STAT 243: Problem Set 2

Jin Rou New

September 21, 2014

1 Problem 1

First open an input file connection to the `bz2` file using `bzfile`, then read in the first block of data with `read.csv`. Subset the data according to the given column index and subset value, then stratify the data with `split` and output stratified data to respective `bz2` files by opening output file connections using again `bzfile` and writing to the connections with `read.table`. Close output file connections, then read in the next block of data. While this reading step does not produce an error, repeat previous data processing steps.

```
#!/usr/bin/Rscript

# Parse arguments
args <- commandArgs(TRUE)
stopifnot(length(args) == 4)
data_filepath <- args[1]
var_stratify <- as.integer(args[2]) # Column index to stratify on
var_subset <- as.integer(args[3]) # Column index to subset on
value_subset <- as.character(args[4]) # Value to subset on

output_dir <- "output"
dir.create(output_dir, showWarnings = FALSE) # Set up output directory
con <- bzfile(data_filepath, "rt") # Open input file connection (for reading in text mode)
nrows_block <- 50000 # Read from input file in blocks of size nrows_block
header <- read.csv(con, header = FALSE, nrows = 1)
data <- read.csv(con, header = FALSE, nrows = nrows_block) # Read in first block
# Read and process blocks while there are still rows to be read and processed
while (!class(data) == "try-error") {
  data_subset <- data[as.character(data[, var_subset]) == value_subset, ] # Subset data
  data_strata_list <- split(data_subset, data_subset[, var_stratify]) # Stratify data
  for (stratum in names(data_strata_list)) { # Output data strata to respective bz2 files
    data_filepath_out <- file.path(output_dir, paste0(stratum, ".csv.bz2"))
    con_out <- bzfile(data_filepath_out, "at") # Open output file connection (for appending)
    write.table(data_strata_list[[stratum]], row.names = FALSE, col.names = FALSE, sep = ",",
                file = con_out, append = TRUE) # Write to file connection
    close(con_out) # Close output file connection
  }
  data <- try(read.csv(con, header = FALSE, nrows = nrows_block)) # Read next block
}
closeAllConnections()
```

```
chmod ugo+x subset-and-stratify.R
data_filepath="data/AirlineData2006-2008.csv.bz2"
var_stratify=1 # Year
```

```
var_subset=18 # Flight destination
value_subset="SFO"
./subset-and-stratify.R $data_filepath $var_stratify $var_subset $value_subset
```

2 Problem 2

(a) `myFuns` is a list of 3 functions, each of which returns the value of `i`. Since `i` is not found in the function environment, `i` in the global environment is used. At the first evaluation, the value of `i` is 3 from the last run of the `for` loop. Hence, that value is returned and printed out thrice.
(b) Again, `i` is being found in the global environment and its value is 1 at the first iteration of the `for` loop, 2 at the second iteration and so on.
(c) `i` is now being found in the environment of the function `funGenerator` during both the third and fourth evaluations. The value of `i` is 3 at the end of the evaluation of the `for` loop.

3 Problem 3

Functions are called in the following order:

1. `sapply` with frame number 1 and objects in the frame are `FUN`, `simplify`, `USE.NAMES`, `X`
2. `lapply` with frame number 2 and objects in the frame are `FUN`, `X`
3. `FUN` with frame number 3 and the only object in the frame is `x`