problem set 2

Malvika Rajeev

9/11/2018

Question 1

Basically, whoever reads your code (possibly you in the future), should find it coherent. This means following conventions on naming, style etc. Small functions that have a specific utlity should be created and then collated to a bigger function. Variables' names shold be meaningful and succint. Functionality tests like assert and testthat should be made use of for efficiency.

Question 2

```
n <- 1e7
a <- matrix(rnorm(n), ncol = 100)
b <- matrix(rnorm(n), ncol = 1)
b <- round(b, 10)
a <- round(a, 10)
```

I wanted to see the difference between sorting it with a multicharatcer delimiter. Also, I tried a separate code where I found out the difference between files stored with different delimiters (',',/, and '//').

part(a)

10M numbers * 1 byte = 12 bytes each, 10M commas (or a new line) = 1 byte So total = between 130M to 140M

```
write.table(a, file = '/tmp/tmp.csv', quote=FALSE, row.names=FALSE, col.names = FALSE, sep=',')
write.table(a, file = '/tmp/tmp!.csv', quote=FALSE, row.names=FALSE, col.names = FALSE, sep='!!')
write.table(b, file = '/tmp/tmp2.csv', quote=FALSE, row.names=FALSE, col.names = FALSE)
save(a, file = '/tmp/tmp.Rda', compress = FALSE)
save(b, file = '/tmp/tmp2.Rda', compress = FALSE)

file.size('/tmp/tmp.csv') - file.size('/tmp/tmp!.csv') #theres a difference of 9900000 bytes!  #I tried
```

```
## [1] -9900000
```

```
file.size('/tmp/tmp.Rda')
```

```
## [1] 80000087
```

```
file.size('/tmp/tmp2.csv')
```

```
## [1] 133888663
```

```
file.size('/tmp/tmp2.Rda')
```

```
## [1] 80000087
```

```
write.table(c(a), file = '/tmp/tmp3.csv', quote=FALSE, row.names=FALSE, col.names = FALSE)
file.size('/tmp/tmp3.csv')
```

```
## [1] 133885774
```

part(b)

There won't be a difference in size because commas have been replaced by new line characters.

part(c)

So read.csv determines type too (as opposed to scan). When you specify the type of columns, read.csv obviously takes less time. Load just reloads the data already saved as Rda by the 'save' function.

```
system.time(a0 <- read.csv('/tmp/tmp.csv', header = FALSE)) #took around 21 seconds.
```

```
##    user  system elapsed
##  28.754   0.702  30.247
```

```
system.time(a1 <- scan('/tmp/tmp.csv', sep = ',')) #like 8 seconds?Unlike the read.table() function, th
```

```
##    user  system elapsed
##   2.817   0.176   3.233
```

```
system.time(a0 <- read.csv('/tmp/tmp.csv',header = FALSE, colClasses = 'numeric')) #when R is told to r
```

```
##    user  system elapsed
##   2.891   0.146   3.248
```

```
system.time(a1 <- scan('/tmp/tmp.csv', sep = ',')) #this is identical to first case?
```

```
##    user  system elapsed
##   2.798   0.140   2.988
```

```
system.time(a1 <- scan('/tmp/tmp.csv', sep = ',')) #same
```

```
##    user  system elapsed
##   2.711   0.070   2.799
```

```
system.time(load('/tmp/tmp.Rda')) #Reload datasets written with the function save. this isnt really rea
```

```
##    user  system elapsed
##   0.192   0.038   0.242
```

part(d)

```
save(a, file = '/tmp/tmp1.Rda')
file.size('/tmp/tmp1.Rda')
```

```
## [1] 76778726
```

```
b <- rep(rnorm(1), 1e7)
save(b, file = '/tmp/tmp2.Rda') ##b is basially a number repeated 100000 times
file.size('/tmp/tmp2.Rda')
```

```
## [1] 116495
```

Rda finds it easier to compress values that are the same vector again and again (through duplicaiton), as opposed to different values.

Question 3

part(a)

```
##creating a function
library(rvest)
```

```
## Loading required package: xml2
```

```
scholarpage <- function(scholar_name){
  URL="https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q="
  name=gsub(" ", "+", scholar_name)       #replace the spaces in input with a plus sign
  URL2 <- paste(URL,name,sep="")           #concatenate the expression to make it URL-friendly


  html <- read_html(URL2)
  tbls <- html_table(html_nodes(html, "table"))
  links <- html %>% html_nodes(xpath = "//a[@href]") %>%     #find all the links in the page
    html_attr('href')

  index=as.list(grep("user=[[:alpha:]]+.+=", links, value=FALSE)) #making a list of links with "user="
  scholar_id <- unlist(strsplit(links[as.integer(index[1])],'='))[[2]] #splitting the first link by '='
  #using the second field
  page <- paste("https://scholar.google.com",links[as.integer(index[1])], sep = "") #required URL
  #page
  output <- read_html(page)
  return(output)
}
```

part(b)

Upon inspection, I found that every field of the tabe of citations had a unique html tag (except of authors and citations, which were alternating)

```
createdataset <- function(scholar_name){
  library(assertthat)
  is_legit <- function(x){
  assert_that(is.character(scholar_name))
  }
  assert_that(is_legit(scholar_name))
  on_failure(is_legit) <- function(call, env){
  paste0(deparse(call$x), " is not a name.") #error message
  }
  output <- scholarpage(scholar_name)  #calling the function I made earlier
  title <- output %>% html_nodes(".gsc_a_t a") %>% html_text()
  title <- title[-1]
  year <- output %>% html_nodes(".gsc_a_h.gsc_a_hc.gs_ibl") %>% html_text()
  authors_journals <- output %>% html_nodes(".gsc_a_t .gs_gray") %>% html_text()
  l=length(authors_journals)
  a <- seq(2,l,2)
  b <- seq(1,l,2)
  authors <- authors_journals[b]
  journals <- authors_journals[a]
  citations <- output %>% html_nodes(".gsc_a_c ac") %>% html_text()
  dataset2 <- cbind(title, year, authors, journals, citations)

  return(dataset2)
}
createdataset("Andrew Ng")
```

```
##       title
##  [1,] "Latent dirichlet allocation"
##  [2,] "On spectral clustering: Analysis and an algorithm"
##  [3,] "ROS: an open-source Robot Operating System"
##  [4,] "Distance metric learning with application to clustering with side-information"
##  [5,] "Efficient sparse coding algorithms"
##  [6,] "Recursive deep models for semantic compositionality over a sentiment treebank"
##  [7,] "On discriminative vs. generative classifiers: A comparison of logistic regression and naive ba
##  [8,] "Convolutional deep belief networks for scalable unsupervised learning of hierarchical represe
##  [9,] "Cheap and fast---but is it good?: evaluating non-expert annotations for natural language task
## [10,] "Large scale distributed deep networks"
## [11,] "Map-reduce for machine learning on multicore"
## [12,] "An analysis of single-layer networks in unsupervised feature learning"
## [13,] "Apprenticeship learning via inverse reinforcement learning"
## [14,] "Multimodal deep learning"
## [15,] "Self-taught learning: transfer learning from unlabeled data"
## [16,] "Algorithms for inverse reinforcement learning."
## [17,] "Rectifier nonlinearities improve neural network acoustic models"
## [18,] "Learning word vectors for sentiment analysis"
## [19,] "Feature selection, L 1 vs. L 2 regularization, and rotational invariance"
## [20,] "Learning hierarchical invariant spatio-temporal features for action recognition with independe
##       year
##  [1,] "2003"
##  [2,] "2002"
##  [3,] "2009"
##  [4,] "2003"
##  [5,] "2007"
##  [6,] "2013"
##  [7,] "2002"
##  [8,] "2009"
##  [9,] "2008"
## [10,] "2012"
## [11,] "2007"
## [12,] "2011"
## [13,] "2004"
## [14,] "2011"
## [15,] "2007"
## [16,] "2000"
## [17,] "2013"
## [18,] "2011"
## [19,] "2004"
## [20,] "2011"
##       authors
##  [1,] "DM Blei, AY Ng, MI Jordan"
##  [2,] "AY Ng, MI Jordan, Y Weiss"
##  [3,] "M Quigley, K Conley, B Gerkey, J Faust, T Foote, J Leibs, R Wheeler, ..."
##  [4,] "EP Xing, MI Jordan, SJ Russell, AY Ng"
##  [5,] "H Lee, A Battle, R Raina, AY Ng"
##  [6,] "R Socher, A Perelygin, J Wu, J Chuang, CD Manning, A Ng, C Potts"
##  [7,] "AY Ng, MI Jordan"
##  [8,] "H Lee, R Grosse, R Ranganath, AY Ng"
##  [9,] "R Snow, B O'Connor, D Jurafsky, AY Ng"
## [10,] "J Dean, G Corrado, R Monga, K Chen, M Devin, M Mao, A Senior, ..."
## [11,] "CT Chu, SK Kim, YA Lin, YY Yu, G Bradski, K Olukotun, AY Ng"
## [12,] "A Coates, A Ng, H Lee"
## [13,] "P Abbeel, AY Ng"
## [14,] "J Ngiam, A Khosla, M Kim, J Nam, H Lee, AY Ng"
## [15,] "R Raina, A Battle, H Lee, B Packer, AY Ng"
## [16,] "AY Ng, SJ Russell"
## [17,] "AL Maas, AY Hannun, AY Ng"
## [18,] "AL Maas, RE Daly, PT Pham, D Huang, AY Ng, C Potts"
## [19,] "AY Ng"
## [20,] "QV Le, WY Zou, SY Yeung, AY Ng"
##       journals
##  [1,] "Journal of machine Learning research 3 (Jan), 993-1022, 2003"
##  [2,] "Advances in neural information processing systems, 849-856, 2002"
##  [3,] "ICRA workshop on open source software 3 (3.2), 5, 2009"
##  [4,] "Advances in neural information processing systems, 521-528, 2003"
##  [5,] "Advances in neural information processing systems, 801-808, 2007"
##  [6,] "Proceedings of the 2013 conference on empirical methods in natural language<U+00A0><U+2026>, 2
##  [7,] "Advances in neural information processing systems, 841-848, 2002"
##  [8,] "Proceedings of the 26th annual international conference on machine learning<U+00A0><U+2026>, 2
##  [9,] "Proceedings of the conference on empirical methods in natural language<U+00A0><U+2026>, 2008"
## [10,] "Advances in neural information processing systems, 1223-1231, 2012"
## [11,] "Advances in neural information processing systems, 281-288, 2007"
## [12,] "Proceedings of the fourteenth international conference on artificial<U+00A0><U+2026>, 2011"
## [13,] "Proceedings of the twenty-first international conference on Machine learning, 1, 2004"
## [14,] "Proceedings of the 28th international conference on machine learning (ICML<U+00A0><U+2026>, 20
## [15,] "Proceedings of the 24th international conference on Machine learning, 759-766, 2007"
## [16,] "Icml, 663-670, 2000"
## [17,] "Proc. icml 30 (1), 3, 2013"
## [18,] "Proceedings of the 49th annual meeting of the association for computational<U+00A0><U+2026>, 2
## [19,] "Proceedings of the twenty-first international conference on Machine learning, 78, 2004"
## [20,] "Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on<U+00A0><U+2026>, 2011
```

Testing my function here

```
####using testthat

library(testthat)
dat <- createdataset('Andrew Ng')

#expect_length(nrow(dat), 20)
test_that("Testing", {

  dat <- createdataset('Andrew Ng')
  expect_is(createdataset("hey man"), 'matrix')
  expect_error(createdataset(FALSE))
})
```

Question 4

A robots.txt file is a set of instructions for the "robots" (codes in general) when they reach that website. Twitter and facebook are apparently not allowed to web crawl on Google scholars. Personally, I have no concern about the tools we used for this assignment because we're literally just fetching the data you anyway see in a cursory search. According to scholars.google.com/robots.txt, the actions we performed are allowed by the website. As far as ethics in webcrawling are concerned, it is essentially about ensuring you aren't violating the terms and conditions. I think it's hard to pinpoint exactly where you're being unethical outside of that stipulation though: where do you draw the line between unethical and ethical web crawling and outside (or even inside) the realm of legality? (and who assumes responsiblity when the fact that almost no personal users actually read legalities is a soft fact) I'm not too sure. It's also important to distinguish between web crawling and data breaches. If any such tools are being used to scrape through datasets/files online, its imperative to ensure their authenticity and that the fact they're available to public is intentional.

In India, there is a a very robust ID system called Aadhar. It basically has assigned a unique 12 digit number for every citizen (it is, for all intents and purposes, mandatory) and with it the names, addresses and believe it or not, biometric information. Intitally the way the data was believed to be stored is that there was no central repository of key value pair system. But it started getting used more, by private companies that would build on its API, routine hacks and data leaks have become prevalent, thanks to tools associated with web scraping and crawling.

But then again I think this is a problematic issue of storing senstivie public data in open directories without having the legal minimum requirement of what constitutes best security practices for a website that deals with such data. So when I examine the function we all made, its obvious that webcrawling in itself cannot be labelled ethical or unethical - its the reason why you end up doing it.
