Assignment 6

Max Wagner

March 7, 2016

PS1.

1. The number of outcomes is 216 for a fair dice rolled three times.

2. The potential outcomes for two rolls is 36, and there are 2 potential rolls that give a sum of 3. That means the chances of getting a sum of 3 is 0.0555556.

3. .

For 25 people in a room:

```
pbirthday(25) # There's a base function that does this, weirdly enough
```

```
## [1] 0.5686997
```

```
1 - prod(1 - (0:24)/365) # testing with my own math
```

```
## [1] 0.5686997
```

For 50 people in a room:

```
pbirthday(50) # There's a base function that does this, weirdly enough
```

```
## [1] 0.9703736
```

```
1 - prod(1 - (0:49)/365) # testing with my own math
```

```
## [1] 0.9703736
```

PS2. For part 1 we are looking at just one word, so it's not too complicated.

```
library("stringr")

singleProb <- function(corpus){
  corp <- iconv(corpus, "UTF-8", "ASCII", sub="?") # convert to ASCII in order to use gsub
  corp <- gsub("[^[:alpha:]]", "", corp) # remove punctuation
  corp <- tolower(corp) # make lowercase
  corp <- corp[corp != ''] # remove first blank line
  prob <- as.data.frame(table(corp)) # make it a data frame
  prob$prob <- prob$Freq / sum(prob$Freq) # add a column to sort by
  return(prob)
}

corpus <- scan('https://raw.githubusercontent.com/maxwagner/605/master/assign/assign6.sample.txt', chara
probability <- singleProb(corpus)
head(probability) # sample output
```

```
##        corp Freq         prob
## 1         a   45 0.0337331334
## 2     about    6 0.0044977511
## 3 abundance    1 0.0007496252
## 4  abundant    1 0.0007496252
## 5     abuse    3 0.0022488756
## 6   abysmal    1 0.0007496252
```

The two word one is a bit more complicated.

```
doubleProb <- function(corpus, word1, word2) {
  corp <- iconv(corpus, "UTF-8", "ASCII", sub="?") # convert to ASCII in order to use gsub
  corp <- gsub("[^[:alpha:]]", "", corp) # remove punctuation
  corp <- tolower(corp) # make lowercase
  corp <- corp[corp != ''] # remove first blank line

  prob_frame <- data.frame(corp) # make a frame with just words in it
  prob_frame$prev <- c(NA, corp[1:length(corp)-1]) # check whats previous to the word
  prob_frame <- prob_frame[2:nrow(prob_frame),] # check whats after the word

  prob_frame$match <- ifelse((prob_frame$corp == word1 & prob_frame$prev == word2) | (prob_frame$corp ==
  return(sum(prob_frame$match) / nrow(prob_frame))
}

corpus <- scan('https://raw.githubusercontent.com/maxwagner/605/master/assign/assign6.sample.txt', chara
doubleProb(corpus, "to", "the") # sample output 1
```

```
## [1] 0.0007501875
```

```
doubleProb(corpus, "to", "cat") # sample output 2
```

```
## [1] 0
```