

# AClimate R API

The AClimate R API package is a comprehensive R interface for accessing the AClimate Web API. This package seamlessly integrates with R, providing users with convenient access to a wide range of agro-climatic forecasts offered by the AClimate platform.

- [Official website \(https://www.aclimate.org\)](https://www.aclimate.org).
- Source code [source code \(https://github.com/CIAT-DAPA/aclimaterapi/\)](https://github.com/CIAT-DAPA/aclimaterapi/).
- Documentation of Web API and Dictionary of variable [documentation \(https://docs.aclimate.org/en/latest/08-webapi.html\)](https://docs.aclimate.org/en/latest/08-webapi.html).
- Depends **R (>= 4.2)**
- Author [stevenotelo \(https://github.com/stevenotelo\)](https://github.com/stevenotelo)

## Key Features

- **Data Access:** Access all relevant information from the AClimate platform, including accurate and up-to-date seasonal climate forecasts.
- **User-Friendly:** The package is designed with ease of use in mind, ensuring that users can effortlessly retrieve the information they need.
- **Seamless Integration:** Integrate AClimate data directly into R workflows, making it simple for users to incorporate climate and agroclimatic forecasts into their existing data analysis and decision-making processes

## Install

The easiest way to install the package is from [Github repository \(https://github.com/CIAT-DAPA/aclimaterapi/\)](https://github.com/CIAT-DAPA/aclimaterapi/) and using devtools.

```
devtools::install_github("CIAT-DAPA/aclimaterapi")
```

The above command, when executed in R, downloads and installs the AClimate R API from official repository CIAT-DAPA.

## Remove

The easiest way to remove the package is:

```
remove.packages("aclimaterapi")
```

# How to use

The following list are recommendations which should be take into account when you try to use the package.

## Import library

Once you have installed the library, you should import it in order to get access to all functions

```
library("aclimaterapi")
```

## Url of the Web API

The first thing that you have to identify is the url which is located the Web API. This parameter will be asked in all methods.

You can create a global variable with this url:

```
url_root = "https://webapi.aclimate.org/api/"
```

## Functions

### Get countries

The method **get\_geographic\_country** allows to users get a list of all countries.

```
df = get_geographic_country(url_root)
print(head(df))
```

### Get weather stations

The method **get\_geographic** allows to users get a list of all weather stations available in the system.

```
df = get_geographic(url_root)
print(head(df))
```

### Get agronomy setup

The method **get\_agronomy** allows to users get a list of cultivars and soils available into the AClimate platform.

```
df = get_agronomy(url_root)
print(df)
```

## Get climate forecast

The method **get\_forecast\_climate**, function which gets the forecast climate for a set of weather stations available into the AClimate platform.

You can find the ids of the weather stations in the method **get\_geographic**

```
stations=c("58504f1a006cb93ed40eebe2", "58504f1a006cb93ed40eebe3")
obj_f = get_forecast_climate(url_root, stations)
print(obj_f$probabilities)
print(obj_f$performance)
print(obj_f$scenarios)
```

## Get crop forecast

The method **get\_forecast\_crop**, function which gets the crop forecast for a set of weather stations available into the AClimate platform.

You can find the ids of the weather stations in the method **get\_geographic**

```
stations=c("58504f1a006cb93ed40eebe2", "58504f1a006cb93ed40eebe3")
df = get_forecast_crop(url_root, stations)
print(head(df))
```

# Developer

## Configure dev environment

Install the following packages

```
install.packages("devtools")
install.packages("roxygen2")
```

## Build package

Import libraries

```
library(devtools)  
library(roxygen2)
```

**Call the packages**

```
setwd("/path/source_code/")
```

**Build documentation**

```
devtools::document()
```

**Build package**

```
devtools::build()
```