Package ‘JacobiEigen’

March 10, 2019

Type Package

Title Classical Jacobi Eigenvalue Algorithm

Version 0.3-3

Date 2019-03-10

Author Bill Venables

Imports Rcpp

Maintainer Bill Venables <Bill.Venables@gmail.com>

Description Implements the classical Jacobi algorithm for the eigenvalues and eigenvectors of a real symmetric matrix, both in pure 'R' and in 'C++' using 'Rcpp'. Mainly as a programming example for teaching purposes.

License GPL (>= 2)

LinkingTo Rcpp

Suggests stats, knitr, dplyr, tidyr, ggplot2, rbenchmark

VignetteBuilder knitr

NeedsCompilation yes

RoxygenNote 6.1.1

Repository CRAN

Date/Publication 2019-03-10 07:12:44 UTC

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jacobi</td>
<td>2</td>
</tr>
<tr>
<td>JacobiR</td>
<td>3</td>
</tr>
<tr>
<td>JacobiS</td>
<td>4</td>
</tr>
</tbody>
</table>

Index 5
The Jacobi Algorithm using Rcpp

Description

The Classical Jacobi Algorithm

Usage

Jacobi(x, symmetric = TRUE, only.values = FALSE, eps = 0)

Arguments

x
A real symmetric matrix

symmetric
a logical value. Is the matrix symmetric? (Only symmetric matrices are allowed.)

only.values
A logical value: do you want only the eigenvalues?

eps
an error tolerance. 0.0 implies .Machine$double.eps and sqrt(.Machine$double.eps)
if only.values = TRUE

Details

Eigenvalues and optionally, eigenvectors, of a real symmetric matrix using the classical Jacobi algorithm, (Jacobi, 1854)

Value

a list of two components as for base::eigen

Examples

V <- crossprod(matrix(rnorm(40, -1, 1), 8))
Jacobi(V)
identical(Jacobi(V), JacobiR(V))
all.equal(Jacobi(V)$values, base::eigen(V)$values)
Description

The Jacobi Algorithm

Usage

`JacobiR(x, symmetric = TRUE, only.values = FALSE, eps = if (!only.values) .Machine$double.eps else sqrt(.Machine$double.eps))`

Arguments

- `x`: a real symmetric matrix
- `symmetric`: a logical value. Is the matrix symmetric? (Only symmetric matrices are allowed.)
- `only.values`: A logical value: Do you want only the eigenvalues?
- `eps`: a small positive error tolerance

Details

Eigenvalues and optionally, eigenvector of a real symmetric matrix using the classical Jacobi algorithm, (Jacobi, 1854)

Value

a list of two components as for `base::eigen`

Examples

```r
V <- crossprod(matrix(rnorm(25), 5))
JacobiR(V)
identical(Jacobi(V), JacobiR(V))
all.equal(Jacobi(V)$values, base::eigen(V)$values)
```
Description

The Classical Jacobi Algorithm with a stagewise protocol

Usage

```r
Jacobis(x, symmetric = TRUE, only.values = FALSE, eps = 0)
```

Arguments

- `x` A real symmetric matrix
- `symmetric` A logical value. Is the matrix symmetric? (Only symmetric matrices are allowed.)
- `only.values` A logical value: do you want only the eigenvalues?
- `eps` an error tolerance. 0.0 implies `.Machine$double.eps` and `sqrt(.Machine$double.eps)` if `only.values = TRUE`

Details

Eigenvalues and optionally, eigenvectors, of a real symmetric matrix using the classical Jacobi algorithm, (Jacobi, 1846) using a stagewise rotation protocol

Value

a list of two components as for `base::eigen`

Examples

```r
V <- crossprod(matrix(runif(40, -1, 1), 8))
Jacobis(V)
all.equal(Jacobis(V)$values, Jacobi(V)$values)
zapsmall(crossprod(Jacobis(V)$vectors, Jacobi(V)$vectors))
```
Index

Jacobi, 2
JacobiR, 3
JacobiS, 4