

# Package ‘Pijavski’

October 22, 2020

**Type** Package

**Title** Global Univariate Minimization

**Version** 1.0.3

**Date** 2020-10-19

**Maintainer** Gleb Beliakov <gleb@deakin.edu.au>

**Author** Gleb Beliakov [aut, cre],  
Gita Das [aut],  
Jonathan Wells [ctb],  
Hewlett-Packard Company [ctb],  
Silicon Graphics Computer Systems Inc. [ctb]

**Description** Global univariate minimization of Lipschitz functions is performed by using Pijavski method, which was published in Pijavski (1972) <DOI:10.1016/0041-5553(72)90115-2>.

**License** LGPL-3

**LazyData** TRUE

**Imports** Rcpp (>= 0.12.3)

**LinkingTo** Rcpp

**RoxygenNote** 5.0.1

**NeedsCompilation** yes

**Copyright** Implementation of the binary heap is by Hewlett-Packard Company (1994), Silicon Graphics Computer Systems, Inc.(1996-1999). Modifications to the code by Johnathan Wells (2002)

**Repository** CRAN

**Date/Publication** 2020-10-22 21:40:09 UTC

## R topics documented:

Pijavski . . . . .	2
Pijavski.example . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

---

Pijavski

*Global univariate minimization of Lipschitz functions using Pijavski method*


---

### Description

Pijavski performs global univariate optimization of a Lipschitz function `fn`. The return value is a list containing `x`, `val=fn(x)`, precision reached and number of iterations made. Pijavski code in C++ is being called from R environment with multiple arguments

### Usage

```
Pijavski( fn, Lips, a, b, iter, prec, env)
```

### Arguments

<code>fn</code>	input, a pointer to the objective function <code>fn</code>
<code>Lips</code>	input, an overestimate of the Lipschitz constant of <code>fn</code>
<code>a,b</code>	input, left and right boundaries of the interval of minimization
<code>iter</code>	input and output, the maximum number of function evaluations, on return, the number of iterations made
<code>prec</code>	input and output, the desired precision in terms of the value of <code>fn</code> , on return the difference between best <code>fn</code> and the lower estimate on the minimum. If negative, the Lipschitz constant is too small
<code>env</code>	input, environment variable passed from R containing a reference to <code>fn</code> , should be defined as <code>new.env(list(fn = myfunction))</code>

### Value

A list with components

<code>x</code>	The global minimizer of <code>fn</code> .
<code>value</code>	The final value of the function being optimized.
<code>precision</code>	The precision of the result in terms of the difference of <code>value</code> and the lower estimate on <code>fn</code> .
<code>iterations</code>	Number of function evaluations performed.

### Author(s)

Gleb Beliakov and Gita Das

**Examples**

```
optimize_funcR <- function(x,y){
  y <- x * x
  return(y)
}

output<-Pijavski(optimize_funcR, 5, -2.0, 1.0, 1000, 10^-3,
  new.env(list(fn = optimize_funcR)))
output

# named parameters
output<-Pijavski(fn= optimize_funcR, Lips=4, a=-2.0, b=1.0,
  iter=1000, prec=10^-3, env=new.env(list(fn = optimize_funcR)))

output
```

---

Pijavski.example

*Illustrates using Pijavski method*

---

**Description**

Pijavski.example illustrates using Pijavski algorithm

**Usage**

```
Pijavski.example()
```

**Author(s)**

Gleb Beliakov and Gita Das

# Index

\* **optimize**

Pijavski, [2](#)

Pijavski.example, [3](#)

Pijavski, [2](#)

Pijavski.example, [3](#)