Stat243 : Problem Set 1

Thibault Doutre

September 11, 2015

Sometimes, I had to split lines of code. Be sure to copy line after line before pasting it into the bash shell.

1 Problem 1

1.1 Part a

Load the data into a tar.gz file using wget.

```
wget -O test.tar.gz "http://data.un.org/Handlers/Download
Handler.ashx?DataFilter=itemCode:526&DataMartId=FAO&Format
=csv&c=2,3,4,5,6,7&s=countryName:asc,elementCode:asc,year
:desc"
```

Decompress the data and rename it.

```
tar -zvxf test.tar.gz
mv UNdata_Export_20150902_002111523.csv apricots.csv
```

Regions of the world have a "+" at the end of their names. In order to make the distinction between them and single countries is :
Select lines containing + and put them into a file while displaying the content of the file using tee.

```
grep .+ apricots.csv | tee data_areas.txt
```

Select lines without + using the -v flag — remove the last 7 lines containing the legend — remove the first line which is useless — use tee command to put this into a file and display it

```
grep -v .+ apricots.csv | ghead -n -7 | sed '1d' | tee dat
a_countries.txt
```

Subset data according to year 2005 using awk. Make a condition over the 4th colum of the data.

```
awk -F'[,]' '$4~"2005"' data_countries.txt
```

Select a subset of data corresponding to the year 2005 — select lines containing "Area Harvested" — remove quotation marks in all the data — sort by numerical value the 6th column (decreasing) — select the top 5 countries — only display names

```
awk -F'[,]' '$4~"2005"' data_countries.txt | grep "Area
Harvested" | sed 's/\\"//g' | sort -t"," -k6 -n -r | head
-n 5 | cut -d',' -f1
```

Creating script with vim editor

```
vim scriptA.sh
```

Content of the script :

- For every year listed below, display it then display the top five countries found as described above.

```
for i in 1965 1975 1985 1995 2005
do
        echo $i
        echo $(awk -F'[,]' '$4~'$i'' data_countries.txt |
grep "Area Harvested" | sed 's/\\"//g' | sort -t"," -k6
-n -r | head -n 5 | cut -d',' -f1)
done
```

Execute the script with the bash command.

```
bash scriptA.sh
```

Here are the rankings for the years 1965, 1975, 1985, 1995 and 2005 :

1965 - USSR Turkey United States of America Spain Tunisia
1975 - USSR Turkey Spain Tunisia Italy
1985 - Turkey USSR Spain Tunisia Italy
1995 - Turkey Spain Ukraine Tunisia Russian Federation
2005 - Turkey Pakistan Uzbekistan Algeria Spain

1.2 Part b

Creating the script

```
vim Bscript.sh
```

Content of the script :

- Download the file with wget and put it into a tar.gz file
- Decompress the file
- Get the name of the file and display it on the screen
- Delete both files in order to avoid keeping every single file.
- This ensure that there is only one file beginning with "UNdata". It is convenient for the grep command.

```
#!/bin/bash
function foo1
{
wget -O test0.tar.gz "http://data.un.org/Handlers/Download
Handler.ashxDataFilter=itemCode:$1&DataMartId=FAO&Format=c
sv&c=2,3,4,5,6,7&s=countryName:asc,elementCode:asc,year:de
sc"

tar -zxvf  test0.tar.gz

ls UNdata_* | xargs cat

ls UNdata_* | xargs rm

rm test0.tar.gz
}
```

Display avocados.csv from the script for example (572)

```
source Bscript.sh ; foo1 572
```

1.3 Part c

Open script

```
vim Dscript.sh
```

Content of the Script :

- Download the meta data from the FAO website.
- Grep for the only line who matters here i.e. the one containing the correspondence between numbers and the names of the files.
- Delete everything after the keyword because it is useless.
- Take the 3 last characters using tail.
- Then, call foo1 function from Bscript.sh and run it with the good corresponding number.

```
#!/bin/bash

getData(){


local locvar=$(curl "http://data.fao.org/dimension-member?
entryId=8462e2d6-ba52-492c-a0a7-e08893a59aac" | grep ">Def
ault composition" | sed -n -e 's/ '$1'.*//p' | tail -c
4)

source Bscript.sh
foo1 $locvar > $1.txt
}
```

Run the function getData() with Avocados as an example

```
source Dscript.sh ; getData Avocados
```

2 Problem 2

Create the script

```
vim Cscript.sh
```

Content of the script :

How to get names of the files :

- Download URL
- Look for text files
- Remove part before "href="
- Remove part after ".txt"

How to download them :

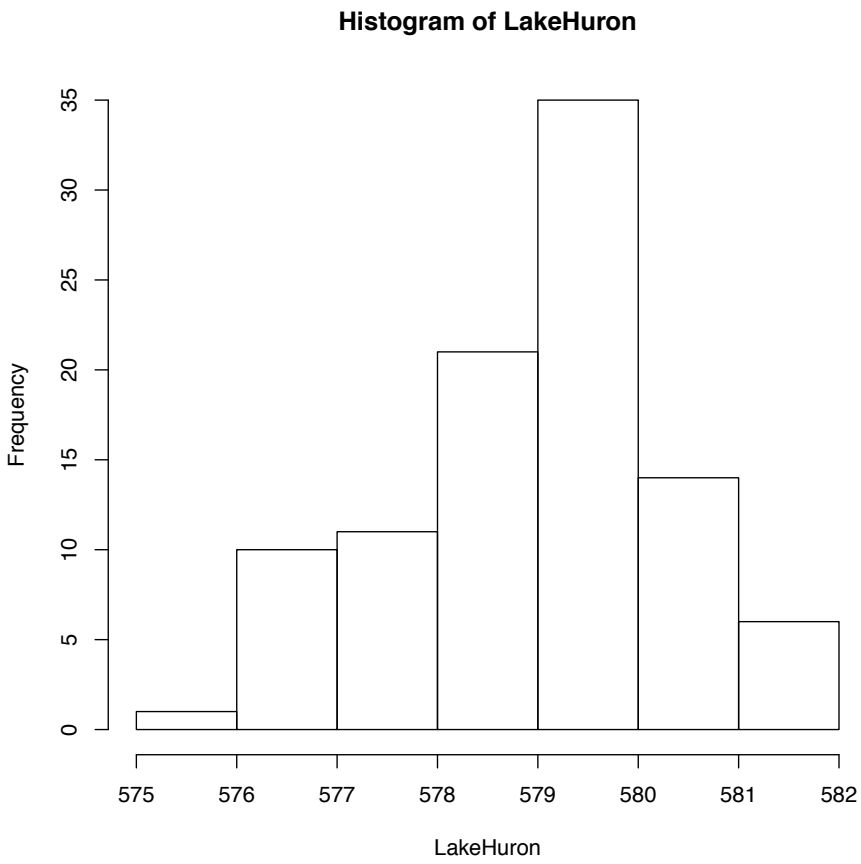
- For every name of the text files we want to download, download it by adding the name at the end of the corresponding URL.
- Display message which shows the name of the file

```
#!/bin/bash
for a in $(curl http://www1.ncdc.noaa.gov/pub/data/ghcn/
daily/ | grep .'\.txt' | sed -n -e 's/.*href=\" *//p' | s
ed -n -e 's/\">.*//p')
do

curl -o $a "http://www1.ncdc.noaa.gov/pub/data/ghcn/daily
/"$a

echo "Download [file :\ "$a"] in progress..."
done
\end{lstlisting}
Run script
\begin{lstlisting}[frame=single]
bash Cscript.sh
```

3 Problem 3

The height of the water level in Lake Huron fluctuates over time. Here I analyze the variation using R. I show a histogram of the lake levels for the period 1875 to 1972.

```
hist(LakeHuron)
```




```r
lowHi <- c(which.min(LakeHuron), which.max(LakeHuron))
yearExtrema <- attributes(LakeHuron)$tsp[1]-1 + lowHi
```