Announcements

WEEK-BY-WEEK

Week 1: Getting Started and R Nuts and Bolts

Week 2: Programming with R

Week 3: Loop Functions and Debugging

Week 4: Simulation and Profiling

LECTURES

Lecture Videos

Lecture Slides (link to GitHub)

EXERCISES

Quizzes

Programming Assignments 1, 3, & swirl

Programming Assignment 2

ABOUT THE COURSE

Syllabus

About the Instructor

COMMUNITY

DSS Community Site

Discussion Forums

Course Wiki

Help Articles

Forums / Programming Assignment 2 (peer assessment): Lexical Scoping

Help

Testing cacheSolve:verify that cacheSolve works and will actually store the inverse

Subscribe for email updates.

No tags yet. + Add Tag

Sort replies by: Oldest first Newest first Most popular

Bryan Signature Track · 7 days ago

I am not sure how helpful this will be for people but for me, it helped bring home the usefulness of implementing cacheSolve. I made a set of test scripts containing the following:

```
## first script: test cacheSolve repeatedly
x <- matrix(rnorm(160000),400,400)
x_ <- makeCacheMatrix(x)
for (i in 1:1000) {
  x__ <- cacheSolve(x_)
}
```

```
## second script: test solve repeatedly
x <- matrix(rnorm(160000),400,400)
for (i in 1:1000) {
  x_ <- solve(x)
}
```

```
## third script: verify equality
x <- matrix(rnorm(160000),400,400)
x_ <- makeCacheMatrix(x)
cacheSolve(x_)
print(identical(x_$getInv(),solve(x)))
```

Using system.time, I have verified that the time elapsed on calling cacheSolve, for example, on a 400 by 400 cached matrix 1000 times in a loop is about .1 second (assumes that there is no output in cacheSolve, as there was in cachemean). The time elapsed on calling solve on a 400 by 400 non cached matrix 1000 times in a loop is more like 69 seconds. The third script prints out TRUE, as expected.

9 · flag

Dmitry Grekov · 6 days ago

Good job, should be of great help.

I'd also suggest making one script which runs your scripts #1 and #2, measures time for both cases and provides the result in a readable form.

0 · flag

+ Comment

Anders Monrad Signature Track · 3 days ago

Nice job, thanks

0 · flag

+ Comment

Yeo Sze Wee Signature Track · 3 days ago

I use the following script to test...

```
B = matrix(4:7,nrow=2,ncol=2)
fList <- makeCacheMatrix(B)
cacheSolve(fList) #first call, this will compute the inverse
cacheSolve(fList) #second call, this will get the result from the cache
```

1 · flag

+ Comment

New post

To ensure a positive and productive discussion, please read our forum posting policies before posting.

B I Link <code> Pic Math Edit: Rich Preview

Make this post anonymous to other students

Subscribe to this thread at the same time

Add post