

HW04

DerekRBratcher

10/21/2021

Homework 4

1. (3 points) Use the rvest R package to scrape the schedule and materials table into R from the course webpage (https://introdatasci.dlilab.com/schedule_materials/). Read the documentation of rvest so you get a better idea about the functions provided by rvest and their usages.

```
library(rvest)
#designate the target /w read html and assign a name to the function
dilab_sched_mats<-read_html("https://introdatasci.dlilab.com/schedule_materials/")

#determine corresponding css label for the sched/materials table
SnM_table <- dilab_sched_mats %>% html_elements("table") |> html_table()

'SMdf'<-as.data.frame(SnM_table)
```

2. (2 points) With the extracted data frame, create two new columns based on the Date column: month and day. month would be the month abbreviations from the Date column; day would be the numeric numbers from the Date column. Although you can use whatever approach to get this done (do not enter them by hand...), I suggest you try to practice regular expression here (sub() or stringr::str_extract()).

```
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

#extract days
days<-stringi::stri_sub(SMdf[, 1], -2,)
days<-as.numeric(days)
SMdf_new <- cbind(SMdf, days)
```

```

#create month columns
Month<-stringi::stri_sub(SMdf[, 1], from = 1, to = -3)
Month<- as.factor(Month)
Month_trim<- stringr::str_trim(Month, side = ("both"))
#not that it matters but how do you manipulate factor levels?
Months_<- factor(Month_trim, levels = c("Aug", "Sep", "Oct", "Nov", "Dec"))
SMdf_MD<-cbind(SMdf_new, Months_)

#eliminate reading, notes, hw column
SMdf_MD_trim<- SMdf_MD |> select(-Notes, -HW, -Reading)
SMdf_MD_trim

```

##	Date	Topic	days	Months_
## 1	Aug 24	About the course	24	Aug
## 2	Aug 26	Data science project cycle	26	Aug
## 3	Aug 31	Class cancelled because of Hurricane Ida	31	Aug
## 4	Sep 2	Class cancelled because of Hurricane Ida	2	Sep
## 5	Sep 7	Introduction and install tools	7	Sep
## 6	Sep 9	Version control with Git	9	Sep
## 7	Sep 14	Introduction to GitHub	14	Sep
## 8	Sep 16	RStudio project and dynamic documents with R Markdown	16	Sep
## 9	Sep 21	The file system and basic unix shell	21	Sep
## 10	Sep 23	R basics: data types, vectors, matrix, data frame, etc.	23	Sep
## 11	Sep 28	More R basics: lists, dates, etc.	28	Sep
## 12	Sep 30	R programming basics: conditional statements	30	Sep
## 13	Oct 5	R programming basics: loops, apply	5	Oct
## 14	Oct 7	Strings and Regular expressions	7	Oct
## 15	Oct 12	API and data scraping	12	Oct
## 16	Oct 14	Data input and output	14	Oct
## 17	Oct 19	Data manipulation with R	19	Oct
## 18	Oct 26	More data manipulation with R	26	Oct
## 19	Oct 28	Data visualization with R	28	Oct
## 20	Nov 2	Exploratory data analysis	2	Nov
## 21	Nov 4	Regression methods	4	Nov
## 22	Nov 9	More on Regression methods	9	Nov
## 23	Nov 11	Write your own functions	11	Nov
## 24	Nov 16	Write your own R package	16	Nov
## 25	Nov 18	Open Science and automating things with Makefile	18	Nov
## 26	Nov 23	Ethics in data science (virtual)	23	Nov
## 27	Nov 25	Thanksgiving, no class	25	Nov
## 28	Nov 30	Final project presentation	30	Nov
## 29	Dec 2	Final project presentation and wrap up	2	Dec
## 30	Dec 14	Final grades due	14	Dec

3. (2 points) With the data frame generated from Q2, use `group_by()` and `summarise()` to find out the number of lectures for each month, order the results by the number of lectures (high to low).

```

#use group by month, the summarise to make new df with data from "month trim", count the number of mont
lec_per_m<- SMdf_MD_trim |> group_by(Months_) |> summarise( Months_, n =n()) |> distinct() |> arrange

```

'summarise()' has grouped output by 'Months_'. You can override using the '.groups' argument.

```
lec_per_m
```

```
## # A tibble: 5 x 2
## # Groups:   Months_ [5]
##   Months_     n
##   <fct>   <int>
## 1 Sep         9
## 2 Nov         9
## 3 Oct         7
## 4 Aug         3
## 5 Dec         2
```

4. (3 points) For the Topic column, split all values into words (hint: `stringr::str_split()`). Observe the values in the Topic column and use regular expression to specify the pattern in the `stringr::str_split()` or `strsplit()` function. Once this is done, you should get a list of list, you can use `unlist()` to convert it into a vector and name it as words. Use `table()` and `sort()` to find the top 5 most frequent words.

```
#use str_split to itemize each word into strings, make a list using regex and str_split, use unlist to
```

```
topic_wrds<-stringr::str_split(SMdf_MD$Topic, " ")
#why doesnt following work help me understand subsetting
test_subset<- stringr::str_split(SMdf_MD["Topic"], " ")
```

```
## Warning in stri_split_regex(string, pattern, n = n, simplify = simplify, :
## argument is not an atomic vector; coercing
```

```
#topic_wrds is now a list of strings each string corresponding to a topic and each character value is a
topic_wrds
```

```
## [[1]]
## [1] "About" "the" "course"
##
## [[2]]
## [1] "Data" "science" "project" "cycle"
##
## [[3]]
## [1] "Class" "cancelled" "because" "of" "Hurricane" "Ida"
##
## [[4]]
## [1] "Class" "cancelled" "because" "of" "Hurricane" "Ida"
##
## [[5]]
## [1] "Introduction" "and" "install" "tools"
##
## [[6]]
## [1] "Version" "control" "with" "Git"
##
## [[7]]
## [1] "Introduction" "to" "GitHub"
##
## [[8]]
```

```

## [1] "RStudio"      "project"      "and"          "dynamic"      "documents" "with"
## [7] "R"             "Markdown"
##
## [[9]]
## [1] "The"          "file"         "system" "and"          "basic"      "unix"      "shell"
##
## [[10]]
## [1] "R"            "basics:"      "data"         "types,"      "vectors," "matrix,"  "data"
## [8] "frame,"      "etc."
##
## [[11]]
## [1] "More"        "R"            "basics:"      "lists,"      "dates,"    "etc."
##
## [[12]]
## [1] "R"           "programming" "basics:"      "conditional" "statements"
##
## [[13]]
## [1] "R"           "programming" "basics:"      "loops,"      "apply"
##
## [[14]]
## [1] "Strings"     "and"          "Regular"      "expressions"
##
## [[15]]
## [1] "API"         "and"          "data"         "scraping"
##
## [[16]]
## [1] "Data"        "input"        "and"          "output"
##
## [[17]]
## [1] "Data"        "manipulation" "with"          "R"
##
## [[18]]
## [1] "More"        "data"         "manipulation" "with"          "R"
##
## [[19]]
## [1] "Data"        "visualization" "with"          "R"
##
## [[20]]
## [1] "Exploratory" "data"         "analysis"
##
## [[21]]
## [1] "Regression"  "methods"
##
## [[22]]
## [1] "More"        "on"           "Regression"  "methods"
##
## [[23]]
## [1] "Write"       "your"         "own"         "functions"
##
## [[24]]
## [1] "Write"       "your"         "own"         "R"            "package"
##
## [[25]]
## [1] "Open"        "Science"      "and"          "automating"  "things"

```

```
## [6] "with"          "Makefile"
##
## [[26]]
## [1] "Ethics"      "in"          "data"        "science"     "(virtual)"
##
## [[27]]
## [1] "Thanksgiving," "no"          "class"
##
## [[28]]
## [1] "Final"        "project"     "presentation"
##
## [[29]]
## [1] "Final"        "project"     "presentation" "and"         "wrap"
## [6] "up"
##
## [[30]]
## [1] "Final"  "grades" "due"
```

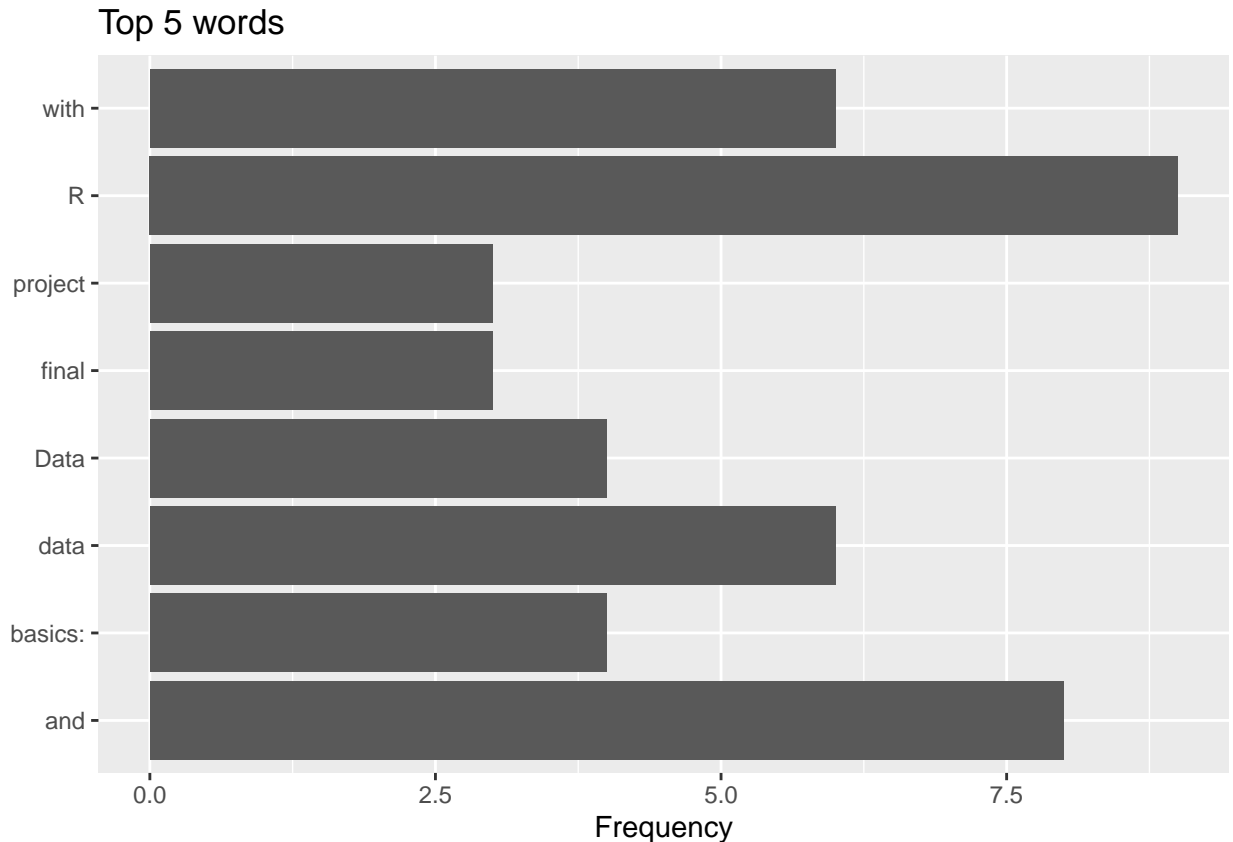
```
as_words<-unlist(topic_wrds)
top5<-sort(prop.table(table(as_words)), decreasing = T)
top_5<-sort(table(as_words), decreasing = T)
top5
```

```
## as_words
##           R           and           data           with           basics:
## 0.064285714 0.057142857 0.042857143 0.042857143 0.028571429
##      Data      project      Final      More      because
## 0.028571429 0.028571429 0.021428571 0.021428571 0.014285714
## cancelled    Class      etc.    Hurricane    Ida
## 0.014285714 0.014285714 0.014285714 0.014285714 0.014285714
## Introduction manipulation methods      of      own
## 0.014285714 0.014285714 0.014285714 0.014285714 0.014285714
## presentation programming Regression science Write
## 0.014285714 0.014285714 0.014285714 0.014285714 0.014285714
##      your      (virtual)      About      analysis      API
## 0.014285714 0.007142857 0.007142857 0.007142857 0.007142857
##      apply      automating      basic      class      conditional
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      control      course      cycle      dates,      documents
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      due      dynamic      Ethics      Exploratory      expressions
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      file      frame,      functions      Git      GitHub
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      grades      in      input      install      lists,
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      loops,      Makefile      Markdown      matrix,      no
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      on      Open      output      package      Regular
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      RStudio      Science      scraping      shell      statements
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      Strings      system Thanksgiving,      the      The
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
```

```
##      things      to      tools      types,      unix
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857
##      up      vectors,      Version visualization      wrap
## 0.007142857 0.007142857 0.007142857 0.007142857 0.007142857

#make a graph with ggplot?
library(ggplot2)
library(ggplot2)
top5df <- data.frame(stringsAsFactors=FALSE,
                      word = c("R", "and", "data", "with", "basics:", "Data", "project", "final"),

graph_top5_wrds<-ggplot(top5df, aes(word, freq)) +
geom_bar(stat="identity") +
coord_flip() +
labs(x = NULL, y = "Frequency") +
labs(title="Top 5 words ")
graph_top5_wrds
```



I was thinking to have a homework to get all plant occurrence data within Baton Rouge from GBIF. But it will require you to register an account and have account name and password when you use the rgbif package. This may have the risk of get your password leaked (you can avoid this by reading the documentation of rgbif); so I decided not to do so. If you are interested, you may want to run some example codes from the rgbif package's documentation.

This homework will be due at October 28th, 9am.