Week 3 Assignment

Samriti Malhotra

February 17, 2019

Introduction

Using regular expression and string manipulation to extract relevant information from structured, un-structed text data using R. The below few examples from Week3 assignment, shows how we can use regular expression to extract strings, on which we can then apply R packages like stringr / concatenation / BaseR etc.. to manipulate the strings and construct something meangingfull.

```
library(stringr)
library(concatenate)
```

```
## Warning: package 'concatenate' was built under R version 3.5.2
```

Question 3:-

Copy the introductory example. The vector name stores the extracted names.

(a) Use the tools of this chapter to rearrange the vector so that all elements conform to the standard first_name last_name.
(b) Construct a logical vector indicating whether a character has a title (i.e., Rev. and Dr.).
(c) Construct a logical vector indicating whether a character has a second name.

Answer 3:-

```
raw.data <- "555-1239Moe Szyslak(636) 555-0113Burns, C. Montgomery555-6542Rev. Timothy Lovejoy555 8904N

name <- unlist(str_extract_all(raw.data, "[[:alpha:]., ]{2,}"))
phoneNum <- unlist(str_extract_all(raw.data, "\\(?(\\d{3})?\\)?(-| )?(\\d{3})(-| )?(\\d{4})"))

# a)
name1 <- str_replace(name ,pattern = "C. " , replacement = "")
name1 <- str_replace(name1 ,pattern = "," , replacement = "")
unlist(str_extract_all(name1, "[[:alpha:],]{2,}[:space:]+[:alpha:]{2,}"))
```

```
## [1] "Moe Szyslak"      "Burns Montgomery" "Timothy Lovejoy"
## [4] "Ned Flanders"     "Simpson Homer"    "Julius Hibbert"
```

```
# b)
str_detect(name,"[[:alpha:]]{2,3}[\\.]")
```

```
## [1] FALSE FALSE  TRUE FALSE FALSE  TRUE
```

```
# c)
str_detect(name,"[:SPACE:][[:alpha:]]{1,}[\\.][:SPACE:]")
```

```
## [1] FALSE  TRUE FALSE FALSE FALSE FALSE
```

Question 4:-

Describe the types of strings that conform to the following regular expressions and construct an example that is matched by the regular expression. (a) [0-9]+\$ (b) \b[a-z]{1,4}\b (c) .*?\.txt$ (d) \d{2}/\d{2}/\d{4} (e)

<(.+?)>.+?</\1>

Answer 4 :-

```
# (i)
str_4i <- c("ADGYRE67575$", "1245638ABCV$","123BAC45")
str_detect(str_4i,"[0-9]+\\$" )
```

```
## [1]  TRUE FALSE FALSE
```

```
# (ii)
str_4ii <- c( "test in the week","Cat", "1234")
str_detect(str_4ii,"\\b[a-z]{1,4}\\b")
```

```
## [1]  TRUE FALSE FALSE
```

```
# (iii)
strPattern <- ".*?\\.txt$"
str_4iii <- c("Assingment.txt","assignment")
str_detect(str_4iii,strPattern)
```

```
## [1]  TRUE FALSE
```

```
# (iv)
strPatt <- "\\d{2}/\\d{2}/\\d{4}"
str_4iv <-c("2/2/1977","02/02/1977","1977/03/29","03/29/86")
str_detect(str_4iv,strPatt)
```

```
## [1] FALSE  TRUE FALSE FALSE
```

```
# (v)
strPatt <- "<(.+?)>.+?</\\1>"
str_4v <- c("<h1> this is an paragraph header</h1>","<p>test</pt>","<test>b</test")
str_detect(str_4v,strPatt)
```

```
## [1]  TRUE FALSE FALSE
```

Question 9:-

The following code hides a secret message. Crack it with R and regular expressions. Hint: Some of the characters are more revealing than others! The code snippet is also available in the materials at www.r-datacollection.com. clcopCow1zmstc0d87wnkig7OvdicpNuggvhryn92Gjuwczi8hqrfpRxs5Aj5dwpn0Tanwo Uwisdij7Lj8kpf03AT5Idr3coc0bt7yczjatOaootj55t3Nj3ne6c4Sfek.r1w1YwwojigO d6vrfUrbz2.2bkAnbhzgv4R9i05zEcrop.wAgnb.SqoU65fPa1otfb7wEm24k6t3sR9zqe5 fy89n6Nd5t9kc4fE905gmc4Rgxo5nhDk!gr

Answer 9 :-

```
secret_mesg <- c("clcopCow1zmstc0d87wnkig7OvdicpNuggvhryn92Gjuwczi8hqrfpRxs5Aj5dwpn0Tanwo
Uwisdij7Lj8kpf03AT5Idr3coc0bt7yczjatOaootj55t3Nj3ne6c4Sfek.r1w1YwwojigO
d6vrfUrbz2.2bkAnbhzgv4R9i05zEcrop.wAgnb.SqoU65fPa1otfb7wEm24k6t3sR9zqe5
fy89n6Nd5t9kc4fE905gmc4Rgxo5nhDk!gr")

secret_msg <- unlist(str_extract_all(secret_mesg,"[[:upper:].]{1,}"))
secret_msg <- cc(secret_msg)
secret_msg <- str_replace_all(secret_msg,",","")
secret_msg <- str_replace_all(secret_msg,"\\."," ")
secret_msg
```

```
## [1] "C O N G R A T U L AT I O N S  Y O U  A R E  A S U P E R N E R D"
```

Summary

During these examples we learnt various techniques of regular expression(back referencing, fixed, {}repeators, [:alpha/space:]) . Also used various methods of stringr , concetanation packages to manipulate the strings.