Retrieving and processing agrometeorological data from API-client sources using R software

Adrian A. Correndo[†], Luiz H. Moro Rosso[†] & Ignacio A. Ciampitti[†]

Department of Agronomy, Kansas State University, Manhattan, KS-66506
[†]Contact: correndo@ksu.edu; lhmrosso@ksu.edu; ciampitti@ksu.edu.

Compiled on 2021-01-28 .

Abstract

The purpose of this publication is to help end-users of weather data with agronomic purposes (students, researchers, farmers, advisors, etc.) to download and process gridded weather data from different Application Programming Interfaces (API client) sources using R software. This document is a tutorial developed in R that is part of the data-curation process from numerous research projects carried out at the Ciampitti's Lab, Department of Agronomy, Kansas State University. We make use of three weather databases for which specific libraries where developed in R language: i) DAYMET, ii) NASA-POWER, and iii) Climate Hazards Group InfraRed Precipitation with Station Data (CHIRPS). These databases offer different weather variables, and vary in terms of spatio-temporal coverage and resolution. This tutorial shows and explains how to retrieve weather data from multiple locations at once using latitude and longitude coordinates. In addition, it offers the possibility to create relevant secondary variables and summaries that are of agronomic interest such as Shannon diversity index of precipitation, growing degree days, extreme precipitation and temperature events, reference evapotranspiration, among others. This tutorial may serve for multiple purposes, including but not limited to research, crop yield forecast models, crop simulation models, and crop advising.

Table of contents

1 INTRODUCTION

1.1 Description

This code is intended to help end-users of agroclimatological data (e.g. students, researchers, farmers, advisors) to download and process open-source gridded-weather data from different Application Programming Interfaces sources (API client) using R-software.

The tutorial makes use of three existing libraries developed for R-software (Table 1): i) *daymetr*[a], ii) *nasapower*[b], and iii) *chirps*[c]. Basic details are provided in **Table 1**. Specific features and functionalities can be further explored at the links below ($^{a-b-c}$). These libraries come with different variables by default. Particularly, DAYMET offers the best combination of agrometeorological variables at the highest spatial resolution (~1 km^{-2}). However, the spatial coverage of DAYMET only includes North America. At a global scale, NASA-POWER offers the most complete set of variables, nonetheless, at a much lower spatial resolution (~50 km^{-2}). Lastly, CHIRPS offers global data only for precipitations, however, with a better spatial resolution (~5 km^{-2}) than NASA-POWER.

Table 1. Basic description of API-client weather databases available in R software.

Database	Coverage	Spatio-Temporal Resolution	Time Coverage	Reference
DAYMET[a]	Canada, United States, Mexico	daily, 0.01°, ~1 km^2	1980 - present	Thornton et al. (2019)
NASA-POWER[b]	Global	daily, 0.5°, ~50 km^2	1981 - present	Sparks (2018)
CHIRPS[c]	50°S to 50°N, all longitudes	daily, 0.05°, ~5 km^2	1981 - present (45 days-lag)	Funk et al. (2015)

[a] https://github.com/bluegreen-labs/daymetr;

[b] https://github.com/ropensci/nasapower;

[c] https://github.com/ropensci/chirps

During the tutorial: i) we provide lines of code showing how to download daily-weather data (**Section 2**), and ii) we offer the option to generate new variables and summaries for different time intervals or periods either historical or during the cropping season (**Sections 3 to 5**).

This code was generated using R version 4.0.3 (Linux-GNU, 64-bit) and R-studio v1.2.5042. Original file is R Markdown (*rmd) with code in chunks.

1.2 Loading packages

```
library(easypackages) # Load and/or install packages if not installed
libraries('tidyverse') # Data wrangling
libraries('lubridate') # Dates operations
libraries('kableExtra') # Table formatting
libraries('daymetr','chirps','nasapower') # Weather databases
libraries('vegan') # Shannon Diversity Index
```

1.3 Input example

1.3.1 Creating within R

In the next chunk we create a data-table with the required formats. Please, note that we use YYYY_MM_DD format, using "_" as separator to avoid format conflicts if data is generated in Spreadsheet software such as Excel, LibreCalc, or similar. Data will be later transformed to Date-format during the code.

The user could use either the provided example, or he/she might use it as a template to fill it out with pertinent data. Each row will represent a unique site/location, and key metadata such as lat-lon coordinates, and key dates will be represented by columns. At least, user must provided "Start" and "End" dates.

```
# Coordinates of each site (site names must be unique)
# Each site is a row
# Date for each site in columns
df.site <- data.frame(ID = c('1','2','3'),
                      Crop = c('Corn','Wheat','Soy'),
                      Site = c('Scandia','Belleville','Ottawa'),
                      # Both coordinates in decimal format
                      latitude = c(39.8291,39.8158,38.5398),
                      longitude = c(-97.8458,-97.6720,-95.2446))

# Specify key dates. Typically, dates relate to phenological stages
# Each date must be a column
df.time <- data.frame(ID = c('1','2','3'),
                      # Dates as YYYY_MM_DD, using "_" to separate
                      Start = c('2002_04_25','2005_10_15','2010_05_20'),
                      Flo = c('2002_07_15','2006_04_15','2010_07_05'),
                      SeFi =c('2002_08_15','2006_05_01','2010_08_15'),
                      End = c('2002_09_30','2006_06_20','2010_10_10'))

# For historical data
df.historical <- data.frame(ID = c('1','2','3'),
                      # Dates as YYYY_MM_DD, using "_" to separate
                      Start = c('2000_01_01','2000_01_01','2000_01_01'),
```

```
                     End = c('2019_12_31','2019_12_31','2019_12_31'))

# Merging sites and dates
# Seasonal
df.input <- df.site %>% left_join(df.time)

# Historical
df.historical <- df.site %>% left_join(df.historical)
```

1.3.2 Creating a .csv template

Here we export the example tables to "csv" format to use as templates

```
write.csv(df.input, 'Example_input.csv', row.names = F, na='')
write.csv(df.historical, 'Example_input_historical.csv', row.names = F, na='')
```

1.3.3 Importing a .csv file

Here we import your table from the csv file, and show how input tables should look like right after importing:

1.3.3.1 Seasonal

```
path = paste0(getwd(), '/') # Current directory or any path.
# Place your file in the current working directory (getwd)

# Input seasonal data
file_input = paste0(path, 'Example_input.csv') # Change to your file

# Open seasonal file
df.input <- read.table(file_input, sep=',', header = TRUE) %>%
  mutate_at(vars(6:ncol(.)), ~as.Date(., format='%Y_%m_%d'))

# View Seasonal
kable(df.input) %>%
  kable_styling(latex_options = c("striped"), position = "center", font_size = 10)
```

ID	Crop	Site	latitude	longitude	Start	Flo	SeFi	End
1	Corn	Scandia	39.8291	-97.8458	2002-04-25	2002-07-15	2002-08-15	2002-09-30
2	Wheat	Belleville	39.8158	-97.6720	2005-10-15	2006-04-15	2006-05-01	2006-06-20
3	Soy	Ottawa	38.5398	-95.2446	2010-05-20	2010-07-05	2010-08-15	2010-10-10

1.3.3.2 Historical

```r
path = paste0(getwd(), '/') # Current directory or any path.
# Place your file in the current working directory (getwd)

# Input historical data
file_historical = paste0(path, 'Example_input_historical.csv') # Change to your file

# Open historical file
df.historical <- read.table(file_historical, sep=',', header = TRUE) %>%
  mutate_at(vars(6:ncol(.)), ~as.Date(., format='%Y_%m_%d'))

# View Historical
kable(df.historical) %>%
  kable_styling(latex_options = c("striped"), position = "center", font_size = 10)
```

ID	Crop	Site	latitude	longitude	Start	End
1	Corn	Scandia	39.8291	-97.8458	2000-01-01	2019-12-31
2	Wheat	Belleville	39.8158	-97.6720	2000-01-01	2019-12-31
3	Soy	Ottawa	38.5398	-95.2446	2000-01-01	2019-12-31

2 RETRIEVING & PROCESSING DATA

During the next chunks of code we will retrieve and process the weather data from the above-mentioned sources.

Starting dates

If the user is interested in weather of periods PRIOR to planting, he/she can define the number of Days Prior Planting (dpp) inside each "weather.source" function. By default, dpp = 0.

Historical weather

If the user is interested in retrieving weather from multiple years at each location, there are two main options: i) define the years as "rows" of the input data table with Start and End dates as Jan 1st and Dec 31st, respectively; or ii) define the Start date of the initial year, and End date of the final year of the series.

The example here includes a separated input for historical weather (df.historical).

Extra variables

Neither of the databases provide data on reference evapotranspiration (ET$_0$). However, using DAYMET and NASA-POWER, it is possible to estimate ET$_0$ using the Hargreaves and Samani approach, which only requires temperature information (Hargreaves and Samani, 1985; Raziei and Pereira, 2013). However, the ET$_{0-HS}$ equation is reported togive unreliable estimates for daily $ET0$ and therefore it should be used for 10-day periods at the shortest (Cobaner et al., 2017).

$$\mathrm{ET}_0 = 0.0135\ k_{Rs}\ \frac{R_a}{\lambda}\sqrt{(T_{max} - T_{min})}(T_{mean} + 17.8)$$

where, ET$_0$ is the reference evapotranspiration (mm d $^{-1}$), k_{Rs} is the radiation adjustment coefficient (Hargreaves and Samani, 1982), R_a is the extraterrestrial radiation (Ra) values as suggested by Cobaner et al. (2017), and λ is the latent heat of vapourization (MJ kg $^{-1}$) for the mean temperature (T_mean).

$$R_a = \frac{24(60)}{\pi}\ Gsc\ d_r\ [\omega_s sin(\phi)sin(\delta) + cos(\phi)cos(\delta)sin(\omega_s)]$$

where, R_a is extraterrestrial radiation [MJ m^{-2}d^{-1}], Gsc is the solar constant = 0.0820 MJ m^{-2}min^{-1}, d_r is the inverse relative distance Earth–Sun , ω_s is the sunset hour angle (rad), ϕ is latitude expressed in radians, and δ is the solar declination (rad).

$$d_r = 1 + 0.033cos(\frac{2\pi}{365}DOY)$$

$$\omega_s = arccos[-tan(\phi)tan(\delta)]$$

$$\lambda = 0.409sin(\frac{2\pi}{365}DOY - 1.39)$$

```
  # Constants for ET0 (Cobaner et al., 2017)
# Solar constant
Gsc = 0.0820 # (MJ m-2 min-1)
# Radiation adjustment coefficient (Samani, 2004)
kRs = 0.17
```

2.1 DAYMET function

Here we download the daily-weather data from the DAYMET database, and we process it to obtain common variables of agronomic value.

```
# Function
weather.daymet <- function(input, dpp=0){ input %>%
  mutate(Weather = pmap(list(ID = ID,
                             lat = latitude,
                             lon = longitude,
                             sta = Start - dpp,
                             end = End),

        # Retrieving daymet data
        function(ID,lat,lon,sta,end){
                download_daymet(site = ID,
                                lat = lat, lon = lon,
                                # Extracting year
                                start = as.numeric(substr(sta,1,4)),
                                end = as.numeric(substr(end,1,4)),
                                internal = T, simplify = T)})) %>%

  mutate(Weather = Weather %>%

          map(~mutate(.,
                      Date = as.Date(as.numeric(yday)-1, # Day of the year
                                     origin = paste0(year,'-01-01')),
                      Year = year(Date),
                      Month = month(Date),
                      Day = mday(Date)))  %>%

          map(~dplyr::select(., yday, Year, Month, Day, Date,
                             measurement,value)) %>%
          map(~spread(., 'measurement','value'))  %>%
          map(~rename_all(., ~c("DOY", # Date as Day of the year
                                "Year", # Year
                                "Month", # Month
                                "Day", # Day of the month
                                "Date", # Date as normal format
                                "DL", # Day length (sec)
                                "PP", # Precipitation (mm)
                                "Rad", # Radiation (W/m2)
                                "SWE", # Snow water (kg/m2)
                                "Tmax", # Max. temp. (degC)
                                "Tmin", # Min. temp. (degC)
                                "VPD"))))  %>% # Vap Pres Def (Pa)

  mutate(Weather = pmap(list(sta=Start-dpp,
```

```r
                                    end = End,data=Weather), # Requested period
                               #~filter(..3, Date>=..1 & Date<= ..2))) %>% unnest() %>%
                               function(sta, end, data){
                                 filter(data, Date >= sta & Date <= end)
                               } )) %>% unnest(cols = c(Weather)) %>%

  # Converting units or adding variables
  mutate(Rad = Rad*0.000001*DL, # Radiation (W/m2 to MJ/m2)
         Tmean = (Tmax+Tmin)/2, # Mean temperature (degC),
         VPD = VPD / 1000, # VPD (Pa to kPa),
         # Data for ETO
         lat_rad = latitude*0.0174533,
         dr = 1 + 0.033*cos((2*pi/365)*DOY),
         Sd = 0.409*sin((2*pi/365)*DOY - 1.39),
         ws = acos(-tan(lat_rad)*tan(Sd)),
         Ra = (24*60)/(pi) * Gsc * dr * (ws*sin(lat_rad)*sin(Sd)+
                                   cos(lat_rad)*sin(ws)),
         ETO_HS = 0.0135 * kRs * (Ra / 2.45) * (sqrt(Tmax-Tmin)) * (Tmean + 17.8),
         DL = (DL/60)/60 # Day length (hours)
          ) %>% dplyr::select(-lat_rad,-dr,-Sd,-ws,-Ra)
}
```

2.1.1 Run "weather.daymet"

```r
# Specify input = dataframe containing sites-data
# Specify Days prior planting. Default is dpp = 0. Here we use dpp = 30.

weather.daymet(input = df.input, dpp = 30) -> df.weather.daymet

# Exporting data as a .csv file
write.csv(df.weather.daymet, row.names = F, na='',
          file = paste0(path, 'Output_daymet.csv'))

#View(df.weather.daymet)
```

2.2 NASA-POWER function

Here we download the daily-weather data from the NASA-POWER database, and we process it to obtain common variables of agronomic value. For the specific case of vapour pressure deficit (VPD), it is not a default weather variable reported by NASA-POWER. However, it is possible to estimate VPD (kPa) using the approach suggested by Allen et al. (1998):

$$\text{VPD (kPa)} = e_s - e_a$$

where, e_s and e_a are the saturated and actual vapour pressures.

$$e_s = 0.6108 * e^{\frac{17.27 * T_{mean}}{T_{mean} + 237.3}}$$

$$e_a = e_s * \frac{RH}{100}$$

```
weather.nasapower <- function(input, dpp=0){input %>%

  # Retrieving the data from nasapower
  mutate(Weather = pmap(list(ID = ID,
                             lat = latitude,
                             lon = longitude,
                             sta = Start - dpp,
                             end = End),
          function(ID,lat,lon,sta,end){
          get_power(community = "AG",
                    dates = c(sta,end),
                    lonlat = c(lon,lat),
                    temporal_average = "DAILY",
                    # Variables (see package documents)
                    pars = c("T2M_MIN", # Min. temp. (degC)
                             "T2M_MAX", # Max temp. (degC)
                             "RH2M", # Relative Humidity 2M
                             "PRECTOT", # Precipitation (mm)
                             "ALLSKY_SFC_SW_DWN"))} ) )  %>% # Radiation (MJ/m2)

  mutate(Weather = Weather %>%
           map(~as.data.frame(.)) %>%
           # Dates operations
           map(~mutate(., yday = lubridate::yday(YYYYMMDD),
                        Year = year(YYYYMMDD),
                        Month = month(YYYYMMDD),
                        Day = mday(YYYYMMDD)))  %>%
           map(~dplyr::select(., yday,Year, Month, Day,YYYYMMDD,
                               T2M_MIN,T2M_MAX,RH2M,
                               PRECTOT,ALLSKY_SFC_SW_DWN)) %>%

           map(~rename_all(., ~c("DOY", # Day of the Year
```

```
                                  "Year", # Year
                                  "Month", # Month
                                  "Day", # Day of the month
                                  "Date", # Date
                                  "Tmin", # Min. temp. (degC)
                                  "Tmax", # Max. temp. (degC)
                                  "RH", # Relative Humidity (%)
                                  "PP", # Precipitation (mm)
                                  "Rad")))) %>% # Radiation (MJ/m2)

  unnest(cols = c(Weather)) %>% ungroup() %>%
  # Converting units or adding variables
  mutate(Tmean = (Tmax+Tmin)/2, # Mean temp. (degC)
         # Nasapower does not provide VPD values
         # However, it is possible to estimate it with Temp and RH.
         es = 0.6108 * exp((17.27*Tmean) / (Tmean+237.3)),
         ea = es * (RH / 100),
         # vapour Pressure deficit (kPa)
         VPD = es - ea,
         # Data for ET0
         lat_rad = latitude*0.0174533,
         dr = 1 + 0.033*cos((2*pi/365)*DOY),
         Sd = 0.409*sin((2*pi/365)*DOY - 1.39),
         ws = acos(-tan(lat_rad)*tan(Sd)),
         Ra = (24*60)/(pi) * Gsc * dr * (ws*sin(lat_rad)*sin(Sd)+
                                  cos(lat_rad)*sin(ws)),
         ET0_HS = 0.0135 * kRs * (Ra / 2.45) * (sqrt(Tmax-Tmin)) * (Tmean + 17.8)
         ) %>% dplyr::select(-es,-ea,-lat_rad,-dr,-Sd,-ws,-Ra)
}
```

2.2.1 Run "weather.nasapower"

```
# Specify input = dataframe containing sites-data
# Specify Days prior planting. Default is dpp = 0. Here we use dpp = 30.

weather.nasapower(input = df.input, dpp = 30) -> df.weather.nasapower

# Exporting data as a .csv file
write.csv(df.weather.nasapower, row.names = F, na='',
          file = paste0(path, 'Output_nasapower.csv'))

#View(df.weather.nasapower)
```

2.3 CHIRPS function

Here we download the daily-weather data from the CHIRPS database, and we process it to obtain common variables of agronomic value.

```
weather.chirps <- function(input, dpp=0){ input %>%

  # Retrieving the data from CHIRPS
  mutate(Weather = pmap(list(ID = ID,
                             lat = latitude,
                             lon = longitude,
                             sta = Start - dpp,
                             end = End),
                        function(ID,lat,lon,sta,end){
      get_chirps(data.frame(lon = c(lon), lat = c(lat)),
                 c(as.character(sta),as.character(end)))}) ,
  # Get prec. indices
      Indices = Weather %>% map(~precip_indices(., timeseries = TRUE,
                                               intervals = 30))) %>%

  # Organizing dataframe
  mutate(Weather = Weather %>%
         map(~as.data.frame(.)) %>%
         map(~dplyr::select(., date,chirps)) %>%
         # Dates operations
         map(~mutate(., yday = lubridate::yday(date),
                     Year = year(date),
                     Month = month(date),
                     Day = mday(date))) %>%
         map(~dplyr::select(., yday, Year, Month, Day, date,chirps)) %>%
         map(~rename_all(., ~c("DOY", "Year", "Month", "Day","Date", "PP"))),

        Indices = Indices %>%
         map(~as.data.frame(.)) %>%
         map(~spread(., 'index','value')) %>%
         map(~dplyr::select(., -id,-lon,-lat)) %>%
         map(~rename(., Date = date))) %>%

  mutate(Full = map2(.x=Weather, .y=Indices, ~left_join(.x,.y))) %>%
  dplyr::select(-Weather, -Indices) %>% unnest(cols = c(Full))

}
```

2.3.1 Run "weather.chirps"

```r
# Specify input = dataframe containing sites-data
# Specify Days prior planting. Default is dpp = 0. Here we use dpp = 30.

weather.chirps(input = df.input, dpp = 30) -> df.weather.chirps

# Exporting data as a .csv file
write.csv(df.weather.chirps, row.names = F, na='',
          file = paste0(path, 'Output_chirps.csv'))

#View(df.weather.chirps)
```

3 TIME INTERVALS

In this section we create time intervals during the cropping season using pre-specified dates as columns at the initial data table with site information. The user can apply: i) a unique seasonal interval (season), ii) even intervals (even), or iii) customized intervales (custom).

3.1 FULL SEASON interval

```
season <- df.input %>%
  mutate(Intervals = # Create
           map2(.x=Start,.y=End,
                ~data.frame( # New data
                  Interval = c("Season"),
                  Start.in = c(.x),
                  End.in = c(.y) ) )) %>%

  dplyr::select(ID,Site,Intervals) %>% unnest(cols = c(Intervals))

kable(season) %>%
  kable_styling(latex_options = c("striped"), position = "center", font_size = 10)
```

ID	Site	Interval	Start.in	End.in
1	Scandia	Season	2002-04-25	2002-09-30
2	Belleville	Season	2005-10-15	2006-06-20
3	Ottawa	Season	2010-05-20	2010-10-10

3.2 EVEN intervals

```
n = 4 # Number of intervals
dpp = 30 # Number of days prior planting, if needed.

even <- df.input %>%
  mutate(Intervals = # Create
       map2(.x=Start,.y=End,
       ~data.frame( # New data
       Interval = c("Prev",LETTERS[1:n+1]),
       Start.in = c(.x-dpp, seq.Date(.x,.y+1,length.out=n+1)[1:n] ),
       End.in = c(.x-1,seq.Date(.x,.y+1,length.out=n+1)[2:(n+1)]))) ) %>%

  dplyr::select(ID,Site,Intervals) %>% unnest(cols = c(Intervals))

kable(even) %>%
  kable_styling(latex_options = c("striped"), position = "center", font_size = 10)
```

ID	Site	Interval	Start.in	End.in
1	Scandia	Prev	2002-03-26	2002-04-24
1	Scandia	B	2002-04-25	2002-06-03
1	Scandia	C	2002-06-03	2002-07-13
1	Scandia	D	2002-07-13	2002-08-22
1	Scandia	E	2002-08-22	2002-10-01
2	Belleville	Prev	2005-09-15	2005-10-14
2	Belleville	B	2005-10-15	2005-12-16
2	Belleville	C	2005-12-16	2006-02-16
2	Belleville	D	2006-02-16	2006-04-19
2	Belleville	E	2006-04-19	2006-06-21
3	Ottawa	Prev	2010-04-20	2010-05-19
3	Ottawa	B	2010-05-20	2010-06-25
3	Ottawa	C	2010-06-25	2010-07-31
3	Ottawa	D	2010-07-31	2010-09-05
3	Ottawa	E	2010-09-05	2010-10-11

3.3 CUSTOM intervals

```
# Counting # intervals
i = ncol(df.input[,6:ncol(df.input)]) # Number of intervals

#
df.input = df.input %>%
  # Reformat Reference dates for operations
  # Modify names and Number of dates as needed
  # Here we follow the example of df.input
  # with 4 dates named as Start (Plant), Flo, SeFi, & End
  mutate_at(vars(6:ncol(.)),
            ~str_replace_all(as.character(.), '-','_')) %>%
  mutate_at(vars(6:ncol(.)), ~as.Date(., format='%Y_%m_%d')) %>% data.frame()

custom <- df.input %>%
  mutate(Intervals = # Create
           pmap(list(x = Start - dpp,
                     y = Start,
                     z = Flo,
                     m = SeFi,
                     k = End),
                function(x,y,z,m,k){
                  data.frame( # New data
                  Interval = c(LETTERS[1:i]),
                  Name = c("Prev", "Plant-Flo","Flo-SeFi","SeFi-End"),
                  Start.in = c(x,y,z,m),
                  End.in = c(y-1,z-1,m-1,k) ) } ) ) %>%

  dplyr::select(ID,Site,Intervals) %>% unnest(cols = c(Intervals))
```

```r
kable(custom) %>%
  kable_styling(latex_options = c("striped"), position = "center", font_size = 10)
```

ID	Site	Interval	Name	Start.in	End.in
1	Scandia	A	Prev	2002-03-26	2002-04-24
1	Scandia	B	Plant-Flo	2002-04-25	2002-07-14
1	Scandia	C	Flo-SeFi	2002-07-15	2002-08-14
1	Scandia	D	SeFi-End	2002-08-15	2002-09-30
2	Belleville	A	Prev	2005-09-15	2005-10-14
2	Belleville	B	Plant-Flo	2005-10-15	2006-04-14
2	Belleville	C	Flo-SeFi	2006-04-15	2006-04-30
2	Belleville	D	SeFi-End	2006-05-01	2006-06-20
3	Ottawa	A	Prev	2010-04-20	2010-05-19
3	Ottawa	B	Plant-Flo	2010-05-20	2010-07-04
3	Ottawa	C	Flo-SeFi	2010-07-05	2010-08-14
3	Ottawa	D	SeFi-End	2010-08-15	2010-10-10

4 SEASONAL SUMMARIES

For each of the period or interval of interest a variety of variables can be created. Here, we present a set of variables that can capture environmental variations that might be missing by analyzing standard weather data (precipitations, temperature, radiation). These variables represent an example that was used for studying influence of weather in corn yields by Correndo et al. (2021).

Table 2. Secondary weather variables that summarize specified time intervals during the cropping-season.

Variable	Units	Description	Reference
SDI	-	*Shannon Diversity Index*. Measures distribution during a period of time. 0 = complete unevenness (all rain in 1 day), 1 = complete evenness (equal rain each day of the period).	Tremblay et al. (2012)
AWDR	mm	*Abundant and Well-Distributed Water*. Proxy of rainfall effectiveness. Weighs PP*SDI.	Tremblay et al. (2012)
EPE	No.	*Extreme Precipitation Events*. Number of days with precipitation > 25 mm (~1"). Proxy of excessive rainfall ocurrence	Puntel et al. (2019); Correndo et al. (2021)
ETE	No.	*Extreme Temperature Events*. Number of days with Tmax > a certain thershold (e.g., 30°C). Proxy of heat stress risk	Butler and Huybers (2013); Ye et al. (2017)
CHU	°C	*Accumulated Crop Heat Units*. Thermal time. Assumes no crop growth if day temperature (Tmax) < 10°C, and night temperature (Tmin) < 4.4°C. Crop development halts with Tmax > 30°C. Developed for corn	Bootsma et al. (2005)
GDD	°C	*Accumulated Growing Degree Days*. Thermal time over base temperature. Assumes constant growth with Tmin < 10°C and Tmax >30. Tmin and Tmax thresholds depends on the crop.	Kumudini et al. (2014)
Q	MJ m^{-2} $°C^{-1}$	*Photothermal quotient*. Availability of radiation per unit of thermal time (CHU or GDD)	Bannayan et al. (2004)

4.1 Summary function - DAYMET & NASA-POWER

```
# Defining the function to summarize DAYMET and/or NASA-POWER
summary.daymet.nasapower <- function(input, intervals) {

  intervals %>%

  # Merging weather data
  left_join(input %>%
              # Nesting weather data back for each site-ID
  dplyr::select_if(
    names(.) %in% c("ID", "Crop", "Site", "Date","DL", "PP",
                    "Rad", "Tmax", "Tmin", "Tmean", "VPD", "ETO_HS")) %>%
  group_by(ID,Crop,Site) %>% nest(.key = 'Weather') %>% ungroup()) %>%

  mutate(Weather = pmap(list(x = Start.in,y = End.in, data = Weather),
                        function(x, y, data){
                          filter(data, Date >= x & Date < y)} ) )  %>%

  mutate(Weather = Weather %>% # User must addapt depending on the crop
           map(~mutate(.,
                 # Ext. Prec. event
                 EPEi = case_when(PP>25~1, TRUE~0),
                 # Ext.Temp. event
                 ETEi = case_when(Tmax >= 30~1, TRUE~0),
                 # Tmax factor,  crop heat units (CHU)
                 Ymax = case_when(Tmax < 10~0,
                            TRUE ~ 3.33*(Tmax-10)-0.084*(Tmax-10)),
                 # Tmin factor, Crop heat units (CHU)
                 Ymin = case_when(Tmin<4.44~0,
                            TRUE ~ 1.8*(Tmin-4.44)),
                 # Daily CHU
                 Yavg = (Ymax+Ymin)/2,
                 # Tmin threshold Growing Degrees.
                 Gmin = case_when(Tmin >= 10 ~ Tmin,
                            TRUE ~ 10),
                 # Tmax threshold Growing Degrees.
                 Gmax = case_when(Tmax <= 30 ~ Tmax,
                            TRUE ~ 30),
                 # Daily Growing Degree Units.
                 GDU = ((Gmin + Gmax)/2) - 10
                     ) ) ) %>%

# Summary for each variable

  mutate(# Duration of interval (days)
       Dur = Weather %>% map(~nrow(.)),
```

```r
        # Accumulated PP (mm)
        PP = Weather %>% map(~sum(.$PP)),
        # Mean Temp (C)
        Tmean = Weather %>% map(~mean(.$Tmean)),
        # Accumulated Rad (MJ/m2)
        Rad = Weather %>% map(~sum(.$Rad)),
        # Accumulated VPD (kPa)
        VPD = Weather %>% map(~sum(.$VPD)),
        # Accumulated ETO (mm)
        ET0_HS = Weather %>% map(~sum(.$ET0_HS)),
        # Number of ETE (#)
        ETE = Weather %>% map(~sum(.$ETEi)),
        # Number of EPE (#)
        EPE = Weather %>% map(~sum(.$EPEi)),
        # Accumulated Crop Heat Units (CHU)
        CHU = Weather %>% map(~sum(.$Yavg)),
        # Shannon Diversity Index for PP
        SDI = Weather %>% map(~diversity(.$PP, index="shannon")/
                                log(length(.$PP))),
        # Accumulated Growing Degree Days (GDD)
        GDD =  Weather %>% map(~sum(.$GDU))) %>%

# Additional indices and final units
  dplyr::select(-Weather) %>% unnest() %>%
  mutate(# Photo-thermal quotient (Q)
        Q_chu = Rad/CHU,
        Q_gdd = Rad/GDD,
        # Abundant and Well Distributed Water
        AWDR = PP*SDI)
}
```

4.2 Summary function - CHIRPS.

```
#########

# Defining function to summarize CHIRPS data
summary.chirps <- function(input, intervals) {

intervals %>%

  # Merging weather data
  left_join(input %>%
              # Nesting weather data back for each site-ID
  dplyr::select(c(ID, Crop, Site, Date, PP)) %>%
  group_by(ID,Crop,Site) %>% nest(.key = 'Weather') %>% ungroup()) %>%

  mutate(Weather = pmap(list(x = Start.in,y = End.in, data = Weather),
                      function(x, y, data){
                        filter(data, Date >= x & Date < y)} ) )  %>%

  mutate(Weather = Weather %>% # User must addapt depending on the crop
           map(~mutate(., EPEi = case_when(PP>25~1, TRUE~0) # Ext. Prec. event
                      ) ) )   %>%

# Summary for each variable

  mutate(# Duration of interval (days)
         Dur = Weather %>% map(~nrow(.)),
         # Accumulated PP (mm)
         PP = Weather %>% map(~sum(.$PP)),
         # Number of EPE (#)
         EPE = Weather %>% map(~sum(.$EPEi)) ,
         # Shannon Diversity Index for precipitation data
         SDI = Weather %>% map(~diversity(.$PP, index="shannon")/
                                 log(length(.$PP)))) %>%

# Additional indices and final units
  dplyr::select(-Weather) %>% unnest() %>%
  mutate(AWDR = PP*SDI) # Abundant and Well Distributed Water

}
```

4.3 DAYMET summary

```
# Run the summary
# input = dataframe containing the data (from daymet or nasapower).
# intervals = type of intervals (season, custom or even)
df.summary.daymet <-
  summary.daymet.nasapower(input = df.weather.daymet,
                           intervals = custom)

kbl(df.summary.daymet) %>%
  kable_styling(font_size = 7, position = "center", latex_options = c("scale_down"))
```

ID	Site	Interval	Name	Start.in	End.in	Crop	Dur	PP	Tmean	Rad	VPD	ET0_HS	ETE	EPE	CHU	SDI	GDD	Q_chu	Q_gdd	AWDR
1	Scandia	A	Prev	2002-03-26	2002-04-24	Corn	29	38.20	11.685000	553.6734	13.60862	108.00850	5	0	517.2172	0.4800688	145.620	1.0704853	3.802180	18.338628
1	Scandia	B	Plant-Flo	2002-04-25	2002-07-14	Corn	80	198.69	20.843250	1683.1232	127.84635	453.11234	37	1	2996.6882	0.5559818	852.210	0.5616611	1.975010	110.468026
1	Scandia	C	Flo-SeFi	2002-07-15	2002-08-14	Corn	30	52.26	28.066667	609.8361	69.27578	205.86950	28	0	1686.1824	0.4436756	452.625	0.3616668	1.347332	23.186485
1	Scandia	D	SeFi-End	2002-08-15	2002-09-30	Corn	46	64.35	22.307391	760.6737	74.80019	209.14710	24	1	1896.7187	0.3467073	536.965	0.4010472	1.416617	22.310618
2	Belleville	A	Prev	2005-09-15	2005-10-14	Wheat	29	89.20	18.093448	420.3018	39.98581	99.43622	9	1	905.3896	0.0473277	252.420	0.4642220	1.665089	4.221631
2	Belleville	B	Plant-Flo	2005-10-15	2006-04-14	Wheat	181	148.46	4.269309	1842.2036	93.24580	312.03325	2	1	1293.4542	0.5072027	126.775	1.4242511	14.531285	75.299308
2	Belleville	C	Flo-SeFi	2006-04-15	2006-04-30	Wheat	15	40.26	14.125333	284.2412	11.47954	63.41565	1	0	328.2973	0.4582681	91.560	0.8658043	3.104425	18.449873
2	Belleville	D	SeFi-End	2006-05-01	2006-06-20	Wheat	50	107.63	20.361300	1068.3480	74.36166	287.09083	19	0	1817.9448	0.5452137	508.210	0.5876680	2.102178	58.681355
3	Ottawa	A	Prev	2010-04-20	2010-05-19	Soy	29	196.54	14.749483	488.6842	35.02715	109.20673	0	3	599.4266	0.5518063	153.545	0.8152528	3.182677	108.452003
3	Ottawa	B	Plant-Flo	2010-05-20	2010-07-04	Soy	45	175.50	24.475333	941.0043	99.79490	255.56061	21	4	2047.2903	0.5132675	628.560	0.4596340	1.497079	90.078442
3	Ottawa	C	Flo-SeFi	2010-07-05	2010-08-14	Soy	40	172.16	27.600500	761.1400	106.85095	231.35308	33	2	2135.6287	0.5709079	635.065	0.3564009	1.198523	98.287501
3	Ottawa	D	SeFi-End	2010-08-15	2010-10-10	Soy	56	190.23	21.037321	885.1340	95.86908	230.17417	18	2	2113.2837	0.5153631	613.730	0.4188430	1.442221	98.037530

```
# Exporting data as a .csv file
# Daymet
write.csv(df.summary.daymet, row.names = F, na='',
          file = paste0(path, 'Summary_daymet.csv'))
```

4.4 NASA-POWER summary

```
# Run the summary
# data = dataframe containing the data (from daymet or nasapower).
# intervals = type of intervals (season, custom or even)
df.summary.nasapower <-
  summary.daymet.nasapower(input = df.weather.nasapower,
                           intervals = custom)

kbl(df.summary.nasapower) %>%
  kable_styling(font_size = 7, position = "center", latex_options = c("scale_down"))
```

ID	Site	Interval	Name	Start.in	End.in	Crop	Dur	PP	Tmean	Rad	VPD	ET0_HS	ETE	EPE	CHU	SDI	GDD	Q_chu	Q_gdd	AWDR
1	Scandia	A	Prev	2002-03-26	2002-04-24	Corn	29	40.41	12.000862	539.31	19.36580	108.10773	3	0	526.4603	0.6752988	145.805	1.0244078	3.698844	27.28882
1	Scandia	B	Plant-Flo	2002-04-25	2002-07-14	Corn	80	191.12	21.604625	1759.25	99.02537	463.93380	39	1	3147.8183	0.7616254	891.975	0.5588791	1.972309	145.56185
1	Scandia	C	Flo-SeFi	2002-07-15	2002-08-14	Corn	30	61.90	29.319000	719.36	71.80175	207.10434	29	0	1774.0018	0.6549487	476.255	0.4055013	1.510451	40.54133
1	Scandia	D	SeFi-End	2002-08-15	2002-09-30	Corn	46	85.40	23.086739	825.83	64.10093	207.25658	25	1	1973.7204	0.6707515	558.635	0.4184129	1.478300	57.28218
2	Belleville	A	Prev	2005-09-15	2005-10-14	Wheat	29	34.16	18.752241	450.42	26.32645	97.08391	10	0	931.8973	0.3925158	264.475	0.4833365	1.703072	13.40834
2	Belleville	B	Plant-Flo	2005-10-15	2006-04-14	Wheat	181	170.29	4.927541	2019.34	65.75696	302.54437	1	1	1228.8620	0.5970578	98.965	1.6432602	20.404588	101.67297
2	Belleville	C	Flo-SeFi	2006-04-15	2006-04-30	Wheat	15	38.10	15.024000	310.06	11.95981	64.26104	1	0	351.3449	0.5179927	97.155	0.8824946	3.191395	19.73552
2	Belleville	D	SeFi-End	2006-05-01	2006-06-20	Wheat	50	134.11	21.290300	1165.59	62.38361	288.52675	24	0	1926.2308	0.7085698	544.155	0.6051144	2.142018	95.02629
3	Ottawa	A	Prev	2010-04-20	2010-05-19	Soy	29	173.57	14.777586	522.60	12.35072	114.60773	0	3	610.7046	0.6584440	158.700	0.8557328	3.293006	114.28613
3	Ottawa	B	Plant-Flo	2010-05-20	2010-07-04	Soy	45	208.47	23.952000	1049.16	26.93406	239.63254	14	3	1970.7530	0.6811555	619.515	0.5323650	1.693518	142.00049
3	Ottawa	C	Flo-SeFi	2010-07-05	2010-08-14	Soy	40	169.12	26.962250	938.65	30.34546	209.15967	30	2	2046.0128	0.6828772	638.375	0.4587704	1.470374	115.48819
3	Ottawa	D	SeFi-End	2010-08-15	2010-10-10	Soy	56	219.47	21.315446	NA	42.69979	223.86218	17	1	2127.6913	0.6374600	629.340	NA	NA	139.90334

```
# Exporting data as a .csv file
write.csv(df.summary.nasapower, row.names = F, na='',
          file = paste0(path, 'Summary_nasapower.csv'))
```

4.5 CHIRPS summary

```
# Run the summary
# data = dataframe containing the data.
# intervals = type of intervals (season, custom or even)

df.summary.chirps <-
  summary.chirps(input = df.weather.chirps,
                 intervals = custom)

kbl(df.summary.chirps) %>%
  kable_styling(font_size = 7, position = "center", latex_options = c("scale_down"))
```

ID	Site	Interval	Name	Start.in	End.in	Crop	Dur	PP	EPE	SDI	AWDR
1	Scandia	A	Prev	2002-03-26	2002-04-24	Corn	29	34.66944	0	0.3114334	10.797223
1	Scandia	B	Plant-Flo	2002-04-25	2002-07-14	Corn	80	195.22630	1	0.6073623	118.573094
1	Scandia	C	Flo-SeFi	2002-07-15	2002-08-14	Corn	30	72.27709	0	0.6120070	44.234083
1	Scandia	D	SeFi-End	2002-08-15	2002-09-30	Corn	46	78.97220	1	0.3832844	30.268813
2	Belleville	A	Prev	2005-09-15	2005-10-14	Wheat	29	21.30135	0	0.3096348	6.595638
2	Belleville	B	Plant-Flo	2005-10-15	2006-04-14	Wheat	181	188.89972	1	0.5865065	110.790913
2	Belleville	C	Flo-SeFi	2006-04-15	2006-04-30	Wheat	15	31.88896	0	0.3843310	12.255917
2	Belleville	D	SeFi-End	2006-05-01	2006-06-20	Wheat	50	125.31568	0	0.7156719	89.684915
3	Ottawa	A	Prev	2010-04-20	2010-05-19	Soy	29	159.14508	3	0.5866904	93.368889
3	Ottawa	B	Plant-Flo	2010-05-20	2010-07-04	Soy	45	198.73473	1	0.6000652	119.253787
3	Ottawa	C	Flo-SeFi	2010-07-05	2010-08-14	Soy	40	164.25588	2	0.3220948	52.905962
3	Ottawa	D	SeFi-End	2010-08-15	2010-10-10	Soy	56	202.29837	3	0.5145636	104.095369

```
# Exporting data as a .csv file
write.csv(df.summary.chirps, row.names = F, na='',
          file = paste0(path, 'Summary_chirps.csv'))
```

5 HISTORICAL WEATHER

5.1 Historical "weather.daymet"

For retrieving historical weather, user must specify the input containing the historical target dates by site.

```
# Specify input = dataframe containing historical dates from sites

weather.daymet(input = df.historical) -> hist.weather.daymet

# Exporting data as a .csv file
write.csv(hist.weather.daymet, row.names = F, na='',
          file = paste0(path, 'Hist_output_daymet.csv'))

#View(hist.weather.daymet)
```

5.2 Historical "weather.nasapower"

```
# Specify input = dataframe containing historical dates from sites

weather.nasapower(input = df.historical) -> hist.weather.nasapower

# Exporting data as a .csv file
write.csv(hist.weather.nasapower, row.names = F, na='',
          file = paste0(path, 'Hist_output_nasapower.csv'))

#View(hist.weather.nasapower)
```

5.3 Historical "weather.chirps"

```
# Specify input = dataframe containing historical dates from sites

weather.chirps(input = df.historical) -> hist.weather.chirps

# Exporting data as a .csv file
write.csv(hist.weather.chirps, row.names = F, na='',
          file = paste0(path, 'Hist_output_chirps.csv'))

#View(hist.weather.chirps)
```

5.4 Intervals functions

```
historical.years <- function(hist.data) {

# By year
hist.data %>% group_by(ID,Year) %>%
  dplyr::select(ID, Crop, Site, latitude, longitude, Start, End, Date, Year, Month) %>%
  summarise(Start.in = min(Date),
            End.in = max(Date))

}

historical.years.months <- function(hist.data) {

# By year*month
hist.data %>% group_by(ID, Year, Month) %>%
  dplyr::select(ID, Crop, Site, latitude, longitude, Start, End, Date, Year, Month) %>%
  summarise(Start.in = min(Date),
            End.in = max(Date))

}
```

5.5 DAYMET Historical summary

Summary can be obtained by years or by years.months. User must specify this option at the "intervals" argument of the summary function.

5.5.1 Intervals

```
# Specify hist.data = dataframe containing the historical weather data to summarize

years = historical.years(hist.data = hist.weather.daymet)

# Specify hist.data = dataframe containing the historical weather data to summarize
years.months = historical.years.months(hist.data = hist.weather.daymet)
```

5.5.2 Summary

```
# input = dataframe containing the historical weather data.
# intervals = type of historical intervals (years, years.months)

historical.summary.daymet <-
  summary.daymet.nasapower(input = hist.weather.daymet,
```

```
                            intervals = years)

kbl(historical.summary.daymet) %>%
  kable_styling(font_size = 7, position = "center", latex_options = c("scale_down"))
```

ID	Year	Start.in	End.in	Crop	Site	Dur	PP	Tmean	Rad	VPD	ET0_HS	ETE	EPE	CHU	SDI	GDD	Q_chu	Q_gdd	AWDR
1	2000	2000-01-01	2000-12-30	Corn	Scandia	364	430.87	12.50508	5476.197	355.5420	1278.666	95	2	8587.692	0.6599856	2009.660	0.6376797	2.724937	284.3680
1	2001	2001-01-01	2001-12-31	Corn	Scandia	364	896.24	12.48468	5312.814	417.7785	1211.823	77	11	8171.777	0.6654668	1981.870	0.6501418	2.680707	596.4180
1	2002	2002-01-01	2002-12-31	Corn	Scandia	364	570.49	12.30462	5397.851	369.6487	1250.200	98	4	8096.128	0.6210552	2004.585	0.6667201	2.692753	354.3058
1	2003	2003-01-01	2003-12-31	Corn	Scandia	364	834.22	11.85797	5336.451	379.1205	1228.314	77	6	7838.901	0.5744805	1842.680	0.6807653	2.896027	479.2432
1	2004	2004-01-01	2004-12-30	Corn	Scandia	364	775.67	11.81563	5263.105	390.2710	1183.946	64	10	7699.707	0.6703507	1865.610	0.6835461	2.821117	519.9709
1	2005	2005-01-01	2005-12-31	Corn	Scandia	364	810.82	12.32692	5281.648	406.8515	1236.839	88	8	8295.397	0.6266284	2010.835	0.6366962	2.626594	508.0828
1	2006	2006-01-01	2006-12-31	Corn	Scandia	364	742.45	12.75533	5242.722	384.0918	1276.253	84	6	8187.467	0.6442724	2130.425	0.6403349	2.460881	478.3400
1	2007	2007-01-01	2007-12-31	Corn	Scandia	364	864.72	11.88191	5353.206	406.3426	1179.145	68	8	8120.108	0.6634598	1913.810	0.6592530	2.797146	573.7069
1	2008	2008-01-01	2008-12-30	Corn	Scandia	364	975.92	10.50554	5320.336	359.3900	1138.635	49	11	7038.313	0.6585597	1625.125	0.7559106	3.273801	642.7016
1	2009	2009-01-01	2009-12-31	Corn	Scandia	364	616.82	10.57141	5377.617	321.2247	1161.106	49	6	7035.696	0.6650616	1647.880	0.7643333	3.263355	410.2233
1	2010	2010-01-01	2010-12-31	Corn	Scandia	364	761.74	11.63665	5197.417	398.5268	1182.199	77	6	7908.053	0.6413833	1819.250	0.6572310	2.856901	488.5673
1	2011	2011-01-01	2011-12-31	Corn	Scandia	364	757.30	11.37154	5355.147	369.9904	1206.832	77	7	7805.566	0.6471593	1855.165	0.6860678	2.886615	490.0938
1	2012	2012-01-01	2012-12-30	Corn	Scandia	364	615.21	13.32372	5396.663	392.3879	1348.612	96	6	8846.515	0.5787965	2230.335	0.6100326	2.419665	356.0814
1	2013	2013-01-01	2013-12-31	Corn	Scandia	364	696.99	10.80997	5318.018	369.4580	1165.633	66	7	7461.667	0.6319254	1721.225	0.7127118	3.089671	440.4457
1	2014	2014-01-01	2014-12-31	Corn	Scandia	364	648.50	10.81908	5377.303	333.7607	1194.459	67	5	7605.359	0.6425404	1750.420	0.7070413	3.072007	416.6875
1	2015	2015-01-01	2015-12-31	Corn	Scandia	364	883.43	12.39194	5271.458	392.4868	1207.448	64	9	8186.675	0.6345908	2044.665	0.6439070	2.578152	560.6165
1	2016	2016-01-01	2016-12-30	Corn	Scandia	364	869.96	12.95982	5244.471	404.9502	1232.522	70	7	8447.385	0.6809664	2163.705	0.6208396	2.423838	592.4135
1	2017	2017-01-01	2017-12-31	Corn	Scandia	364	647.44	12.55620	5250.261	383.7066	1232.920	69	8	8053.745	0.6286623	2037.820	0.6519030	2.576410	407.0211
1	2018	2018-01-01	2018-12-31	Corn	Scandia	364	947.52	11.12897	5181.924	366.9640	1174.721	81	9	7621.219	0.6609851	1806.035	0.6799338	2.869227	626.2966
1	2019	2019-01-01	2019-12-31	Corn	Scandia	364	999.57	10.98386	5205.186	397.8354	1120.105	66	10	7437.354	0.6902476	1721.300	0.6998706	3.023985	689.9508
2	2000	2000-01-01	2000-12-30	Wheat	Belleville	364	496.87	12.44040	5395.768	383.8191	1266.803	94	2	8518.275	0.6697404	1993.335	0.6334344	2.706905	332.7739
2	2001	2001-01-01	2001-12-31	Wheat	Belleville	364	938.67	12.42883	5306.008	420.6572	1201.169	74	10	8098.520	0.6772810	1959.135	0.6551824	2.708342	635.7434
2	2002	2002-01-01	2002-12-31	Wheat	Belleville	364	571.91	12.30687	5381.252	373.4799	1240.406	97	5	8047.798	0.6311478	1991.460	0.6686615	2.702164	360.9597
2	2003	2003-01-01	2003-12-31	Wheat	Belleville	364	844.66	11.80463	5330.413	382.1917	1217.745	75	5	7773.442	0.5868726	1816.550	0.6857211	2.934361	495.7078
2	2004	2004-01-01	2004-12-30	Wheat	Belleville	364	756.70	11.75357	5272.054	394.3482	1165.406	58	7	7597.293	0.6663236	1834.250	0.6939385	2.874229	504.2070
2	2005	2005-01-01	2005-12-31	Wheat	Belleville	364	877.89	12.36390	5291.557	414.6776	1217.633	84	10	8231.897	0.6214964	2008.065	0.6428114	2.635152	545.6055
2	2006	2006-01-01	2006-12-31	Wheat	Belleville	364	786.66	12.84018	5252.586	391.3249	1261.571	80	5	8135.789	0.6482459	2127.815	0.6456149	2.468535	509.9491
2	2007	2007-01-01	2007-12-31	Wheat	Belleville	364	1075.54	11.97801	5332.735	416.6738	1168.714	68	13	8128.007	0.6566895	1916.125	0.6560939	2.783083	706.2958
2	2008	2008-01-01	2008-12-30	Wheat	Belleville	364	1159.68	10.56824	5321.420	370.7535	1129.700	47	17	7022.260	0.6491877	1624.045	0.7577930	3.276645	752.8500
2	2009	2009-01-01	2009-12-31	Wheat	Belleville	364	683.27	10.55257	5359.954	332.3693	1147.734	47	7	6972.738	0.6639736	1624.345	0.7687015	3.299764	453.6732
2	2010	2010-01-01	2010-12-31	Wheat	Belleville	364	809.58	11.59067	5206.913	402.7480	1164.938	75	10	7854.444	0.6327294	1792.675	0.6629256	2.904549	512.2451
2	2011	2011-01-01	2011-12-31	Wheat	Belleville	364	764.92	11.38503	5360.560	372.3768	1195.423	73	9	7757.097	0.6373220	1839.100	0.6910524	2.914774	487.5003
2	2012	2012-01-01	2012-12-30	Wheat	Belleville	364	618.42	13.29808	5402.416	393.5945	1332.681	94	9	8774.278	0.5824671	2212.265	0.6157106	2.442029	360.2093
2	2013	2013-01-01	2013-12-31	Wheat	Belleville	364	683.12	10.71444	5303.381	368.9465	1149.930	64	8	7363.757	0.6174274	1682.490	0.7202004	3.152102	421.7770
2	2014	2014-01-01	2014-12-31	Wheat	Belleville	364	599.45	10.78378	5377.217	328.3103	1182.196	67	4	7550.992	0.6367361	1721.990	0.7121206	3.122676	381.6915
2	2015	2015-01-01	2015-12-31	Wheat	Belleville	364	815.90	12.35922	5274.701	396.7896	1193.184	63	9	8122.617	0.6438683	2022.920	0.6493844	2.607469	525.3322
2	2016	2016-01-01	2016-12-30	Wheat	Belleville	364	893.22	12.97978	5237.432	410.6357	1223.018	71	9	8442.207	0.6831610	2153.700	0.6203866	2.431830	610.2131
2	2017	2017-01-01	2017-12-31	Wheat	Belleville	364	730.40	12.49541	5249.386	389.9764	1222.022	67	8	7971.558	0.6233700	2004.655	0.6585145	2.618598	455.3095
2	2018	2018-01-01	2018-12-31	Wheat	Belleville	364	944.67	11.11979	5198.976	372.0482	1168.097	81	12	7601.345	0.6634684	1790.430	0.6839547	2.903758	626.7587
2	2019	2019-01-01	2019-12-31	Wheat	Belleville	364	1033.08	10.98037	5199.047	401.8592	1110.761	65	10	7432.313	0.6891426	1713.190	0.6995194	3.034717	711.9394
3	2000	2000-01-01	2000-12-30	Soy	Ottawa	364	689.08	13.69025	5322.620	457.6826	1243.000	88	3	8962.132	0.6686865	2253.045	0.5939011	2.362412	460.7785
3	2001	2001-01-01	2001-12-31	Soy	Ottawa	364	923.58	13.88330	5272.813	469.0804	1209.489	75	11	8703.382	0.6663936	2307.130	0.6058350	2.285443	615.4678
3	2002	2002-01-01	2002-12-31	Soy	Ottawa	364	782.77	13.44680	5268.540	446.8951	1209.372	88	7	8345.598	0.6622767	2236.325	0.6312957	2.355892	518.4103
3	2003	2003-01-01	2003-12-31	Soy	Ottawa	364	872.94	13.09783	5317.561	435.8266	1212.401	62	11	8282.616	0.6463907	2075.825	0.6420146	2.561661	564.2603
3	2004	2004-01-01	2004-12-30	Soy	Ottawa	364	1189.19	13.01896	5203.824	447.5230	1131.363	38	13	7889.195	0.7058715	2075.030	0.6596141	2.507831	839.4154
3	2005	2005-01-01	2005-12-31	Soy	Ottawa	364	1145.15	13.47507	5254.697	454.8703	1179.751	72	15	8428.881	0.6577909	2226.010	0.6234157	2.360590	753.2692
3	2006	2006-01-01	2006-12-31	Soy	Ottawa	364	802.63	14.12995	5272.789	447.8425	1246.283	73	7	8639.430	0.6693564	2331.005	0.6103167	2.262024	537.2455
3	2007	2007-01-01	2007-12-31	Soy	Ottawa	364	1299.49	13.34286	5208.913	473.9347	1159.490	71	13	8720.902	0.6615834	2236.920	0.5972906	2.328610	859.7210
3	2008	2008-01-01	2008-12-30	Soy	Ottawa	364	1195.79	11.91095	5259.767	425.6149	1113.511	41	18	7481.205	0.6927486	1883.330	0.7030641	2.792802	828.3819
3	2009	2009-01-01	2009-12-31	Soy	Ottawa	364	1342.50	12.20199	5069.419	426.8225	1099.185	38	14	7403.735	0.6988706	1882.710	0.6847110	2.692618	938.2338
3	2010	2010-01-01	2010-12-31	Soy	Ottawa	364	1093.09	12.95784	5165.664	461.7282	1160.906	73	16	8495.064	0.6737458	2091.860	0.6080783	2.469412	736.4648
3	2011	2011-01-01	2011-12-31	Soy	Ottawa	364	928.67	13.05956	5217.933	455.9065	1220.183	88	9	8570.147	0.6926861	2114.930	0.6088498	2.467189	643.2768
3	2012	2012-01-01	2012-12-30	Soy	Ottawa	364	631.82	15.11397	5316.917	442.3282	1329.461	89	3	9582.677	0.6370328	2548.240	0.5548467	2.086505	402.4900
3	2013	2013-01-01	2013-12-31	Soy	Ottawa	364	983.28	11.94393	5210.946	425.3260	1127.139	70	11	7603.615	0.6806146	1863.990	0.6853247	2.795587	669.2348
3	2014	2014-01-01	2014-12-31	Soy	Ottawa	364	834.77	11.89621	5249.363	405.0908	1157.040	56	8	7862.848	0.6666827	1917.070	0.6676159	2.738222	556.5267
3	2015	2015-01-01	2015-12-31	Soy	Ottawa	364	1162.72	13.54743	5151.945	453.6231	1171.251	56	15	8568.485	0.6854885	2266.005	0.6012668	2.273581	797.0311
3	2016	2016-01-01	2016-12-30	Soy	Ottawa	364	1056.19	14.21709	5132.503	465.2574	1208.928	75	11	8973.415	0.6870032	2422.285	0.5719677	2.118868	725.6059
3	2017	2017-01-01	2017-12-31	Soy	Ottawa	364	1182.35	13.87076	5160.887	438.5015	1203.262	52	13	8466.807	0.6535312	2299.905	0.6095435	2.243957	772.7026
3	2018	2018-01-01	2018-12-31	Soy	Ottawa	364	1089.09	12.79159	5181.370	449.0397	1184.532	93	12	8246.124	0.6826056	2129.615	0.6283401	2.433008	743.4189
3	2019	2019-01-01	2019-12-31	Soy	Ottawa	364	1771.08	12.41896	5127.400	448.7489	1108.090	64	21	7783.457	0.6802279	1983.980	0.6587562	2.584401	1204.7380

```
# Exporting data as a .csv file
# Daymet
write.csv(historical.summary.daymet, row.names = F, na='',
          file = paste0(path, 'Historical_summary_daymet.csv'))
```

5.6 NASA-POWER Historical summary

Summary can be obtained by years or by years.months. User must specify this option at the "intervals" argument of the summary function.

5.6.1 Intervals

```r
# Specify hist.data = dataframe containing the historical weather data to summarize

years = historical.years(hist.data = hist.weather.nasapower)

# Specify hist.data = dataframe containing the historical weather data to summarize
years.months = historical.years.months(hist.data = hist.weather.nasapower)

# Run the summary
# input = dataframe containing the historical weather data.
# intervals = type of historical intervals (years, years.months)
```

5.6.2 Summary

```r
# input = dataframe containing the historical weather data.
# intervals = type of historical intervals (years, years.months)

historical.summary.nasapower <-
  summary.daymet.nasapower(input = hist.weather.nasapower,
                           intervals = years)

kbl(historical.summary.nasapower) %>%
  kable_styling(font_size = 7, position = "center", latex_options = c("scale_down"))
```

ID	Year	Start.in	End.in	Crop	Site	Dur	PP	Tmean	Rad	VPD	ET0_HS	ETE	EPE	CHU	SDI	GDD	Q_chu	Q_gdd	AWDR
1	2000	2000-01-01	2000-12-31	Corn	Scandia	365	602.32	13.24638	5753.76	336.3510	1292.612	93	1	8937.944	0.7707448	2120.575	0.6437454	2.713302	464.2350
1	2001	2001-01-01	2001-12-31	Corn	Scandia	364	859.08	12.93495	5745.10	264.5661	1222.578	81	3	8331.061	0.7661693	2043.595	0.6896001	2.811271	658.2007
1	2002	2002-01-01	2002-12-31	Corn	Scandia	364	573.47	12.75551	5720.85	309.7433	1247.132	100	4	8259.408	0.7519184	2014.885	0.6926465	2.839294	431.2026
1	2003	2003-01-01	2003-12-31	Corn	Scandia	364	686.43	12.49056	5733.45	284.1095	1217.471	75	5	8084.866	0.7120453	1909.685	0.7091584	3.002301	488.7692
1	2004	2004-01-01	2004-12-31	Corn	Scandia	365	769.70	12.38059	5684.73	233.5174	1172.289	65	1	7901.955	0.7678605	1954.545	0.7194080	2.908467	591.0222
1	2005	2005-01-01	2005-12-31	Corn	Scandia	364	808.81	12.78176	5702.94	254.7706	1199.523	81	5	8295.770	0.7342180	2053.860	0.6874516	2.776694	593.8428
1	2006	2006-01-01	2006-12-31	Corn	Scandia	364	748.98	13.52328	5854.79	293.2485	1247.206	82	5	8378.264	0.7223758	2187.955	0.6988070	2.675919	541.0450
1	2007	2007-01-01	2007-12-31	Corn	Scandia	364	993.17	12.20837	5573.23	193.3783	1124.611	53	6	8029.643	0.7574844	1943.440	0.6940819	2.867714	752.3108
1	2008	2008-01-01	2008-12-31	Corn	Scandia	365	947.63	10.98212	NA	183.5211	1102.758	43	5	7030.394	0.7504728	1653.135	NA	NA	711.1705
1	2009	2009-01-01	2009-12-31	Corn	Scandia	364	697.58	11.37130	5774.87	208.8476	1128.201	51	1	7193.089	0.7583914	1699.855	0.8028359	3.397272	529.0387
1	2010	2010-01-01	2010-12-31	Corn	Scandia	364	825.16	12.33984	5837.68	222.3663	1155.458	75	5	8090.658	0.7243511	1939.625	0.7215334	3.009695	597.7056
1	2011	2011-01-01	2011-12-31	Corn	Scandia	364	803.97	12.32176	5939.37	264.1967	1189.969	77	5	8109.493	0.7176807	1948.835	0.7323972	3.047652	576.9937
1	2012	2012-01-01	2012-12-31	Corn	Scandia	365	547.28	14.62341	NA	366.2841	1346.743	103	5	9487.757	0.6577771	2395.205	NA	NA	359.9883
1	2013	2013-01-01	2013-12-31	Corn	Scandia	364	721.78	11.92681	NA	255.9873	1168.029	70	4	7819.464	0.7372302	1875.275	NA	NA	532.1180
1	2014	2014-01-01	2014-12-31	Corn	Scandia	364	659.18	11.82257	NA	256.8268	1185.396	73	2	7988.337	0.7382284	1879.220	NA	NA	486.6254
1	2015	2015-01-01	2015-12-31	Corn	Scandia	364	873.42	13.00084	NA	232.9642	1167.382	64	5	8378.936	0.7338279	2120.690	NA	NA	640.9400
1	2016	2016-01-01	2016-12-31	Corn	Scandia	365	810.93	13.51496	NA	232.1920	1191.999	76	2	8564.112	0.7472877	2213.515	NA	NA	605.9980
1	2017	2017-01-01	2017-12-31	Corn	Scandia	364	739.83	13.18391	NA	265.0667	1219.477	75	5	8323.556	0.7071833	2107.850	NA	NA	523.1954
1	2018	2018-01-01	2018-12-31	Corn	Scandia	364	915.12	12.03764	NA	240.4321	1177.964	86	8	8015.438	0.7185792	1933.395	NA	NA	657.5862
1	2019	2019-01-01	2019-12-31	Corn	Scandia	364	1036.89	10.99271	NA	135.0564	1039.597	39	7	6995.073	0.7541236	1669.270	NA	NA	781.9432
2	2000	2000-01-01	2000-12-31	Wheat	Belleville	365	602.32	13.24638	5753.76	336.3510	1292.769	93	1	8937.944	0.7707448	2120.575	0.6437454	2.713302	464.2350
2	2001	2001-01-01	2001-12-31	Wheat	Belleville	364	859.08	12.93495	5745.10	264.5661	1222.731	81	3	8331.061	0.7661693	2043.595	0.6896001	2.811271	658.2007
2	2002	2002-01-01	2002-12-31	Wheat	Belleville	364	573.47	12.75551	5720.85	309.7433	1247.282	100	4	8259.408	0.7519184	2014.885	0.6926465	2.839294	431.2026
2	2003	2003-01-01	2003-12-31	Wheat	Belleville	364	686.43	12.49056	5733.45	284.1095	1217.620	75	5	8084.866	0.7120453	1909.685	0.7091584	3.002301	488.7692
2	2004	2004-01-01	2004-12-31	Wheat	Belleville	365	769.70	12.38059	5684.73	233.5174	1172.440	65	1	7901.955	0.7678605	1954.545	0.7194080	2.908467	591.0222
2	2005	2005-01-01	2005-12-31	Wheat	Belleville	364	808.81	12.78176	5702.94	254.7706	1199.673	81	5	8295.770	0.7342180	2053.860	0.6874516	2.776694	593.8428
2	2006	2006-01-01	2006-12-31	Wheat	Belleville	364	748.98	13.52328	5854.79	293.2485	1247.365	82	5	8378.264	0.7223758	2187.955	0.6988070	2.675919	541.0450
2	2007	2007-01-01	2007-12-31	Wheat	Belleville	364	993.17	12.20837	5573.23	193.3783	1124.754	53	6	8029.643	0.7574844	1943.440	0.6940819	2.867714	752.3108
2	2008	2008-01-01	2008-12-31	Wheat	Belleville	365	947.63	10.98212	NA	183.5211	1102.895	43	5	7030.394	0.7504728	1653.135	NA	NA	711.1705
2	2009	2009-01-01	2009-12-31	Wheat	Belleville	364	697.58	11.37130	5774.87	208.8476	1128.339	51	1	7193.089	0.7583914	1699.855	0.8028359	3.397272	529.0387
2	2010	2010-01-01	2010-12-31	Wheat	Belleville	364	825.16	12.33984	5837.68	222.3663	1155.602	75	5	8090.658	0.7243511	1939.625	0.7215334	3.009695	597.7056
2	2011	2011-01-01	2011-12-31	Wheat	Belleville	364	803.97	12.32176	5939.37	264.1967	1190.116	77	5	8109.493	0.7176807	1948.835	0.7323972	3.047652	576.9937
2	2012	2012-01-01	2012-12-31	Wheat	Belleville	365	547.28	14.62341	NA	366.2841	1346.910	103	5	9487.757	0.6577771	2395.205	NA	NA	359.9883
2	2013	2013-01-01	2013-12-31	Wheat	Belleville	364	721.78	11.92681	NA	255.9873	1168.174	70	4	7819.464	0.7372302	1875.275	NA	NA	532.1180
2	2014	2014-01-01	2014-12-31	Wheat	Belleville	364	659.18	11.82257	NA	256.8268	1185.540	73	2	7988.337	0.7382284	1879.220	NA	NA	486.6254
2	2015	2015-01-01	2015-12-31	Wheat	Belleville	364	873.42	13.00084	NA	232.9642	1167.538	64	5	8378.936	0.7338279	2120.690	NA	NA	640.9400
2	2016	2016-01-01	2016-12-31	Wheat	Belleville	365	810.93	13.51496	NA	232.1920	1192.156	76	2	8564.112	0.7472877	2213.515	NA	NA	605.9980
2	2017	2017-01-01	2017-12-31	Wheat	Belleville	364	739.83	13.18391	NA	265.0667	1219.633	75	5	8323.556	0.7071833	2107.850	NA	NA	523.1954
2	2018	2018-01-01	2018-12-31	Wheat	Belleville	364	915.12	12.03764	NA	240.4321	1178.102	86	8	8015.438	0.7185792	1933.395	NA	NA	657.5862
2	2019	2019-01-01	2019-12-31	Wheat	Belleville	364	1036.89	10.99271	NA	135.0564	1039.727	39	7	6995.073	0.7541236	1669.270	NA	NA	781.9432
3	2000	2000-01-01	2000-12-31	Soy	Ottawa	365	741.07	13.70579	5672.74	298.5132	1276.557	97	2	9124.475	0.7600555	2210.445	0.6217059	2.566334	563.2543
3	2001	2001-01-01	2001-12-31	Soy	Ottawa	364	1020.72	13.50849	5809.51	211.5631	1193.977	62	4	8483.679	0.7550656	2190.105	0.6847866	2.652617	770.7106
3	2002	2002-01-01	2002-12-31	Soy	Ottawa	364	735.73	13.42022	5714.59	252.1175	1234.282	91	4	8422.441	0.7544644	2172.500	0.6784957	2.630421	555.0821
3	2003	2003-01-01	2003-12-31	Soy	Ottawa	364	799.39	13.14639	5570.23	261.3332	1238.705	69	5	8373.076	0.7228071	2031.900	0.6652550	2.741390	577.8047
3	2004	2004-01-01	2004-12-31	Soy	Ottawa	365	1135.10	12.53034	5618.81	153.9516	1103.198	27	8	7610.845	0.7587902	1962.590	0.7382636	2.862957	861.3027
3	2005	2005-01-01	2005-12-31	Soy	Ottawa	364	1104.53	12.82806	5690.20	174.7178	1134.029	49	12	7976.424	0.7367288	2065.170	0.7133773	2.755318	813.7391
3	2006	2006-01-01	2006-12-31	Soy	Ottawa	364	825.40	14.15922	5784.79	255.6557	1237.783	79	4	8658.404	0.7456299	2292.065	0.6681127	2.523833	615.4429
3	2007	2007-01-01	2007-12-31	Soy	Ottawa	364	1187.25	12.76114	5454.84	148.4151	1086.552	42	10	8187.997	0.7509775	2097.210	0.6661996	2.600999	891.5981
3	2008	2008-01-01	2008-12-31	Soy	Ottawa	365	1202.85	11.40427	NA	125.9982	1055.514	28	7	7056.751	0.7715914	1740.780	NA	NA	928.1087
3	2009	2009-01-01	2009-12-31	Soy	Ottawa	364	1217.25	11.69330	5577.69	128.2331	1048.948	26	7	6931.011	0.7699742	1712.620	0.8047441	3.256817	937.2512
3	2010	2010-01-01	2010-12-31	Soy	Ottawa	364	1056.96	12.75880	NA	159.1429	1110.307	62	10	8192.357	0.7439814	2051.650	NA	NA	786.3586
3	2011	2011-01-01	2011-12-31	Soy	Ottawa	364	865.58	13.54266	5810.20	275.3057	1253.737	92	3	8940.954	0.7551310	2186.150	0.6498411	2.657732	653.6262
3	2012	2012-01-01	2012-12-31	Soy	Ottawa	365	644.47	15.41844	NA	329.0573	1340.130	96	4	9798.964	0.7004707	2571.965	NA	NA	451.4323
3	2013	2013-01-01	2013-12-31	Soy	Ottawa	364	925.08	12.29738	NA	199.3420	1141.330	65	4	7801.187	0.7591886	1933.880	NA	NA	702.3102
3	2014	2014-01-01	2014-12-31	Soy	Ottawa	364	894.01	12.12967	NA	205.6149	1152.047	58	5	7991.891	0.7420734	1924.040	NA	NA	663.4211
3	2015	2015-01-01	2015-12-31	Soy	Ottawa	364	1135.28	13.32724	NA	167.5090	1123.507	52	7	8333.505	0.7671181	2195.735	NA	NA	870.8938
3	2016	2016-01-01	2016-12-31	Soy	Ottawa	365	1024.99	14.00532	NA	169.4432	1143.270	60	8	8667.481	0.7528693	2333.945	NA	NA	771.6835
3	2017	2017-01-01	2017-12-31	Soy	Ottawa	364	1047.91	13.87806	NA	195.9278	1180.193	58	10	8391.184	0.7218007	2252.375	NA	NA	756.3822
3	2018	2018-01-01	2018-12-31	Soy	Ottawa	364	958.78	13.14831	NA	220.8660	1185.180	99	7	8507.374	0.7466583	2179.970	NA	NA	715.8811
3	2019	2019-01-01	2019-12-31	Soy	Ottawa	364	1413.04	12.12419	NA	110.9091	1027.552	28	13	7451.871	0.7623617	1922.015	NA	NA	1077.2475

```
# Exporting data as a .csv file
# Daymet
write.csv(historical.summary.nasapower, row.names = F, na='',
          file = paste0(path, 'Historical_summary_nasapower.csv'))
```

5.7 CHIRPS Historical summary

Summary can be obtained by years or by years.months. User must specify this option at the "intervals" argument of the summary function.

5.7.1 Intervals

```
# Specify hist.data = dataframe containing the historical weather data to summarize

years = historical.years(hist.data = hist.weather.chirps)

# Specify hist.data = dataframe containing the historical weather data to summarize
years.months = historical.years.months(hist.data = hist.weather.chirps)
```

```
# Run the summary
# input = dataframe containing the historical weather data.
# intervals = type of historical intervals (years, years.months)
```

5.7.2 Summary

```
# input = dataframe containing the historical weather data.
# intervals = type of historical intervals (years, years.months)

historical.summary.chirps <-
  summary.chirps(input = hist.weather.chirps,
                              intervals = years)

kbl(historical.summary.chirps) %>%
  kable_styling(font_size = 7, position = "center", latex_options = c("scale_down"))
```

ID	Year	Start.in	End.in	Crop	Site	Dur	PP	EPE	SDI	AWDR
1	2000	2000-01-01	2000-12-31	Corn	Scandia	365	499.3485	2	0.6881628	343.6331
1	2001	2001-01-01	2001-12-31	Corn	Scandia	364	870.3336	7	0.6925287	602.7310
1	2002	2002-01-01	2002-12-31	Corn	Scandia	364	585.3025	4	0.6578964	385.0684
1	2003	2003-01-01	2003-12-31	Corn	Scandia	364	757.4231	6	0.6425145	486.6554
1	2004	2004-01-01	2004-12-31	Corn	Scandia	365	727.5973	2	0.7014822	510.3965
1	2005	2005-01-01	2005-12-31	Corn	Scandia	364	729.7513	7	0.6521099	475.8780
1	2006	2006-01-01	2006-12-31	Corn	Scandia	364	684.6518	5	0.6687303	457.8474
1	2007	2007-01-01	2007-12-31	Corn	Scandia	364	815.4451	5	0.6774749	552.4436
1	2008	2008-01-01	2008-12-31	Corn	Scandia	365	886.3654	6	0.6967637	617.5873
1	2009	2009-01-01	2009-12-31	Corn	Scandia	364	646.8506	1	0.7049619	456.0050
1	2010	2010-01-01	2010-12-31	Corn	Scandia	364	750.7935	4	0.6805154	510.9266
1	2011	2011-01-01	2011-12-31	Corn	Scandia	364	732.4323	3	0.6725762	492.6166
1	2012	2012-01-01	2012-12-31	Corn	Scandia	365	592.2249	5	0.6331376	374.9598
1	2013	2013-01-01	2013-12-31	Corn	Scandia	364	710.1979	4	0.6936898	492.6571
1	2014	2014-01-01	2014-12-31	Corn	Scandia	364	676.9023	4	0.6900753	467.1136
1	2015	2015-01-01	2015-12-31	Corn	Scandia	364	833.3981	3	0.7322068	610.2197
1	2016	2016-01-01	2016-12-31	Corn	Scandia	365	844.1357	2	0.7306711	616.7856
1	2017	2017-01-01	2017-12-31	Corn	Scandia	364	709.4869	7	0.6687871	474.4957
1	2018	2018-01-01	2018-12-31	Corn	Scandia	364	812.6801	10	0.6878171	558.9753
1	2019	2019-01-01	2019-12-31	Corn	Scandia	364	910.1402	6	0.7086933	645.0103
2	2000	2000-01-01	2000-12-31	Wheat	Belleville	365	525.1730	3	0.6900727	362.4076
2	2001	2001-01-01	2001-12-31	Wheat	Belleville	364	876.7466	6	0.7053497	618.4130
2	2002	2002-01-01	2002-12-31	Wheat	Belleville	364	580.7330	5	0.6610052	383.8675
2	2003	2003-01-01	2003-12-31	Wheat	Belleville	364	782.3805	6	0.6413441	501.7751
2	2004	2004-01-01	2004-12-31	Wheat	Belleville	365	716.1294	3	0.6937718	496.8304
2	2005	2005-01-01	2005-12-31	Wheat	Belleville	364	761.6930	6	0.6530140	497.3962
2	2006	2006-01-01	2006-12-31	Wheat	Belleville	364	735.7132	4	0.6874233	505.7464
2	2007	2007-01-01	2007-12-31	Wheat	Belleville	364	878.1005	7	0.6764843	594.0212
2	2008	2008-01-01	2008-12-31	Wheat	Belleville	365	893.1808	9	0.6963247	621.9438
2	2009	2009-01-01	2009-12-31	Wheat	Belleville	364	696.2938	1	0.7007190	487.9063
2	2010	2010-01-01	2010-12-31	Wheat	Belleville	364	805.9291	6	0.6751747	544.1429
2	2011	2011-01-01	2011-12-31	Wheat	Belleville	364	750.7867	7	0.6619973	497.0188
2	2012	2012-01-01	2012-12-31	Wheat	Belleville	365	590.3264	4	0.6344293	374.5204
2	2013	2013-01-01	2013-12-31	Wheat	Belleville	364	724.7179	6	0.6830068	494.9872
2	2014	2014-01-01	2014-12-31	Wheat	Belleville	364	699.2984	4	0.7067073	494.1993
2	2015	2015-01-01	2015-12-31	Wheat	Belleville	364	845.4522	3	0.7309626	617.9939
2	2016	2016-01-01	2016-12-31	Wheat	Belleville	365	850.1045	2	0.7328990	623.0407
2	2017	2017-01-01	2017-12-31	Wheat	Belleville	364	763.5838	8	0.6608411	504.6076
2	2018	2018-01-01	2018-12-31	Wheat	Belleville	364	839.4139	9	0.6818291	572.3368
2	2019	2019-01-01	2019-12-31	Wheat	Belleville	364	953.9690	6	0.7063422	673.8286
3	2000	2000-01-01	2000-12-31	Soy	Ottawa	365	690.8306	4	0.6822080	471.2901
3	2001	2001-01-01	2001-12-31	Soy	Ottawa	364	967.1720	7	0.7154856	691.9977
3	2002	2002-01-01	2002-12-31	Soy	Ottawa	364	839.3448	7	0.6992869	586.9429
3	2003	2003-01-01	2003-12-31	Soy	Ottawa	364	852.5324	6	0.6743419	574.8983
3	2004	2004-01-01	2004-12-31	Soy	Ottawa	365	1101.9620	10	0.7138454	786.6305
3	2005	2005-01-01	2005-12-31	Soy	Ottawa	364	1057.7508	12	0.6850213	724.5818
3	2006	2006-01-01	2006-12-31	Soy	Ottawa	364	767.4704	6	0.6727282	516.2990
3	2007	2007-01-01	2007-12-31	Soy	Ottawa	364	1110.2389	9	0.6817829	756.9419
3	2008	2008-01-01	2008-12-31	Soy	Ottawa	365	1126.0232	9	0.7028945	791.4756
3	2009	2009-01-01	2009-12-31	Soy	Ottawa	364	1213.4274	11	0.7292031	884.8349
3	2010	2010-01-01	2010-12-31	Soy	Ottawa	364	1030.0084	9	0.6641430	684.0729
3	2011	2011-01-01	2011-12-31	Soy	Ottawa	364	878.0863	5	0.7046768	618.7670
3	2012	2012-01-01	2012-12-31	Soy	Ottawa	365	663.7940	5	0.6378462	423.3985
3	2013	2013-01-01	2013-12-31	Soy	Ottawa	364	980.1644	9	0.6898653	676.1814
3	2014	2014-01-01	2014-12-31	Soy	Ottawa	364	892.1045	11	0.6853283	611.3845
3	2015	2015-01-01	2015-12-31	Soy	Ottawa	364	1177.7812	9	0.7319779	862.1098
3	2016	2016-01-01	2016-12-31	Soy	Ottawa	365	1058.0802	8	0.7095237	750.7330
3	2017	2017-01-01	2017-12-31	Soy	Ottawa	364	1006.9949	8	0.6833515	688.1315
3	2018	2018-01-01	2018-12-31	Soy	Ottawa	364	946.6831	9	0.6761854	640.1333
3	2019	2019-01-01	2019-12-31	Soy	Ottawa	364	1506.2817	16	0.7383182	1112.1153

```r
# Exporting data as a .csv file
# Daymet
write.csv(historical.summary.chirps, row.names = F, na='',
          file = paste0(path, 'Historical_summary_chirps.csv'))
```

6 REFERENCES

Allen, R.G., L.S. Pereira, D. Raes, M. Smith. 1998. Crop evapotranspiration - Guidelines for computing crop water requirements. *FAO Irrigation and drainage*, 56. FAO - Food and Agriculture Organization of the United Nations. Rome, Italy. ISBN 92-5-104219-5. http://www.fao.org/3/x0490e/x0490e00.htm#Contents

Bannayan, M., Hoogenboom, G., & Crout, N.M.J., 2004. Photothermal impact on maize performance: a simulation approach. *Ecol. Modell.*, 180 (2-3), 277-290. https://doi.org/10.1016/j.ecolmodel.2004.04.022

Bootsma, A., S. Gameda, & D.W. McKenney. 2005. Potential impacts of cli- mate change on corn, soybeans and barley yields in Atlantic Canada. *Can J. Soil Sci.* 85:345–357. https://doi.org/10.4141/S04-025

Butler, E.E., & Huybers, P. 2013. Adaptation of US maize to temperature variations. *Nat. Clim. Chang.* 3N, 68–72. https://doi.org/10.1038/nclimate1585

Cobaner M., H. Citakoğlu, T. Haktanir, & O. Kisi. 2017. Modifying Hargreaves–Samani equation with meteorological variables for estimation of reference evapotranspiration in Turkey. *Hydrol. Res.* 1 April 2017, 48 (2), 480–497. https://doi.org/10.2166/nh.2016.217

Correndo, A.A., J.L. Rotundo, N. Tremblay, S. Archontoulis, J.A. Coulter, D. Ruiz-Diaz, D. Franzen, A.J. Franzluebbers, E. Nafziger, R. Schwalbert, K. Steinke, J. Williams, C.D. Messina, & I.A. Ciampitti. 2021. Assessing the uncertainty of maize yield without nitrogen fertilization. *Field Crops Res.* 260, 2021, 107985. https://doi.org/10.1016/j.fcr.2020.107985.

Funk C., P. Peterson, M. Landsfeld, D. Pedreros, J. Verdin, S. Shukla, & J. Michaelsen. 2015. The climate hazards infrared precipitation with stations—a new environmental record for monitoring extremes. *Scientific Data* 2, 150066. https://doi.org/10.1038/sdata.2015.66.

Gilmore, E.C., & Rogers, J.S. 1958. Heat units as a method of measuring maturity in corn. *Agron. J.* 50:611–615. https://doi.org/10.2134/agronj1958.00021962005000100014x

Hargreaves,G.H., & Z.A. Samani. 1985. Reference crop evapotranspiration from temperature. *Appl. Eng. Agric.* 1(2),96–99. https://doi.org/10.13031/2013.26773

Raziei, T., & L.S. Pereira. 2013. Estimation of ET0 with Hargreaves–Samani and FAO-PM temperature methods for a wide range of climates in Iran. *Agric. Water Manag.* 121 (2013), 1-18. https://doi.org/10.1016/j.agwat.2012.12.019

Sparks, A. 2018. nasapower: A NASA POWER Global Meteorology, Surface Solar ENergy and Climatology Data Client for R. *J. of Open Source Softw.*, 3(30), 1035. https://doi.org/10.21105/joss.01035

Thornton, P.E., M. Thornton, B. Mayer, Y. Wei, R. Devarakonda, R. Vose, & R.B. Cook. 2019. Daymet: daily surface weather data on a 1-km Grid for North America, Version3. ORNL DAAC, Oak Ridge, Tennessee, USA. https://daymet.ornl.gov/

Tremblay, N., Bouroubi, Y.M., Bélec, C., Mullen, R.W., Kitchen, N.R., Thomason, W.E., Ebelhar, S., Mengel, D.B., Raun, W.R., Francis, D.D., Vories, E.D., & Ortiz-Monasterio, I., 2012. Corn response to nitrogen is influenced by soil texture and weather. *Agron. J.*, 104, 1658-1671. https://doi.org/10.2134/agronj2012.0184

Ye, Q., Lin, X., Adee, E., Min, D., Assefa Mulisa, Y., O'Brien, D., & Ciampitti, I.A., 2017. Evaluation of climatic variables as yield-limiting factors for maize in Kansas. *Int. J. Climatol.* 37.S1, 464-75. https://doi.org/10.1002/joc.5015