

## Google Groups

---

### Re: [D-RUG]: using mclapply in shiny app

---

Alex Mandel

Nov 2, 2017 5:17 PM

Posted in group: **Davis R Users' Group**

Aside from what Noam pointed out.

You might want to detect the number of cores actually available, and vary based on that. I like `cores-1` to keep the machine happy. Also in `mclapply` there is an arg called `mc.preschedule` that you might want to toggle for this use case.

Last, if you want to guarantee performance, you need to run your own Shiny server. Which is not actually that hard, though I don't think you can use AWS EBS with it.

Enjoy,  
Alex Mandel, PhD

Center for Spatial Sciences  
<http://spatial.ucdavis.edu>  
Geospatial and Farming Systems Research Consortium  
<http://gfc.ucdavis.edu>  
University of California, Davis

On 10/30/2017 05:02 PM, Noam Ross wrote:

```
> If you're running this on shinyapps.io, you aren't guaranteed more than a
> single core to run on. From the manual (
> http://docs.rstudio.com/shinyapps.io/applications.html):
>
>> "By default, shinyapps.io deploys all applications on 'medium' instances,
> which are allowed to use 512 MB of memory.....shinyapps.io does not make
> any guarantees regarding the number of CPU cores, or the speed of the CPUs
> that are allocated to the deployed applications."
>
> My guess is that shinyapp.io is spinning up small Amazon or other cloud
> instances when applications fire up, and they're grabbing whatever the
> lowest-cost instance is that's available, which could be of varied number
> of cores or memory. If you specify more cores than are available, it'll
> work, but you'll be using up memory resources you don't need to by copying
> the objects in memory into different processes. It's possible that for
> your larger, more complex shapefiles you are hitting memory limits as a
> result, and those processes are failing so you are only getting back a
> subset of the results. If you just use lapply() it might be slow, but
> you'll be more likely to get all the results back.
>
> Unless you're using Shiny on high-power servers, you'll probably have
> *fewer* resources in the cloud than on your machine. Even the Pro accounts
> for shinyapps.io aren't so much designed for compute-intense jobs so much
```

> as applications that are being accessed by lots of people at once and have  
> to be scaled up to many small servers.  
>  
> On Fri, Oct 27, 2017 at 6:02 PM Jens Stevens <jtst...@ucdavis.edu> wrote:  
>  
>> This question may be specific to folks who have used shiny before...  
>> I have written an app that reads in a shapefile, and does a series of  
>> internal buffering operations on that shapefile, calculating the resulting  
>> areas. Because the buffering operation is fairly computationally intensive,  
>> I have been using mclapply to buffer at different distances in parallel,  
>> which speeds up the processing time. However I seem to be having issues  
>> with migrating this process to shiny- it works properly on my own machine  
>> but on the shiny server it only accomplishes the first 1-2 distances (out  
>> of 5-10 or more possible distances), and sometimes returns the resulting  
>> areas in different orders for different runs on the shiny server. This  
>> problem seems to occur for large complex shapefiles but not for smaller  
>> simpler shapefiles. I think that the issue has to do with how mclapply is  
>> implemented on the shinyapps server but I have no idea how to troubleshoot  
>> this.  
>>  
>> The attached script is a distilled version of the code with everything but  
>> the core operations stripped away. It should be reproducible, and the  
>> attached shapefile can be used to test it. When the app is modified and  
>> synced to shinyapps.io the resulting webpage is  
>> [https://stevensjt.shinyapps.io/toy\\_app/](https://stevensjt.shinyapps.io/toy_app/) , however if you would like to  
>> test it you would need to sync to your own shinyapps account, or somehow  
>> get a token to sync to my account (not sure about this).  
>>  
>> When I sync to the shinyapps server, I modify the code in mclapply to  
>> specify 16 cores instead of 4 (which I specify when running the app on my  
>> local machine).  
>>  
>> Ideas as to where the bug might be?  
>>  
>> --  
>> Check out our R resources at <http://d-rug.github.io/>  
>> ---  
>> You received this message because you are subscribed to the Google Groups  
>> "Davis R Users' Group" group.  
>> To unsubscribe from this group and stop receiving emails from it, send an  
>> email to [davis-rug+...@googlegroups.com](mailto:davis-rug+...@googlegroups.com).  
>> Visit this group at <https://groups.google.com/group/davis-rug>.  
>> For more options, visit <https://groups.google.com/d/optout>.  
>>  
>