The University of British Columbia

Data Science 581 Modelling and Simulation II

Lab Assignment 3

Submit your answers to questions 8, 9(d) and 10 to Canvas.

1. The file *text.txt* contains some technical writing with punctuation stripped away, so it is just a sequence of words. You can read in the file with the `readLines()` function:

```
MyText <- readLines("text.txt")
```

2. R has some functions for handling character strings, including `strsplit` which splits strings at given characters. The following separates strings at spaces, resulting in lists of single words:

```
MyTextStrings <- strsplit(MyText, " ")
```

   We will find it easier to work with a single vector of the words, so we use the `unlist()` function to remove the list structure, resulting in a plain vector of strings (none of which contain blank spaces):

```
MyTextStrings <- unlist(MyTextStrings)
```

   Finally, to do some Markov chain analysis of the text strings, we might be interested in the lengths of each word. We can count the number characters in each string using the `nchar()` function:

```
lettercounts <- nchar(MyTextStrings)
```

   To see the lengths of the first few words in the document, try

```
lettercounts[1:12]

##  [1]  3  4  2 13  2  2  7  3 10  2  3  5
```

   Later, we will set up a Markov chain which models the sequence of word lengths. States for this Markov chain will be the word lengths, so the state space will be all possible word lengths.

3. Use the `table()` function to determine the number of words exceeding 12 characters, for example, and to see if there are any blank spaces remaining (and there are).

   The following code will remove the remaining blank spaces:

```
lettercounts <- lettercounts[lettercounts > 0]
```

4. We will use the following truncation to reduce the size of the state space:

```
lettercountsT <- lettercounts
lettercountsT[lettercounts > 11] <- 12
```

   This means that the state '12' actually contains 13 and 14 as well as 12. Thus, our state space is now $\{1, 2, 3, \ldots, 12\}$. We only need to estimate 144 elements of our transition matrix instead of 196, so we gain some accuracy by giving up this degree of precision.

5. Construct the transition matrix as follows:

```
P <- matrix(0, nrow=12, ncol=12)
for (i in 2:length(lettercountsT)) {
  P[lettercountsT[i-1], lettercountsT[i]] <- P[lettercountsT[i-1],lettercountsT[i]] + 1
}
P <- P/as.numeric(table(lettercountsT[-length(lettercountsT)]))
length(table(lettercountsT[-length(lettercountsT)]))

## [1] 12
```

Note that we are estimating the entries of the transition matrix by calculating the proportion of the time that each type of transition occurs. In calculating these proportions, we are using the `table()` function to determine the numbers of times that we are in each state, and since we don't know what comes after the final observation, we do not include that in our count.

6. Find the mean and standard deviation of the word lengths in the original data set as well as the proportion of the time that the difference in subsequent word lengths exceeds 7, i.e.

```
mean(diff(lettercountsT) > 7)

## [1] 0.05199516
```

These are examples of statistics that could be used to make comparisons with other samples of text, e.g. in cases where one might be checking for forgeries.

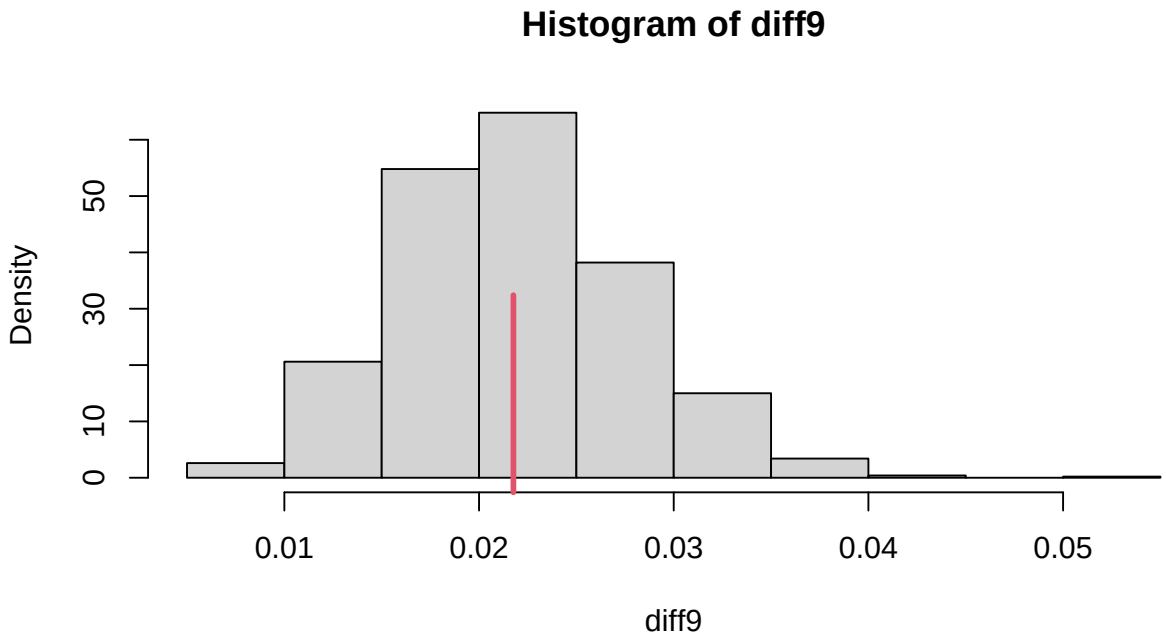
7. To check another text to see if its authorship would be different from the given text, we might run a simulation as follows:

```
wordlengthAVGs <- numeric(1000)
wordlengthSDs <- numeric(1000)
diff9 <- numeric(1000)
for (i in 1:1000) {
    Ntransitions <- length(lettercounts) # number of words
    wordlength <- numeric(Ntransitions)#initializing the Markov chain
    current.state <- lettercountsT[1] # initial wordlength
    for (j in 1:Ntransitions) {
        current.state <- sample(1:12,
           size = 1, prob = P[current.state, ])
        wordlength[j] <- current.state
    }
wordlengthAVGs[i] <- mean(wordlength)
wordlengthSDs[i] <- sd(wordlength)
diff9[i] <- mean(abs(diff(wordlength)) > 9)
}
```

In the above simulation, we have simulated 1000 realizations of the fitted Markov chain model. In each case, we have calculated the mean and standard deviation of the wordlength, as well as the proportion of time the absolute value of the subsequent difference in wordlength changes by more than 9 characters, a fairly extreme type of statistic.

The following is a histogram of the extreme changes, together with a rug plot indicating the location of the observed data:

```
hist(diff9, freq=FALSE)
rug(mean(abs(diff(lettercountsT)) > 9), col=2, lwd=3, ticksize=.5)
```

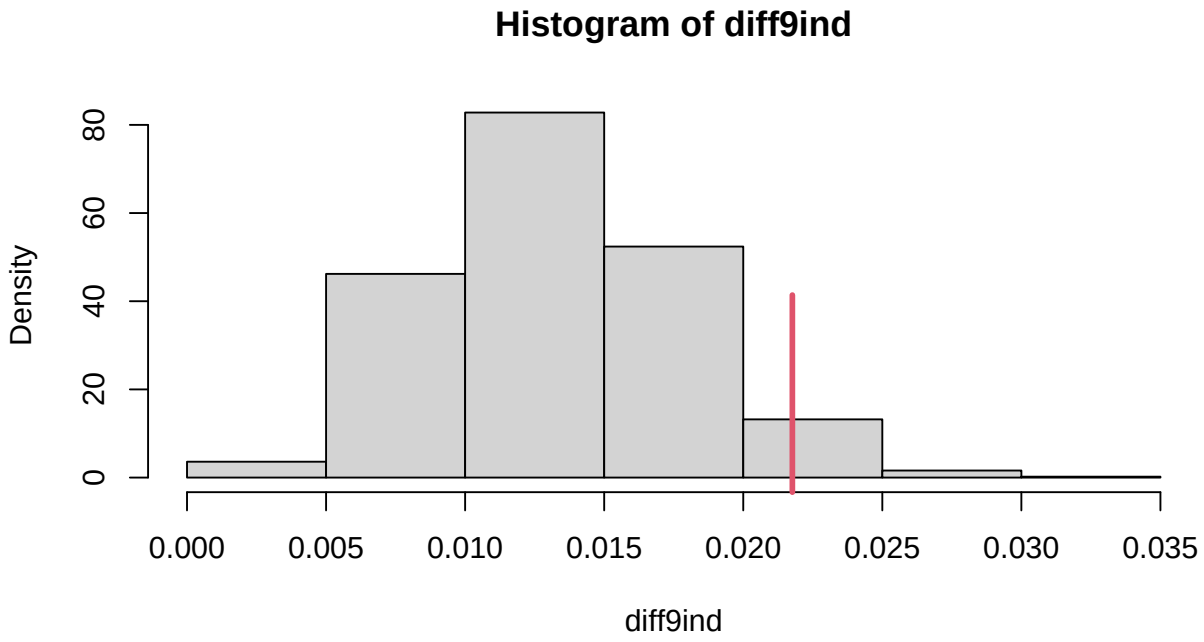


Note how the statistic lies in an area of high probability density. This is an indicator that our model is fitting well, and we would be suspicious of any text where the extreme statistic was larger than .04 or less than .01.

But was it necessary to use a Markov chain model? What if we just sample from the marginal distribution of the observed states as below (assuming that the changes in wordlength are independent from word to word)?

```
diff9ind <- numeric(1000)
for (i in 1:1000) {
    indepcounts <- sample(1:12, size=length(lettercounts),
         replace=TRUE, prob=table(lettercountsT)/
         length(lettercounts))
    diff9ind[i] <- mean(abs(diff(indepcounts)) > 9)
}
```

```
hist(diff9ind, freq=FALSE)
rug(mean(abs(diff(lettercountsT)) > 9), col=2, lwd=3, ticksize=.5)
```



Note how the observed value of the extreme statistic is no longer in an area of high probability density. We would judge the actual author as a forger. We would also fail to detect forgery in a work where the extreme statistic is less than .01 – a further indication of inaccuracy.

8. Repeat the above graphical analyses in the cases of the mean and the standard deviation. What rule would you use to identify a forgery on the basis of the mean wordlength? How about on the basis of the standard deviation of the wordlength?

9. ~~**Problem from Lecture 3.**~~ At the beginning of each day, a batch of containers arrives at a stockyard having capacity to store 6 containers. The batch size has the discrete probability distribution $\{q_0 = .4, q_1 = 0.3, q_2 = 0.2, q_3 = 0.1\}$. If the stockyard does not have sufficient space to store the whole batch, the batch as a whole is taken elsewhere. Each day, as long as there are containers in the stockyard, exactly one container is removed from the stockyard.

   (a) Find the transition matrix for the Markov chain $\{X_1, X_2, \ldots\}$, where X_t = the number of containers in the stockyard at the beginning of the tth day.

   (b) Find the long run distribution for this Markov chain.

   (c) Suppose a profit of \$100 is realized for each container that spends a night at the stockyard. Calculate the long-run average weekly profit.

   (d) Write R code to simulate this Markov chain, and run a simulation of 100000 transitions, starting in state 1 (i.e. where there is 1 container in the yard).

10. Consider the Markov chain $X_0, X_1, X_2, \ldots$, with transition matrix and state space $\{1, 2, 3\}$ and

$$\mathbf{P} = \begin{bmatrix} 0.5 & 0 & 0.5 \\ 0.25 & 0.75 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

   (a) Find the probability that $X_2 = 1$, given that $X_0 = 2$.

   (b) Do states 1 and 3 communicate? Explain briefly.

(c) Use the fact that

$$\mathbf{P}^5 = \begin{bmatrix} 0.2891 & 0.5703 & 0.1406 \\ 0.2842 & 0.5732 & 0.1426 \\ 0.2852 & 0.5664 & 0.1484 \end{bmatrix}.$$

to find the probability that $X_5 = 3$, given that $X_0 = 1$.

(d) Is $\mathbf{P}$ a regular matrix?

(e) Find the stationary distribution for the given Markov chain.