STAT 443: Lab 1

Flora Zhang 52135365

17 January, 2022

Question 1

(a)

```
# read the data
dat <-  read_csv("LakeLevels.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Date = col_character(),
##   LakeLevel = col_double()
## )
```

```
head(dat, 10)
```

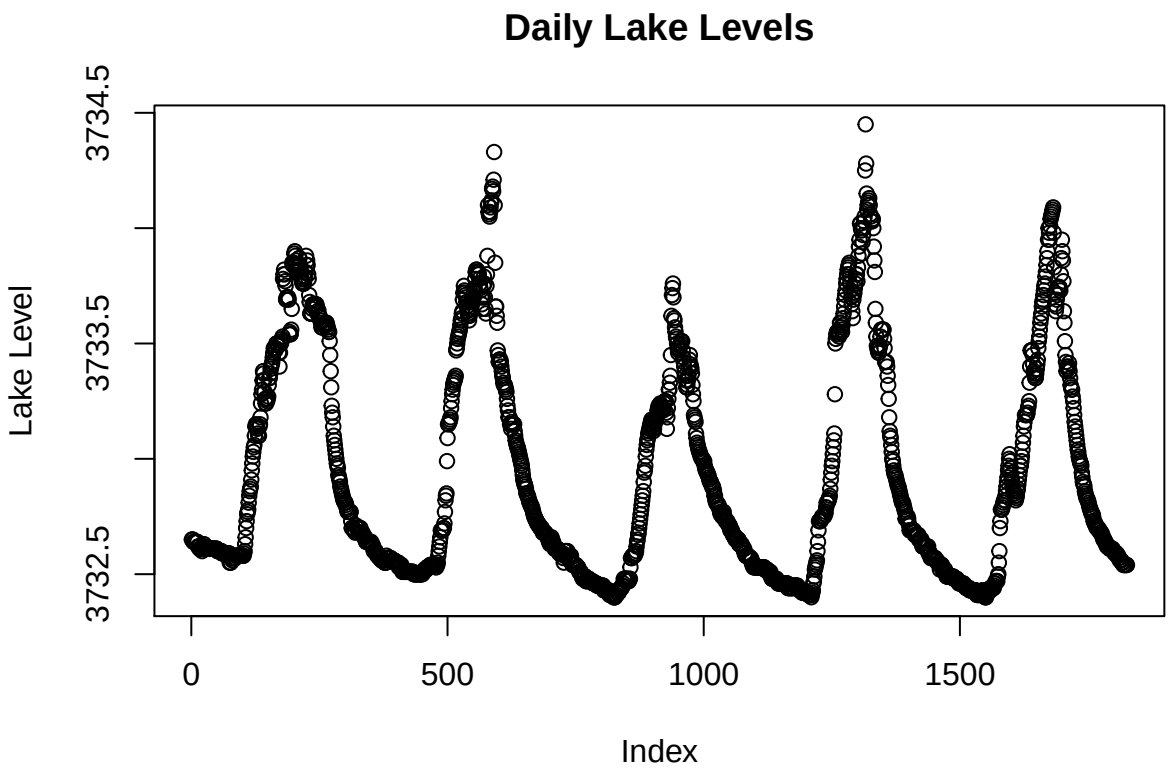
```
## # A tibble: 10 x 2
##    Date      LakeLevel
##    <chr>         <dbl>
##  1 1/1/2007      3733.
##  2 1/2/2007      3733.
##  3 1/3/2007      3733.
##  4 1/4/2007      3733.
##  5 1/5/2007      3733.
##  6 1/6/2007      3733.
##  7 1/7/2007      3733.
##  8 1/8/2007      3733.
##  9 1/9/2007      3733.
## 10 1/10/2007     3733.
```

```
names(dat)
```

```
## [1] "Date"      "LakeLevel"
```

```
dl = dat$LakeLevel

#create plot
plot(dl, main = "Daily Lake Levels", ylab = "Lake Level")
```



This plot differs from one we would like to create as it is not continuous.

(b)

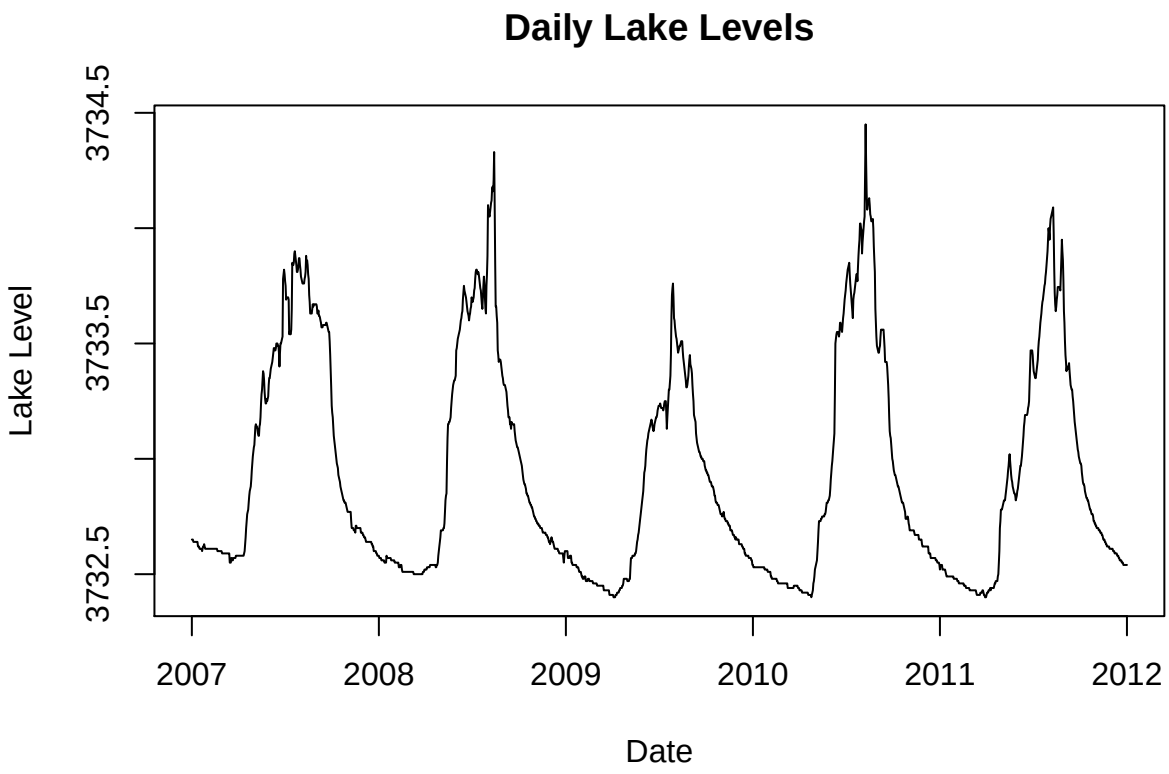
```
# check the time series object
is.ts(dat)
```

```
## [1] FALSE
```

```
x = ts(data=dat$LakeLevel, start = c(2007, 1), frequency = 365)
```

(c)

```
# plot the data
plot(x, main = "Daily Lake Levels", ylab = "Lake Level", xlab = "Date")
```



This new time series has a smoothed, continuous line and visually, looks better. The x variables have now been properly switched to the corresponding year and date.

Question 2

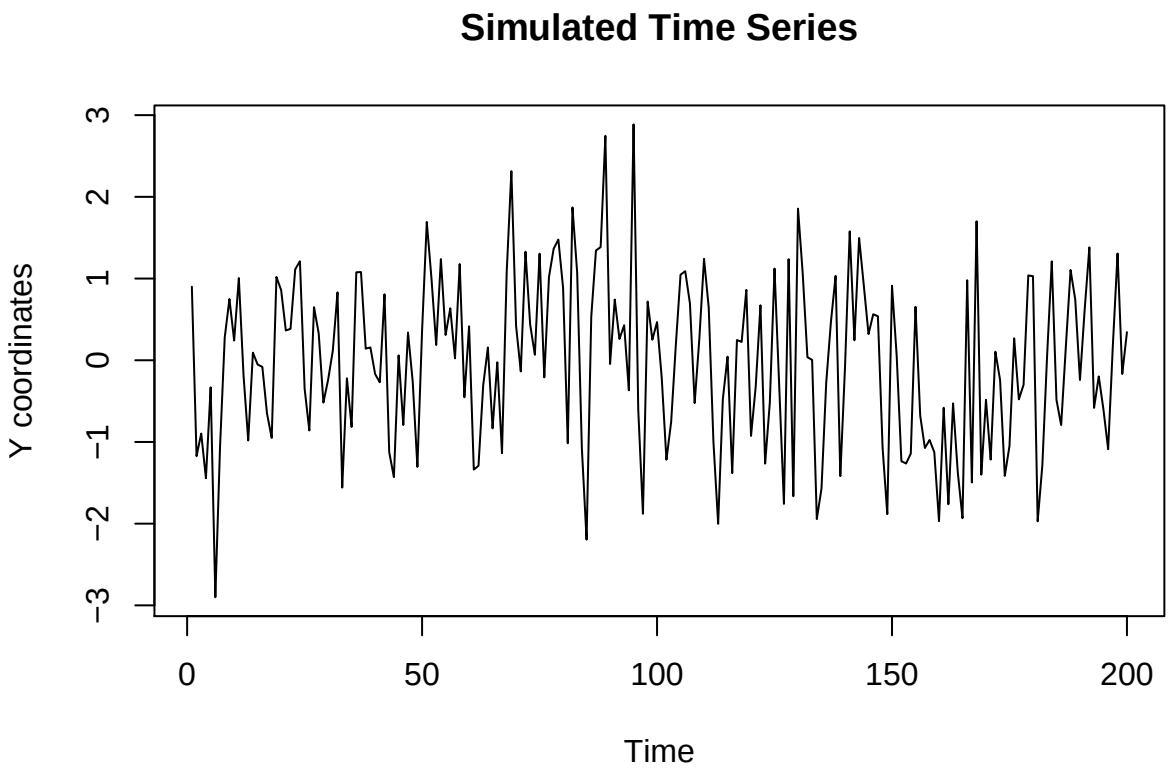
(a)

```
# 200 independent observations
set.seed(2022)
x = rnorm(200)

#create ts
y = ts(x)
```

(b)

```
# plot simulated time series
plot(x = y, ylab = "Y coordinates", main = "Simulated Time Series")
```



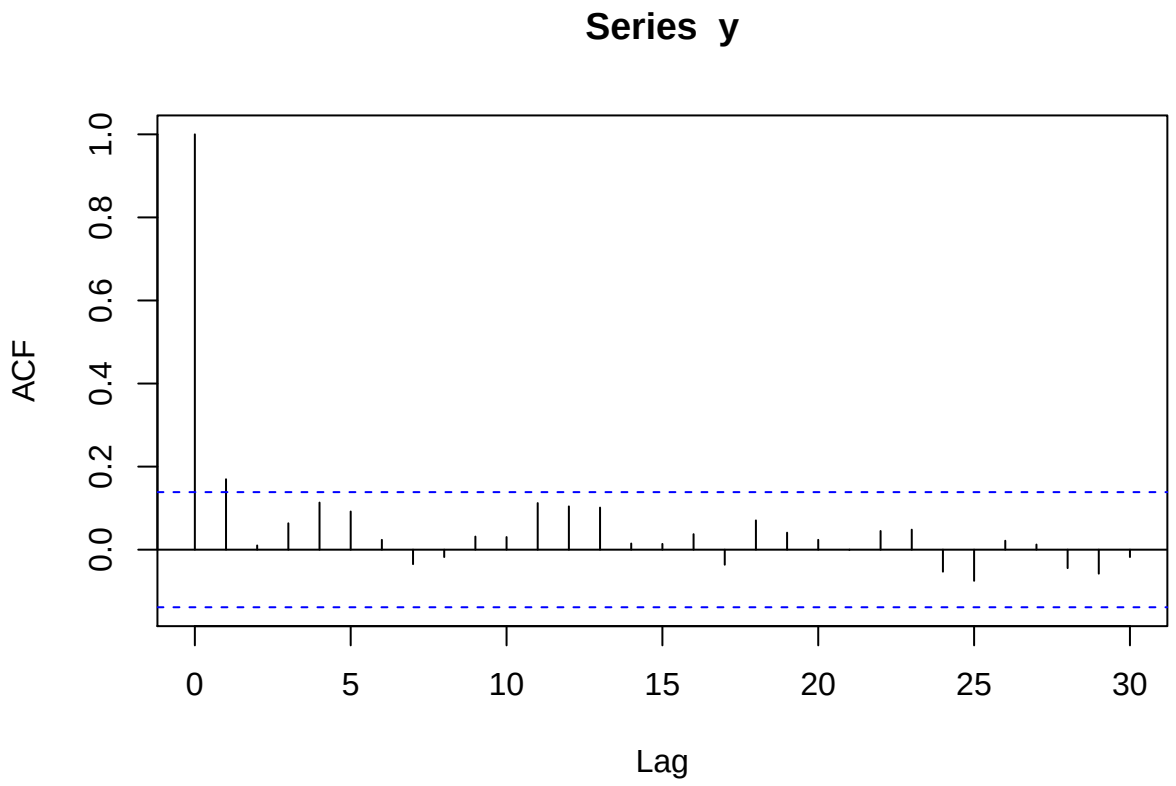

```
# check the outliers
sum(abs(y) > 2)
```

```
## [1] 6
```

There are 6 observations outside the range ± 2. We would expect a small number of values to be outside ± 2.

(c)

```
# create sample autocorrelation function
acf(y, lag.max = 30)
```



The sample acf has a lag of 30, and shows an alternating pattern.

More information on R Markdown

This is an R Markdown document, which can be used as a template for STAT 443 labs and assignments. 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. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

Using the function *kable*, it produces a nicer table

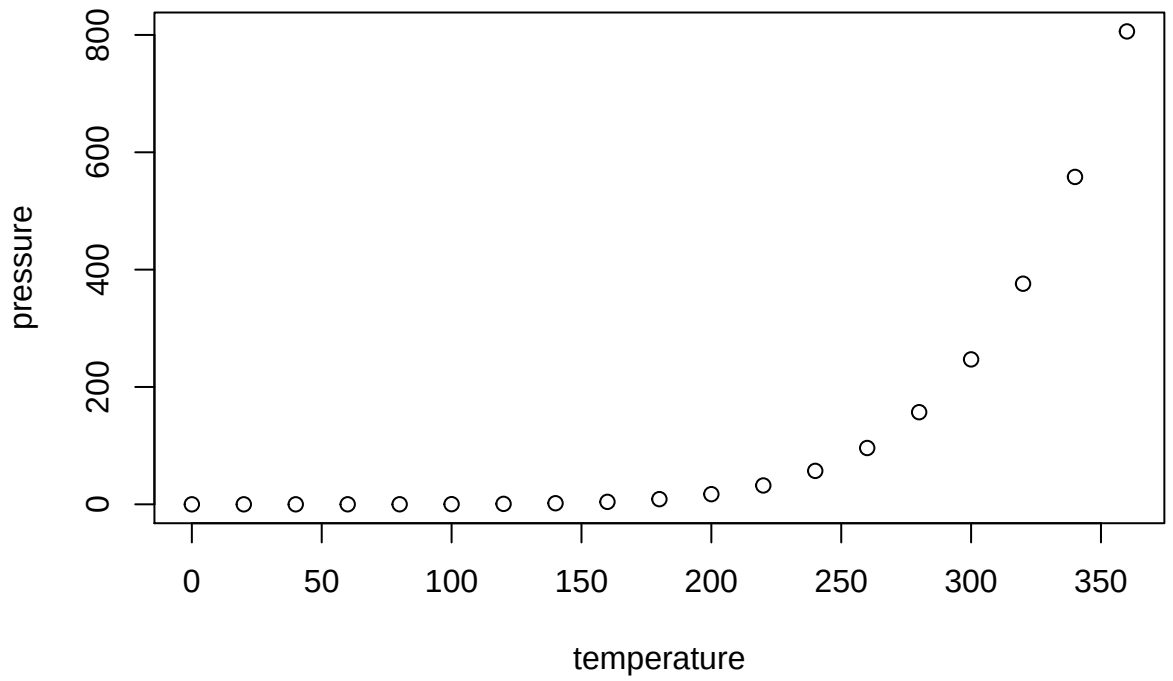
```
kable(summary(cars))
```

speed	dist
Min. : 4.0	Min. : 2.00
1st Qu.:12.0	1st Qu.: 26.00
Median :15.0	Median : 36.00
Mean :15.4	Mean : 42.98
3rd Qu.:19.0	3rd Qu.: 56.00
Max. :25.0	Max. :120.00

Including Plots

You can also embed plots, for example:

```
plot(pressure)
```



Note that specifying `echo = FALSE` parameter would prevent printing of the R code that generated the plot. This is something you may want to do for larger reports that would not require display of the R code.

You can also modify the size and alignment of the figure.

```
plot(pressure)
```

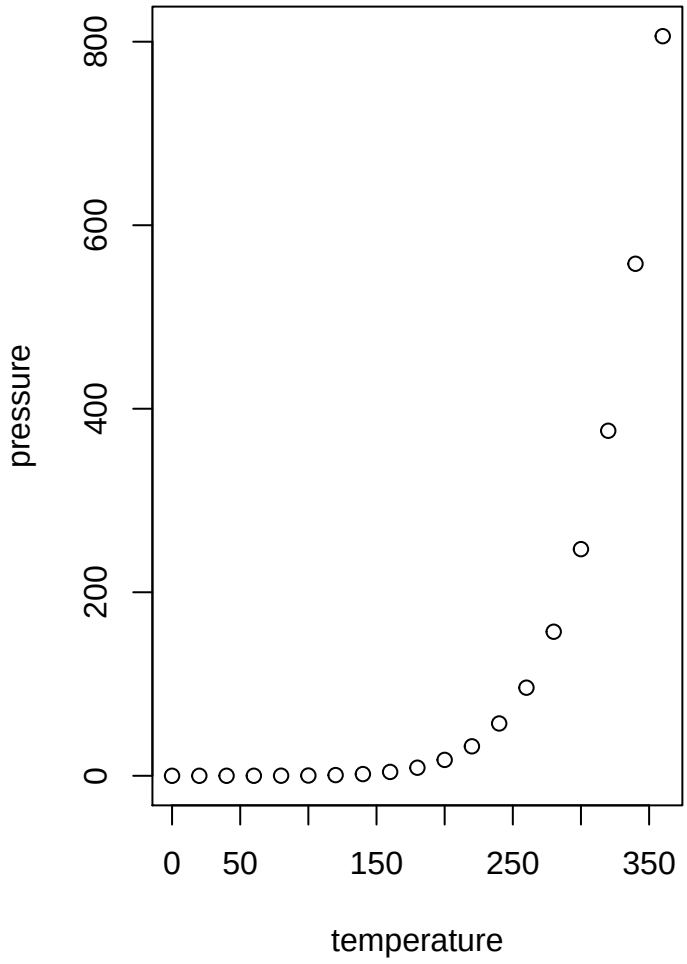


Figure 1: title