Appendix

Nikita Stempniewicz

October 24, 2017

```
# this is the code I used to scrape glassdoor

## Used some of stephens code from lab for help, need to
## figure out how to cite?

## Load Packages

library(rvest)
library(stringr)

#############################################################################################################

## Urls for data scientist positions on glassdoor note for
## some reason glassdoor does not let you go past job post
## page 33?


url <- paste0("https://www.glassdoor.com/Job/new-york-data-scientist-jobs-SRCH_IL.0,8_IC1132348_KO9,23_
    1, ".htm")
urls <- paste0("https://www.glassdoor.com/Job/new-york-data-scientist-jobs-SRCH_IL.0,8_IC1132348_KO9,23
    1:33, ".htm")

df1 <- data.frame()

#############################################################################################################

## First we will get data from summary data for each post from
## the search result urls


#############################################################################################################

## creates fields which are the individual job posts from
## glassdoor
for (i in 1:33) {
    paste(i)
    download.file(urls[i], destfile = "scrapedpage.html", quiet = TRUE)
    fields <- read_html("scrapedpage.html") %>% html_nodes(xpath = "//*[contains(concat( \" \", @class,

    ###########################################################################################################
    title <- fields %>% html_nodes(".flexbox .jobLink") %>% html_text() %>%
        trimws()

    ###########################################################################################################
    ###########################################################################################################
    salaries <- sapply(fields, function(x) {
        tmp <- html_nodes(x, ".small") %>% html_text()
```

```
        ifelse(length(tmp) == 0, "Not listed", trimws(tmp))
    })
    salaries <- trimws(gsub("(Glassdoor est.)", "", salaries))

    ##############################################################################################################

    locations <- fields %>% html_nodes(".loc") %>% html_text() %>%
        trimws()


    ##############################################################################################################

    ## can;t figure out how to get gsub or strreplace to recognize
    ## the - to remove the city name
    employer <- fields %>% html_nodes(".empLoc") %>% html_text() %>%
        trimws()

    employer_id <- fields %>% html_attr("data-emp-id")

    job_id <- fields %>% html_attr("data-id")


    ##############################################################################################################
    df1a <- cbind(job_id, employer_id, title, salaries, locations,
        employer)

    df1 <- rbind(df1, df1a)
}

save(df1, file = "glassdoor_df1.r")

############################################################################################################## Nov
############################################################################################################## inc


############################################################################################################## cre
############################################################################################################## jus

df1$job.urls <- paste0("https://www.glassdoor.com/job-listing/data-scientist-emc-research-JV_IC1145845_K
    df1$job_id)

############################################################################################################## get
############################################################################################################## fro

df1$job_desc_raw <- ""
df1$employer2 <- ""
# used to compare null results in loop
a <- character(0)
n <- nrow(df1)
for (i in 1:n) {

    Sys.sleep(1)
```

```
    download.file(df1[i, ]$job.urls, destfile = "scrapedpage.html",
        quiet = TRUE)

    website <- read_html("scrapedpage.html")

    df1[i, ]$job_desc_raw <- (if (identical(a, (website %>% html_nodes("#JobDescContainer") %>%
        html_text()))) {
        "NO DESCRIPTION LISTED"
    } else {
        website %>% html_nodes("#JobDescContainer") %>% html_text()
    })

    df1[i, ]$employer2 <- (if (identical(a, (website %>% html_nodes(".padRtSm") %>%
        html_text()))) {

        "NO EMPLOYER LISTED"
    } else {
        website %>% html_nodes(".padRtSm") %>% html_text()
    })
}

saveRDS(df1, file = "glassdoor_df1")

############################################################################################################ Fi
############################################################################################################ gla

############################################################################################################ cr
############################################################################################################ br
############################################################################################################ an

unique_emp_ids <- unlist(unique(df1$employer_id[df1$employer_id !=
    0]))

emp.urls <- paste0("https://www.glassdoor.com/Overview/Working-at-ID-Analytics-EI_IE",
    unique_emp_ids, ".11,23.htm")

df2 <- data.frame(unique_emp_ids, emp.urls)

############################################################################################################ ge

n <- nrow(df2)

df2$emp_desc_raw <- ""

# df2<-df2[-c(387),]

for (i in 1:n) {

    Sys.sleep(1)

    download.file(as.character(df2$emp.urls[i]), destfile = "scrapedpage.html",
        quiet = TRUE)
```

```
    website <- read_html("scrapedpage.html")

    df2[i, ]$emp_desc_raw <- (if (identical(a, (website %>% html_nodes("#EmpBasicInfo") %>%
        html_text()))) {
        "NO DESCRIPTION LISTED"
    } else {
        website %>% html_nodes("#EmpBasicInfo") %>% html_text()
    })

}

saveRDS(df2, file = "glassdoor_df2")


################################################################################################################ me

colnames(df2)[1] <- "employer_id"

glassdoor_df <- merge(x = df1, y = df2, by = "employer_id", all.x = TRUE)

saveRDS(glassdoor_df, file = "glassdoor_df")
```