fill_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  ggtitle("2015") +

  th1+
  theme(legend.position = "none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'),  axis.text.x = element_blank(), axis.title.x =
element_blank()) +
  guides(fill = guide_legend(nrow = 1))

zoom.15 <-
  SHAPs_cast_CDH_LAE %>% filter(DOY_day %in% c(188:225)) %>%
  filter(Year == 2015) %>%
  ggplot() +

  geom_col(aes(x = DOY_day, y = Rg_f, fill = "Rg"), alpha = 0.4) +
  geom_col(aes(x = DOY_day , y = Tair_f, fill= "Ta"), alpha = 0.5) +
  geom_col(aes(x = DOY_day, y = VPD_f, fill = "VPD"), alpha = 0.3) +
  geom_col(aes(x = DOY_day, y = SWC_stnd, fill="SWC"), alpha = 0.7) +
  annotate("rect", xmin = 203, xmax = 213, ymin = -3.5, ymax = 2, fill = "white", alpha = 0.6)
+
  geom_vline(xintercept = 203, col = "black", linetype = "dashed") +
  geom_vline(xintercept = 213, col = "black", linetype = "dashed") +
  scale_x_continuous(breaks= c(190, 201, 213, 223), labels = c("09-07", "20-07", "01-08", "11-
08")) +
  scale_y_continuous(breaks = seq(-3, 2, by = 1), labels = c("",-2, "", 0, "", 2), limits = c(-
3.5, 2)) +
  scale_fill_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(),
              axis.text.x = element_text(size = 10),
              axis.text.y = element_text(size = 10),
              axis.title.y = element_blank())

col.15 <-
  ggplot() +
  geom_col(data = sum15.lae.csad, aes(x = vars, y = mean_value, fill = vars), position =
position_dodge(preserve = "single"),  width = 0.7) +
  geom_errorbar(data = sum15.lae.csad, aes(x = vars, ymin = mean_value - se_value,  ymax =
mean_value + se_value),
                position = position_dodge(preserve = "single", width = 0.7), col = "black",
size = 0.5, width = 0.5) +
  scale_fill_manual(name = "", breaks = c("Tair_f","SWC_stnd", "Rg_f", "VPD_f"), labels =
c("Ta", "SWC", "Rg", "VPD"),
                    values = col.fun)+
  scale_x_discrete( limit = c("Tair_f",  "SWC_stnd","Rg_f", "VPD_f"), labels = c("Tair", "SWC",
"Rg", "VPD")) +
  geom_hline(yintercept= 0) +
  scale_y_continuous(breaks = seq(-1.5,1, by = 0.5), labels = c("", -1, "", 0, "", 1), limits =
c(-2, 1.1))+
  xlab("") +
  ylab("") +
  th1 +
  theme(legend.position ="none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.title.x = element_blank(),
        axis.text.x = element_text(size = 10),
        axis.text.y = element_text(size = 10),
        axis.title.y = element_blank()) +
  guides(fill = guide_legend(nrow = 1))
```

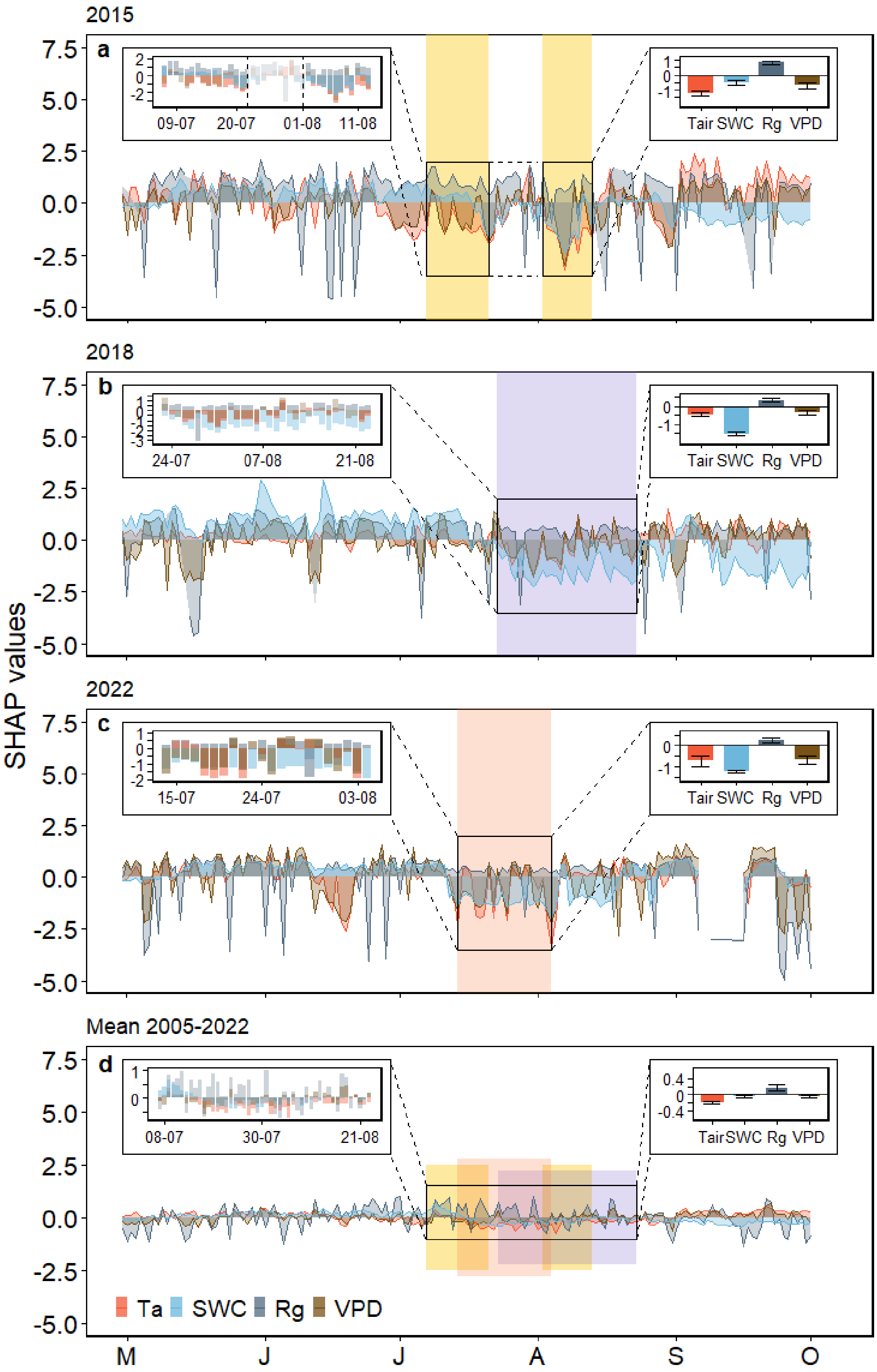
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
cb15 <-
  shap.15 +
  annotation_custom(ggplotGrob(zoom.15), xmin = 120, xmax = 180, ymin = 3, ymax = 7.5) + # plot
sx
  geom_rect(aes(xmin = 188, xmax = 202, ymin = -3.5, ymax = 2), color='black',
linetype='solid', alpha=0) + # interval csad
  geom_rect(aes(xmin = 214, xmax = 225, ymin = -3.5, ymax = 2), color='black',
linetype='solid', alpha=0) + # interval csad

  geom_rect(aes(xmin = 120, xmax = 180, ymin = 3, ymax = 7.5), color='black', linetype='solid',
alpha=0) + #plot square sx

  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(202, 213, 202, 213), y=c(-3.5, -3.5, 2, 2),grp=c(1, 1, 2,
2)),
            linetype='dashed') +

  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(187, 180, 187, 180), y=c(-3.5, 3, 2, 7.5),grp=c(1, 1, 2, 2)),
            linetype='dashed') + # connections to the left

  annotation_custom(ggplotGrob(col.15), xmin = 238, xmax = 280, ymin = 3, ymax = 7.5) + # plot
dx
  geom_rect(aes(xmin = 238, xmax = 280, ymin = 3, ymax = 7.5), color='black', linetype='solid',
alpha=0) + #plot square dx
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(225, 238, 225, 238), y=c(-3.5,3,2, 7.5),grp=c(1, 1, 2, 2)),
            linetype='dashed') # connections to the right

# Plot for 2018 -----------

shap.18 <- SHAPs_cast_CDH_LAE %>% filter(Year == 2018) %>%
  ggplot() +
  annotate("rect", xmin = 204, xmax = 235, ymin = -Inf, ymax = +Inf, fill = "#6248BF", alpha =
0.2) +

  geom_area(aes(x = DOY_day, y = Tair_f, fill= "Ta"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Tair_f, col= "Ta"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = Rg_f, fill = "Rg"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Rg_f, col = "Rg"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = VPD_f, fill = "VPD"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = VPD_f, col = "VPD"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = SWC_stnd, fill="SWC"), alpha = 0.4) +
  geom_line(aes(x = DOY_day, y = SWC_stnd, col="SWC"), linewidth = 0.5) +


  scale_y_continuous(breaks = seq(-5, 7.5, by = 2.5), limits = c(-5, 7.5)) +
  scale_x_continuous(breaks= c(121, 152, 182, 213, 244, 274),
                     labels = c("M", "J", "J", "A", "S", "O")) +
  scale_fill_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  ggtitle("2018") +

  th1+
  theme(legend.position = "none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'),  axis.text.x = element_blank(), axis.title.x =
element_blank()) +
  guides(fill = guide_legend(nrow = 1))

zoom.18 <- SHAPs_cast_CDH_LAE %>% filter(DOY_day %in% c(204:235)) %>%
  filter(Year == 2018) %>%
  ggplot() +
  geom_col(aes(x = DOY_day, y = Rg_f, fill = "Rg"), alpha = 0.4) +
  geom_col(aes(x = DOY_day, y = SWC_stnd, fill="SWC"), alpha = 0.4) +
  geom_col(aes(x = DOY_day , y = Tair_f, fill= "Ta"), alpha = 0.5) +
  geom_col(aes(x = DOY_day, y = VPD_f, fill = "VPD"), alpha = 0.3) +

  scale_x_continuous(breaks= c(205, 219, 233), labels = c("24-07", "07-08", "21-08")) +
  scale_y_continuous(breaks = seq(-3,1.5, by = 0.5), labels = c(-3, "", -2, "", -1, "", 0, "",
1, ""), limits = c(-3.2, 1.5)) +
  scale_fill_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(),
              axis.text.x = element_text(size = 10),
              axis.text.y = element_text(size = 10),
              axis.title.y = element_blank())



col.18 <- ggplot() +
  geom_col(data = sum18.lae.csad, aes(x = vars, y = mean_value, fill = vars), position =
position_dodge(preserve = "single"), width = 0.7) +
  geom_errorbar(data = sum18.lae.csad, aes(x = vars, ymin = mean_value - se_value,  ymax =
mean_value + se_value),
                position = position_dodge(preserve = "single", width = 0.7), col = "black",
size = 0.5, width = 0.5) +
  scale_fill_manual(name = "", breaks = c("Tair_f","SWC_stnd", "Rg_f", "VPD_f"), labels =
c("Ta", "SWC", "Rg", "VPD"),
                    values = col.fun)+
  scale_x_discrete(limit = c("Tair_f",  "SWC_stnd","Rg_f", "VPD_f"), labels = c("Tair", "SWC",
"Rg", "VPD")) +
  scale_y_continuous(breaks = seq(-1.5, 0.5, by = 0.5), labels = c("", -1, "", 0, ""), limits =
c(-2, 0.6)) +

  geom_hline(yintercept= 0) +
  xlab("") +
  ylab("") +
  th1 +
  theme(legend.position ="none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.title.x = element_blank(),
        axis.text.x = element_text(size = 10),
        axis.text.y = element_text(size = 10),
        axis.title.y = element_blank()) +
  guides(fill = guide_legend(nrow = 1))



cb18 <- shap.18 +
  annotation_custom(ggplotGrob(zoom.18), xmin = 120, xmax = 180, ymin = 3, ymax = 7.5)+  # plot
sx
  geom_rect(aes(xmin = 204, xmax = 235, ymin = -3.5, ymax = 2), color='black',
linetype='solid', alpha=0) + # interval

  geom_rect(aes(xmin = 120, xmax = 180, ymin = 3, ymax = 7.5), color='black', linetype='solid',
alpha=0) + #plot square sx
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(204, 180, 204, 180), y=c(-3.5, 3, 2, 7.5),grp=c(1, 1, 2, 2)),
            linetype='dashed') + # connections to the left
  annotation_custom(ggplotGrob(col.18), xmin = 238, xmax = 280, ymin = 3, ymax = 7.5) + # plot
dx
  geom_rect(aes(xmin = 238, xmax = 280, ymin = 3, ymax = 7.5), color='black', linetype='solid',
alpha=0) + #plot square dx
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(235, 238, 235, 238), y=c(-3.5, 3,2, 7.5), grp=c(1, 1, 2, 2)),
            linetype='dashed') # connections to the rightconnections to the right

# Plot for 2022 ----------

shap.22 <- SHAPs_cast_CDH_LAE %>% filter(Year == 2022) %>%
  ggplot() +
  annotate("rect", xmin = 195, xmax = 216, ymin = -Inf, ymax = +Inf, fill = "#F86624", alpha =
0.2) +
  geom_area(aes(x = DOY_day, y = Tair_f, fill= "Ta"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Tair_f, col= "Ta"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = Rg_f, fill = "Rg"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Rg_f, col = "Rg"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = VPD_f, fill = "VPD"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = VPD_f, col = "VPD"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = SWC_stnd, fill="SWC"), alpha = 0.4) +
  geom_line(aes(x = DOY_day, y = SWC_stnd, col="SWC"), linewidth = 0.5) +

  annotate("rect", xmin = 249, xmax = 259, ymin = -3, ymax = 3.2, fill = "white") +
  scale_y_continuous(breaks = seq(-5, 7.5 , by = 2.5), limits = c(-5, 7.5)) +
  scale_x_continuous(breaks= c(121, 152, 182, 213, 244, 274),
                     labels = c("M", "J", "J", "A", "S", "O")) +
  scale_fill_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  ggtitle("2022") +
  th1+
  theme(legend.position = "none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'),  axis.text.x = element_blank(), axis.title.x =
element_blank()) +
  guides(fill = guide_legend(nrow = 1))

zoom.22 <- SHAPs_cast_CDH_LAE %>% filter(DOY_day %in% c(195:216)) %>%
  filter(Year == 2022) %>%
  ggplot() +
  geom_col(aes(x = DOY_day, y = SWC_stnd, fill="SWC"), alpha = 0.5) +
  geom_col(aes(x = DOY_day, y = Rg_f, fill = "Rg"), alpha = 0.5) +
  geom_col(aes(x = DOY_day , y = Tair_f, fill= "Ta"), alpha = 0.5) +
  geom_col(aes(x = DOY_day, y = VPD_f, fill = "VPD"), alpha = 0.5) +
  scale_x_continuous(breaks= c(196, 205, 215),
                     labels = c("15-07", "24-07", "03-08")) +
  scale_y_continuous(breaks = c(-2,-1.5, -1,-0.5, 0, 0.5, 1, 1.5, 2), labels = c(-2,"", -1,"",
0, "", 1, "", 2), limits = c(-2.0, 1)) +
  scale_fill_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(),
              axis.text.x = element_text(size = 10),
              axis.text.y = element_text(size = 10),
              axis.title.y = element_blank())

col.22 <- ggplot() +
  geom_col(data = sum22.lae.csad, aes(x = vars, y = mean_value, fill = vars), position =
position_dodge(preserve = "single"), width = 0.7) +
  geom_errorbar(data = sum22.lae.csad, aes(x = vars, ymin = mean_value - se_value,  ymax =
mean_value + se_value),
                position = position_dodge(preserve = "single", width = 0.7),size = 0.5, width =
0.5, col = "black") +
  scale_fill_manual(name = "", breaks = c("Tair_f","SWC_stnd", "Rg_f", "VPD_f"), labels =
c("Ta", "SWC", "Rg", "VPD"), values = col.fun)+
  scale_x_discrete( limit = c("Tair_f", "SWC_stnd","Rg_f", "VPD_f"), labels = c("Tair", "SWC",
"Rg", "VPD")) +
  scale_y_continuous(breaks = seq(-1.5, 0.5, by = 0.5), labels = c("", -1, "", 0, ""), limits =
c(-1.6, 0.6))+
  geom_hline(yintercept= 0) +
  xlab("") +
  ylab("") +
  th1 +
  theme(legend.position ="none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.title.x = element_blank(),
        axis.text.x = element_text(size = 10),
        axis.text.y = element_text(size = 10),
        axis.title.y = element_blank()) +
  guides(fill = guide_legend(nrow = 1))



cb22 <- shap.22 +
  annotation_custom(ggplotGrob(zoom.22), xmin = 120, xmax = 180, ymin = 3, ymax = 7.5)+  # plot
sx
```

```
  geom_rect(aes(xmin = 195, xmax = 216, ymin = -3.5, ymax = 2), color='black',
linetype='solid', alpha=0) + # interval

  geom_rect(aes(xmin = 120, xmax = 180, ymin = 3, ymax = 7.5), color='black', linetype='solid',
alpha=0) + #plot square sx
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(195, 180, 195, 180), y=c(-3.5, 3, 2, 7.5),grp=c(1, 1, 2, 2)),
            linetype='dashed') + # connections to the left
  annotation_custom(ggplotGrob(col.22), xmin = 238, xmax = 280, ymin = 3, ymax = 7.5) + # plot
dx
  geom_rect(aes(xmin = 238, xmax = 280, ymin = 3, ymax = 7.5), color='black', linetype='solid',
alpha=0) + #plot square dx
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(216, 238, 216, 238), y=c(-3.5, 3,2, 7.5), grp=c(1, 1, 2, 2)),
            linetype='dashed')
```

```
## Warning: Removed 3 rows containing missing values (`position_stack()`).

## Warning: Removed 1 rows containing missing values (`position_stack()`).
```

```
# Plot for the mean 2005-2022

SHAP_mean.LAE = SHAPs_cast_CDH_LAE %>% group_by(DOY_day) %>% summarise_if(is.numeric, mean,
na.rm = TRUE)
SHAP_sd.LAE = SHAPs_cast_CDH_LAE %>% group_by(DOY_day) %>% summarise_if(is.numeric, sd, na.rm =
TRUE)

lae.shap <-
  SHAP_mean.LAE  %>%
  ggplot() +
  th1 +
  annotate("rect", xmin = 188, xmax = 202, ymin = -2.5, ymax = 2.5, fill = "#F9C80E", alpha =
0.4) +
  annotate("rect", xmin = 214, xmax = 225, ymin = -2.5, ymax = 2.5, fill = "#F9C80E", alpha =
0.4) +
  annotate("rect", xmin = 204, xmax = 235, ymin = -2.2, ymax = +2.2, fill = "#6248BF", alpha =
0.2) +
  annotate("rect", xmin = 195, xmax = 216, ymin = -2.8, ymax = +2.8, fill = "#F86624", alpha =
0.2) +
  geom_area(aes(x = DOY_day, y = Tair_f, fill= "Ta"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Tair_f, col= "Ta"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = Rg_f, fill = "Rg"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Rg_f, col = "Rg"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = VPD_f, fill = "VPD"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = VPD_f, col = "VPD"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = SWC_stnd, fill="SWC"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = SWC_stnd, col="SWC"), linewidth = 0.5) +
  scale_x_continuous(breaks= c(121, 152, 182, 213, 244, 274),
                     labels = c("M", "J", "J", "A", "S", "O")) +
  scale_y_continuous(breaks = seq(-5, 7.5, by = 2.5), limits = c(-5, 7.5)) +
  scale_fill_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                     values = col.fun)+
  xlab("") +
  ylab("")+
  ggtitle("Mean 2005-2022") +
  theme(legend.position = c(0.2, 0.1), legend.direction = "horizontal", legend.key.size =
unit(0.3, 'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'),  legend.background = element_rect(fill = "white"),
axis.text.x = element_text(), axis.title.x = element_blank()) +
  guides(fill = guide_legend(nrow = 1))


lae.zoom <- SHAP_mean.LAE %>% filter(DOY_day %in% c(188:235)) %>%
  ggplot() +
  geom_col(aes(x = DOY_day, y = Tair_f, fill= "Ta"), alpha = 0.4) +
  geom_col(aes(x = DOY_day, y = Rg_f, fill = "Rg"), alpha = 0.3) +
  geom_col(aes(x = DOY_day, y = VPD_f, fill = "VPD"), alpha = 0.4) +
  geom_col(aes(x = DOY_day, y = SWC_stnd, fill="SWC"), alpha = 0.5) +
  scale_x_continuous(breaks= c(189,  211, 233),
                     labels = c("08-07", "30-07", "21-08")) +
  scale_y_continuous(breaks = c(-0.5, 0, 0.5, 1, 1.5, 2, 2.5), labels = c("", 0, "", 1, "", 2,
"")) +
  scale_fill_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("Ta","SWC", "Rg", "VPD"), labels = c("Ta", "SWC",
"Rg", "VPD"),
                     values = col.fun)+
  xlab("") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(),
              axis.text.x = element_text(size = 10),
              axis.text.y = element_text(size = 10),
              axis.title.y = element_blank())

lae.col <- ggplot() +
  geom_col(data = sum.lae.csad, aes(x = vars, y = mean_value, fill = vars), position =
position_dodge(preserve = "single"),  width = 0.7) +
  geom_errorbar(data = sum.lae.csad, aes(x = vars, ymin = mean_value - se_value,  ymax =
mean_value + se_value),
                position = position_dodge(preserve = "single", width = 0.7), col = "black",
size = 0.5, width = 0.5) +
  scale_fill_manual(name = "", breaks = c("Tair_f","SWC_stnd", "Rg_f", "VPD_f"), labels =
c("Ta", "SWC", "Rg", "VPD"),
                    values = col.fun)+
  scale_x_discrete(limit =  c("Tair_f", "SWC_stnd","Rg_f", "VPD_f"), labels = c("Tair", "SWC",
"Rg", "VPD")) +
  geom_hline(yintercept= 0) +
  scale_y_continuous(breaks = seq(-0.4, 0.4, by = 0.2), labels = c(-0.4, "", 0, "", 0.4),
limits = c(-0.6, 0.6))+
  xlab("") +
  th1 +
  theme(legend.position ="none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.title.x = element_blank(),
        axis.text.x = element_text(size = 10),
        axis.text.y = element_text(size = 10),
        axis.title.y = element_blank()) +
  guides(fill = guide_legend(nrow = 1))



cb.mean <-
  lae.shap +
  annotation_custom(ggplotGrob(lae.zoom), xmin = 120, xmax = 180, ymin = 3, ymax = 7.5)+  #
plot sx
  geom_rect(aes(xmin = 188, xmax = 235, ymin = -1, ymax = 1.5), color='black',
linetype='solid', alpha=0) + # interval

  geom_rect(aes(xmin = 120, xmax = 180, ymin = 3, ymax = 7.5), color='black', linetype='solid',
alpha=0) + #plot square sx
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(188, 180, 188, 180), y=c(-1, 3, 1.5, 7.5),grp=c(1, 1, 2, 2)),
            linetype='dashed') + # connections to the left
  annotation_custom(ggplotGrob(lae.col), xmin = 238, xmax = 280, ymin = 3, ymax = 7.5) + # plot
dx
  geom_rect(aes(xmin = 238, xmax = 280, ymin = 3, ymax = 7.5), color='black', linetype='solid',
alpha=0) + #plot square dx
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(235, 238, 235, 238), y=c(-1, 3, 1.5, 7.5), grp=c(1, 1, 2,
2)),
            linetype='dashed')

### FINAL PLOT ###
pp <- plot_grid(cb15, cb18, cb22, cb.mean, nrow = 4, ncol = 1, rel_heights = c(1,1,1,1.1),
labels = "auto", label_x = 0.11, label_y = 0.9, axis = "l")

## Warning: Removed 9 rows containing non-finite values (`stat_align()`).

## Warning: Removed 6 rows containing non-finite values (`stat_align()`).

## Warning: Removed 1 rows containing non-finite values (`stat_align()`).

annotate_figure(pp, left = textGrob("SHAP values", rot = 90, vjust = 2.2, gp = gpar(cex =
1.5)))
```


2015
a
7.5
5.0
2.5
0.0
-2.5
-5.0
2
0
-2
09-07
20-07
01-08
11-08
1
0
-1
Tair SWC Rg VPD
2018
b
1
0
-1
-2
-3
24-07
07-08
21-08
0
-1
Tair SWC Rg VPD
2022
c
1
0
-1
-2
15-07
24-07
03-08
0
-1
Tair SWC Rg VPD
SHAP values
Mean 2005-2022
d
1
0
08-07
30-07
21-08
0.4
0
-0.4
Tair SWC Rg VPD
Ta
SWC
Rg
VPD
M
J
J
A
S
O

6. Figure 6: SHAP for Rff

We repeat the same analysis for Rff:

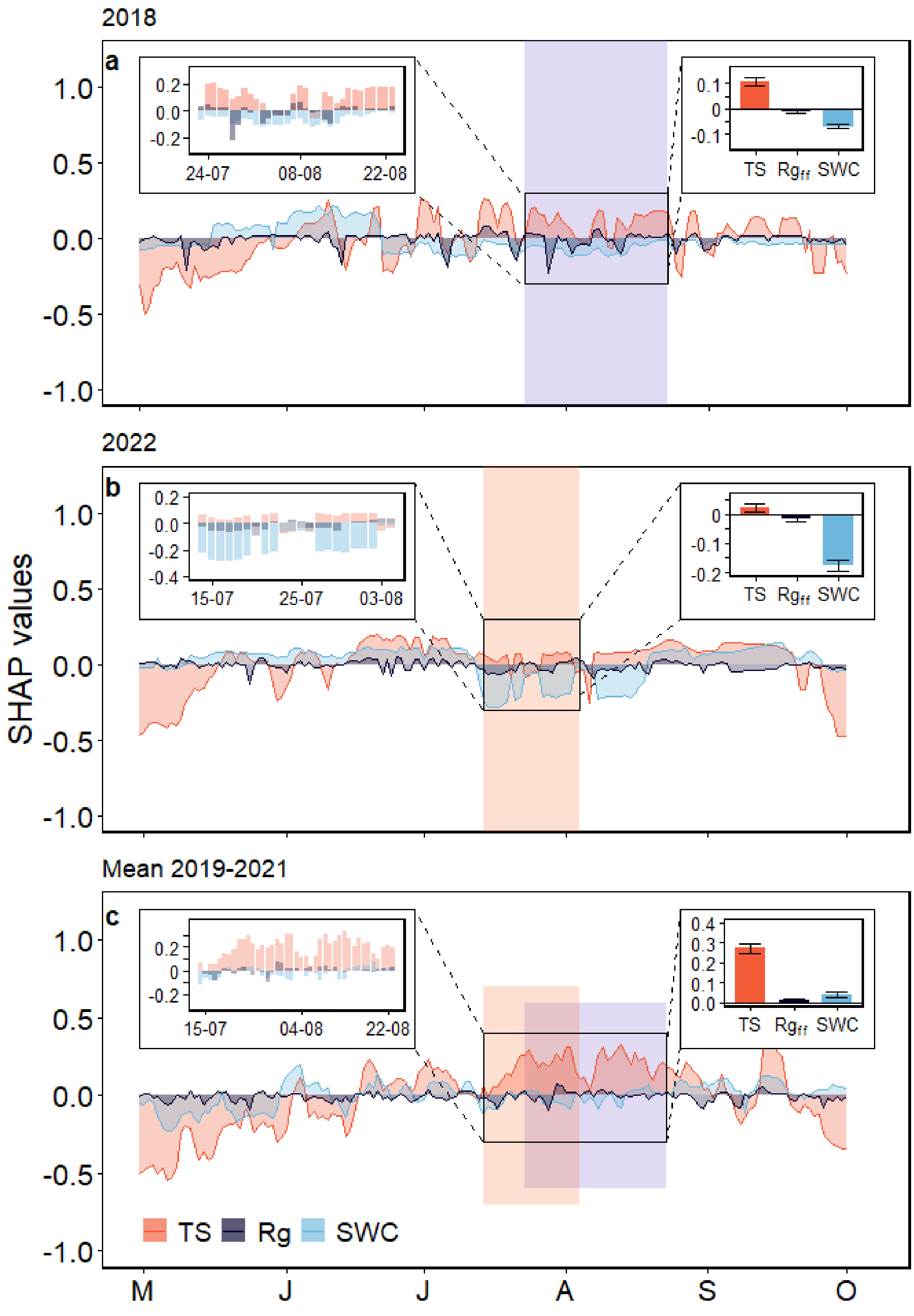
```
# las$mean_Rg <- PPFD.to.Rg(las$mean_PPFD_IN_FF, J_to_mol = 4.6, frac_PAR = 0.5)
#
# las_shap <- las %>%
#   mutate(doy = yday(Date)) %>% mutate(year = year(Date)) %>% filter(year >= 2018) %>%
#   dplyr::select(doy, mean_Reco_DT_U50, mean_TS_5cm, mean_SWC_stnd_20, mean_Rg, year) %>%
#   arrange(doy)
#
#
# colnames(las_shap) = c("doy", "Rff", "TS", "SWC","Rg", "year")
#
# for(yr in c(2018:2022))
# {
#   y_var <-  "Rff"
#   dataXY = las_shap %>% filter(year == yr) %>% filter(as.numeric(doy) %in% 120:274) %>%
#     na.omit()
#   dataX <- as.matrix(dataXY %>% dplyr::select(!(Rff)))
#   # hyperparameter tuning results
#   params <- list(objective = "reg:squarederror",  # For regression
#                  eta = 0.02,
#                  max_depth = 10,
#                  gamma = 0.01,
#                  subsample = 0.98,
#                  colsample_bytree = 0.86)
#
#   mod <- xgboost::xgboost(data = dataX,
#                           label =  as.matrix(dataXY$Rff),
#                           params = params,
#                           nrounds = 200,
#                           verbose = FALSE,
#                           early_stopping_rounds = 8)
#
#   shap_values <- shap.values(xgb_model = mod, X_train = dataX)
#   str(shap_values)
#   # remember that it only works with matrix, so add here o before a matrix conversion
#   # The ranked features by mean |SHAP|
#
#   # To prepare the long-format data:
#   shap_long <- shap.prep(xgb_model = mod, X_train = dataX)
#   shap.plot.summary(shap_long)
#   shap_long_cast = inner_join(reshape2::dcast(shap_long, ID ~ variable, value.var =
c('value')),
#                               shap_long %>% filter(variable == 'doy') %>% mutate(DOY_day =
rfvalue) %>%
#                                 dplyr::select(ID, DOY_day), by = 'ID')
#
#   shap_long_cast$Year = yr
#   shap_long$Year = yr
#   if(yr == 2018)
#   {
#     SHAPs_cast_CDH_LAS = shap_long_cast
#     SHAPs_CDH_LAS = shap_long
#   }
#
#   if(yr > 2018)
#   {
#     SHAPs_cast_CDH_LAS = rbind(SHAPs_cast_CDH_LAS, shap_long_cast)
#     SHAPs_CDH_LAS = rbind(SHAPs_CDH_LAS, shap_long)
#   }
#   print(yr)
# }
#
# save(SHAPs_cast_CDH_LAS, file = 'SHAP_values_casted_Rff_2018_2022.RData')
# save(SHAPs_CDH_LAS, file = 'SHAP_values_long_Rff_2018_2022.RData')
```

Now we plot the results:

```
load( 'SHAP_values_casted_Rff_2018_2022.RData')
load('SHAP_values_long_Rff_2018_2022.RData')

# calculation of the mean values over the CSAD periods

SHAPs_CDH_LAS$DOY_day <- SHAPs_CDH_LAS$ID + 119
sum18.las.csad = SHAPs_CDH_LAS %>% filter(Year == 2018) %>% filter(DOY_day %in% c(204:235)) %>%
  group_by(variable) %>% summarise(mean_SHAP = mean(value, na.rm = TRUE), se_SHAP =
std.error(value, na.rm = TRUE))

sum22.las.csad = SHAPs_CDH_LAS%>% filter(Year == 2022) %>% filter(DOY_day %in% c(195:216)) %>%
  group_by(variable) %>% summarise(mean_SHAP = mean(value, na.rm = TRUE), se_SHAP =
std.error(value, na.rm = TRUE))

summean.las.csad <-SHAPs_CDH_LAS %>% filter(Year %in% 2019:2021) %>%
  filter(DOY_day %in% 195:235) %>% group_by(variable) %>%
  summarise(mean_SHAP = mean(value, na.rm = TRUE), se_SHAP = std.error(value, na.rm = TRUE))
sum18.las.csad <- sum18.las.csad[sum18.las.csad$variable %in% c("TS", "Rg", "SWC"), ]
sum22.las.csad <- sum22.las.csad[sum22.las.csad$variable %in% c("TS", "Rg", "SWC"), ]
summean.las.csad = summean.las.csad[summean.las.csad$variable %in% c("TS", "Rg", "SWC"), ]

sum18.las.csad$year = "2018"
sum22.las.csad$year = "2022"
summean.las.csad$year = "mean"

# Plotting the results

col.fun <- c("#F25A38", "#050533", "#6CB7DB")

### 2018 ----
# all year Timeseries
s18 <- SHAPs_cast_CDH_LAS%>%
  filter(Year == 2018) %>%
  ggplot() +
  annotate("rect", xmin = 204, xmax = 235, ymin = -Inf, ymax = +Inf, fill = "#6248BF", alpha =
0.2) +
  geom_area(aes(x = DOY_day, y = TS, fill= "TS"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = TS, col= "TS"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = SWC, fill="SWC"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = SWC, col="SWC"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = Rg, fill = "Rg"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Rg, col = "Rg"), linewidth = 0.5) +
  scale_y_continuous(breaks = c(-1, -0.5, 0, 0.5, 1), limits = c(-1, 1.2)) +
  scale_x_continuous(breaks= c( 120, 152, 182, 213, 244, 274),
                     labels = c("M", "J", "J", "A", "S", "O")) +
  scale_fill_manual(name = "", breaks = c("TS", "Rg", "SWC"), labels = c("TS", "Rg"["ff"]~"",
"SWC"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("TS",  "Rg", "SWC"), labels = c("TS",
"Rg"["ff"]~"", "SWC"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  ggtitle("2018") +
  th1+  theme(legend.position = "none", legend.direction = "horizontal", legend.key.size =
unit(0.3, 'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.text.x = element_blank(), axis.title.x =
element_blank()) +
  guides(fill = guide_legend(nrow = 1))


# Zoom plot for CSAD
zoom.sh18 <- SHAPs_cast_CDH_LAS%>% filter(DOY_day %in% c(204:235)) %>%
  filter(Year == 2018) %>%
  ggplot() +
  geom_col(aes(x = DOY_day, y = TS, fill= "TS"), alpha = 0.4) +

  geom_col(aes(x = DOY_day, y = SWC, fill="SWC"), alpha = 0.4) +

  geom_col(aes(x = DOY_day, y = Rg, fill = "Rg"), alpha = 0.4) +
  scale_x_continuous(breaks= c(205, 220, 234),
                     labels = c("24-07", "08-08", "22-08")) +
  scale_y_continuous(breaks = c(-0.2, 0, 0.2), limits = c(-0.3, 0.3)) +
  scale_fill_manual(name = "", breaks = c("TS", "Rg", "SWC"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("TS",  "Rg", "SWC"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(),
              axis.text.x = element_text(size = 10),
              axis.text.y = element_text(size = 10),
              axis.title.y = element_blank())

# Zoom mean col
zoom.col18 <- ggplot() +
  geom_col(data = sum18.las.csad, aes(x = variable, y = mean_SHAP, fill = variable), position =
position_dodge(preserve = "single"),  width = 0.7) +
  geom_errorbar(data = sum18.las.csad, aes(x = variable, ymin = mean_SHAP - se_SHAP,  ymax =
mean_SHAP + se_SHAP, group = variable),
                position = position_dodge(preserve = "single", width = 0.7), col = "black",
size = 0.5, width = 0.5) +
  scale_fill_manual(name = "", breaks = c("TS", "Rg", "SWC"), labels = c("TS", "Rg", "SWC"),
                    values = col.fun)+
  scale_x_discrete(limit = c("TS", "Rg", "SWC"), labels = c("TS", "Rg"["ff"]~"", "SWC") )+
  scale_y_continuous(breaks = seq(-0.1, 0.1, by = 0.05), labels = c(-0.1, "", 0, "", 0.1),
limits = c(-0.15, 0.15)) +
  geom_hline(yintercept= 0) +
  xlab("") +
  ylab("") +
  th1 +
  theme(legend.position ="none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.title.x = element_blank(),
        axis.text.x = element_text(size = 10),
        axis.text.y = element_text(size = 10),
        axis.title.y = element_blank()) +
  guides(fill = guide_legend(nrow = 1))

cb1 <- s18 +
  annotation_custom(ggplotGrob(zoom.sh18), xmin =120, xmax = 180, ymin = 0.3, ymax = 1.2) +
  geom_rect(aes(xmin = 204, xmax = 235, ymin = -0.3, ymax =0.3), color='black',
linetype='solid', alpha=0) + # interval
  geom_rect(aes(xmin = 120, xmax = 180, ymin = 0.3, ymax = 1.2), color='black',
linetype='solid', alpha=0) + #plot square
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(203, 180, 203, 180), y=c(-0.3, 0.3, 0.3, 1.2),grp=c(1, 1, 2,
2)),
            linetype='dashed') +
  annotation_custom(ggplotGrob(zoom.col18), xmin = 238, xmax = 280, ymin = 0.3, ymax = 1.2) +
  geom_rect(aes(xmin = 238, xmax = 280, ymin = 0.3, ymax = 1.2), color='black',
linetype='solid', alpha=0) +
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(235, 238, 235, 238), y=c(-0.2, 0.3, 0.3, 1.2),grp=c(1, 1, 2,
2)),
            linetype='dashed')



# 2022 ----------

s22 <- SHAPs_cast_CDH_LAS%>% #filter(DOY_day %in% c(152:243)) %>%
  filter(Year == 2022) %>%
  ggplot() +
  annotate("rect", xmin = 195, xmax = 216, ymin = -Inf, ymax = +Inf, fill = "#F86624", alpha =
0.2) +
  geom_area(aes(x = DOY_day, y = TS, fill= "TS"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = TS, col= "TS"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = SWC, fill="SWC"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = SWC, col="SWC"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = Rg, fill = "Rg"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Rg, col = "Rg"), linewidth = 0.5) +

  scale_y_continuous(breaks = c(-1, -0.5, 0, 0.5, 1), limits = c(-1, 1.2)) +
  scale_x_continuous(breaks= c( 121, 152, 182, 213, 244, 274),
                     labels = c("M", "J", "J", "A", "S", "O")) +
  scale_fill_manual(name = "", breaks = c("TS", "Rg", "SWC"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("TS",  "Rg", "SWC"),
                     values = col.fun)+

  xlab("") +
  ylab("SHAP values") +
  ggtitle("2022") +
  th1+
  theme(legend.position = "none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.text.x = element_blank(), axis.title.x =
element_blank()) +
  guides(fill = guide_legend(nrow = 1))

# Zoom for CSAD
zoom.sh22 <- SHAPs_cast_CDH_LAS%>% filter(DOY_day %in% c(195:216)) %>%
  filter(Year == 2022) %>%
  ggplot() +
  geom_col(aes(x = DOY_day, y = TS, fill= "TS"), alpha = 0.3) +
  geom_col(aes(x = DOY_day, y = SWC, fill="SWC"), alpha = 0.4) +
  geom_col(aes(x = DOY_day, y = Rg, fill = "Rg"), alpha = 0.3) +
  scale_x_continuous(breaks= c(196, 206, 215),
                     labels = c("15-07", "25-07", "03-08")) +
  scale_y_continuous(breaks = c(-0.4, -0.2, 0, 0.2), limits = c(-0.4, 0.2)) +
  scale_fill_manual(name = "", breaks = c("TS", "Rg", "SWC"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("TS",  "Rg", "SWC"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(),
              axis.text.x = element_text(size = 10),
              axis.text.y = element_text(size = 10),
              axis.title.y = element_blank())

# zoom for col plot

zoom.col22 <- ggplot() +
  geom_col(data = sum22.las.csad, aes(x = variable, y = mean_SHAP, fill = variable), position =
position_dodge(preserve = "single"), width = 0.7) +
  geom_errorbar(data = sum22.las.csad, aes(x = variable, ymin = mean_SHAP - se_SHAP,  ymax =
mean_SHAP + se_SHAP, group = variable),
                position = position_dodge(preserve = "single", width = 0.7), col = "black",
size = 0.5, width = 0.5) +
  scale_fill_manual(name = "", breaks =  c("TS", "Rg", "SWC"), labels = c("TS", "Rg", "SWC"),
                    values = col.fun)+
    scale_x_discrete(limit = c("TS", "Rg", "SWC"), labels = c("TS", "Rg"["ff"]~"", "SWC") )+
  scale_y_continuous(breaks = seq(-0.2, 0.05, by = 0.05), labels = c(-0.2, "", -0.1, "", 0,
""), limits = c(-0.2, 0.06)) +
  geom_hline(yintercept= 0) +
  xlab("") +
  ylab("") +
  th1 +
  theme(legend.position ="none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.title.x = element_blank(),
        axis.text.x = element_text(size = 10),
        axis.text.y = element_text(size = 10),
        axis.title.y = element_blank()) +
  guides(fill = guide_legend(nrow = 1))

cb2 <- s22 +
  annotation_custom(ggplotGrob(zoom.sh22), xmin =120, xmax = 180, ymin = 0.3, ymax = 1.2) +
  geom_rect(aes(xmin = 195, xmax = 216, ymin = -0.3, ymax =0.3), color='black',
linetype='solid', alpha=0) + # interval
  geom_rect(aes(xmin = 120, xmax = 180, ymin = 0.3, ymax = 1.2), color='black',
linetype='solid', alpha=0) + #plot square
  geom_path(aes(x,y,group=grp),
          data=data.frame(x = c(195, 180, 195, 180), y=c(-0.3, 0.3, 0.3, 1.2),grp=c(1, 1, 2,
2)),
          linetype='dashed') +
  annotation_custom(ggplotGrob(zoom.col22), xmin = 238, xmax = 280, ymin = 0.3, ymax = 1.2) +
  geom_rect(aes(xmin = 238, xmax = 280, ymin = 0.3, ymax = 1.2), color='black',
linetype='solid', alpha=0) +
  geom_path(aes(x,y,group=grp),
          data=data.frame(x = c(216, 238, 216, 238), y=c(-0.2, 0.3, 0.3, 1.2),grp=c(1, 1, 2,
2)),
          linetype='dashed')


## Mean 2019-2021

SHAP_mean.las = SHAPs_cast_CDH_LAS %>% group_by(DOY_day) %>% summarise_if(is.numeric, mean,
na.rm = TRUE)
smean <- SHAP_mean.las  %>% #filter(DOY_day %in% c(152:243)) %>%
  ggplot() +
  th1 +
  annotate("rect", xmin = 204, xmax = 235, ymin = -0.6, ymax = +0.6, fill = "#6248BF", alpha =
0.2) +
  annotate("rect", xmin = 195, xmax = 216, ymin = -0.7, ymax = +0.7, fill = "#F86624", alpha =
0.2) +
  geom_area(aes(x = DOY_day, y = TS, fill= "TS"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = TS, col= "TS"), linewidth = 0.5) +
  geom_area(aes(x = DOY_day, y = SWC, fill="SWC"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = SWC, col="SWC"), linewidth = 0.5) +
```

```r
  geom_area(aes(x = DOY_day, y = Rg, fill = "Rg"), alpha = 0.3) +
  geom_line(aes(x = DOY_day, y = Rg, col = "Rg"), linewidth = 0.5) +
  scale_y_continuous(breaks = c(-1, -0.5, 0, 0.5, 1), limits = c(-1, 1.2)) +
  scale_x_continuous(breaks= c(121, 152, 182, 213, 244, 274),
                     labels = c("M", "J", "J", "A", "S", "O"), limits = c(120, 280)) +
  scale_fill_manual(name = "", breaks = c("TS", "Rg", "SWC"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("TS", "Rg", "SWC"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  ggtitle("Mean 2019-2021") +
   theme(legend.position = c(0.20, 0.10), legend.direction = "horizontal", legend.key.size =
unit(0.5, 'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.5, 'cm'), axis.text.x = element_text(), axis.title.x =
element_blank()) +
  guides(fill = guide_legend(nrow = 1))

#zoom for CSAD
zoom.shmean<- SHAP_mean.las %>% filter(DOY_day %in% c(195:235)) %>%
  ggplot() +
  geom_col(aes(x = DOY_day, y = TS, fill= "TS"), alpha = 0.3) +
  geom_col(aes(x = DOY_day, y = SWC, fill="SWC"), alpha = 0.4) +
  geom_col(aes(x = DOY_day, y = Rg, fill = "Rg"), alpha = 0.3) +
  scale_x_continuous(breaks= c(196,  216,  234),
                     labels = c("15-07", "04-08", "22-08")) +
  scale_y_continuous(breaks = c(-0.2, -0.1, 0, 0.1, 0.2, 0.3), limits = c(-0.3, 0.4), labels =
c(-0.2, "", 0, "", 0.2, "")) +
  scale_fill_manual(name = "", breaks = c("TS", "Rg", "SWC"),
                    values = col.fun)+
  scale_color_manual(name = "", breaks = c("TS","Rg", "SWC"),
                     values = col.fun)+
  xlab("") +
  ylab("") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(),
              axis.text.x = element_text(size = 10),
              axis.text.y = element_text(size = 10),
              axis.title.y = element_blank())

# zoom for cols
zoom.colmean<- ggplot() +
  geom_col(data = summean.las.csad, aes(x = variable, y = mean_SHAP, fill = variable), position
= position_dodge(preserve = "single"), width = 0.7) +
  geom_errorbar(data = summean.las.csad, aes(x = variable, ymin = mean_SHAP - se_SHAP,  ymax =
mean_SHAP + se_SHAP, group = variable),
                position = position_dodge(preserve = "single", width = 0.7), col = "black",
size = 0.5, width = 0.5) +
  scale_fill_manual(name = "", breaks =  c("TS", "Rg", "SWC"), labels = c("TS", "Rg", "SWC"),
                    values = col.fun)+
    scale_x_discrete(limit = c("TS", "Rg", "SWC"), labels = c("TS", "Rg"["ff"]~"", "SWC") )+
  geom_hline(yintercept= 0) +
  ylim(0, 0.4)+
  xlab("") +
  ylab("") +
  th1 +
  theme(legend.position ="none", legend.direction = "horizontal", legend.key.size = unit(0.3,
'cm'), legend.key.height= unit(0.5, 'cm'),
        legend.key.width= unit(0.3, 'cm'), axis.title.x = element_blank(),
        axis.text.x = element_text(size = 10),
        axis.text.y = element_text(size = 10),
        axis.title.y = element_blank()) +
  guides(fill = guide_legend(nrow = 1))
```

```
cb3<- smean +
  annotation_custom(ggplotGrob(zoom.shmean), xmin =120, xmax = 180, ymin = 0.3, ymax = 1.2) +
  geom_rect(aes(xmin = 195, xmax = 235, ymin = -0.3, ymax =0.4), color='black',
linetype='solid', alpha=0) + # interval
  geom_rect(aes(xmin = 120, xmax = 180, ymin = 0.3, ymax = 1.2), color='black',
linetype='solid', alpha=0) + #plot square
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(195, 180, 195, 180), y=c(-0.3, 0.3, 0.4, 1.2),grp=c(1, 1, 2,
2)),
            linetype='dashed') +
  annotation_custom(ggplotGrob(zoom.colmean), xmin = 238, xmax = 280, ymin = 0.3, ymax = 1.2) +
  geom_rect(aes(xmin = 238, xmax = 280, ymin = 0.3, ymax = 1.2), color='black',
linetype='solid', alpha=0) +
  geom_path(aes(x,y,group=grp),
            data=data.frame(x = c(235, 238, 235, 238), y=c(-0.3, 0.3, 0.3, 1.2),grp=c(1, 1, 2,
2)),
            linetype='dashed')

## FINAL COMBINED PLOT
gs_rff <- plot_grid(cb1, cb2, cb3, nrow = 3, ncol = 1, rel_heights = c(1,1,1.1), labels =
"auto", label_x = 0.11, label_y = 0.9)
gs_rff
```



7.

Figure 7: SHAP thresholds for NEP

Finally we plot the SHAP values calculated for VPD and SWC against the VPD and SWC values and we use a local regression with a 0.8 span. Where the curve intersects the SHAP = 0 line we evidence the peak of the line and define it as the point in which there is the maximum rate of NEP.

```
SHAPs_CDH_LAE$DOY_day <- SHAPs_CDH_LAE$ID + 119
#SHAPs_CDH_LAE[SHAPs_CDH_LAE$Year == 2022, "DOY_day"] <- SHAPs_CDH_LAE[SHAPs_CDH_LAE$Year ==
2022, "DOY_day"] + 119 # for the January gap
# there is a gap between doy 250 and 258
SHAPs_CDH_LAE[SHAPs_CDH_LAE$Year == 2022 & SHAPs_CDH_LAE$DOY_day > 249, "DOY_day"] <-
SHAPs_CDH_LAE[SHAPs_CDH_LAE$Year == 2022 & SHAPs_CDH_LAE$DOY_day > 249, "DOY_day"] + 9


SHAPs_CDH_LAE <- SHAPs_CDH_LAE %>% mutate(season = dplyr::case_when(DOY_day %in% 275:290 ~
"Fall",
                                                            DOY_day %in% 100:181  ~ "Spring",
                                                            DOY_day %in% 182:274 ~ "Summer",
                                                            DOY_day%in% c(291:365, 1:99) ~
"Winter"))

### Finding the optimum of the curves
# SWC

lm15.SWC = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "SWC_stnd") %>%
filter(Year == 2015),
               span = 1)
opt.SWC.15 <- lm15.SWC$x[which(lm15.SWC$fitted == max(lm15.SWC$fitted))]
se.SWC.15 <- summary(lm15.SWC)$s

lm18.SWC = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "SWC_stnd") %>%
filter(Year == 2018),
                 span = 1)
opt.SWC.18 <- lm18.SWC$x[which(lm18.SWC$fitted == max(lm18.SWC$fitted))]
se.SWC.18 <- summary(lm18.SWC)$s

lm22.SWC = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "SWC_stnd") %>%
filter(Year == 2022),
                 span = 1)
opt.SWC.22 <- lm22.SWC$x[which(lm22.SWC$fitted == max(lm22.SWC$fitted))]
se.SWC.22 <- summary(lm22.SWC)$s

lm.SWC = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "SWC_stnd"),
                 span = 1)
opt.SWC <- lm.SWC$x[which(lm.SWC$fitted == max(lm.SWC$fitted))]
se.SWC<- summary(lm.SWC)$s


lm.SWC = loess(mean_value ~ mean_rfvalue, data = SHAPs_CDH_LAE %>% filter(variable ==
"SWC_stnd") %>% group_by(DOY_day) %>%
                 summarise(mean_value = mean(value, na.rm = T), mean_rfvalue = mean(rfvalue,
na.rm = T)),
               span = 1)
opt.SWC <- lm.SWC$x[which(lm.SWC$fitted == max(lm.SWC$fitted))]
se.SWC <- summary(lm15.SWC)$s


# VPD
lm15.VPD = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "VPD_f") %>%
filter(Year == 2015),
                span = 1)
opt.VPD.15 <- lm15.VPD$x[which(lm15.VPD$fitted == max(lm15.VPD$fitted))]
se.VPD.15 <- summary(lm15.VPD)$s

lm18.VPD = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "VPD_f") %>%
filter(Year == 2018),
                span = 1)
```

```r
opt.VPD.18 <- lm18.VPD$x[which(lm18.VPD$fitted == max(lm18.VPD$fitted))]
se.VPD.18 <- summary(lm18.VPD)$s

lm22.VPD = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "VPD_f") %>%
filter(Year == 2022),
                span = 1)
opt.VPD.22 <- lm22.VPD$x[which(lm22.VPD$fitted == max(lm22.VPD$fitted))]
se.VPD.22 <- summary(lm22.VPD)$s

lm.VPD = loess(mean_value ~ mean_rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "VPD_f")
%>% group_by(DOY_day) %>%
                 summarise(mean_value = mean(value, na.rm = T), mean_rfvalue = mean(rfvalue,
na.rm = T)),
               span = 1)
opt.VPD <- lm.VPD$x[which(lm.VPD$fitted == max(lm.VPD$fitted))]
se.VPD <- summary(lm.VPD)$s


# TA
lm15.TA = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "Tair_f") %>%
filter(Year == 2015),
                 span = 1)
opt.TA.15 <- lm15.TA$x[which(lm15.TA$fitted == max(lm15.TA$fitted))]
se.TA.15 <- summary(lm15.TA)$s

lm18.TA = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "Tair_f") %>%
filter(Year == 2018),
                 span = 1)
opt.TA.18 <- lm18.TA$x[which(lm18.TA$fitted == max(lm18.TA$fitted))]
se.TA.18 <- summary(lm18.TA)$s

lm22.TA = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "Tair_f") %>%
filter(Year == 2022),
                 span = 1)
opt.TA.22 <- lm22.TA$x[which(lm22.TA$fitted == max(lm22.TA$fitted))]
se.TA.22 <- summary(lm22.TA)$s

lm.TA = loess(value ~ rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "Tair_f"),
               span = 1)
opt.TA <- lm.TA$x[which(lm.TA$fitted == max(lm.TA$fitted))]
se.TA<- summary(lm.TA)$s

lm.TA = loess(mean_value ~ mean_rfvalue, data = SHAPs_CDH_LAE %>% filter(variable == "Tair_f")
%>% group_by(DOY_day) %>%
                 summarise(mean_value = mean(value, na.rm = T), mean_rfvalue = mean(rfvalue,
na.rm = T)),
               span = 1)
opt.TA <- lm.TA$x[which(lm.TA$fitted == max(lm.TA$fitted))]
se.ta <- summary(lm.TA)$s


## PLOTTING ##
vpd15 <- SHAPs_CDH_LAE %>% filter(variable == "VPD_f") %>% filter(Year == 2015) %>% ggplot() +
  annotate(geom = "rect", xmin = (opt.VPD.15 -se.VPD.15)/10, xmax = (opt.VPD.15 +
se.VPD.15)/10, ymin = -Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +
  geom_vline(xintercept = opt.VPD.15/10, lty = "dashed") +
  geom_point(data = .%>%  filter(!(DOY_day %in% c(188:202, 214:225))), aes(x = rfvalue/10, y =
value, col  = "Rest of the year", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(188:202, 214:225)),  aes(x = rfvalue/10, y =
value, col = "CDH", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue/10, y = value), col = "grey2", fill = "grey20", alpha =0.2, span
= 1) +
  scale_color_manual(values = c("#C7980A","#FBD446"),labels = c("CSAD 2015", "2015"), name =
"") +
  scale_shape_manual(values = c(17,19), name = "", guide = "none") +
  scale_x_continuous(breaks = seq(0, 2.5, by = 0.5), limits = c(0, 2.5)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2), labels = c(-4, "", 0, "", 4), limits = c(-5,
5)) +  ylab("") +
  xlab("VPD (kPa)") +
  th1 + guides(color = guide_legend(label.position = "left", override.aes = list(shape = 16)))+
  theme(legend.position = c(0.67, 0.80), axis.title.x = element_blank(),
        legend.box.just = "right", axis.text.x = element_blank())

vpd18 <- SHAPs_CDH_LAE %>% filter(variable == "VPD_f") %>% filter(Year == 2018) %>% ggplot() +
  annotate(geom = "rect", xmin = (opt.VPD.18 -se.VPD.18)/10, xmax = (opt.VPD.18 +
se.VPD.18)/10, ymin = -Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +

  geom_vline(xintercept = opt.VPD.18/10, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(204:235))), aes(x = rfvalue/10, y = value,
col  = "Rest of the year", shape = season),size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(204:235)),  aes(x = rfvalue/10, y = value, col
= "CDH", shape = season),size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue/10, y = value), col = "grey2", fill = "grey20", alpha =0.2, span
= 1) +
  scale_color_manual(values = c("#6248BF","#9789CA"),labels = c("CSAD 2018", "2018"), name =
"") +
  scale_shape_manual(values = c(17,19), name = "", guide = "none") +
  scale_x_continuous(breaks = seq(0, 2.5, by = 0.5), limits = c(0, 2.5)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2), labels = c(-4, "", 0, "", 4), limits = c(-5,
5)) +  ylab("SHAP values") +
  xlab("") +
  th1 + guides(col = guide_legend(label.position = "left", override.aes = list(shape = 16))) +
  theme(legend.position = c(0.67, 0.80), axis.title.x = element_blank(),  legend.box.just =
"right",
        axis.text.x = element_blank())

vpd22 <- SHAPs_CDH_LAE %>% filter(variable == "VPD_f") %>% filter(Year == 2022) %>% ggplot() +
  annotate(geom = "rect", xmin = (opt.VPD.22 -se.VPD.22)/10, xmax = (opt.VPD.22+ se.VPD.22)/10,
ymin = -Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +

  geom_vline(xintercept = opt.VPD.22/10, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(195:216))), aes(x = rfvalue/10, y = value,
col  = "Rest of the year", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(195:216)),  aes(x = rfvalue/10, y = value, col
= "CDH", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue/10, y = value), col = "grey2", fill = "grey20", alpha =0.2, span
= 1) +
  scale_color_manual(values = c("#F86624","#FAA67A"),labels = c("CSAD 2022", "2022"), name =
"") +
  scale_shape_manual(values = c(17,19),  name = "", guide = "none") +
  scale_x_continuous(breaks = seq(0, 2.5, by = 0.5), limits = c(0, 2.5)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2), labels = c(-4, "", 0, "", 4), limits = c(-5,
5)) +  ylab("SHAP values") +  ylab("") +
  xlab("VPD (kPa)") +
  th1 + guides(col = guide_legend(label.position = "left", override.aes = list(shape = 16))) +
  theme(legend.position = c(0.67, 0.80),  legend.box.just = "right")

swc15<- SHAPs_CDH_LAE %>% filter(variable == "SWC_stnd") %>% filter(Year == 2015) %>% ggplot()
+
  annotate(geom = "rect", xmin = opt.SWC.15 -se.SWC.15, xmax = opt.SWC.15+ se.SWC.15, ymin = -
Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +

  geom_vline(xintercept = opt.SWC.15, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(188:202, 214:225))), aes(x = rfvalue, y =
value, col  = "Rest of the year", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(188:202, 214:225)),  aes(x = rfvalue, y =
value, col = "CDH", shape = season),size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span =
1) +
  scale_color_manual(values = c("#C7980A","#FBD446"),labels = c("CDH", "Rest of the year"),
name = "", guide = "none") +
  scale_shape_manual(values = c(17,19), name = "") +
  scale_x_continuous(breaks = seq(-2, 2, by = 1), limits = c(-2.5, 2.5)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2), labels = c(-4, "", 0, "", 4), limits = c(-5,
5)) +  ylab("SHAP values") +  ylab("SHAP value") +
  xlab("SWC") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(), axis.title.y =
element_blank(),
              axis.text.x = element_blank(), axis.text.y = element_blank())

swc18 <- SHAPs_CDH_LAE %>% filter(variable == "SWC_stnd") %>% filter(Year == 2018) %>% ggplot()
+
  annotate(geom = "rect", xmin = opt.SWC.18 -se.SWC.18, xmax = opt.SWC.18+ se.SWC.18, ymin = -
Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +

  geom_vline(xintercept = opt.SWC.18, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(204:235))), aes(x = rfvalue, y = value, col
= "Rest of the year", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(204:235)),  aes(x = rfvalue, y = value, col =
"CDH", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span =
1) +
  scale_color_manual(values = c("#6248BF","#9789CA"),labels = c("CDH", "Rest of the year"),
name = "", guide = "none") +
  scale_shape_manual(values = c(17,19), name = "") +
  scale_x_continuous(breaks = seq(-2, 2, by = 1), limits = c(-2.5, 2.5)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2), labels = c(-4, "", 0, "", 4),  limits = c(-5,
5)) +  ylab("SHAP values") +  ylab("SHAP value") +
  xlab("SWC") +
  th1 + theme(legend.position = c(0.76, 0.16), axis.title.x = element_blank(), axis.title.y =
element_blank(), axis.text.x = element_blank(),
              legend.box.background = element_rect(fill = "transparent"), legend.key.size =
unit(5, "mm"),
              legend.text = element_text(size = 14),  axis.text.y = element_blank()) +
guides(shape = guide_legend(nrow = 2, label.position = "left"))

swc22 <- SHAPs_CDH_LAE %>% filter(variable == "SWC_stnd") %>% filter(Year == 2022) %>% ggplot()
+
  annotate(geom = "rect", xmin = opt.SWC.22 -se.SWC.22, xmax = opt.SWC.22+ se.SWC.22, ymin = -
Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +

  geom_vline(xintercept = opt.SWC.22, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(195:216))), aes(x = rfvalue, y = value, col
= "Rest of the year", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(195:216)),  aes(x = rfvalue, y = value, col =
"CDH", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span =
1) +
  scale_color_manual(values = c("#F86624","#FAA67A"),labels = c("CDH", "Rest of the year"),
name = "") +
  scale_shape_manual(values = c(17,19), name = "") +
  scale_x_continuous(breaks = seq(-2, 2, by = 1), limits = c(-2.5, 2.5)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2), labels = c(-4, "", 0, "", 4), limits = c(-5,
5)) +  ylab("SHAP values") +
  ylab("SHAP value") +
  xlab("Norm. SWC") +
  th1 + theme(legend.position = "none", axis.title.y = element_blank(), axis.text.y =
element_blank())

ta15<- SHAPs_CDH_LAE %>% filter(variable == "Tair_f") %>% filter(Year == 2015) %>% ggplot() +
  annotate(geom = "rect", xmin = opt.TA.15 -se.TA.15, xmax = opt.TA.15+ se.TA.15, ymin = -Inf,
ymax = Inf, alpha = 0.2,
           fill = "grey50") +

  geom_vline(xintercept = opt.TA.15, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(188:202, 214:225))), aes(x = rfvalue, y =
value, col  = "Rest of the year", shape = season),size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(188:202, 214:225)),  aes(x = rfvalue, y =
value, col = "CDH", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span =
1) +
  scale_color_manual(values = c("#C7980A","#FBD446"),labels = c("CDH", "Rest of the year"),
name = "") +
  scale_shape_manual(values = c(17,19), name = "") +
  scale_x_continuous(breaks = seq(5, 30, by = 5), limits = c(5, 30)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2), labels = c(-4, "", 0, "", 4), limits = c(-5,
5)) +  ylab("SHAP values") +
  ylab("SHAP value") +
  xlab("SWC") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(), axis.title.y =
element_blank(),
              axis.text.x = element_blank(), axis.text.y = element_blank())

ta18 <- SHAPs_CDH_LAE %>% filter(variable == "Tair_f") %>% filter(Year == 2018) %>% ggplot() +
  annotate(geom = "rect", xmin = opt.TA.18 -se.TA.18, xmax = se.TA.18+ opt.TA.18, ymin = -Inf,
ymax = Inf, alpha = 0.2,
           fill = "grey50") +

  geom_vline(xintercept = opt.TA.18, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(204:235))), aes(x = rfvalue, y = value, col
= "Rest of the year", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(204:235)),  aes(x = rfvalue, y = value, col =
"CDH", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span =
1) +
  scale_color_manual(values = c("#6248BF","#9789CA"),labels = c("CDH", "Rest of the year"),
name = "", guide = "none") +
  scale_shape_manual(values = c(17,19), name = "") +
  scale_x_continuous(breaks = seq(5, 30, by = 5), limits = c(5, 30)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2), labels = c(-4, "", 0, "", 4),  limits = c(-5,
5)) +  ylab("SHAP values") +
  ylab("SHAP value") +
  xlab("SWC") +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(), axis.title.y =
element_blank(), axis.text.x = element_blank(),
              legend.background = element_rect(colour = "white"), legend.key.size = unit(5,
"mm"),
              legend.text = element_text(size = 7), axis.text.y = element_blank())

ta22 <- SHAPs_CDH_LAE %>% filter(variable == "Tair_f") %>% filter(Year == 2022) %>% ggplot() +
  annotate(geom = "rect", xmin = opt.TA.22 - se.TA.22, xmax = opt.TA.22+ se.TA.22, ymin = -Inf,
ymax = Inf, alpha = 0.2,
           fill = "grey50") +
  geom_vline(xintercept = opt.TA.22, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(195:216))), aes(x = rfvalue, y = value, col
= "Rest of the year", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(195:216)),  aes(x = rfvalue, y = value, col =
"CDH", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span =
1) +
  scale_color_manual(values = c("#F86624","#FAA67A"),labels = c("CDH", "Rest of the year"),
name = "") +
  scale_shape_manual(values = c(17,19), name = "") +
  scale_x_continuous(breaks = seq(5, 30, by = 5), limits = c(5,30)) +
  scale_y_continuous(breaks = seq(-4, 5, by = 2),labels = c(-4, "", 0, "", 4), limits = c(-5,
5)) +  ylab("SHAP values") +
  ylab("SHAP value") +
  xlab("Tair (°C)") +
  th1 + theme(legend.position = "none", axis.title.y = element_blank(), axis.text.y =
element_blank())

gg <- ggarrange(vpd15, swc15, ta15, vpd18, swc18, ta18, vpd22, swc22, ta22,
                ncol = 3, nrow = 3,  labels = "auto", label.x = c(0.25, 0.10, 0.10), label.y =
0.96, heights = c(1,1,1.3),
                widths = c(1.2, 1, 1))

## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 1 rows containing non-finite values (`stat_smooth()`).

## Warning: Removed 1 rows containing missing values (`geom_point()`).

## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 2 rows containing non-finite values (`stat_smooth()`).

## Warning: Removed 2 rows containing missing values (`geom_point()`).

## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 2 rows containing non-finite values (`stat_smooth()`).

## Warning: Removed 1 rows containing missing values (`geom_point()`).
## Removed 1 rows containing missing values (`geom_point()`).

## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

gg
```

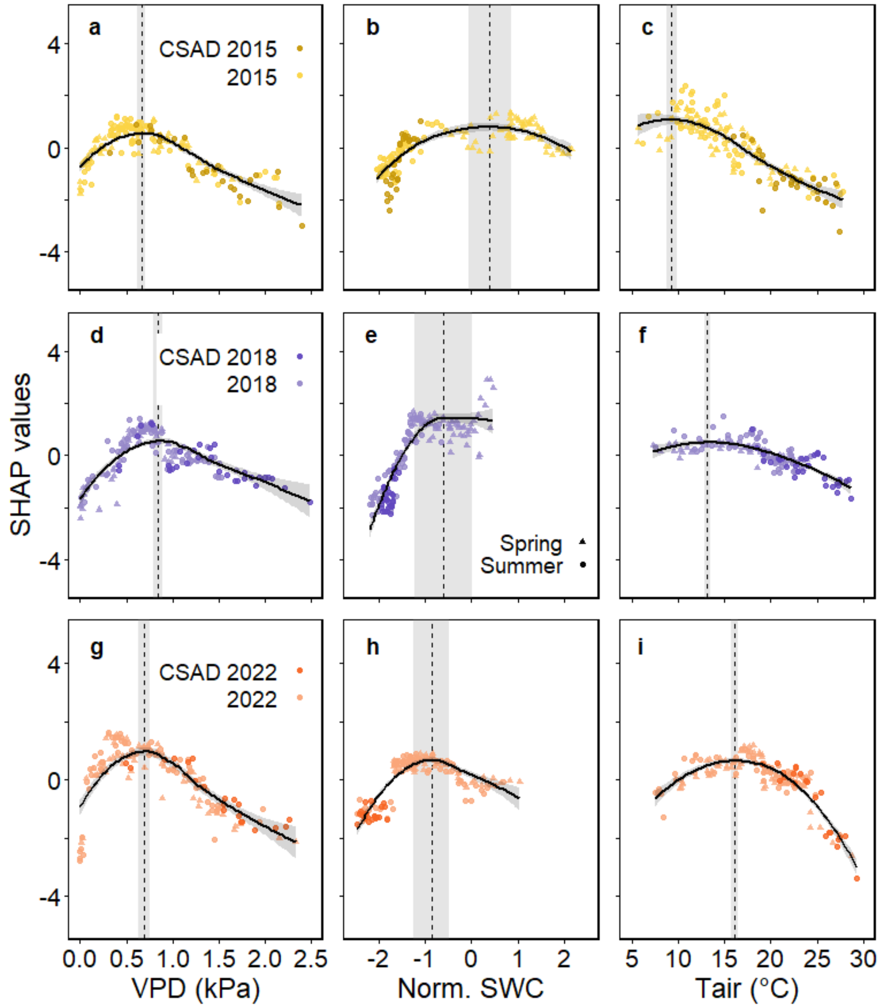


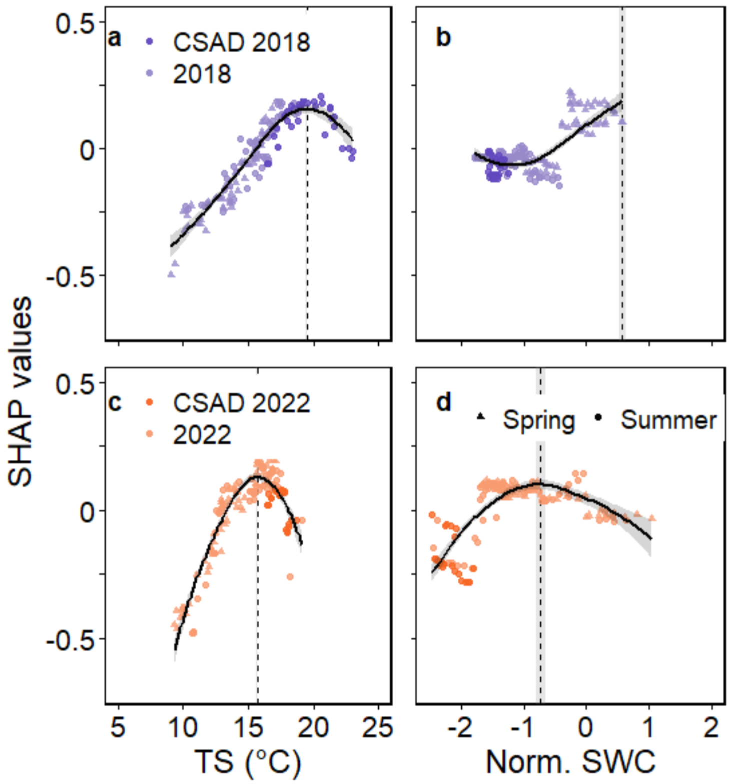
***Figure 8**: SHAP thresholds for Rff*

Finally we look into the thresholds for TS and SWC

```
SHAPs_CDH_LAS$DOY_day <- SHAPs_CDH_LAS$ID + 119
SHAPs_CDH_LAS <- SHAPs_CDH_LAS %>% mutate(season = dplyr::case_when(DOY_day %in% 275:290 ~
"Fall",
                                                            DOY_day %in% 100:181  ~ "Spring",
                                                            DOY_day %in% 182:274 ~ "Summer",
                                                            DOY_day%in% c(291:365, 1:99) ~
"Winter"))
# calculating the optimum of TS and SWC

lm18.TS = loess(value ~ rfvalue, data = SHAPs_CDH_LAS %>% filter(variable == "TS") %>%
filter(Year == 2018),
                span = 1)
opt.TS.18 <- lm18.TS$x[which(lm18.TS$fitted == max(lm18.TS$fitted))]
se.TS.18 <- summary(lm18.TS)$s

lm22.TS = loess(value ~ rfvalue, data = SHAPs_CDH_LAS %>% filter(variable == "TS") %>%
filter(Year == 2022),
                span = 1)
opt.TS.22 <- lm22.TS$x[which(lm22.TS$fitted == max(lm22.TS$fitted))]
se.TS.22 <- summary(lm22.TS)$s
```

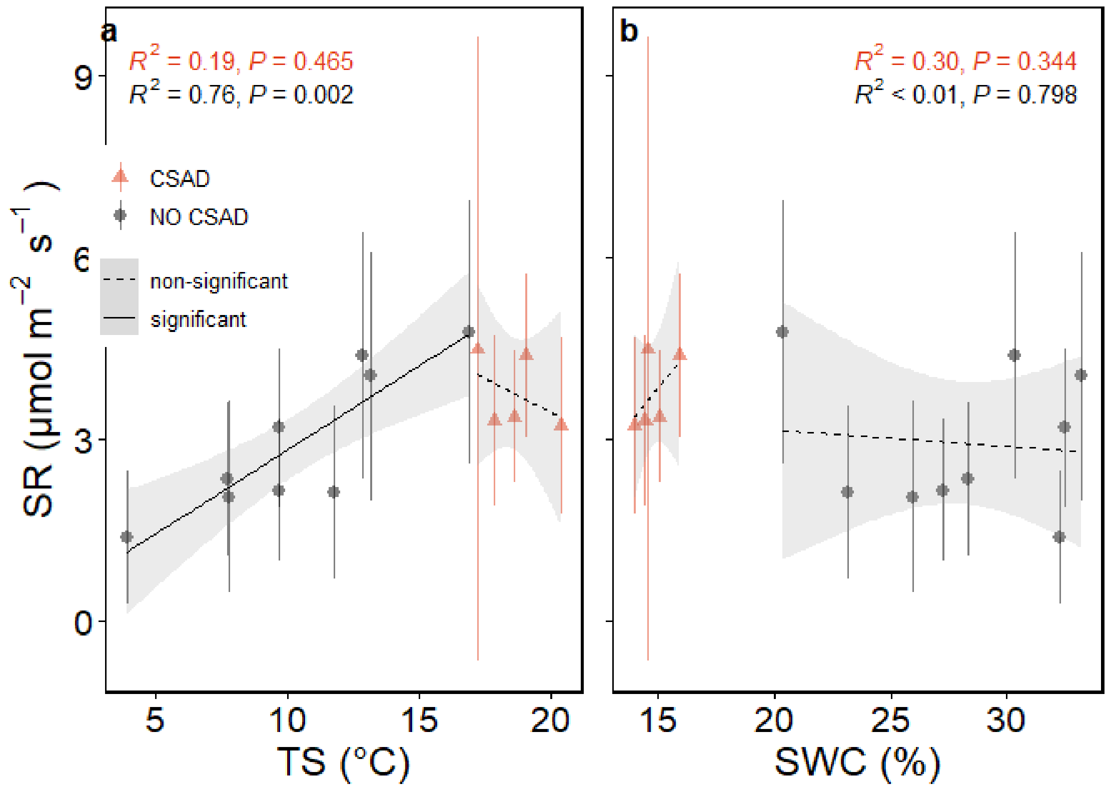
```
lm.TS = loess(value ~ rfvalue, data = SHAPs_CDH_LAS %>% filter(variable == "TS") %>%
filter(Year %in% 2019:2021), span = 1)
opt.TS<- lm.TS$x[which(lm.TS$fitted == max(lm.TS$fitted))]
se.TS <- summary(lm.TS)$s

lm18.SWC5 = loess(value ~ rfvalue, data = SHAPs_CDH_LAS %>% filter(variable == "SWC") %>%
filter(Year == 2018),
                 span = 1)
opt.SWC5.18 <- lm18.SWC5$x[which(lm18.SWC5$fitted == max(lm18.SWC5$fitted))]
se.SWC5.18 <- summary(lm18.SWC5)$s

lm22.SWC5 = loess(value ~ rfvalue, data = SHAPs_CDH_LAS %>% filter(variable == "SWC") %>%
filter(Year == 2022),
                 span = 1)
opt.SWC5.22 <- lm22.SWC5$x[which(lm22.SWC5$fitted == max(lm22.SWC5$fitted))]
se.SWC5.22 <- summary(lm22.SWC5)$s

lm.SWC5 = loess(value ~ rfvalue, data = SHAPs_CDH_LAS %>% filter(variable == "SWC") %>%
filter(Year %in% 2019:2021),
                 span = 1)
opt.SWC5 <- lm.SWC5$x[which(lm.SWC5$fitted == max(lm.SWC5$fitted))]
se.SWC5 <- summary(lm.SWC5)$s
## PLOTTING ##

SHAPs_CDH_LAS$Year <- as.factor(SHAPs_CDH_LAS$Year)

ts18 <- SHAPs_CDH_LAS %>% filter(variable == "TS") %>% filter(Year == 2018) %>% ggplot() +
  annotate(geom = "rect", xmin = opt.TS.18 - se.TS.18, xmax = opt.TS.18 + se.TS.18, ymin = -
Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +
  geom_vline(xintercept = opt.TS.18, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(204:235))), aes(x = rfvalue, y = value, col
= "Rest of the year", shape = season),  size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(204:235)),  aes(x = rfvalue, y = value, col =
"CDH", shape = season),  size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span =
1) +
  scale_color_manual(values = c("#6248BF","#9789CA"),labels = c("CSAD 2018", "2018"), name =
"") +
  scale_shape_manual(values = c(17,19), name = "", guide = "none") +
  scale_x_continuous(breaks = seq(5, 25, by = 5), limits = c(5, 25)) +
  scale_y_continuous(breaks = seq(-0.5, 0.5, by = 0.25), limits = c(-0.7, 0.5), labels = c(-
0.5, "", 0, "",0.5)) +
  ylab("") +
  xlab("") +
  th1 + guides(col = guide_legend(label.position = "right", override.aes = list(shape = 16))) +
  theme(legend.position = c(0.4, 0.85), axis.title.x = element_blank(),  legend.box.just =
"left",
        axis.text.x = element_blank())

ts22 <- SHAPs_CDH_LAS %>% filter(variable == "TS") %>% filter(Year == 2022) %>% ggplot() +
  annotate(geom = "rect", xmin = opt.TS.22 - se.TS.22, xmax = opt.TS.22 + se.TS.22, ymin = -
Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +
  geom_vline(xintercept = opt.TS.22, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(195:216))), aes(x = rfvalue, y = value, col
= "Rest of the year", shape = season),  size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(195:216)),  aes(x = rfvalue, y = value, col =
"CDH", shape = season),  size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span =
1) +
  scale_color_manual(values = c("#F86624","#F99B70"),labels = c("CSAD 2022", "2022"), name =
"") +
  scale_shape_manual(values = c(17,19),  name = "", guide = "none") +
  scale_x_continuous(breaks = seq(5, 25, by = 5), limits = c(5, 25)) +
  scale_y_continuous(breaks = seq(-0.5, 0.5, by = 0.25), limits = c(-0.7, 0.5), labels = c(-
0.5, "", 0, "",0.5)) +
  ylab("") +
  xlab("TS (°C)") +
  th1 + guides(col = guide_legend(label.position = "right",  override.aes = list(shape = 16)))
+
  theme(legend.position = c(0.4, 0.85),  legend.box.just = "left")


swc18.las <- SHAPs_CDH_LAS %>% filter(variable == "SWC") %>%  filter(Year == 2018) %>%
ggplot() +
  annotate(geom = "rect", xmin = opt.SWC5.18 - se.SWC5.18, xmax = opt.SWC5.18 + se.SWC5.18,
ymin = -Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +
  geom_vline(xintercept = opt.SWC5.18, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(204:235))), aes(x = rfvalue, y = value, col
= "Rest of the year", shape = season), size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(204:235)),  aes(x = rfvalue, y = value, col =
"CDH", shape = season),  size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span= 1)
+
  scale_color_manual(values = c("#6248BF", "#9789CA"), breaks = c("CDH","Rest of the year"),
labels = c("CDH 2018", "2018"), name = "", guide = "none") +
  scale_shape_manual(values = c(17,19), name = "") +
  scale_x_continuous(limits = c(-2.5, +2), breaks = seq(-2,2, by = 1), labels = seq(-2, 2, by =
1)) +
  scale_y_continuous(breaks = seq(-0.5, 0.5, by = 0.25), limits = c(-0.7, 0.5), labels = c(-
0.5, "", 0, "",0.5)) +
  ylab("SHAP value") +
  xlab("SWC") + guides(shape = guide_legend(label.position = "left") ) +
  th1 + theme(legend.position = "none", axis.title.x = element_blank(), axis.title.y =
element_blank(),
              axis.text.x = element_blank(), axis.text.y = element_blank())

swc22.las <- SHAPs_CDH_LAS %>% filter(variable == "SWC") %>%  filter(Year == 2022) %>%
ggplot() +
  annotate(geom = "rect", xmin = opt.SWC5.22 - se.SWC5.22, xmax = opt.SWC5.22 + se.SWC5.22,
ymin = -Inf, ymax = Inf, alpha = 0.2,
           fill = "grey50") +
  geom_vline(xintercept = opt.SWC5.22, lty = "dashed") +
  geom_point(data = . %>%  filter(!(DOY_day %in% c(195:216))), aes(x = rfvalue, y = value, col
= "Rest of the year", shape = season),  size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_point(data = . %>%  filter(DOY_day %in% c(195:216)),  aes(x = rfvalue, y = value, col =
"CDH", shape = season),  size = 1.3, stroke = 0.8, alpha = 0.8) +
  geom_smooth(aes(x = rfvalue, y = value), col = "grey2", fill = "grey20", alpha =0.2, span= 1)
+
  scale_color_manual(values = c("#F86624","#F99B70"),labels = c("CDH 2022", "2022"), name = "",
guide = "none") +
  scale_shape_manual(values = c(17,19), name = "") +
  scale_y_continuous(breaks = seq(-0.5, 0.5, by = 0.25), limits = c(-0.7, 0.5), labels = c(-
0.5, "", 0, "",0.5)) +
  scale_x_continuous(limits = c(-2.5, +2), labels = seq(-2, 2, by = 1), breaks = seq(-2,2, by =
1)) +
  ylab("") +
  xlab("Norm. SWC") + guides(shape = guide_legend(label.position = "right", ncol = 2)) +
  th1 + theme(legend.position = c(0.55, 0.85),  legend.box.just = "right", legend.direction =
"horizontal", legend.background = element_rect(colour = "white"),
                legend.key.size = unit(5, "mm"), axis.text.y = element_blank(), axis.title.y =
element_blank())
```

```
pl <- plot_grid(ts18, swc18.las, ts22, swc22.las, ncol = 2, nrow = 2,
                labels = "auto", label_x =c(0.25,0.08), label_y = 0.95, rel_widths = c(1.2, 1),
rel_heights = c(1, 1.2))

## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 1 rows containing non-finite values (`stat_smooth()`).

## Warning: Removed 1 rows containing missing values (`geom_point()`).

annotate_figure(pl, left = textGrob("SHAP values", rot = 90, vjust = 2, gp = gpar(cex = 1.5)))
```



9. Figure 9: SR survey

```
surv.day <- sr.22 %>% filter(flux>0) %>% mutate(doy = yday(timestamp)) %>% group_by(doy) %>%
  summarise(mean_flux = mean(flux, na.rm = T),
            sd_flux = sd(flux, na.rm = T),
            mean_ST = mean(ST, na.rm = T),
            sd_ST = sd(ST, na.rm = T),
            mean_SWC = mean(SWC, na.rm = T),
            sd_SWC = sd(SWC, na.rm = T)) %>% mutate(CSAD = ifelse(doy %in% 194:224, "CSAD", "NO
CSAD"))

# After using the values obtained by the conversion equation to get the SWC in the days th
instrument was not working, the calulated values are used to replace the Nan in the data
surv.day[surv.day[,"doy"] == 69, "mean_SWC"] <- 32.27
surv.day[surv.day[,"doy"] == 203, "mean_SWC"] <- 15.88
surv.day[surv.day[,"doy"] == 216, "mean_SWC"] <- 13.98

surv.day[surv.day[,"doy"] == 69, "sd_SWC"] <- 8.732482
surv.day[surv.day[,"doy"] == 203, "sd_SWC"] <- 5.90697
surv.day[surv.day[,"doy"] == 216, "sd_SWC"] <- 5.90697


ts <- surv.day  %>%
  ggplot(aes(x = mean_ST, y = mean_flux)) +
  geom_pointrange(aes(x = mean_ST, y = mean_flux, ymax = mean_flux + sd_flux, ymin = mean_flux
- sd_flux, col  = CSAD, pch = CSAD),
                  size = 0.5, alpha = 0.5) +
  geom_smooth(data = . %>%  filter (CSAD == "NO CSAD"), aes(linetype = "significant"), col =
"black" ,
              span = 1,  alpha = 0.2, method = "lm", formula = y ~ x, linewidth = 0.5) +
  geom_smooth(data = . %>%  filter (CSAD == "CSAD"), aes( linetype = "non-significant"),col =
"black",
              span = 1,  alpha = 0.2, method = "lm", formula = y ~ x, linewidth = 0.5) +
  scale_color_manual(values = c("#E03916", "black")) +
  scale_fill_manual(values = c("#E03916", "black")) +
  scale_linetype_manual(values = c(2, 1)) +
  scale_shape_manual(values = c(17,16)) +
  stat_poly_eq(aes(group = CSAD, col = CSAD, label =  paste(after_stat(rr.label),
                                                            after_stat(p.value.label), sep =
"*\", \"*"))) +
  scale_y_continuous(breaks = c(0, 3, 6, 9)) +
  ylab("SR (μmol m" ^-2~" s"^-1~")") +
  xlab("TS (°C)") +
  th1 + theme(legend.position = c(0.20, 0.65), legend.text = element_text(size = 10))

swc <- surv.day %>%
  ggplot(aes(x = mean_SWC, y = mean_flux)) +
  geom_pointrange(aes(x = mean_SWC, y = mean_flux, ymax = mean_flux + sd_flux, ymin = mean_flux
- sd_flux,, col  = CSAD, pch = CSAD),
                  size = 0.5, alpha = 0.5) +
  geom_smooth(data = . %>%  filter (CSAD == "NO CSAD"), aes(linetype = "non-significant"), col
= "black" ,
              span = 1,  alpha = 0.2, method = "lm", formula = y ~ x, linewidth = 0.5) +
  geom_smooth(data = . %>%  filter (CSAD == "CSAD"), aes( linetype = "non-significant"),col =
"black",
              span = 1,  alpha = 0.2, method = "lm", formula = y ~ x, linewidth = 0.5) +
  stat_poly_eq(aes(group = CSAD, col = CSAD, label =  paste(after_stat(rr.label),
                                                            after_stat(p.value.label), sep =
"*\", \"*")),
               label.x = 0.95) +
  scale_y_continuous(breaks = c(0, 3, 6, 9)) +
  scale_color_manual(values = c("#E03916", "black")) +
  scale_fill_manual(values = c("#E03916", "black")) +
  scale_linewidth_manual(values = c(1.2, 0.5)) +
  scale_shape_manual(values = c(17,16)) +
  scale_linetype_manual(values = c(2, 2)) +
  ylab("SR (μmol m" ^-2~" s"^-1~")") +
  xlab("SWC (%)") +
  th1 + theme(legend.position = "none", axis.text.y = element_blank(), axis.title.y =
element_blank())

ggarrange(ts, swc, ncol = 2, widths = c(1.15, 1), labels = c("a", "b"), label.x = c(0.16,
0.03), label.y = 0.99)
```



Extra content

Figure A1

```
#names(lae_30MIN)

EC_Daily <- lae_30MIN %>% ungroup() %>% arrange(Date) %>%
              mutate(NEP_15ma = zoo::rollapply(NEP, width = 15, FUN=function(x) mean(x,
na.rm=TRUE), partial=TRUE, align = 'center')) %>%
              group_by(doy) %>%
              summarise(NEPm = mean(NEP_15ma, na.rm = TRUE), NEPse = std.error(NEP_15ma, na.rm
= TRUE))
summary(EC_Daily)

##       doy               NEPm              NEPse
##  Min.   :  1.00   Min.   :-3.1267   Min.   :0.05971
##  1st Qu.: 92.25   1st Qu.:-1.6504   1st Qu.:0.09448
##  Median :183.50   Median :-0.1632   Median :0.13418
##  Mean   :183.50   Mean   : 0.4705   Mean   :0.16476
##  3rd Qu.:274.75   3rd Qu.: 2.7444   3rd Qu.:0.24267
##  Max.   :366.00   Max.   : 5.2670   Max.   :0.31361

##1.2.2 Growing season detection-----------------
names(EC_Daily)

## [1] "doy"   "NEPm"  "NEPse"

EC_Daily %>% ggplot(., aes(x = doy, y = NEPm, ymin = NEPm-NEPse, ymax = NEPm+NEPse), linewidth
= 1) +
  geom_ribbon(alpha = 0.4,show.legend = FALSE, color = NA, fill = "#33C55C") +
  geom_line(show.legend = FALSE, linewidth =0.5, col = "#33C55C") +
  scale_x_continuous(breaks= c(1, 32, 60, 91, 121, 152, 182, 213, 244, 274, 305, 335, 366),
                     labels = c("J", "F", "M", "A", "M", "J", "J", "A", "S", "O", "N", "D",
"J"))+
  labs(x = 'Day of year', y = "Mean NEP (µmol m" ^-2~" s"^-1~")") +
  geom_hline(yintercept = 0, lty = 'dashed') +
  th1
```

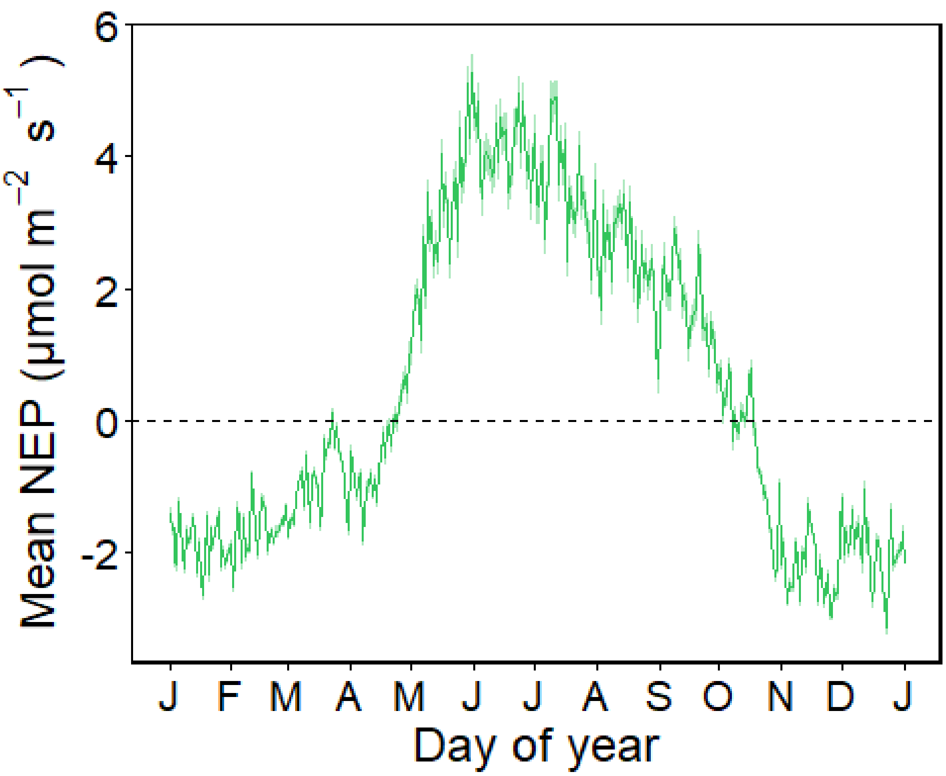
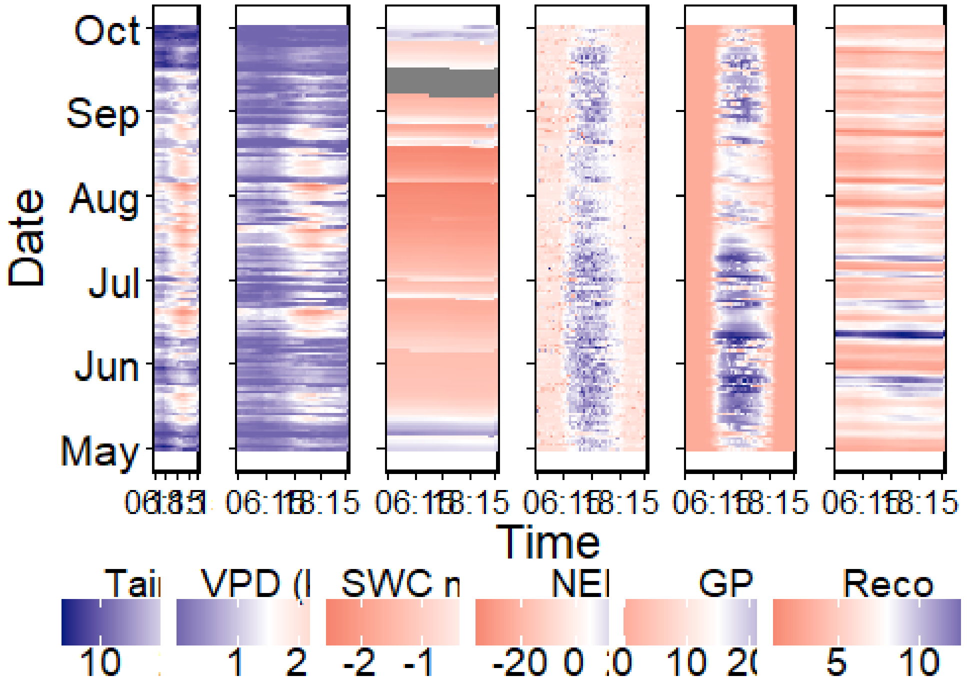


Figure A3

```
finger_22 = lae_30MIN %>% mutate(Time = as.character(format(Timestamp-15*60, '%H:%M')),
                                                Year = as.numeric(format(Timestamp-15*60,'%Y')))
%>%
  filter(Year == 2022) %>% filter (doy %in% 120:274)


fin.nep <- ggplot(finger_22, aes(x = Time, y = Date, fill = NEP)) +
  geom_tile() +
  scale_fill_gradient2(mid = 'white', high = '#031E82', low = '#F5836D', midpoint = 5, name =
"NEP") +
  scale_x_discrete(breaks = c( "00:15", "06:15", "12:15", "18:15", "23:15"), labels = c("",
"06:15","", "18:15", "")) +
  th1 + theme(axis.text.x=element_text(size = 12),legend.position = "bottom", legend.direction
= "horizontal",
              legend.title = element_text(size = 14),
              panel.border = element_rect(fill = NA), axis.text.y = element_blank(),
axis.title.y = element_blank(), axis.title.x = element_blank()) +
  guides(fill = guide_colourbar(title.position="top", title.hjust = 0.5))

fin.gpp <- ggplot(finger_22, aes(x = Time, y = Date, fill = GPP_DT)) +
  geom_tile() +
  scale_fill_gradient2(mid = 'white', high = '#031E82', low = '#F5836D', midpoint = 15, name =
"GPP") +
  scale_x_discrete(breaks = c( "00:15", "06:15", "12:15", "18:15", "23:15"), labels = c("",
"06:15","", "18:15", "")) +
  th1 + theme(axis.text.x=element_text(size = 12), legend.position = "bottom", legend.direction
= "horizontal",
              legend.title = element_text(size = 14),
              axis.text.y = element_blank(), axis.title.y = element_blank(), panel.border =
element_rect(fill = NA), axis.title.x = element_blank()) +
  guides(fill = guide_colourbar(title.position="top", title.hjust = 0.5))

fin.reco<- ggplot(finger_22, aes(x = Time, y = Date, fill = Reco_DT)) +
  geom_tile() +
  scale_fill_gradient2(mid = 'white', high = '#031E82', low = '#F5836D', midpoint = 8, name =
"Reco") +
  scale_x_discrete(breaks = c( "00:15", "06:15", "12:15", "18:15", "23:15"), labels = c("",
"06:15","", "18:15", "")) +
  th1 + theme(axis.text.x=element_text(size = 12), legend.position = "bottom", legend.direction
= "horizontal", legend.title = element_text(size = 14),
              axis.text.y = element_blank(), axis.title.y = element_blank(), panel.border =
element_rect(fill = NA), axis.title.x = element_blank()) +
  guides(fill = guide_colourbar(title.position="top", title.hjust = 0.5))


fin.vpd <- ggplot(finger_22, aes(x = Time, y = Date, fill = VPD_f/10)) +
  geom_tile() +
  scale_fill_gradient2(mid = 'white', low = '#031E82', high = '#F5836D', midpoint =1.5, name =
"VPD (kPa)") +
  scale_x_discrete(breaks = c( "00:15", "06:15", "12:15", "18:15", "23:15"), labels = c("",
"06:15","", "18:15", "")) +
  th1 + theme(axis.text.x=element_text(size = 12), legend.position = "bottom", legend.direction
= "horizontal", legend.title = element_text(size = 14),
              axis.text.y = element_blank(), axis.title.y = element_blank(), panel.border =
element_rect(fill = NA), axis.title.x = element_blank()) +
  guides(fill = guide_colourbar(title.position="top", title.hjust = 0.5))

fin.swc <- ggplot(finger_22, aes(x = Time, y = Date, fill = SWC_stnd)) + th1 +
  geom_tile() +
  scale_fill_gradient2(mid = 'white', high = '#031E82', low = '#F5836D', midpoint= 0, name =
"SWC norm.") +
  scale_x_discrete(breaks = c( "00:15", "06:15", "12:15", "18:15", "23:15"), labels = c("",
"06:15","", "18:15", "")) +
  theme(axis.text.x=element_text(size = 12), legend.position = "bottom", legend.direction =
"horizontal",
        legend.title = element_text(size = 14),
        axis.text.y = element_blank(), axis.title.y = element_blank(),panel.border =
element_rect(fill = NA), axis.title.x = element_blank()) +
  guides(fill = guide_colourbar(title.position="top", title.hjust = 0.5))

fin.t <- ggplot(finger_22, aes(x = Time, y = Date, fill = Tair_f)) +
  geom_tile() +
  scale_fill_gradient2(mid = 'white', low = '#031E82', high = '#F5836D', midpoint =20, breaks =
c(10, 20, 30), name ="Tair (°C)") +
  scale_x_discrete(breaks = c( "00:15", "06:15", "12:15", "18:15", "23:15"), labels = c("",
"06:15","", "18:15", "")) +
  th1 + theme(axis.text.x=element_text(size = 12),legend.position = "bottom", legend.direction
= "horizontal",
              panel.border = element_rect(fill = NA), legend.title = element_text(size = 14),
axis.title.x = element_blank()) +
  guides(fill = guide_colourbar(title.position="top", title.hjust = 0.5))


pp2 <-plot_grid(fin.t, fin.vpd, fin.swc, fin.nep, fin.gpp, fin.reco, ncol = 6, rel_widths =
c(1.5, 1, 1, 1 ,1,1))

 annotate_figure(pp2, bottom = textGrob("Time", rot = 0, vjust = -5, hjust = -0.2, gp =
gpar(cex = 1.5)))
```


Oct
Sep
Aug
Jul
Jun
May
Date
06:15
18:15
Time
Tai
VPD (I
SWC n
NEI
GP
Reco
10
1
2
-2
-1
-20
0
10
20
5
10