1.2 - Introduction to medfate

Miquel De Cáceres, Victor Granda, Aitor Ameztegui

Ecosystem Modelling Facility

2022-06-13

Outline

1. Purpose and development context
2. Companion packages
3. Package installation
4. Overview of package functions

1. Purpose and development context

Model scope

Being able to anticipate the impact of global change on forest ecosystems is one of the major environmental challenges in contemporary societies.

1. Purpose and development context

Model scope

Being able to anticipate the impact of global change on forest ecosystems is one of the major environmental challenges in contemporary societies.

The R package **medfate** (ver. 2.7.4) has been designed to study the characteristics and simulate the functioning and dynamics of forest ecosystems.

1. Purpose and development context

Model scope

Being able to anticipate the impact of global change on forest ecosystems is one of the major environmental challenges in contemporary societies.

The R package **medfate** (ver. 2.7.4) has been designed to study the characteristics and simulate the functioning and dynamics of forest ecosystems.

Climatic conditions are the main environmental drivers, with a particular focus on drought impacts under Mediterranean conditions.

1. Purpose and development context

Model scope

Being able to anticipate the impact of global change on forest ecosystems is one of the major environmental challenges in contemporary societies.

The R package **medfate** (ver. 2.7.4) has been designed to study the characteristics and simulate the functioning and dynamics of forest ecosystems.

Climatic conditions are the main environmental drivers, with a particular focus on drought impacts under Mediterranean conditions.

Representation of vegetation accounts for structural and compositional variation but is not spatially-explicit (i.e. trees or shrubs do not have explicit coordinates within forest stands).

1. Purpose and development context

Development context

I have been intensively working on medfate since 2013, when I obtained a Ramon y Cajal research fellowship from the Spanish government. Two other research projects (FORESTCAST, DRESS) have funded developments.

1. Purpose and development context

Development context

I have been intensively working on medfate since 2013, when I obtained a Ramon y Cajal research fellowship from the Spanish government. Two other research projects (FORESTCAST, DRESS) have funded developments.

Developments were also supported by CTFC (until March 2021) and CREAF, where I currently coordinate its *Ecosystem Modelling Facility*.

1. Purpose and development context

Development context

I have been intensively working on medfate since 2013, when I obtained a Ramon y Cajal research fellowship from the Spanish government. Two other research projects (FORESTCAST, DRESS) have funded developments.

Developments were also supported by CTFC (until March 2021) and CREAF, where I currently coordinate its *Ecosystem Modelling Facility*.

A large number of people has contributed with *ideas*, *data* or *code* to the project:

- Jordi Martínez-Vilalta (CREAF-UAB, Spain)
- Maurizio Mencuccini (CREAF-ICREA, Spain)
- Juli G. Pausas (CIDE-CSIC, Spain)
- Pilar Llorens (CSIC, Spain)
- Rafa Poyatos (CREAF, Spain)
- Lluís Brotons (CREAF-CSIC, Spain)
- Antoine Cabon (WSL, Switzerland)
- Roberto Molowny (EMF-CREAF, Spain)
- Victor Granda (EMF-CREAF, Spain)
- Alicia Forner (MNCN-CSIC, Spain)
- Lluís Coll (UdL, Spain)
- Pere Casals (CTFC, Spain)
- Mario Beltrán (CTFC, Spain)
- Aitor Améztegui (UdL, Spain)
- Nicolas Martin-StPaul (INRA, France)
- Shengli Huang (USDA, USA)
- Enric Batllori (UB-CREAF, Spain)
- Santi Sabaté (UB-CREAF, Spain)
- Daniel Nadal-Sala (UB, Spain)
- …

2. Companion packages

During the development of **medfate** ancillary functions were originally included in the package itself…

2. Companion packages

During the development of **medfate** ancillary functions were originally included in the package itself…

.. but many of them were later moved into more specialized packages, which now evolve in parallel to the main package:

2. Companion packages

During the development of **medfate** ancillary functions were originally included in the package itself…

.. but many of them were later moved into more specialized packages, which now evolve in parallel to the main package:

1. Package **meteoland** (ver. 1.0.3) allows generating *daily weather input* for simulation models in medfate.

2. Companion packages

During the development of **medfate** ancillary functions were originally included in the package itself…

.. but many of them were later moved into more specialized packages, which now evolve in parallel to the main package:

1. Package **meteoland** (ver. 1.0.3) allows generating *daily weather input* for simulation models in medfate.

2. Package **medfateland** (ver. 0.4.2) extends medfate by allowing simulations to be performed in a *spatially explicit context*.

2. Companion packages

During the development of **medfate** ancillary functions were originally included in the package itself…

.. but many of them were later moved into more specialized packages, which now evolve in parallel to the main package:

1. Package **meteoland** (ver. 1.0.3) allows generating *daily weather input* for simulation models in medfate.

2. Package **medfateland** (ver. 0.4.2) extends medfate by allowing simulations to be performed in a *spatially explicit context*.

3. Package **medfateutils** (ver. 0.1.3) provides functions to help *initializing* vegetation, soil and species inputs.

3. Package installation and documentation

Installation

From **CRAN** (stable versions; now ver. **2.7.3**):

```
install.packages("medfate")
```

3. Package installation and documentation

Installation

From **CRAN** (stable versions; now ver. **2.7.3**):

```
install.packages("medfate")
```

From **GitHub** (frequent updates; now ver. **2.7.4**):

```
remotes::install_github("emf-creaf/medfate")
```

3. Package installation and documentation

Installation

From **CRAN** (stable versions; now ver. **2.7.3**):

```
install.packages("medfate")
```

From **GitHub** (frequent updates; now ver. **2.7.4**):

```
remotes::install_github("emf-creaf/medfate")
```

Later today we will also use packages **meteoland** and **medfateutils**:

```
install.packages("meteoland")
remotes::install_github("emf-creaf/medfateutils") # Not on CRAN
```

3. Package installation and documentation

Installation

From **CRAN** (stable versions; now ver. **2.7.3**):

```
install.packages("medfate")
```

From **GitHub** (frequent updates; now ver. **2.7.4**):

```
remotes::install_github("emf-creaf/medfate")
```

Later today we will also use packages **meteoland** and **medfateutils**:

```
install.packages("meteoland")
remotes::install_github("emf-creaf/medfateutils") # Not on CRAN
```

Documentation

Several vignettes are available at the package web page.

3. Package installation and documentation

Installation

From **CRAN** (stable versions; now ver. **2.7.3**):

```
install.packages("medfate")
```

From **GitHub** (frequent updates; now ver. **2.7.4**):

```
remotes::install_github("emf-creaf/medfate")
```

Later today we will also use packages **meteoland** and **medfateutils**:

```
install.packages("meteoland")
remotes::install_github("emf-creaf/medfateutils") # Not on CRAN
```

Documentation

Several vignettes are available at the package web page.

A complete documentation of the models included in the package can be found in medfate's reference book.

3. Overview of package functions

Simulation functions

Three main simulation models can be executed in medfate, each building on the preceding ones:

Function	Description
`spwb()`	Water and energy balance
`growth()`	Carbon balance, growth and mortality
`fordyn()`	Forest dynamics, including recruitment and forest management

3. Overview of package functions

Simulation functions

Three main simulation models can be executed in medfate, each building on the preceding ones:

Function	Description
`spwb()`	Water and energy balance
`growth()`	Carbon balance, growth and mortality
`fordyn()`	Forest dynamics, including recruitment and forest management

Functions `spwb()` and `growth()` have single-day counterparts `spwb_day()` and `growth_day()`.

3. Overview of package functions

Simulation functions

Three main simulation models can be executed in medfate, each building on the preceding ones:

Function	Description
`spwb()`	Water and energy balance
`growth()`	Carbon balance, growth and mortality
`fordyn()`	Forest dynamics, including recruitment and forest management

Functions `spwb()` and `growth()` have single-day counterparts `spwb_day()` and `growth_day()`.

Plot/summary functions

Specific `summary()`, `plot()` and `shinyplot()` functions are included to *extract* and *display* the time series included in the output of each simulation function.

3. Overview of package functions

Post-processing functions

Some package functions are meant to be used on simulation results (some of them implementing static ancillary models) and produce time series of additional properties.

3. Overview of package functions

Post-processing functions

Some package functions are meant to be used on simulation results (some of them implementing static ancillary models) and produce time series of additional properties.

Function	Description
`droughtStress()`	Plant drought stress indices
`waterUseEfficiency()`	Water use efficiency metrics
`resistances()`	Hydraulic resistances to water transport
`fireHazard()`	Potential fire behaviour

Other functions could be envisaged (e.g. light use efficiency) but have not been implemented.

3. Overview of package functions

Sub-model functions

A large number of functions implement sub-models (i.e. modules) on which the simulation functions are built. They are included in the package for the sake of transparency, but most users are not expected to use them directly.

Sub-model functions are grouped by *subject*:

3. Overview of package functions

Sub-model functions

A large number of functions implement sub-models (i.e. modules) on which the simulation functions are built. They are included in the package for the sake of transparency, but most users are not expected to use them directly.

Sub-model functions are grouped by *subject*:

Group	Description
`biophysics_*`	Physics and biophysics
`carbon_*`	Carbon balance
`fuel_*`	Fuel properties
`fire_*`	Fire behaviour
`hydraulics_*`	Plant hydraulics
`hydrology_*`	Canopy and soil hydrology
`light_*`	Light extinction and absortion
`moisture_*`	Live tissue moisture

3. Overview of package functions

Sub-model functions

A large number of functions implement sub-models (i.e. modules) on which the simulation functions are built. They are included in the package for the sake of transparency, but most users are not expected to use them directly.

Sub-model functions are grouped by *subject*:

Group	Description
`biophysics_*`	Physics and biophysics
`carbon_*`	Carbon balance
`fuel_*`	Fuel properties
`fire_*`	Fire behaviour
`hydraulics_*`	Plant hydraulics
`hydrology_*`	Canopy and soil hydrology
`light_*`	Light extinction and absortion
`moisture_*`	Live tissue moisture

Group	Description
`pheno_*`	Leaf phenology
`photo_*`	Leaf photosynthesis
`root_*`	Root distribution and conductance calculations
`soil_*`	Soil hydraulics and thermodynamics
`transp_*`	Stomatal regulation, transpiration and photosynthesis
`wind_*`	Canopy turbulence

M.C. Escher - Reptiles, 1943