Cran download logs aggregation time summary

A.Birek, M.Kosiński, N.Ryciak, W.Ryciuk

May 11, 2014

Figure 1: Powered by ! https://github.com/MarcinKosinski/AlmostBigData

CHAPTER ONE

DOWNLOADING DATA

Syntax used for downloading, unzipping and merging data is available in section A.1. More or less downloading looked like this and took about:

```
start <- as.Date("2012-10-01")
today <- as.Date("2014-05-10")
all_days <- seq(start, today, by = "day")
year <- as.POSIXlt(all_days)$year + 1900
urls <- paste0("http://cran-logs.rstudio.com/", year, "/", all_days, ".csv.gz")

destdir <- "D:/bd1/AlmostBigData/cran-logs/"
n <- length(urls)
i = 1
for (i in 1:n) {
    destfile <- stri_paste(destdir, as.character(all_days[i]))
    download.file(urls[i], destfile)
}
```

Unzipping files syntax looked like this and took:

```
lok <- "D:/bd1/AlmostBigData"
gzpath <- character(n)
i <- 1
for (i in 1:n) {
    gzpath[i] <- paste(lok, "/cran-logs", all_days[i], sep = "")
}
install.packages("R.utils")
library(R.utils)
for (i in 1:n) {
    gunzip(gzpath[i], destname = paste(gzpath[i], ".csv"), remove = TRUE)
}
```

Converting CSV files with proper delimiter syntax looked like this and time spent was:

```
for (i in 1:n) {
    write.csv2(read.csv2(paste(gzpath[i], ".csv"), sep = ","), paste(gzpath[i], "_new.csv"))
}
```

CHAPTER
TWO

SAS PATH

Syntax used for importing, merging and summarizing data is available in chapter B.

2.1 Importing data

Importing `csv` files into **SAS** syntax looked like this and took:

```
proc import datafile='D:/bd1/AlmostBigData/cran-logs2012-10-01 _new.csv'
out=CR.cran1 dbms=csv replace;
      delimiter = ';';
      getnames=yes;
      run;

...

proc import datafile='D:/bd1/AlmostBigData/cran-logs2014-05-09 _new.csv'
out=CR.cran586 dbms=csv replace;
      delimiter = ';';
      getnames=yes;
      run;
```

2.2 Merging files

Merging all those files syntax looked like this and time expired was:

```
data Cr.DANE;
set
CR.cran1,
CR.cran2,
....
CR.cran586;
run;
```

2.3 Summary for each variable

Summaries of each variable syntax looked like this and time expired was:

```
12   proc summary data=Cr.DANE2 print;
13   class package;
14   run;

NOTE: There were 41611796 observations read from the data set CR.DANE2.
NOTE: PROCEDURE SUMMARY used (Total process time):
      real time           29.82 seconds
      cpu time            20.06 seconds
```

```
15
16   proc summary data=Cr.DANE2 print;
17   class version;
18   run;

NOTE: There were 41611796 observations read from the data set CR.DANE2.
NOTE: PROCEDURE SUMMARY used (Total process time):
      real time           31.23 seconds
      cpu time            20.18 seconds


19
20   proc summary data=Cr.DANE2 print;
21   class r_arch;
22   run;

NOTE: There were 41611796 observations read from the data set CR.DANE2.
NOTE: PROCEDURE SUMMARY used (Total process time):
      real time           30.65 seconds
      cpu time            10.12 seconds


23
24   proc summary data=Cr.DANE2 print;
25   class r_os;
26   run;

NOTE: There were 41611796 observations read from the data set CR.DANE2.
NOTE: PROCEDURE SUMMARY used (Total process time):
      real time           30.59 seconds
      cpu time            12.58 seconds


27
28   proc summary data=Cr.DANE2 print;
29   class r_version;
30   run;

NOTE: There were 41611796 observations read from the data set CR.DANE2.
NOTE: PROCEDURE SUMMARY used (Total process time):
      real time           30.56 seconds
      cpu time            10.95 seconds


31
32   proc summary data=Cr.DANE2 print;
33   class country;
34   run;

NOTE: There were 41611796 observations read from the data set CR.DANE2.
NOTE: PROCEDURE SUMMARY used (Total process time):
      real time           30.07 seconds
      cpu time            12.48 seconds
```

2.4 Frequency tables

2.4.1 r os

Frequency tables syntax and the time expired for `r os`:

```
    proc freq data=Cr.dane2;
    tables r_os;
    run;

NOTE: Writing HTML Body file: sashtml.htm
NOTE: There were 41611796 observations read from the data set CR.DANE2.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           32.60 seconds
      cpu time            5.46 seconds
```

2.4.2 Packages

Frequency tables syntax and the time expired for `packages`, grouped by `r os`:

```
   proc freq data=Cr.dane2 page;
   by r_os;
   tables package;
   run;

NOTE: There were 41611796 observations read from the data set CR.DANE2.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           53.25 seconds
      cpu time            32.46 seconds
```

CHAPTER THREE

TRADITIONAL $\mathcal{R}$ PATH

3.1 Unmerged $\mathcal{R}$ files Path

3.2 Merged $\mathcal{R}$ files Path

CHAPTER

FOUR

RCPP PATH

APPENDIX

A

PREPARING REPORT

A.1 Data download, unzipp, conversion syntax

```
start <- as.Date("2012-10-01")
today <- as.Date("2014-05-10")
all_days <- seq(start, today, by = "day")
year <- as.POSIXlt(all_days)$year + 1900
urls <- paste0("http://cran-logs.rstudio.com/", year, "/", all_days, ".csv.gz")

destdir <- "D:/bd1/AlmostBigData/cran-logs/"
n <- length(urls)
i = 1
for (i in 1:n) {
    destfile <- stri_paste(destdir, as.character(all_days[i]))
    download.file(urls[i], destfile)
}

lok <- "D:/bd1/AlmostBigData"
gzpath <- character(n)
i <- 1
for (i in 1:n) {
    gzpath[i] <- paste(lok, "/cran-logs", all_days[i], sep = "")
}
install.packages("R.utils")
library(R.utils)
for (i in 1:n) {
    gunzip(gzpath[i], destname = paste(gzpath[i], ".csv"), remove = TRUE)
}

for (i in 1:n) {
    write.csv2(read.csv2(paste(gzpath[i], ".csv"), sep = ","), paste(gzpath[i], "_new.csv"))
}
```

APPENDIX

B

SAS SYNTAX

```
proc import datafile='D:/bd1/AlmostBigData/cran-logs2012-10-01 _new.csv'
out=CR.cran1 dbms=csv replace;
      delimiter = ';';
      getnames=yes;
      run;

...

proc import datafile='D:/bd1/AlmostBigData/cran-logs2014-05-09 _new.csv'
out=CR.cran586 dbms=csv replace;
      delimiter = ';';
      getnames=yes;
      run;

data Cr.DANE;
set
CR.cran1,
CR.cran2,
....
CR.cran586;
run;

proc summary data=Cr.DANE print;
class package;
run;

proc summary data=Cr.DANE print;
class version;
run;

proc summary data=Cr.DANE print;
class r_arch;
run;

proc summary data=Cr.DANE print;
class r_os;
run;

proc summary data=Cr.DANE print;
class r_version;
run;

proc summary data=Cr.DANE print;
class country;
run;
```

```
proc freq data=Cr.dane2 page;
tables r_os;
run;


proc freq data=Cr.dane2 page;
by r_os;
tables package;
run;
```