Homework2

Hsueh-Pin Liu

2023-02-08

Problem 1

Google Trends (https://www.google.com/trends/) provides information about popularity of search phrases. You can enter your phrase in the search box and from the drop down menu your can choose your time frame. For less than 90 days, you can view the daily data, otherwise you will see the weekly or monthly data. You can download the data in csv format by clicking on the [+] symbol from the top right corner (you need to be logged into your Google account). Here is the daily time series plot for the frequency of the search term "Super Bowl" in the last 30 days.

Your job is to think of three search terms each with the following characteristics: 1. a time series with a clear trend component. 2. a time series with a clear seasonal component. 3. a time series with no clear trend or seasonal pattern and is poorly described by a combination of trend and seasonal components. In each case download and plot the data. Explain why the data satisfies the requirements as stated above.

```
library(TSA)
```

```
##
## Attaching package: 'TSA'

## The following objects are masked from 'package:stats':
##
##     acf, arima

## The following object is masked from 'package:utils':
##
##     tar
```

```
library(readr)
```

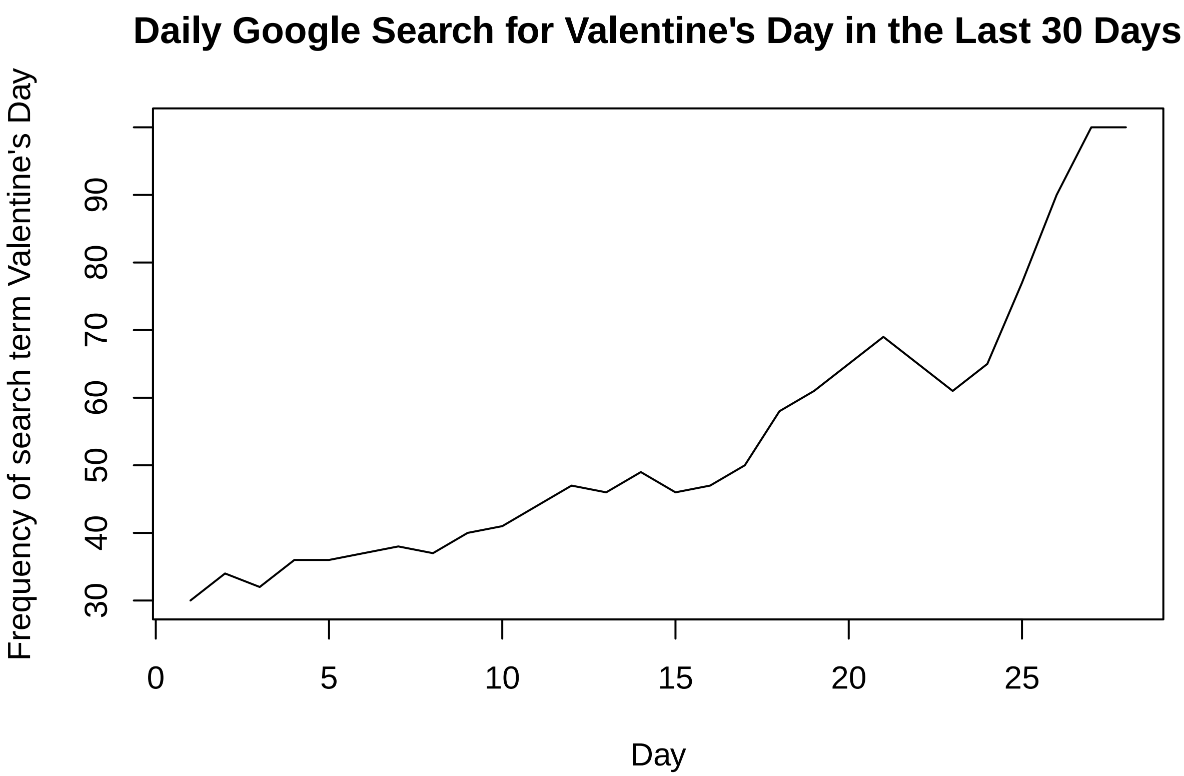
```
##
## Attaching package: 'readr'

## The following object is masked from 'package:TSA':
##
##     spec
```

```
p1<- read_csv("multiTimeline 1.csv")
```

```
## Rows: 28 Columns: 2
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (1): Valentine’s Day: (United States)
## date (1): Day
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
plot.ts(p1[,2],ylab="Frequency of search term Valentine's Day",xlab="Day")
 title("Daily Google Search for Valentine's Day in the Last 30 Days")
```

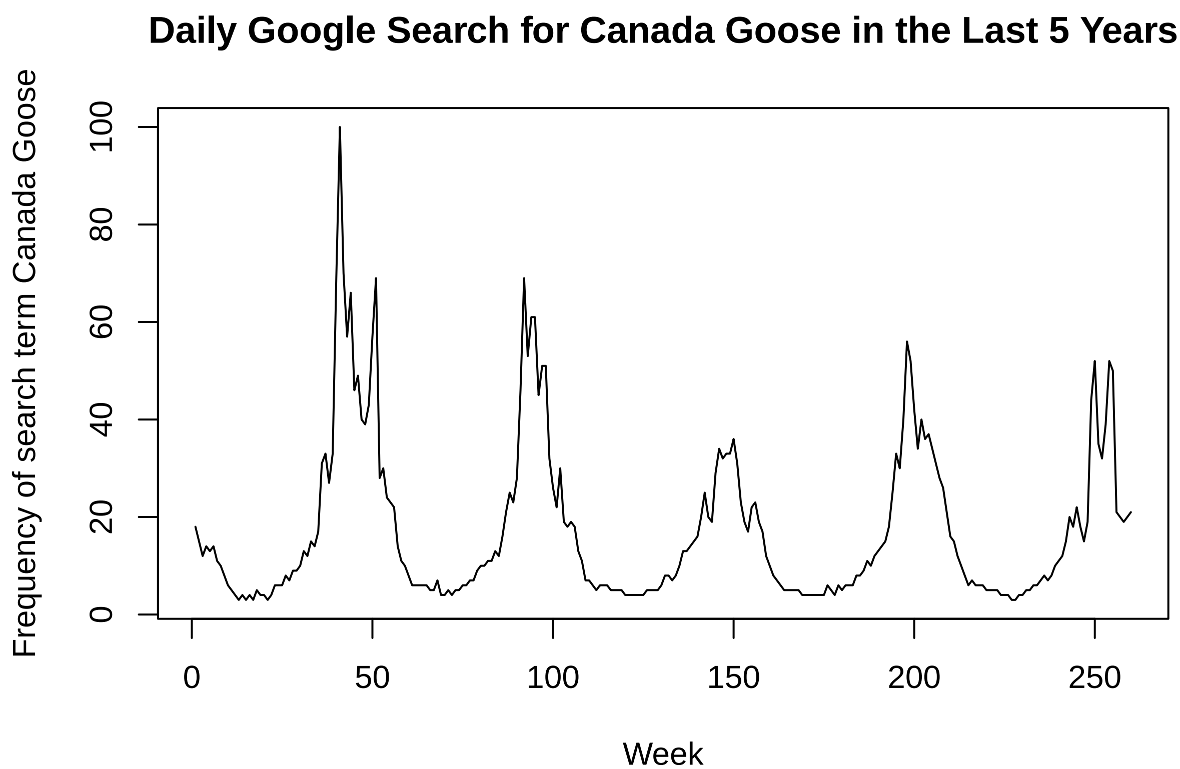


As the plot shows, the trend is increasing.

```
p2 <- read_csv("multiTimeline 2.csv")
```

```
## Rows: 260 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (1): Canada Goose: (United States)
## date (1): Week
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
plot.ts(p2[,2],ylab="Frequency of search term Canada Goose",xlab="Week")
 title("Daily Google Search for Canada Goose in the Last 5 Years")
```

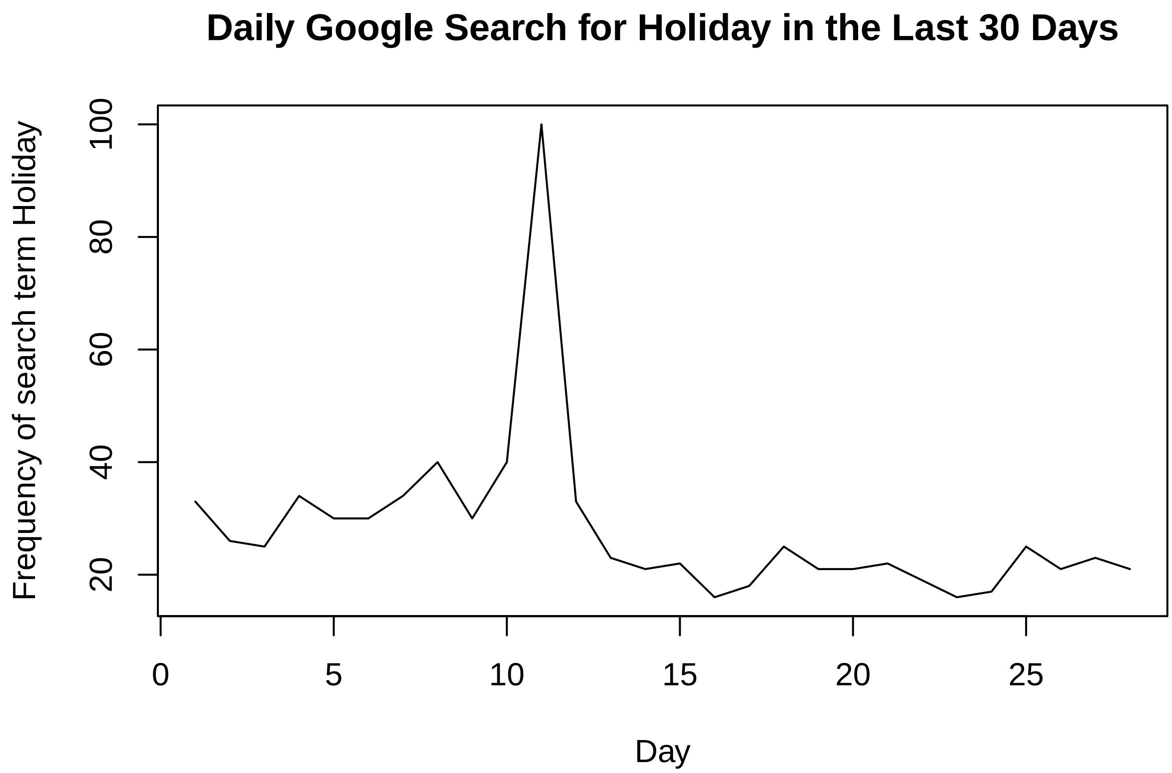


As the plot shows, people will start searching Canada Goose when it's near winter.

```
p3<- read_csv("multiTimeline 3.csv")
```

```
## Rows: 28 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (1): Holiday: (United States)
## date (1): Day
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
plot.ts(p3[,2],ylab="Frequency of search term Holiday",xlab="Day")
 title("Daily Google Search for Holiday in the Last 30 Days")
```



We can see neither a clear trend component nor a time series with a clear seasonal component in the plot.

Problem 2

National Oceanic and Atmospheric Administration (NOAA) is a great resource for atmospheric time series data. Explore the website https://www.ncdc.noaa.gov/cdoweb/. You will find a wealth of time series data on climate. I have downloaded data related to recent climate patterns in Boston (2000 onwards) and it is available as a csv file "BostonClimateData.csv". Load the data in R. You will see that the data was collected from multiple weather stations in Massachusetts. Let's focus on the data collected at the Boston Logan Airport.

The data has four variables. TPCP - Total precipitation amount for the month (tens of mm) TSNW - Total snowfall amount for the month (mm) MMXT - Monthly mean maximum temperature (tenths of degrees of Celsius) MNTM -Monthly mean temperature (tenths of degrees of Celsius)

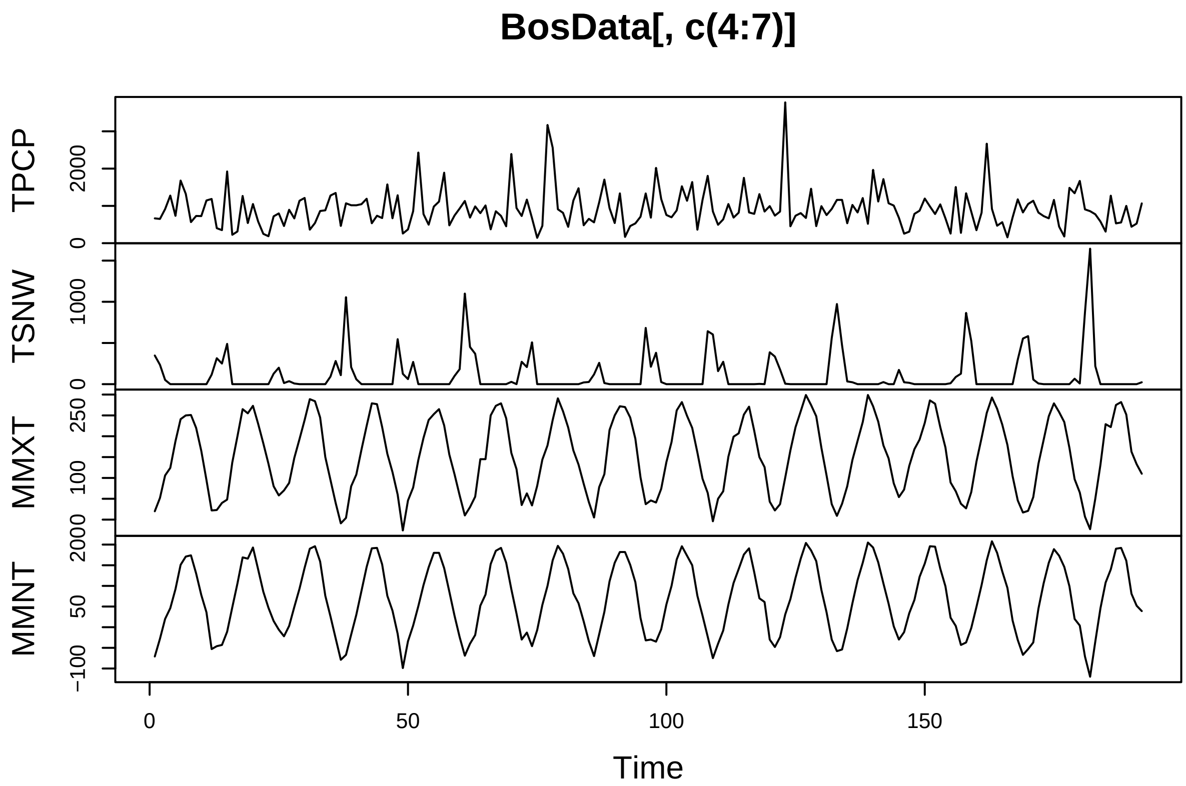
Plot the time series for each variable and write a short description of the key features of each series. What proportion of months had total snowfall greater than a foot? What proportion of months mean maximum temperature exceeded 80F?

```
BosClimateData <- read_csv("BosClimateData.csv")
```

```
## Rows: 6877 Columns: 7
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): STATION, STATION_NAME
```

```
## dbl (5): DATE, TPCP, TSNW, MMXT, MMNT
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
BosData=subset(BosClimateData,BosClimateData$STATION_NAME=="BOSTON LOGAN INTERNATIONAL AIRPORT MA US")
plot.ts(BosData[,c(4:7)])
```



For TCCP, the trend is not so obvious. For TSNW, it has periodicity. For MMXT and MMNT, the figure doesn't change much as time changes.

```
length(subset(BosData$TSNW,BosData$TSNW>304.8))/length(BosData$TSNW)
```

```
## [1] 0.125
```

```
length(subset(BosData$MMXT,BosData$MMXT>80))/length(BosData$MMXT)
```

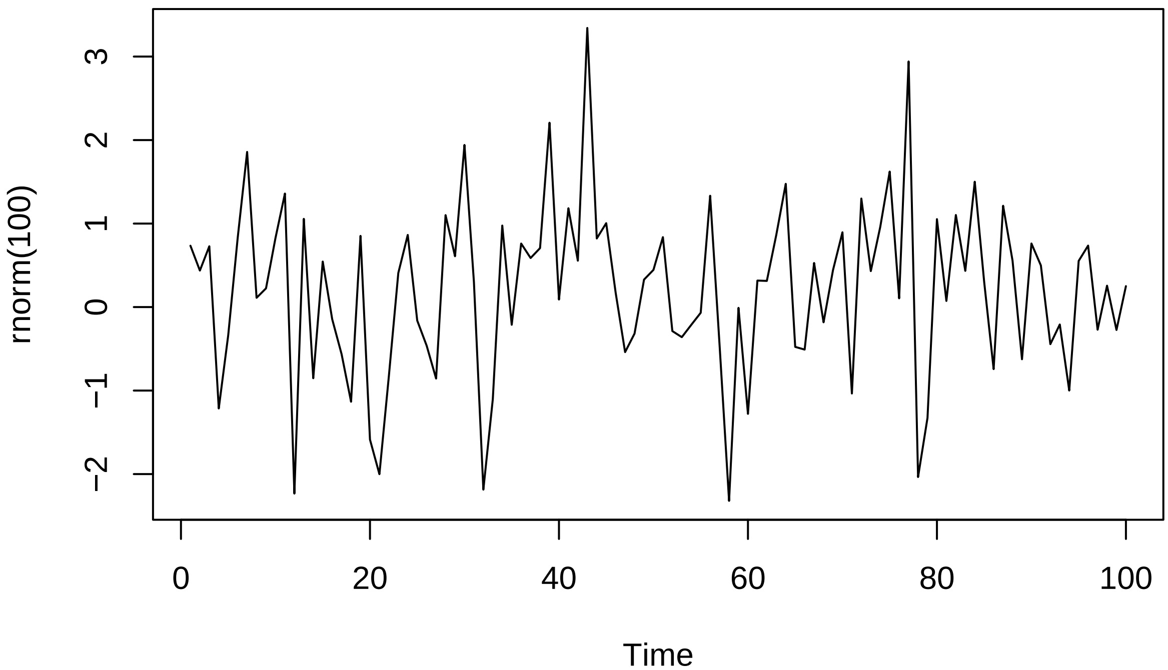
```
## [1] 0.7083333
```

A:12.5% months had total snowfall greater than a foot. 70.8% months mean maximum temperature exceeded 80°F.

Problem 3

a. Simulate a completely random process of length 100 with independent standard normal values. Plot the time series. Does it look "random"? Repeat this exercise a few times with a new simulation each time.

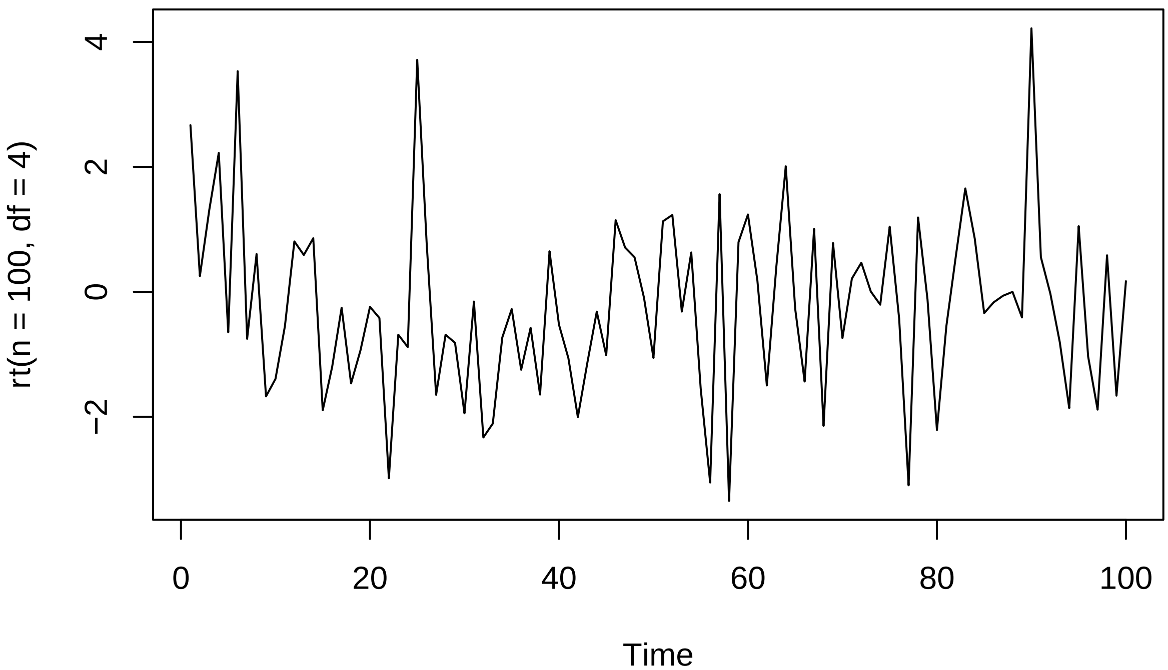
```r
plot.ts(rnorm(100))
```



A:Everytime the plot is different, so it seems random.

b. Repeat the problem in (a), except this time use random samples from a t-distribution with 4 df (replace rnorm(100) by rt(n=100, df=4)). If you didnít know that the data was generated from independent t-distribution, what feature of the data might have made you think that the data generating process may not be Gaussian?

```r
plot.ts(rt(n=100, df=4))
```



A:T-distributed process have heavier tails compared to standard normal process, so it's more likely to observe extreme values in the tails.