

# Package: getCRUCLdata (via r-universe)

February 9, 2025

**Type** Package

**Title** 'CRU' 'CL' v. 2.0 Climatology Client

**Version** 1.0.3

**Description** Provides functions that automate downloading and importing University of East Anglia Climate Research Unit ('CRU') 'CL' v. 2.0 climatology data, facilitates the calculation of minimum temperature and maximum temperature and formats the data into a data.table object or a list of 'terra' 'rast' objects for use. 'CRU' 'CL' v. 2.0 data are a gridded climatology of 1961-1990 monthly means released in 2002 and cover all land areas (excluding Antarctica) at 10 arc minutes (0.1666667 degree) resolution. For more information see the description of the data provided by the University of East Anglia Climate Research Unit, <<https://crudata.uea.ac.uk/cru/data/hrg/tmc/readme.txt>>.

**License** MIT + file LICENSE

**URL** <https://github.com/ropensci/getCRUCLdata>,  
<https://docs.ropensci.org/getCRUCLdata/>

**BugReports** <https://github.com/ropensci/getCRUCLdata/issues>

**Depends** R (>= 4.0.0)

**Imports** cli, curl, data.table, hoardr, rlang, terra, utils

**Suggests** ggplot2, knitr, rappdirs, rmarkdown, roxygen2, roxyglobals, spelling, testthat, viridis, withr

**VignetteBuilder** knitr

**ByteCompile** TRUE

**Config/roxyglobals/filename** globals.R

**Config/roxyglobals/unique** FALSE

**Config/roxylint** list(linters = roxylint::tidy)

**Config/testthat/edition** 3

**Config/testthat/parallel** true

**Encoding** UTF-8

**Language** en-US

**RoxygenNote** 7.3.2

**X-schema.org-applicationCategory** Tools

**X-schema.org-isPartOf** <https://ropensci.org>

**X-schema.org-keywords** anglia-cru, climate-data, cru-cl2, temperature,  
rainfall, elevation, data-access, wind, relative-humidity,  
solar-radiation, diurnal-temperature, frost

**Config/Needs/build** moodymudskipper/devtag

**NeedsCompilation** no

**Author** Adam H. Sparks [aut, cre]  
(<https://orcid.org/0000-0002-0061-8359>), Curtin University of  
Technology [cph] (Provided support through Adam Sparks's  
time.), Grains Research and Development Corporation [cph] (GRDC  
Project CUR2210-005OPX (AAGI-CU))

**Maintainer** Adam H. Sparks <[adamhsparks@gmail.com](mailto:adamhsparks@gmail.com)>

**Date/Publication** 2025-02-08 16:20:16 UTC

**Additional\_repositories** <https://cranhaven.r-universe.dev>

**Config/pak/sysreqs** libgdal-dev gdal-bin libgeos-dev libssl-dev  
libproj-dev libsqlite3-dev

**Repository** <https://cranhaven.r-universe.dev>

**RemoteUrl** <https://github.com/cranhaven/cranhaven.r-universe.dev>

**RemoteRef** package/getCRUCLdata

**RemoteSha** 5d2630b0c27713485a6fb3650cf94183ce76d2f3

**RemoteSubdir** getCRUCLdata

## Contents

create_CRU_df . . . . .	3
create_CRU_stack . . . . .	6
get_CRU_df . . . . .	9
get_CRU_stack . . . . .	12
manage_cache . . . . .	15

<b>Index</b>	<b>16</b>
--------------	-----------

---

create_CRU_df	<i>Create a data.table of CRU CL v. 2.0 climatology elements from local disk files</i>
---------------	--

---

## Description

Automates importing CRU CL v.2.0 climatology data and creates a **data.table** of the data. If requested, minimum and maximum temperature may also be automatically calculated as described in the data [readme.txt](#) file. Data may be cached for later use by this function, saving time downloading files in future using this function. This function can be useful if you have network connection issues that mean automated downloading of the files using R does not work properly.

## Usage

```
create_CRU_df(  
  pre = FALSE,  
  pre_cv = FALSE,  
  rd0 = FALSE,  
  tmp = FALSE,  
  dtr = FALSE,  
  reh = FALSE,  
  tmn = FALSE,  
  tmx = FALSE,  
  sunp = FALSE,  
  frs = FALSE,  
  wnd = FALSE,  
  elv = FALSE,  
  dsn  
)
```

```
create_cru_df(  
  pre = FALSE,  
  pre_cv = FALSE,  
  rd0 = FALSE,  
  tmp = FALSE,  
  dtr = FALSE,  
  reh = FALSE,  
  tmn = FALSE,  
  tmx = FALSE,  
  sunp = FALSE,  
  frs = FALSE,  
  wnd = FALSE,  
  elv = FALSE,  
  dsn  
)
```

### Arguments

<code>pre</code>	Loads precipitation (millimetres/month) from server and returns in the data frame, TRUE. Defaults to FALSE.
<code>pre_cv</code>	Loads cv of precipitation (percent) from server and returns in the data frame, TRUE. Defaults to FALSE. NOTE. Setting this to TRUE will always results in <b>pre</b> being set to TRUE and returned as well.
<code>rd0</code>	Loads wet-days (number days with >0.1 millimetres rain per month) and returns in the data frame, TRUE. Defaults to FALSE.
<code>tmp</code>	Loads temperature (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>dtr</code>	Loads mean diurnal temperature range (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>reh</code>	Loads relative humidity and returns it in the data frame, TRUE. Defaults to FALSE.
<code>tmn</code>	Calculate minimum temperature values (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>tmx</code>	Calculate maximum temperature (degrees Celsius) and return it in the data frame, TRUE. Defaults to FALSE.
<code>sunp</code>	Loads sunshine, percent of maximum possible (percent of day length) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>frs</code>	Loads ground-frost records (number of days with ground- frost per month) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>wnd</code>	Load 10 m wind speed (metres/second) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>elv</code>	Loads elevation (converted to metres) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>dsn</code>	Local file path where CRU CL v.2.0 .dat.gz files are located.

### Value

A `data.table::data.table` object of CRU CL v. 2.0 climatology elements.

### Nomenclature and Units

<b>pre</b>	precipitation (millimetres/month)
<b>cv</b>	cv of precipitation (percent)
<b>rd0</b>	wet-days (number days with >0.1 millimetres rain per month)
<b>tmp</b>	mean temperature (degrees Celsius)
<b>dtr</b>	mean diurnal temperature range (degrees Celsius)
<b>reh</b>	relative humidity (percent)
<b>sunp</b>	sunshine (percent of maximum possible (percent of day length))
<b>frs</b>	ground-frost (number of days with ground-frost per month)
<b>wnd</b>	10 metre windspeed (metres/second)
<b>elv</b>	elevation (automatically converted to metres)

For more information see the description of the data provided by CRU, <https://crudata.uea.ac.uk/cru/data/hrg/tmc/readme.txt>

**Author(s)**

Adam H. Sparks, <adamhsparks@gmail.com>

**Source**

**pre** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_pre.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_pre.dat.gz)

**rd0** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_rd0.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_rd0.dat.gz)

**tmp** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_tmp.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_tmp.dat.gz)

**dtr** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_dtr.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_dtr.dat.gz)

**reh** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_reh.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_reh.dat.gz)

**sunp** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_sunp.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_sunp.dat.gz)

**frs** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_frs.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_frs.dat.gz)

**wnd** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_wnd.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_wnd.dat.gz), areas originally including Antarctica are removed.

**elv** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_elv.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_elv.dat.gz), values are converted from kilometres to metres.

This package crops all spatial outputs to an extent of ymin = -60, ymax = 85, xmin = -180, xmax = 180.

**References**

New, Mark, et al. "A high-resolution data set of surface climate over global land areas." *Climate research* 21.1 (2002): 1-25. [https://crudata.uea.ac.uk/cru/data/hrg/tmc/new\\_et\\_al\\_10minute\\_climate\\_CR.pdf](https://crudata.uea.ac.uk/cru/data/hrg/tmc/new_et_al_10minute_climate_CR.pdf)

**See Also**

[get\\_CRU\\_df](#).

**Examples**

```
# Create a data frame of temperature from locally available files in the
# tempdir() directory.

download.file(
  url = "https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_tmp.dat.gz",
  destfile = file.path(tempdir(), "grid_10min_tmp.dat.gz")
)

CRU_tmp <- create_CRU_df(tmp = TRUE, dsn = tempdir())

CRU_tmp
```

---

create_CRU_stack	<i>Create a list of terra rast objects of CRU CL v. 2.0 climatology elements from local disk files</i>
------------------	--

---

### Description

Automates importing CRU CL v.2.0 climatology data and creates a **terra** `terra::rast` object of the data. If requested, minimum and maximum temperature may also be automatically calculated as described in the data [readme.txt](#) file. Data may be cached for later use by this function, saving time downloading files in future using this function. This function can be useful if you have network connection issues that mean automated downloading of the files using R does not work properly or you have cached the files locally for your own future use.

### Usage

```
create_CRU_stack(  
  pre = FALSE,  
  pre_cv = FALSE,  
  rd0 = FALSE,  
  tmp = FALSE,  
  dtr = FALSE,  
  reh = FALSE,  
  tmn = FALSE,  
  tmx = FALSE,  
  sunp = FALSE,  
  frs = FALSE,  
  wnd = FALSE,  
  elv = FALSE,  
  dsn  
)
```

```
create_cru_stack(  
  pre = FALSE,  
  pre_cv = FALSE,  
  rd0 = FALSE,  
  tmp = FALSE,  
  dtr = FALSE,  
  reh = FALSE,  
  tmn = FALSE,  
  tmx = FALSE,  
  sunp = FALSE,  
  frs = FALSE,  
  wnd = FALSE,  
  elv = FALSE,  
  dsn  
)
```

**Arguments**

<code>pre</code>	Loads precipitation (millimetres/month) from server and returns in the data frame, TRUE. Defaults to FALSE.
<code>pre_cv</code>	Loads cv of precipitation (percent) from server and returns in the data frame, TRUE. Defaults to FALSE. NOTE. Setting this to TRUE will always results in <b>pre</b> being set to TRUE and returned as well.
<code>rd0</code>	Loads wet-days (number days with >0.1 millimetres rain per month) and returns in the data frame, TRUE. Defaults to FALSE.
<code>tmp</code>	Loads temperature (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>dtr</code>	Loads mean diurnal temperature range (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>reh</code>	Loads relative humidity and returns it in the data frame, TRUE. Defaults to FALSE.
<code>tmn</code>	Calculate minimum temperature values (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>tmx</code>	Calculate maximum temperature (degrees Celsius) and return it in the data frame, TRUE. Defaults to FALSE.
<code>sunp</code>	Loads sunshine, percent of maximum possible (percent of day length) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>frs</code>	Loads ground-frost records (number of days with ground- frost per month) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>wnd</code>	Load 10 m wind speed (metres/second) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>elv</code>	Loads elevation (converted to metres) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>dsn</code>	Local file path where CRU CL v.2.0 .dat.gz files are located.

**Value**

A `base::list` of `terra::rast` objects of CRU CL v. 2.0 climatology elements.

**Nomenclature and Units**

<b>pre</b>	precipitation (millimetres/month)
<b>cv</b>	cv of precipitation (percent)
<b>rd0</b>	wet-days (number days with >0.1 millimetres rain per month)
<b>tmp</b>	mean temperature (degrees Celsius)
<b>dtr</b>	mean diurnal temperature range (degrees Celsius)
<b>reh</b>	relative humidity (percent)
<b>sunp</b>	sunshine (percent of maximum possible (percent of day length))
<b>frs</b>	ground-frost (number of days with ground-frost per month)
<b>wnd</b>	10 metre windspeed (metres/second)
<b>elv</b>	elevation (automatically converted to metres)

For more information see the description of the data provided by CRU, <https://crudata.uea.ac.uk/cru/data/hrg/tmc/readme.txt>

**Author(s)**

Adam H. Sparks, <adamhsparks@gmail.com>

**Source**

**pre** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_pre.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_pre.dat.gz)

**rd0** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_rd0.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_rd0.dat.gz)

**tmp** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_tmp.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_tmp.dat.gz)

**dtr** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_dtr.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_dtr.dat.gz)

**reh** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_reh.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_reh.dat.gz)

**sunp** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_sunp.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_sunp.dat.gz)

**frs** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_frs.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_frs.dat.gz)

**wnd** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_wnd.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_wnd.dat.gz), areas originally including Antarctica are removed.

**elv** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_elv.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_elv.dat.gz), values are converted from kilometres to metres.

This package crops all spatial outputs to an extent of ymin = -60, ymax = 85, xmin = -180, xmax = 180.

**References**

New, Mark, et al. "A high-resolution data set of surface climate over global land areas." *Climate research* 21.1 (2002): 1-25. [https://crudata.uea.ac.uk/cru/data/hrg/tmc/new\\_et\\_al\\_10minute\\_climate\\_CR.pdf](https://crudata.uea.ac.uk/cru/data/hrg/tmc/new_et_al_10minute_climate_CR.pdf)

**See Also**

[get\\_CRU\\_stack](#).

**Examples**

```
download.file(  
  url = "https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_tmp.dat.gz",  
  destfile = file.path(tempdir(), "grid_10min_tmp.dat.gz")  
)
```

```
CRU_tmp <- create_CRU_stack(tmp = TRUE, dsn = tempdir())
```

```
CRU_tmp
```



---

get_CRU_df	<i>Download and create a data.table of CRU CL v. 2.0 climatology elements</i>
------------	---

---

## Description

This function automates downloading and importing CRU CL v. 2.0 climatology data and creates a **data.table** of the data. If requested, minimum and maximum temperature may also be automatically calculated as described in the data [readme.txt](#) file. Data may be cached for later use by this function, saving time downloading files in future use of this function.

## Usage

```
get_CRU_df(  
  pre = FALSE,  
  pre_cv = FALSE,  
  rd0 = FALSE,  
  tmp = FALSE,  
  dtr = FALSE,  
  reh = FALSE,  
  tmn = FALSE,  
  tmx = FALSE,  
  sunp = FALSE,  
  frs = FALSE,  
  wnd = FALSE,  
  elv = FALSE,  
  cache = FALSE  
)
```

```
get_cru_df(  
  pre = FALSE,  
  pre_cv = FALSE,  
  rd0 = FALSE,  
  tmp = FALSE,  
  dtr = FALSE,  
  reh = FALSE,  
  tmn = FALSE,  
  tmx = FALSE,  
  sunp = FALSE,  
  frs = FALSE,  
  wnd = FALSE,  
  elv = FALSE,  
  cache = FALSE  
)
```

**Arguments**

<code>pre</code>	Fetches precipitation (millimetres/month) from server and returns it in the data frame, TRUE. Defaults to FALSE.
<code>pre_cv</code>	Fetch cv of precipitation (percent) from server and returns it in the data frame, TRUE. Defaults to FALSE. NOTE Setting this to TRUE will always results in <b>pre</b> being set to TRUE and returned as well.
<code>rd0</code>	Fetches wet-days (number days with >0.1 millimetres rain per month) and returns it in the data frame? Defaults to FALSE.
<code>tmp</code>	Fetches temperature (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>dtr</code>	Fetches mean diurnal temperature range (degrees Celsius) and returns it in the data frame? Defaults to FALSE.
<code>reh</code>	Fetches relative humidity and returns it in the data frame, TRUE. Defaults to FALSE.
<code>tmn</code>	Calculates minimum temperature values (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>tmx</code>	Calculates maximum temperature (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>sunp</code>	Fetch sunshine, percent of maximum possible (percent of day length) and return it in the data frame? Defaults to FALSE.
<code>frs</code>	Fetches ground-frost records (number of days with ground- frost per month) and return it in the data frame? Defaults to FALSE.
<code>wnd</code>	Fetches 10m wind speed (metres/second) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>elv</code>	Fetches elevation (converted to metres) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>cache</code>	Stores CRU CL v. 2.0 data files locally for later use. If FALSE, the downloaded files are removed when the R session is closed. To take advantage of cached files in future sessions, use <code>cache = TRUE</code> even after the initial download and caching. Defaults to FALSE.

**Value**

A `data.table::data.table` object of CRU CL v. 2.0 climatology elements.

**Nomenclature and Units**

<b>pre</b>	precipitation (millimetres/month)
<b>cv</b>	cv of precipitation (percent)
<b>rd0</b>	wet-days (number days with >0.1 millimetres rain per month)
<b>tmp</b>	mean temperature (degrees Celsius)
<b>dtr</b>	mean diurnal temperature range (degrees Celsius)
<b>reh</b>	relative humidity (percent)

**sunp** sunshine (percent of maximum possible (percent of day length))

**frs** ground-frost (number of days with ground-frost per month)

**wnd** 10 metre windspeed (metres/second)

**elv** elevation (automatically converted to metres)

For more information see the description of the data provided by CRU, <https://crudata.uea.ac.uk/cru/data/hrg/tmc/readme.txt>

### Author(s)

Adam H. Sparks, <adamhsparks@gmail.com>

### Source

**pre** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_pre.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_pre.dat.gz)

**rd0** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_rd0.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_rd0.dat.gz)

**tmp** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_tmp.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_tmp.dat.gz)

**dtr** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_dtr.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_dtr.dat.gz)

**reh** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_reh.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_reh.dat.gz)

**sunp** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_sunp.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_sunp.dat.gz)

**frs** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_frs.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_frs.dat.gz)

**wnd** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_wnd.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_wnd.dat.gz), areas originally including Antarctica are removed.

**elv** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_elv.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_elv.dat.gz), values are converted from kilometres to metres.

This package crops all spatial outputs to an extent of ymin = -60, ymax = 85, xmin = -180, xmax = 180.

### References

New, Mark, et al. "A high-resolution data set of surface climate over global land areas." Climate research 21.1 (2002): 1-25. [https://crudata.uea.ac.uk/cru/data/hrg/tmc/new\\_et\\_al\\_10minute\\_climate\\_CR.pdf](https://crudata.uea.ac.uk/cru/data/hrg/tmc/new_et_al_10minute_climate_CR.pdf)

### See Also

[create\\_CRU\\_stack](#), [manage\\_cache](#).

### Examples

```
# Download data and create a data frame of precipitation and temperature
# without caching the data files
CRU_pre_tmp <- get_CRU_df(pre = TRUE, tmp = TRUE)

head(CRU_pre_tmp)
CRU_pre_tmp
```

---

get_CRU_stack	<i>Download and create a list of terra rast objects of CRU CL v. 2.0 climatology elements</i>
---------------	---

---

### Description

This function automates downloading and importing CRU CL v. 2.0 climatology data into R and creates a list of **terra** `terra::rast` objects of the data. If requested, minimum and maximum temperature may also be automatically calculated as described in the data [readme.txt](#) file. Data may be cached for later use by this function, saving time downloading files in future using this function.

### Usage

```
get_CRU_stack(  
  pre = FALSE,  
  pre_cv = FALSE,  
  rd0 = FALSE,  
  tmp = FALSE,  
  dtr = FALSE,  
  reh = FALSE,  
  tmn = FALSE,  
  tmx = FALSE,  
  sunp = FALSE,  
  frs = FALSE,  
  wnd = FALSE,  
  elv = FALSE,  
  cache = FALSE  
)
```

```
get_cru_stack(  
  pre = FALSE,  
  pre_cv = FALSE,  
  rd0 = FALSE,  
  tmp = FALSE,  
  dtr = FALSE,  
  reh = FALSE,  
  tmn = FALSE,  
  tmx = FALSE,  
  sunp = FALSE,  
  frs = FALSE,  
  wnd = FALSE,  
  elv = FALSE,  
  cache = FALSE  
)
```

**Arguments**

<code>pre</code>	Fetches precipitation (millimetres/month) from server and returns it in the data frame, TRUE. Defaults to FALSE.
<code>pre_cv</code>	Fetch cv of precipitation (percent) from server and returns it in the data frame, TRUE. Defaults to FALSE. NOTE Setting this to TRUE will always results in <b>pre</b> being set to TRUE and returned as well.
<code>rd0</code>	Fetches wet-days (number days with >0.1 millimetres rain per month) and returns it in the data frame? Defaults to FALSE.
<code>tmp</code>	Fetches temperature (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>dtr</code>	Fetches mean diurnal temperature range (degrees Celsius) and returns it in the data frame? Defaults to FALSE.
<code>reh</code>	Fetches relative humidity and returns it in the data frame, TRUE. Defaults to FALSE.
<code>tmn</code>	Calculates minimum temperature values (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>tmx</code>	Calculates maximum temperature (degrees Celsius) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>sunp</code>	Fetch sunshine, percent of maximum possible (percent of day length) and return it in the data frame? Defaults to FALSE.
<code>frs</code>	Fetches ground-frost records (number of days with ground- frost per month) and return it in the data frame? Defaults to FALSE.
<code>wnd</code>	Fetches 10m wind speed (metres/second) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>elv</code>	Fetches elevation (converted to metres) and returns it in the data frame, TRUE. Defaults to FALSE.
<code>cache</code>	Stores CRU CL v. 2.0 data files locally for later use. If FALSE, the downloaded files are removed when the R session is closed. To take advantage of cached files in future sessions, use <code>cache = TRUE</code> even after the initial download and caching. Defaults to FALSE.

**Value**

A `base::list` of `terra::rast` objects of CRU CL v. 2.0 climatology elements.

**Nomenclature and Units**

<b>pre</b>	precipitation (millimetres/month)
<b>cv</b>	cv of precipitation (percent)
<b>rd0</b>	wet-days (number days with >0.1 millimetres rain per month)
<b>tmp</b>	mean temperature (degrees Celsius)
<b>dtr</b>	mean diurnal temperature range (degrees Celsius)
<b>reh</b>	relative humidity (percent)

**sunp** sunshine (percent of maximum possible (percent of day length))

**frs** ground-frost (number of days with ground-frost per month)

**wnd** 10 metre windspeed (metres/second)

**elv** elevation (automatically converted to metres)

For more information see the description of the data provided by CRU, <https://crudata.uea.ac.uk/cru/data/hrg/tmc/readme.txt>

### Author(s)

Adam H. Sparks, <adamhsparks@gmail.com>

### Source

**pre** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_pre.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_pre.dat.gz)

**rd0** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_rd0.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_rd0.dat.gz)

**tmp** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_tmp.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_tmp.dat.gz)

**dtr** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_dtr.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_dtr.dat.gz)

**reh** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_reh.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_reh.dat.gz)

**sunp** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_sunp.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_sunp.dat.gz)

**frs** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_frs.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_frs.dat.gz)

**wnd** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_wnd.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_wnd.dat.gz), areas originally including Antarctica are removed.

**elv** [https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid\\_10min\\_elv.dat.gz](https://crudata.uea.ac.uk/cru/data/hrg/tmc/grid_10min_elv.dat.gz), values are converted from kilometres to metres.

This package crops all spatial outputs to an extent of ymin = -60, ymax = 85, xmin = -180, xmax = 180.

### References

New, Mark, et al. "A high-resolution data set of surface climate over global land areas." *Climate research* 21.1 (2002): 1-25. [https://crudata.uea.ac.uk/cru/data/hrg/tmc/new\\_et\\_al\\_10minute\\_climate\\_CR.pdf](https://crudata.uea.ac.uk/cru/data/hrg/tmc/new_et_al_10minute_climate_CR.pdf)

### See Also

[create\\_CRU\\_stack](#), [manage\\_cache](#).

### Examples

```
# Download data and create a list of {terra} `rast` objects of precipitation
# and temperature without caching the data files
CRU_pre_tmp <- get_CRU_stack(pre = TRUE, tmp = TRUE)

CRU_pre_tmp
```

---

manage_cache	<i>Manage locally cached CRU CL v. 2.0 files</i>
--------------	--

---

## Description

Manage cached **getCRUCLdata** files with **hoardr**.

## Details

The default cache directory is `tools::R_user_dir(package = "getCRUCLdata")`, but you can set your own path using `manage_cache$cache_path_set()`.

`manage_cache$cache_delete` only accepts one file name, while `manage_cache$cache_delete_all` does not accept any names, but deletes all files. For deleting many specific files, use `manage_cache$cache_delete` in an `base::lapply` type call.

## Useful user functions

`manage_cache$cache_path_get()` get cache path  
`manage_cache$cache_path_set()` set cache path  
`manage_cache$list()` returns a character vector of full path file names  
`manage_cache$files()` returns file objects with metadata  
`manage_cache$details()` returns files with details  
`manage_cache$delete()` delete specific files  
`manage_cache$delete_all()` delete all files, returns nothing

## Author(s)

Adam H. Sparks, <adamhsparks@gmail.com>

## Examples

```
## Not run:  
  
# list files in cache  
manage_cache$list()  
  
# delete certain database files  
manage_cache$delete("file path")  
manage_cache$list()  
  
# delete all files in cache  
manage_cache$delete_all()  
manage_cache$list()  
  
# set a different cache path from the default  
manage_cache$cache_path_set("~/tmp")  
  
## End(Not run)
```

# Index

`base::lapply`, [15](#)

`base::list`, [7](#), [13](#)

`create_CRU_df`, [3](#)

`create_cru_df` (`create_CRU_df`), [3](#)

`create_CRU_stack`, [6](#), [11](#), [14](#)

`create_cru_stack` (`create_CRU_stack`), [6](#)

`data.table::data.table`, [4](#), [10](#)

`get_CRU_df`, [5](#), [9](#)

`get_cru_df` (`get_CRU_df`), [9](#)

`get_CRU_stack`, [8](#), [12](#)

`get_cru_stack` (`get_CRU_stack`), [12](#)

`manage_cache`, [11](#), [14](#), [15](#)

`terra::rast`, [6](#), [7](#), [12](#), [13](#)