

# Package: RGN (via r-universe)

September 19, 2024

**Type** Package

**Title** Robust-Gauss Newton (RGN) Optimization of Sum-of-Squares Objective Function

**Version** 1.0.0

**Description** Implementation of the Robust Gauss-Newton (RGN) algorithm, designed for solving optimization problems with a sum of least squares objective function. For algorithm details please refer to Qin et. al. (2018) <[doi:10.1029/2017WR022488](https://doi.org/10.1029/2017WR022488)>.

**License** MIT + file LICENSE

**URL** <https://github.com/ClimateAnalytics/RGN/>

**BugReports** <https://github.com/ClimateAnalytics/RGN/issues>

**Depends** R (>= 3.5.0)

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Encoding** UTF-8

**LazyData** true

**NeedsCompilation** yes

**RoxygenNote** 7.2.3

**Author** David McInerney [cre, aut] (<<https://orcid.org/0000-0003-4876-8281>>), Michael Leonard [aut] (<<https://orcid.org/0000-0002-9519-3188>>), Dmitri Kavetski [aut] (<<https://orcid.org/0000-0003-4966-9234>>), Youwei Qin [aut] (<<https://orcid.org/0000-0001-5425-539X>>), George Kuczera [aut]

**Maintainer** David McInerney <dmcinern@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-09-25 14:22:27 UTC

## Contents

BassRiverData . . . . .	2
rgn . . . . .	2
simFunc_hymod . . . . .	4
<b>Index</b>	<b>6</b>

---

BassRiverData	<i>Hydrological data for Bass River catchment in Victoria, Australia</i>
---------------	--

---

### Description

Streamflow, rainfall and PET data for Bass River catchment (227219) in Victoria, Australia. Originally obtained from Francis Chiew.

### Usage

```
data(BassRiver)
```

### Format

List containing numerical vectors for precipitation (Rain.mm), potential evapotranspiration (ET.mm), and runoff (Runoff.mm.day), and date vector (Date)

### References

<https://github.com/eachonly/Robust-Gauss-Newton-Algorithm>, <http://www.bom.gov.au/water/hrs/>

---

rgn	<i>Robust Gauss Newton optimization</i>
-----	---

---

### Description

rgn performs optimization of weighted-sum-of-squares (WSS) objective function using the Robust Gauss Newton algorithm

**Usage**

```

rgn(
  simFunc,
  simTarget = 0,
  weights = NULL,
  par,
  lower,
  upper,
  control = NULL,
  ...
)

```

**Arguments**

simFunc	is a function that simulates a (vector) response, with first argument the vector of parameters over which optimization is performed
simTarget	is the target vector that simFunc is trying to match
weights	is a vector of weights used in the WSS objective function. Defaults to equal weights.
par	is the vector of initial parameters
lower	is the lower bounds on parameters
upper	is the upper bounds on parameters
control	list of RGN settings <ul style="list-style-type: none"> <li>• control\$n.multi is number of multi-starts (i.e. invocations of optimization with different initial parameter estimates). Default is 1.</li> <li>• control\$iterMax is maximum iterations. Default is 100.</li> <li>• control\$dump is level of diagnostic outputs between 0 (none) and 3 (highest). Default is 0.</li> <li>• control\$keep.multi (TRUE/FALSE) controls whether diagnostic output from each multi-start is recorded. Default is FALSE.</li> <li>• control\$logFile is log file name</li> </ul>
...	other arguments to simFunc()

**Details**

rgn minimizes the objective function  $\text{sum}(\text{weights} * (\text{simFunc} - \text{simTarget})^2)$ , which is a sum of squared weighted residuals ( $\text{residuals} = \text{weights} * (\text{simFunc} - \text{simTarget})$ ). Note simFunc corresponds to the vector of residuals when default arguments for simTarget and weights are used.

**Value**

List with

- par, the optimal parameters
- value, the optimal objective function value

- sim, the simulated vector using optimal parameters
- residuals, the vector of residuals using optimal parameters
- counts, the total number of function calls
- convergence, an integer code indicating reason for completion. 1 maximum iterations reached, 2 relative reduction in function value small. 3 absolute reduction in function value small 4 relative change in parameters small

## Examples

```
# Example 1: Rosenbrock
simFunc_rosenbrock=function(x) c(1.0-x[1],10.0*(x[2]-x[1]**2))
rgnOut = rgn(simFunc=simFunc_rosenbrock,
             par=c(-1.0, 0.0), lower=c(-1.5, -1.0), upper=c( 1.5, 3.0),
             simTarget=c(0,0))
rgnOut$par #optimal parameters
rgnOut$value #optimal objective function value

# Example 2: Hymod

data("BassRiver") # load Bass River hydrological data
rgnOut = rgn(simFunc=simFunc_hymod,
             par=c(400.,0.5,0.1,0.2,0.1),
             lower=c(1.,0.1,0.05,0.000001,0.000001),
             upper=c(1000.,2.,0.95,0.99999,0.99999),
             simTarget=BassRiverData$Runoff.mm.day[365:length(BassRiverData$Date)],
             stateVal=c(100.0,30.0,27.0,25.0,30.0,0.0,0.0,0.0), # initial states for hymod
             nWarmUp=365, # warmup period
             rain=BassRiverData$Rain.mm, # precip input
             pet=BassRiverData$ET.mm) # PET input
rgnOut$par #optimal parameters
rgnOut$value #optimal objective function value
```

---

simFunc_hymod	<i>hymod simulation</i>
---------------	-------------------------

---

## Description

Simulation of hymod rainfall-runoff model

## Usage

```
simFunc_hymod(
  x,
  rain,
  pet,
  nWarmUp,
  stateVal = c(100, 30, 27, 25, 30, 0, 0, 0)
)
```

**Arguments**

x	parameter values
rain	precipitation input (mm/day)
pet	potential evapotranspiration (mm/day)
nWarmUp	length of warmup period
stateVal	(optional) initial states

**Value**

Vector of simulated runoff

# Index

## \* datasets

BassRiverData, [2](#)

BassRiverData, [2](#)

rgn, [2](#)

simFunc\_hymod, [4](#)