BAN432 fall 2019 - Second assignment

Goal of this assignment

In this assignment you revisit data collection, preprocessing and key-word based analysis in R. We touched on almost all used functions in our last lectures and the previous assignment. If you are unsure how to use a function, either use `R`'s own documentation (type `?function.name()`) or `www.stackoverflow.com`. We encourage you to code this assignment yourselves, and do not use purpose made solutions or packages as provided on the internet. All tasks in this assignment focus on regular expressions.

Please submit your assignment even though you were not able to find a solution to all tasks.

Learning outcomes

In this assignment you will apply the skills you acquired in the lectures on data gathering and cleaning. In contrast to the first assignment, this time you will work with data from the field of business and finance. Furthermore you will relate the result of your analysis to relevant data from other analyses.

Formalities

This assignment will be handed out on 24 of September, 2019 at 16:00 and has to be submitted no later than the 1st of October, 2019 at 14:00. Please comment your code shortly so that the grader can reconstruct your thinking. You do not need to explain the used functions.

Please work together in groups of four and submit the assignment via Canvas.

Introduction

Sustainable investing is on the rise. Many big mutual funds companies started to offer funds that focus solemnly on ESG firms (ESG is an abbreviation for environmental, social, and governance issues). Evaluating whether firms are sustainable and care about their stakeholders is not an easy task. Several firms offer sustainability ranking these days. For example, Calvert provides a ranking of the top 100 sustainable US firms: http://webreprints.djreprints.com/55916.pdf.

In this assignment, you will try to construct an ESG score for firms based on a key-word analysis. You will download the annual reports for firms within the Calvert ranking and a set of control firms. Subsequently, you will use a KWIC (Key words in context) analysis to decide which keywords you want to use in your analysis. Next you will relate relative word counts of the key words - your ESG score - to the Calvert rating.

Hint: Before starting to code, try to seperate the task into smaller pieces.

Task 1: gather data from EDGAR

Download the most recent annual report for all firms within `company_index.RData`. The supplied data file holds a dataframe with three variables: `ticker` indicates the ticker of a firm, `cik` indicates the central index key of the firm, and `sustainable` indicates whether or not the firm is within the Calvert ranking. We suggest following coding approach (feel free to use your own):

1. Extract the index file from EDGAR for the most recent 12 months.

2. Find the url for the 10-Ks of the listed firms. Note that not all firms will have filed an annual report with the SEC. Make sure that you do not include ammendments (such as 10-K/A).
3. Download the identified files. You might want to use the `download.file(..., mode = "wb")` function.

Task 2: clean the 10Ks

The txt-files you downloaded in task 1 contain a lot of noise. Use regular expressions to perform the following cleaning steps on each txt-file:

- an EDGAR filing often consists of several documents. Keep just that part of the file that contains the actual 10-K form (with the items 1 to 14). Hint: look for the `<TEXT>`-tag.
- remove all html-tags
- remove html-entities
- remove numbers
- remove excessive whitespace

You might want to save the cleaned 10-Ks for later analysis.

Task 3: KWIC analysis

Your ultimate goal is a key-word based ESG score. The score should capture the same dimension as the Calvert ranking (please see the provided link in the introduction). Hence, you need to settle on the right regular expressions. Potential terms to start your investigation are: stakeholder, environment, sustinability, etc. While these terms sound unambiguous in the first place, it is not obvious that the are. For example, your KWIC analysis will reveal the firms often use the word environment to refer to their competition and the legal frameworks governing their activities. Feel free to suggest additional terms.

Given the KWIC analysis decide on a fairly short list of terms for your sustainability score. Make sure that you use regular expressions so that you capture all words you intent to (e.g. environment v.s. environmental).

Task 4: ESG score

For each annual report compute the relative word count of the key words defined in the last task, your ESG score. Relate your ESG score to the one by Calvert.

Optional: Provide a statistical test indicating whether your ESG score has the same information as the one by Calvert.

```r
# packages
require(dplyr)
require(tibble)

# Task 1.1 download index file
download.file(url = "https://www.sec.gov/Archives/edgar/full-index/2019/QTR1/master.idx",
              destfile = "data/2019_QTR1_master.idx", mode = "wb")
download.file(url = "https://www.sec.gov/Archives/edgar/full-index/2019/QTR2/master.idx",
              destfile = "data/2019_QTR2_master.idx", mode = "wb")
download.file(url = "https://www.sec.gov/Archives/edgar/full-index/2019/QTR3/master.idx",
              destfile = "data/2019_QTR3_master.idx", mode = "wb")
download.file(url = "https://www.sec.gov/Archives/edgar/full-index/2018/QTR4/master.idx",
              destfile = "data/2018_QTR4_master.idx", mode = "wb")


# Task 1.2 extract 10K file path for sample firms
load("company_index.RData")

# output
filings.extracted <- NULL
```

```r
f.downloaded <- list.files("data/", full.names = T, pattern = ".idx")

# loop
for(f in f.downloaded){
  # load
  filings <- read.table(file = f,
                        header = F,
                        sep = "|",
                        skip = 11,
                        stringsAsFactors = F,
                        fill =T)
  colnames(filings) <- c("CIK", "Name", "Form", "Date", "URL")
  # limit
  filings <- filings %>% filter(CIK %in% comp$cik &
                       Form %in% ("10-K"))
  # add
  filings.extracted <- rbind(filings.extracted, filings)
}
# append file path
comp$file.path <- filings.extracted$URL[match(comp$cik, filings.extracted$CIK)]

# limit to merged firms
comp <- comp %>% filter(!is.na(file.path))

# adjust file path
comp$file.path <- paste0("https://www.sec.gov/Archives/", comp$file.path)

# save
save(comp, file = "company_index_v2.RData")

#
load(file = "company_index_v2.RData")


# Task 1.3 download the individual 10Ks, filter the right part, clean, and save
dir.create("data/indiv.10Ks/", showWarnings = F)

# download individual files
for(i in 1:nrow(comp)){
  download.file(url = comp$file.path[i],
                destfile = paste0("data/indiv.10Ks/AR_",
                                  i,
                                  "-",
                                  comp$cik[i],
                                  ".txt"),
                mode = "wb")
  #Sys.sleep(1)
}


# Task 2: clean 10Ks

# clean them
```

```r
dir.create("data/indiv.10Ks.cleaned/", showWarnings = F)

# downloaded files
f.downloaded <- list.files("data/indiv.10Ks/", full.names = T)

# load
for(i in seq_along(f.downloaded)){
  # load
  text <- readLines(f.downloaded[i])
  # extract the right lines
  text <- paste(text[(grep("^<TEXT>", text)[1]+1) : (grep("^</TEXT>", text)[1]-1)], collapse = " ")

  # (3.1): Cleaning steps
  text <- gsub(".+?(<text>.+?</text>).+", "\\1", text, ignore.case = T)
  text <- gsub("<.+?>", " ", text)
  text <- gsub("&.+?;", " ", text)
  text <- gsub("[[:digit:]]", " ", text)
  text <- gsub("[[:space:]]{2,}", " ", text)
  text <- tolower(text <- gsub("[[:space:]]{2,}", " ", text))

  # save
  write.csv(text, file  = gsub("indiv.10Ks", "indiv.10Ks.cleaned", f.downloaded[i]))
}



# Task 3: KWIC analysis

# potential root terms
root.term <- "communit"
root.term <- "sustainab"
root.term <- "stakeholder"

# list of cleaned files
f.cleaned <- list.files("data/indiv.10Ks.cleaned/", full.names = T)

# declare KWIC able
KWIC <- NULL

# terms in front of the KWIC
n <- 4

# loop through everything
for(i in seq_along(f.cleaned)){

  # load and tokenize
text <- scan(f.cleaned[i], what = "character", quote="")

# determine which words match the root.term
index.env <- grep(root.term, text)

# Use KWIC table on current filing
KWIC.temp <- tibble(left = sapply(index.env,
                               function(i) {paste(text[i-n:1], collapse = " ")}),
               keyword = text[index.env],
               right = sapply(index.env,
                               function(i) {paste(text[i+1:n], collapse = " ")}))

# append current filing
KWIC <- rbind(KWIC, KWIC.temp)
}

# inspect
View(KWIC)

# insight: go from environment to environmental
# for sustainable, we can use either sustainability / sustainable
```

```
# Task 4: ESG score

# construct measure
load("company_index_v2.RData") # not completely correct!

# file path
comp$file.path.cleaned <- paste0("data/indiv.10Ks.cleaned/AR_", 1:nrow(comp), "-", comp$cik,".txt")

# declare new variables
comp$nr.words <- comp$nr.ESG.term <- NA

# decide on root term
root.term <- "sustainab"
# i use a very simple approach here. More elaborate root terms are encouraged!

# loop through all filings
for(i in seq_along(comp$file.path.cleaned)){

  # load and tokenize
  text <- readLines(comp$file.path.cleaned[i])

  # merge
  text <- paste(text, collapse = " ")

  # determine which words match the root.term
  comp$nr.ESG.term[i] <- length(gregexpr( root.term, text,  ignore.case = T)[[1]])
  comp$nr.words[i] <- length(gregexpr( "\\b\\w+\\b",text, ignore.case = T)[[1]])
}
comp$ESG.score <- comp$nr.ESG.term / comp$nr.words * 1000

# relate the word count measure and the score
summary(lm(comp$ESG.score ~ comp$sustainable))
```