hw 3 - Regular Expressions

Yan Lin

September 17, 2017

Question 3, 4 and 9

```
## Question 3

raw.data <-"555-1239Moe Szyslak(636) 555-0113Burns, C. Montgomery555-6542Rev.
Timothy Lovejoy555 8904Ned Flanders636-555-3226Simpson, Homer5553642Dr. Juliu
s Hibbert"

library(stringr)

name <- unlist(str_extract_all(raw.data,"[[:alpha:]., ]{2,}"))
name

## [1] "Moe Szyslak"          "Burns, C. Montgomery" "Rev. Timothy Lovejoy"
## [4] "Ned Flanders"         "Simpson, Homer"       "Dr. Julius Hibbert"

## remove the titles and middle name with space
name2 <- sub("[A-z]{1,}\\.( )?","",name)
name2

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

## witch the order
name3<- sub("(\\w+), (\\w+)","\\2 \\1",name2)
name3

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

## check if a name has a title
title <- str_detect(name,"[A-z]{2,}\\.")
title

## [1] FALSE FALSE  TRUE FALSE FALSE  TRUE

name_title <- data.frame(name,title)
name_title

##                    name title
## 1          Moe Szyslak FALSE
## 2 Burns, C. Montgomery FALSE
## 3 Rev. Timothy Lovejoy  TRUE
## 4         Ned Flanders FALSE
## 5       Simpson, Homer FALSE
## 6   Dr. Julius Hibbert  TRUE
## check if a name has a second name
middle_name <- str_detect(name,"[A-Z]. ")
middle_name
```

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

```
name_middle <- data.frame(name,middle_name)
name_middle
```

```
##                    name middle_name
## 1          Moe Szyslak       FALSE
## 2 Burns, C. Montgomery        TRUE
## 3 Rev. Timothy Lovejoy       FALSE
## 4         Ned Flanders       FALSE
## 5       Simpson, Homer       FALSE
## 6   Dr. Julius Hibbert       FALSE
```

```
## Question 4
```

```
## (a) [0-9]+\\$
## It extracts any digit number from 0 to 9 and will stop when the sign "$" s
hows up. It will return a format as digital numbers and followed by "$" sign.
```

```
a <- "3452000001234$$$skdfh54565"
unlist(str_extract_all(a,"[0-9]+\\$"))
```

```
## [1] "3452000001234$"
```

```
## (b) \\b[a-z]{1,4}\\b
## "[a-z]" indicates the reture value has to be lowercase letter. "{1,4}" ask
this sequence appears at least once and up to 4 times, such as "a good one" b
ut not "world", or "word8". Notice that digital number bounded with letters h
as been considered to be one letter.
## "\\b" word edge is applied to in the beginning of any word in a string as
well the end of any word in a string. It will skip any words start or end as
capital letter.
```

```
b <- "Are$ $y%ou sU**re *8okay for thiS now, angelababy?"
unlist(str_extract_all(b,"\\b[a-z]{1,4}\\b"))
```

```
## [1] "y"   "ou"  "re"  "for" "now"
```

```
## (c) .*?\\.txt$
## It only returns any string that ends with ".txt"
```

```
c <- "8&>this .is% not ^a g*ood day.txt"
unlist(str_extract_all(c,".*?\\.txt$"))
```

```
## [1] "8&>this .is% not ^a g*ood day.txt"
```

```
## (d) \\d{2}/\\d{2}/\\d{4}
## d{2} asks the function to return 2 digital numbers while d{4} asks for 4 d
```

```
igital numbers. so, it will return any string in the date format, dd/dd/dddd.

d <- "02/21/2009 is your&& birthday!!!"
unlist(str_extract_all(d,"\\d{2}/\\d{2}/\\d{4}"))

## [1] "02/21/2009"

##(e) <(.+?)>.+?</\\1>
## It returns a string that begins with "<" and followes any character that m
atches at least one time but at most one time (which is optional).Using back
reference to return any string starts with <text> ends with </text>.

e <- "<script> a = {1:4} </script> <head> meta </head> "
unlist(str_extract_all(e, '<(.+?)>.+?</\\1>'))

## [1] "<script> a = {1:4} </script>" "<head> meta </head>"

## extra credit - question 9
raw.data <- "clcopCow1zmstc0d87wnkig7OvdicpNuggvhryn92Gjuwczi8hqrfpRxs5Aj5dwp
n0TanwoUwisdij7Lj8kpf03AT5Idr3coc0bt7yczjatOaootj55t3Nj3ne6c4Sfek.r1w1Ywwojig
Od6vrfUrbz2.2bkAnbhzgv4R9i05zEcrop.wAgnb.SqoU65fPa1otfb7wEm24k6t3sR9zqe5fy89n
6Nd5t9kc4fE905gmc4Rgxo5nhDk!gr"
msg <- unlist(str_extract_all(raw.data,"[[:upper:].!]"))
msg

##  [1] "C" "O" "N" "G" "R" "A" "T" "U" "L" "A" "T" "I" "O" "N" "S" "." "Y"
## [18] "O" "U" "." "A" "R" "E" "." "A" "." "S" "U" "P" "E" "R" "N" "E" "R"
## [35] "D" "!"

msg1 <- paste(msg, sep="",collapse="")
msg1

## [1] "CONGRATULATIONS.YOU.ARE.A.SUPERNERD!"

secret_msg <- str_replace_all(msg1, "[\\.]"," ")
secret_msg

## [1] "CONGRATULATIONS YOU ARE A SUPERNERD!"

str_locate(secret_msg,"S")

##      start end
## [1,]    15  15

str_sub(secret_msg,15,15) <- "S!"
secret_msg

## [1] "CONGRATULATIONS! YOU ARE A SUPERNERD!"
```