Stat243: Problem Set 3

Linqing Wei

September 29, 2017

```
> #==============Problem 2(A) ===============
> download.file("http://www.gutenberg.org/cache/epub/100/pg100.txt",
+               destfile = "SHAKE.txt")
> original_file  <- readLines('SHAKE.txt')
> #Trim the first and last chunks are trivial
> original_file = original_file[-c(1:2814,123979:length(original_file ))]
> #generate a character vector from the text file
> original_file = paste(original_file, collapse = " ")
> #Split plays using "THE END" as a marker
> plays = strsplit(original_file, "THE END")[[1]]
> #37th element doesn't contain any play information, remove
> plays = plays[-37]
> num_plays = length(plays)   #36 plays
> print(num_plays)

[1] 36

> #================Problem 2(B)==================
> library(stringr)
> library(gsubfn)
> #Housekeeping, remove all the texts within "<<>>", rm_extra is a cleaned file
> rm_extra = gsub("<<[^>]*>>", "", plays)
> #Extract body part starting from "ACT"
> body = lapply(rm_extra, function(x) str_extract_all(x, ":?ACT.*"))
> #Extract years
> year = lapply(rm_extra, function(x) unlist(str_extract_all(x, "[0-9]{3,}")))
> #Unique() in this case collaborates with length() to get the number of acts
> acts = lapply(rm_extra, function(x) length(unique(unlist(str_extract_all(x,
+                         "(?:ACT)[[:space:]][[:upper:]]{1,}[.]")))))
> #Scenes either starts with"SCENE" or "Scene"
> scenes = lapply(rm_extra, function(x) length(unlist(str_extract_all(x,
+     "((?:SCENE)|(?:Scene))[[:space:]][[:upper:]]{1,}[.]{0,}|((?:SCENE)|(Scene))
+      [[:space:]][1-9][.]{0,}"))))
> #exrtact titles
> titles = strapplyc(rm_extra, "[0-9]{3,}[\n]*([^[:lower:]]{2,})")
> #create dataframe, dimension = 4 rows(attributes) and
> #36 variables (each variable is a play )
> play_table = rbind(year, titles, acts, scenes)
> play_table = matrix(play_table, nrow = 4, ncol = 36)
> play_table = as.data.frame(play_table)
> row.names(play_table) = c("year", "titles", "num_acts", "num_scenes")
> head(play_table)

                                    V1                                       V2
year                              1603                                     1607
titles       ALLS WELL THAT ENDS WELL     THE TRAGEDY OF ANTONY AND CLEOPATRA
num_acts                             5                                        5
num_scenes                           0                                       42
                         V3                        V4
year                   1601                      1593
titles       AS YOU LIKE IT     THE COMEDY OF ERRORS
num_acts                  5                         4
num_scenes               22                         0
                                    V5            V6
year                              1608          1609
titles       THE TRAGEDY OF CORIOLANUS     CYMBELINE
num_acts                             5             5
num_scenes                          29            27
                                                   V7
year                                             1604
titles        THE TRAGEDY OF HAMLET, PRINCE OF DENMARK
num_acts                                            4
num_scenes                                         20
                                                 V8
year                                           1598
titles       THE FIRST PART OF KING HENRY THE FOURTH
num_acts                                          5
num_scenes                                       19
                                        V9
year                                  1598
titles        SECOND PART OF KING HENRY IV
num_acts                                 5
num_scenes                              19
                                          V10
year                                     1599
titles       THE LIFE OF KING HENRY THE FIFTH
num_acts                                    5
num_scenes                                 23
                                           V11
year                                      1592
titles       THE FIRST PART OF HENRY THE SIXTH
```

```
num_acts                                       5
num_scenes                                     0
                                                   V12
year                                              1591
titles       THE SECOND PART OF KING HENRY THE SIXTH
num_acts                                             5
num_scenes                                          24
                                                  V13                    V14
year                                             1591                   1611
titles       THE THIRD PART OF KING HENRY THE SIXTH     KING HENRY THE EIGHTH
num_acts                                            5                         5
num_scenes                                         28                         0
                     V15                               V16
year                1597                              1599
titles       KING JOHN      THE TRAGEDY OF JULIUS CAESAR
num_acts                 5                                 5
num_scenes               0                                18
                                     V17                      V18
year                                1606                     1595
titles        THE TRAGEDY OF KING LEAR     LOVE'S LABOUR'S LOST
num_acts                               5                        5
num_scenes                            26                        9
                                   V19                      V20
year                              1606                     1605
titles       THE TRAGEDY OF MACBETH       MEASURE FOR MEASURE
num_acts                             5                        4
num_scenes                          29                       17
                                  V21                            V22
year                             1597                           1601
titles       THE MERCHANT OF VENICE     THE MERRY WIVES OF WINDSOR
num_acts                            5                              4
num_scenes                         20                              1
                                     V23                          V24
year                                1596                         1599
titles       A MIDSUMMER NIGHT'S DREAM      MUCH ADO ABOUT NOTHING
num_acts                               5                            5
num_scenes                             9                           17
                                                   V25
year                                              1605
titles        THE TRAGEDY OF OTHELLO, MOOR OF VENICE
num_acts                                             5
num_scenes                                          15
                                     V26                  V27
year                                1596                 1593
titles        KING RICHARD THE SECOND      KING RICHARD III
num_acts                               5                    5
```

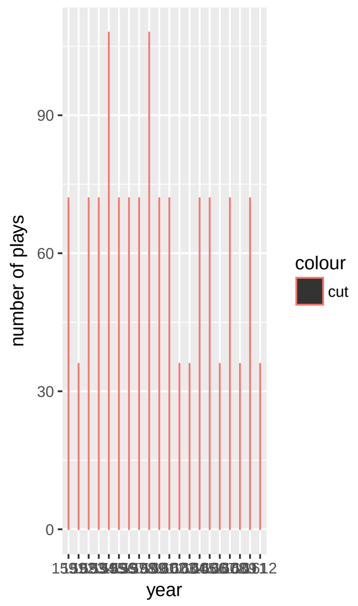
```
num_scenes                             3                    0
                                            V28                             V29
year                                       1595                            1594
titles        THE TRAGEDY OF ROMEO AND JULIET         THE TAMING OF THE SHREW
num_acts                                      5                               4
num_scenes                                   24                              14
                       V30                             V31
year                  1612                            1608
titles       THE TEMPEST     THE LIFE OF TIMON OF ATHENS
num_acts                 5                               5
num_scenes               0                              17
                                           V32
year                                      1594
titles       THE TRAGEDY OF TITUS ANDRONICUS
num_acts                                     4
num_scenes                                  14
                                               V33
year                                          1602
titles       THE HISTORY OF TROILUS AND CRESSIDA
num_acts                                         5
num_scenes                                       0
                                             V34
year                                        1602
titles        TWELFTH NIGHT; OR, WHAT YOU WILL
num_acts                                       5
num_scenes                                    18
                                       V35                   V36
year                                  1595                  1611
titles       THE TWO GENTLEMEN OF VERONA     THE WINTER'S TALE
num_acts                                 5                     5
num_scenes                              20                    15
```

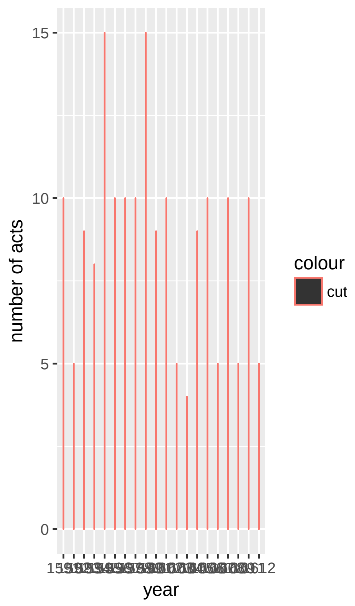
```
> #NOW, put the body information into the dataframe
> play_table = rbind(play_table, body)
> names(body) <- "body"
> #Since the full output would be too long,
> #only show the dimension of final dataframe
> dim(play_table)
```

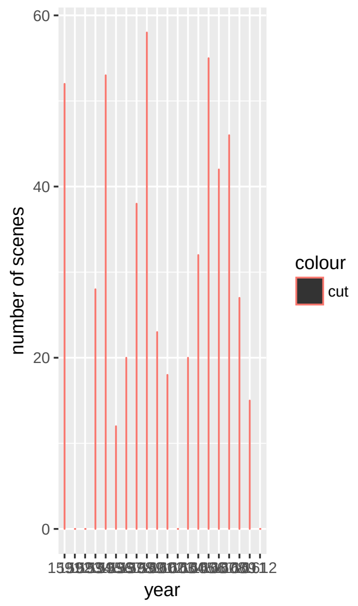
```
[1]  5 36
```

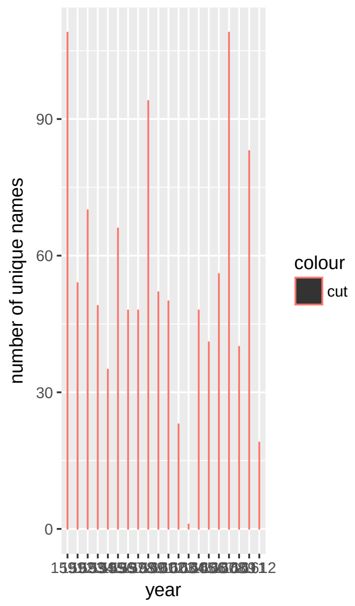
```
> #####============Problem 2(C)======================
> #To make sure this part of processing doesnt interfere the original file,
> #make a copy of the body parts
> clean_file = body
> #Extract the chunk parts after speaker's name
> output_chunk = lapply(clean_file, function(x) str_split(x,
+     "[[:space:]]{2}[[:upper:]]{1}[[:alpha:]]{1,}[[:space:]]{0,}[[:upper:]]{1,}[.]"))
> #Extract stage information
> output_chunk = lapply(output_chunk, function(x) unlist(x)[-1])
> output_chunk = lapply(output_chunk, function(x) x[-(grep("SCENE", x))])
> #Remove trailing spaces
> output_chunk = lapply(output_chunk, function(x) unlist(x)[grepl("[[:alpha:]]", unlist(x))]
> count_chunk = lapply(output_chunk, function(x) length(x))
> #NOTE: several plays do not have universal format, such as play#4
> unlist(count_chunk)

body <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
 933 1213  825   17 1130  879    0    0  916    0  629  810  823  706  538  806
<NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
   0  948  666  910  624 1008  508    0 1190  538 1051    0  823  637  767  573
<NA> <NA> <NA> <NA>
1142  936  872  753

> #Extract speaker names based on two spaces indentation
> output_name = lapply(clean_file, function(x) str_extract_all(x,
+       "[[:space:]]{2}[[:upper:]]{2,}[[:space:]]{0,}[[:upper:]]{0,}[.]"))
> output_name = lapply(output_name, function(x) unlist(x))
> #Extract unwanted names
> output_name = lapply(output_name, function(x)
+   x[-(grep("Exit|ACT|ALL|BOTH|EPILOGUE|Exeunt|SCENE", x))])
> unique_name = lapply(output_name, function(x) unique(x))
> count_name = lapply(unique_name, function(x) length(x))
> #NOTE: several plays without universal format fail to
> #generate names, eg.Romeo and Juliet,
> unlist(count_name)

body <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
  23   56   27   13   58   40    1    0   48   47   54   64   45   48   27   47
<NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
   0   18   41   23   21   25   31    0   25   35   57    0   25   19   51   24
<NA> <NA> <NA> <NA>
  29   21   17   35

> ####============Problem 2(D)===================
> #Get number of lines for each play
> num_lines = lapply(output_chunk, function(x) sum(str_count(x,'[[:alpha:]][.!?]')))
> unlist(num_lines)

body <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
1648 2315 1521  858 2086 2151    0    0 1931    0 1363 1628 1663 1538 1171 1802
<NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
   0 1867 1614 1666 1362 1996 1157    0 2650 1267 1998    0 1480 1261 1642 1307
<NA> <NA> <NA> <NA>
2218 1640 1357 1705
```

```
> #Get number of words for each play
> all_words = lapply(output_chunk, function(x) str_split(x,'[[:space:]\n,.:;!?]'))
> #Remove the ones with only trailing spaces
> all_words = lapply(all_words, function(x) unlist(x)[grepl("[[:alpha:]]", unlist(x))])
> num_words = lapply(all_words, function(x) length(x))
> unlist(num_words)

 body  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
22604 24877 21705  8154 27631 27580     0     0 26018     0 20376 25499 24413
 <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
23892 20203 19689     0 21723 17101 21769 21287 21033 16479     0 26477 21626
 <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
27737     0 18714 16444 18527 20788 25685 19870 17212 24964

> #Get number of unique words
> num_unique_words = lapply(all_words, function(x) length(unique(x)))
> unlist(num_unique_words)

body <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
4053 4761 3770 1899 4810 5064    0    0 4728    0 4350 4777 4197 4360 4073 3355
<NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
   0 4412 3875 3862 3769 3685 3467    0 4376 4244 4729    0 3459 3759 3923 3995
<NA> <NA> <NA> <NA>
4900 3635 3164 4625

> #Compute number of words per chunk
> words_per_chunck = unlist(num_words) / (unlist(count_chunk))
> unlist(words_per_chunck)

     body      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>
 24.22722  20.50866  26.30909 479.64706  24.45221  31.37656       NaN       NaN
     <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>
 28.40393       NaN  32.39428  31.48025  29.66343  33.84136  37.55204  24.42804
     <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>
      NaN  22.91456  25.67718  23.92198  34.11378  20.86607  32.43898       NaN
     <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>      <NA>
 22.24958  40.19703  26.39106       NaN  22.73876  25.81476  24.15515  36.27923
     <NA>      <NA>      <NA>      <NA>
 22.49124  21.22863  19.73853  33.15272

> ##====================Problem 2(E)===========================
> library(ggplot2)
> library(grid)
> library(gridExtra)
> #Create a plot_function to automate ploting for multiple graphs
> plot_func <- function(yearX_axis, var_Yaxis, Yaxis_label){
+   df = data.frame(yearX_axis, var_Yaxis)
+   df = aggregate(var_Yaxis~yearX_axis, data = df, sum)
+   get_plot = ggplot(df, aes(x = yearX_axis, y = var_Yaxis,
+     color = 'cut')) + geom_area() + labs(x = "year", y = Yaxis_label)
+   return(get_plot)
+
+ }
> #year vs number of plays
> plot_1 = plot_func(unlist(year), unlist(num_plays), "number of plays")
> #year vs number of acts
> plot_2 = plot_func(unlist(year), unlist(acts), "number of acts")
> #year vs number of scenes
> plot_3 = plot_func(unlist(year), unlist(scenes), "number of scenes")
> #year vs number of unique names
> plot_4 = plot_func(unlist(year), unlist(count_name), "number of unique names")
> grid.arrange(plot_1, plot_2, plot_3, plot_4, ncol=2)
>
> #Based on the plots, 1595 and 1599 are the two years he wrote the most plays with
> #high number of scenes, acts and unique names. Especially in 1599, he was at a
> #peak time, meaning that his play was the most sophisticated in that year.
> #Interestingly, although 1612 has the lowest number of production in every aespect,
> #he actually maintained a good amount of production right before 1612.
>
```









=================Problem 2(F)=======================

1. I split plays based on "THE END" rather than by year, which gave an accurate result 2. When extracting the number of acts, I used length and unique function rather than only regular expression to get the optimal output. 3. My dataframe created in 2(b) is a bit different. Rather than making attributes as variables, I used each play as a variable. In this case, the data strcutre is a nested list in the dataframe. I could simply extract one varibale to analyze

every attribute of it, which is neat and clean. 4. When extracting chunks and names separately, I noticed that there're irregular patterns. The speaker may have a more than one-word name. "FIRST LORD." My solution avoided missing speaker names under this scenario. 5. I also did a second check "grepl" to remove all the trailing spaces so that unique names won't count "" as a unique term 6. I didn't use any for loop for text processing, but only apply functions which make my program more efficient and run faster. 7. Rather than ploting graphs one by one, I create a plot function and only pass in vectors x and y axis, and the label string I need. This automated the ploting proces and removed tedious code, also ran faster.

```
> #######3(A)
> #Class List: PLAY, SubPlay, PlaySummary, MetaData       *MetaData is a "tool class"
> #Inhearitance Structure: PLAY <-  SubPlay <- PlaySummary   *MetaData Class has no inherita
>
>
> #Class PLAY
> #Fileds: Name(string), txt_input(a text file / character vector)
> #Methods: trim_plays(txt_input, trim_method) -> A list, each element is one play
>           #num_plays(play_list) -> total number of plays
>
>
> #Class SubPlay (Inherit from PLAY Class)
> #Fileds: PLAY.Name (string, get from PLAY class), PLAY.List (list, get from PLAY class), P
> #Methods: body(PLAY.list) -> A list, each element is a play's body
>           #split_chunk (PLAY.list) -> A list, each element is a play's chunk
>           #split_names (PLAY.List) -> A list of speaker's names
>           #unique_names (Name.List) -> A list of unique names in each play
>          #Meta.year(PLAY.list) -> A list, each element is the year of a play
>          #Meta.acts(PLAY.list) -> A list, each element is the number of acts of a play
>          #Meta.scenes(PLAY.list) -> A list, each element is the number of scenes of a play
>          #Make_dataframe(year, acts, scenes, body) -> A dataframe with all the metadata
>
> #Class Meta (Auxiliary Class, called by other classes)
> #Fields: Play(character string), Chunk(character string )
> ##Methods:
>          #year(Play) -> return year
>          #acts(Play) -> return number of acts
>          #scenes(Play) -> return number of scenes
>
>
>
> #Class PlaySummary (Inherit from SubPlay Class)
> #Fileds DataFrame(a dataframe of all metadatas )
> #chunk_list(list, chacteracter vector)
> #Methods:
```

```
>         #num_words(chunk_list) -> return number of words per play
>         #num_lines(chunk_list) -> return number of lines per play
>         #multi_plots(dataframe)  -> return a series of plots regarding the trends in plays
>
```