Code

---

title: 'Assignment #3 Text Analysis'

author: "<Peter Leng>"

date: "10/8/2017"

output: html\_document

runtime: shiny

---

```{r setup, include=FALSE}

install.packages("knitr")

install.packages("ggplot2")

knitr::opts\_chunk$set(echo = TRUE)

getOption("rstudio.markdownToHTML")

```

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the \*\*Knit\*\* button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document.

1. ) [10 pts.] Download the works of Sir Arthur Conan Doyle. Use documentation on the âgutenbergrâ package to help with this.

```{r}

library(tidytext)

library(gutenbergr)

library(dplyr)

#<YOUR CODE HERE>

library(stringr)

gutenberg\_works()

summary(gutenberg\_authors)

doyle <- gutenberg\_works(author == "Doyle, Arthur Conan") %>%

gutenberg\_download(meta\_fields = "title")

```

2.) [15 pts.] Create a plot of the top 15 most relevant words in each book. Annotate each step of your code.

```{r}

##install.packages("magrittr")

library(magrittr)

##install.packages("gutenbergr")

library(gutenbergr)

library(tidytext)

library(ggplot2)

library(dplyr)

## The code below takes the dataframe doyle and seperates the words in text into tokens and it does that in each title

title<- gutenberg\_works(author == "Doyle, Arthur Conan") %>%gutenberg\_download(meta\_fields = "title")

doyle\_words <- doyle %>% unnest\_tokens(word, text) %>%

count(title, word, sort = TRUE) %>% ungroup()

doyle\_words

## The code here starts by binding term frequency with inverse document frequency per word in each title and arange them in descending order, and then creates a coloumn of the words in the 10 books of the author listed below

doyle\_words %>% bind\_tf\_idf(word,title,n) %>% arrange(desc(tf\_idf))

plot\_doyle <- doyle\_words %>% bind\_tf\_idf(word, title, n) %>% arrange(desc(tf\_idf)) %>%

mutate(word = factor(word, levels = rev(unique(word))))%>%

mutate(title = factor(title, levels = c("The New Revelation", "The Vital Message", "The Wanderings of a Spiritualist", "The Coming of the Fairies", "The Crime of the Congo", "The Lost World", "The German War", "The Memoirs of Sherlock Holmes", "The Great Boer War", "The Poison Belt")))

library(ggplot2)

## Draws the plot of the most common 15 words in each book

plot\_doyle %>%

group\_by(title) %>%

top\_n(15, tf\_idf) %>%

ungroup() %>%

mutate(word = reorder(word, tf\_idf)) %>%

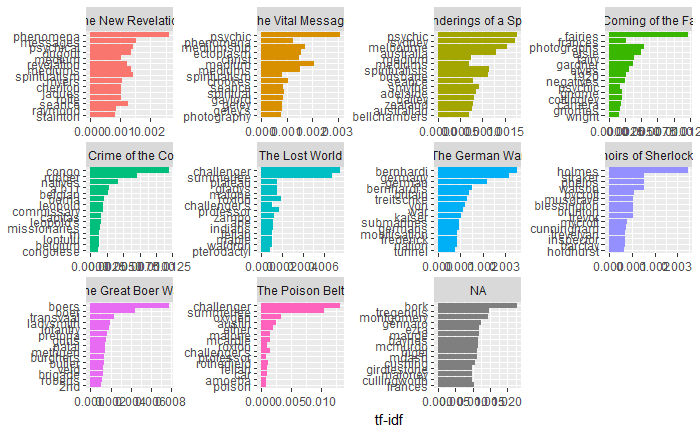
ggplot(aes(word, tf\_idf, fill = title)) +

geom\_col(show.legend = FALSE) +

labs(x = NULL, y = "tf-idf") +

facet\_wrap(~title, ncol = 4, scales = "free") +

coord\_flip()



```

3.) What are the most common streets and manors mentioned in Doyleâs books? Annotate your code.

```{r}

library(tidyr)

##Get all the books by Arthur Conan Doyle

doyle <- gutenberg\_works(author == "Doyle, Arthur Conan") %>%

gutenberg\_download(meta\_fields = "title")

##Put it into a Bigram

doyle\_bigrams <- doyle %>%

unnest\_tokens(bigram, text, token = "ngrams", n = 2)

##Count the Bigram

doyle\_bigrams %>%

count(bigram, sort = TRUE)

##Seperate the bigram by word1 and word 2

bigrams\_separated <- doyle\_bigrams %>%

separate(bigram, c("word1", "word2"), sep = " ")

##Put in the stop word

bigrams\_filtered <- bigrams\_separated %>%

filter(!word1 %in% stop\_words$word) %>%

filter(!word2 %in% stop\_words$word)

##Assign word 2 to be street and count it

bigrams\_filtered %>%

filter(word2 == "street") %>%

count(title, word1, sort = TRUE)

##the most common streets are baker fenchurch and goldlphin

##Assign word 2 to be manor and count it

bigrams\_filtered %>%

filter(word2 == "manor") %>%

count(title, word1, sort = TRUE)

##the most common manors are tilford thorpe and birlstone

```

4.) [20 pts.] Choose one or two of Doyleâs books and try to divide it/them into meaningful topics. Are you able to describe any of the topics? Annotate your code and write your answer as annotation in the code.

```{r}

library(ggplot2)

library(topicmodels)

#downloaded a shelock holmes book from the work of the author Doyle

book <- gutenberg\_download(c(108), meta\_fields = "title")

#this code unnest the text to words

book\_words <- book %>% unnest\_tokens(word, text) %>%

count(title, word, sort = TRUE) %>% ungroup()

#get rid of stop words

doyle\_words <- anti\_join(book\_words, stop\_words)

word\_counts.df <- doyle\_words %>%

count(title, word, sort = TRUE) %>%

ungroup()

#converts the dataframe doyle\_words to document term matrix which is required for topic modeling

doyle.dtm <- cast\_dtm(doyle\_words, title, word, n)

doyle\_lda <- LDA(doyle.dtm, k = 2, control = list(seed=1234))

doyle\_lda

#tidies up the data frame

doyle\_topics <- tidy(doyle\_lda, matrix = "beta")

doyle\_topics

#finds the 10 common terms in each topic

doyle\_top\_terms <- doyle\_topics %>%

group\_by(topic) %>%

top\_n(10, beta) %>%

ungroup() %>%

arrange(topic, -beta)

#creates a plot with the 10 most common terms for each topic

doyle\_top\_terms %>%

mutate(term = reorder(term, beta)) %>%

ggplot(aes(term, beta, fill = factor(topic))) +

geom\_col(show.legend = FALSE) +

facet\_wrap(~topic, scales = "free") +

coord\_flip()

```

