lab2_zejie

2023-10-13

```r
library("readxl")
motor_death_month = read_excel("Deaths by Month Table.xlsx"
                               ,skip=1, n_max=391)[-c(1,2),]
#delete summery information
delete = c(1:30)*13
motor_death_month = motor_death_month[-delete,] #reorder the data.
groups = split(motor_death_month, rep(1:30, each = 12))
motor_death_month = do.call(rbind, rev(groups))
```

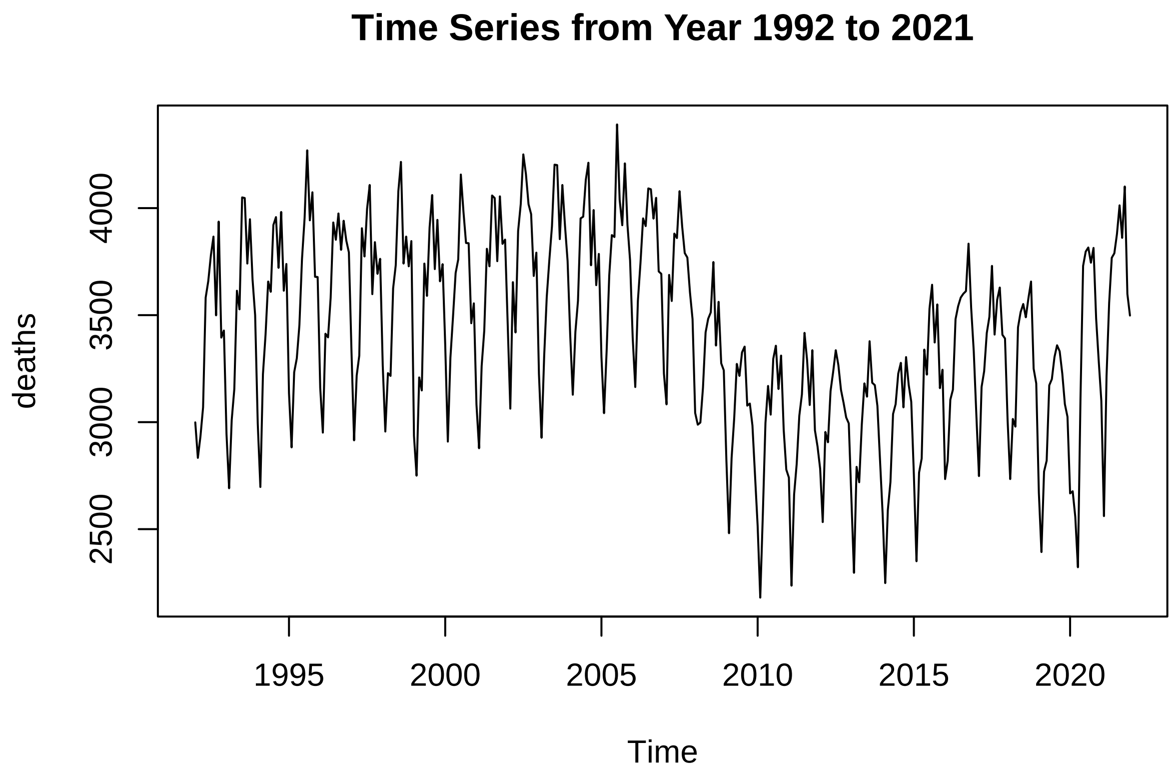
1. Travel patterns largely returned to normal in 2021 compared to the highly unusual patterns experienced during the peak of the pandemic in 2020. The file 'Deaths by Month Table.xlsx' contains information about vehicle-related deaths from January 1992 to December 2021. We can import the data into R by typing: When the data is posted as many columns (e.g. in this data, we have Death,Death rate per 100 million vehicle miles and Vehicle miles in billions.). We will choose one column as our time series data. Instead of using Death rate per 100 million vehicle miles column in the lab material, here we will be using Deaths column in the following analysis. (a).Choose the Deaths column and convert the data to ts format.

```r
# Choose the Deaths column
deaths_data <- motor_death_month$Deaths

# Convert the Deaths data to a time series format with a monthly frequency
deaths <- ts(deaths_data,start = c(1992,1), frequency = 12)
```

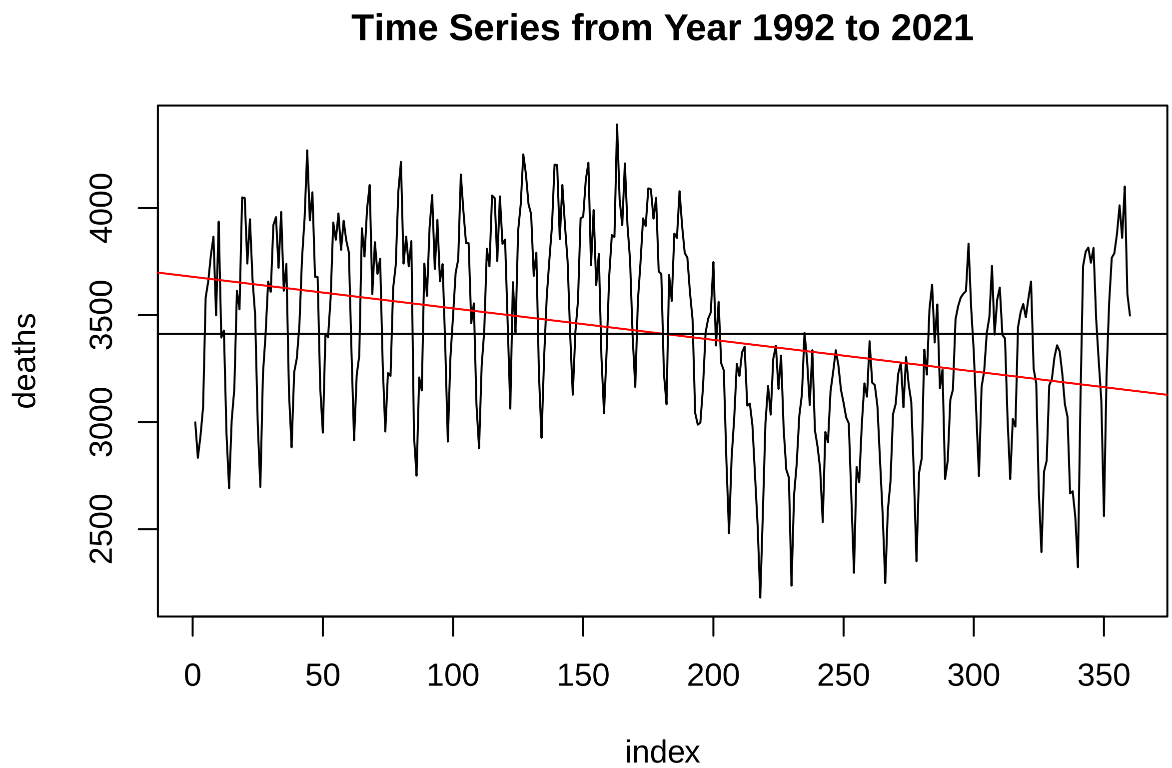
(b). Plot the data versus each month.

```r
plot(deaths, main = "Time Series from Year 1992 to 2021")
```



(c). Plot the data versus index number from 1, 2, . . . n.

```
# Plot the Deaths data versus the index
plot(1:length(deaths),deaths, main =
   "Time Series from Year 1992 to 2021", type = 'l',xlab='index')
index = 1: length(deaths)
trend <- lm(deaths ~ index)
abline(trend, col="red")
abline(h=mean(deaths , col='blue'))
```



(d). Please cut the start of the data and the end, e.g. your data will start at Jan 2009 and end at Dec 2018 and use the last three year as testing data.

```
select <- c(1:(17*12))
data_split <- deaths[-c(select)]
training = data_split[1:(10*12)]
testing = data_split[((10*12)+1):(13*12)]
```

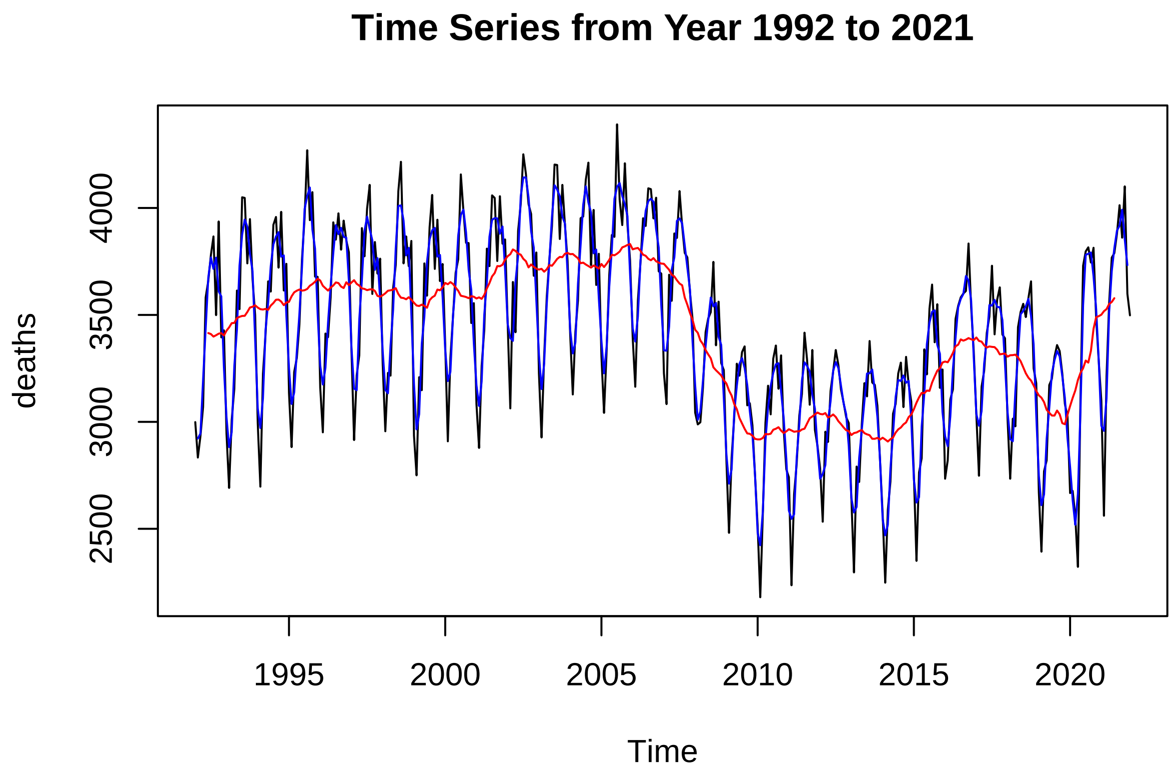
(e). Create quarterly average. Compare roughness/smoothness of the data.

```
library(zoo)
```

```
##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
smooth_data_3m = zoo::rollmean(deaths, k = 3, fill = NA)
smooth_data_12m = zoo::rollmean(deaths, k = 12, fill = NA)
plot(deaths, main = "Time Series from Year 1992 to 2021")
lines(smooth_data_3m, col='blue' )
lines(smooth_data_12m, col='red')
legend(2013, 2, legend=c("Original data","3-month average", "12-month average"),
   col=c("black" ,"blue", "red"), lty=1, cex=0.8)
```

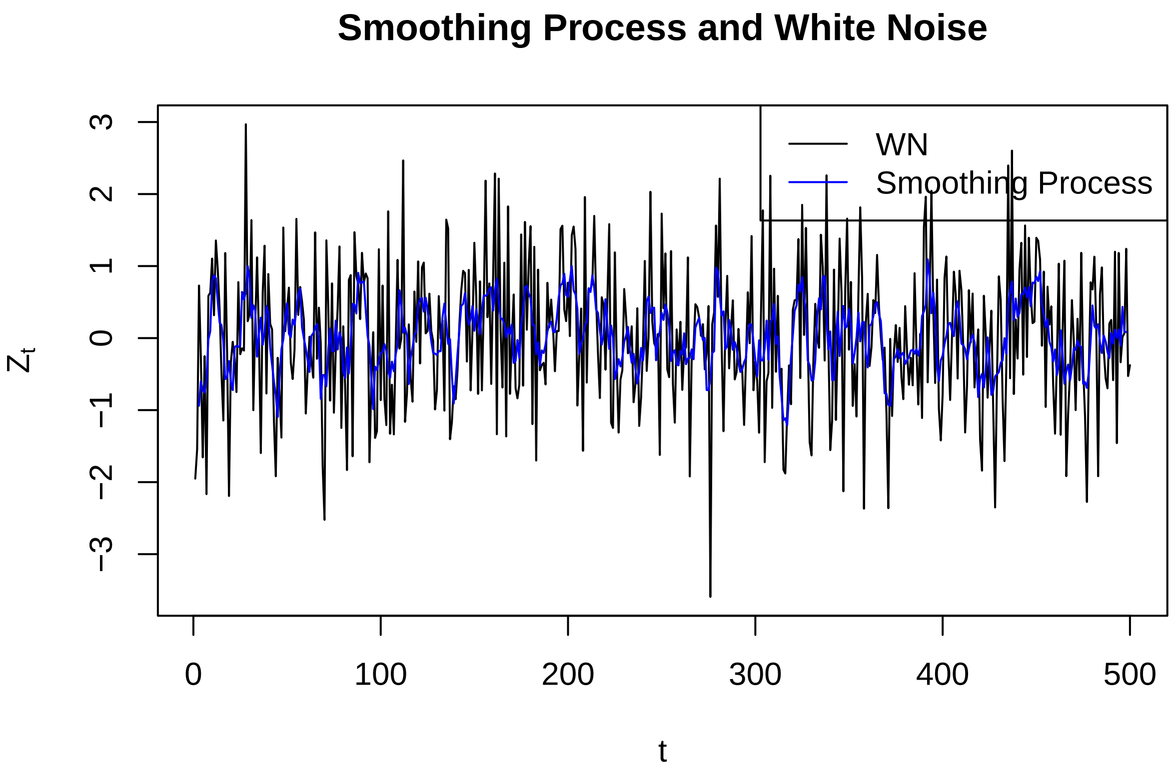


Using longer time intervals, like months, to create average data makes the numbers look steadier and less jumpy. For example, when we average data over a whole year (12 months), it's much smoother and easier to see trends than when we average only over 3 months. 2.Simulate and plot n = 500 values of a Gaussian white-noise process with variance

```
#1. set the random seed
set.seed(888) #any integer number
```

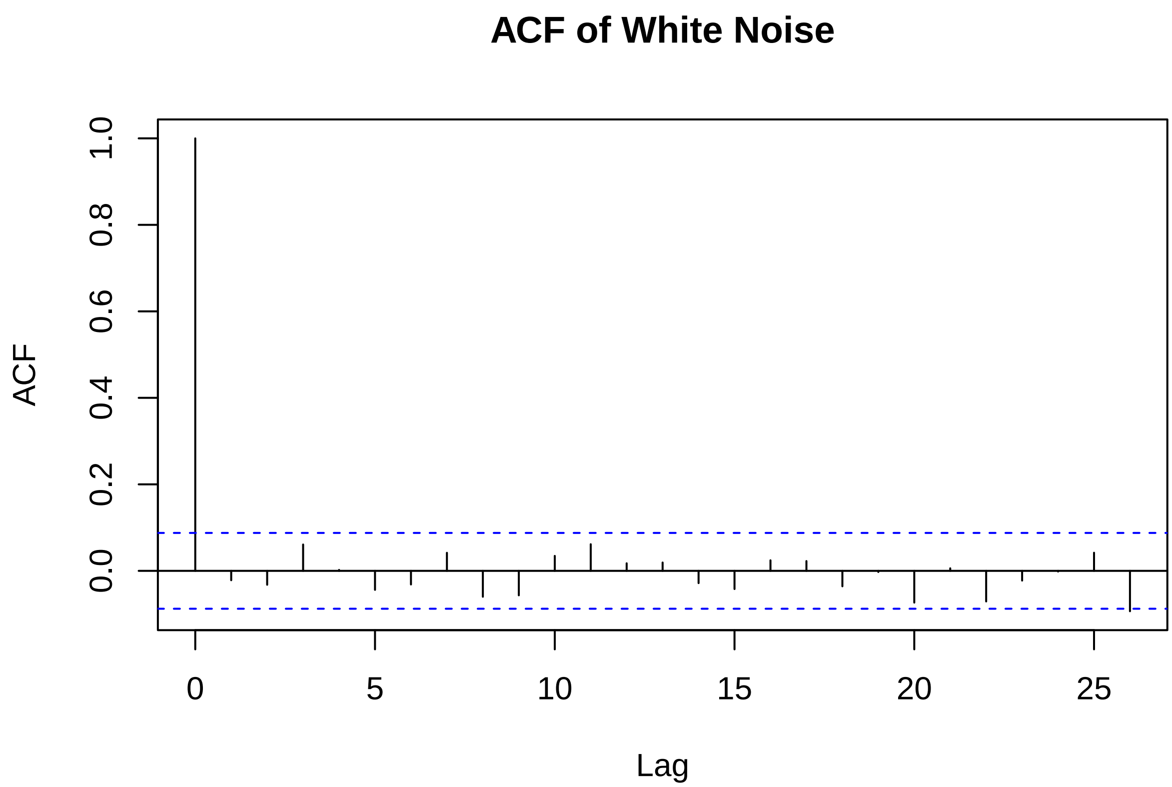
(a). Plot Zt and Xt together with different colors. Do you observe any difference?

```
z_t <- rnorm(500,0,1)
x_t = filter(z_t, filter = rep(1/5,5), sides = 2, method = "convolution")
# Plot of white-noise and x-t
plot(z_t,xlab = "t",ylab = expression(Z[t]),type = "l",
main = "Smoothing Process and White Noise") # Plot of Smoothing Process
lines(x_t,col = "blue")
# Add legend
legend("topright",c("WN", "Smoothing Process"),col = c("black","blue"),
       lty = 1)
```

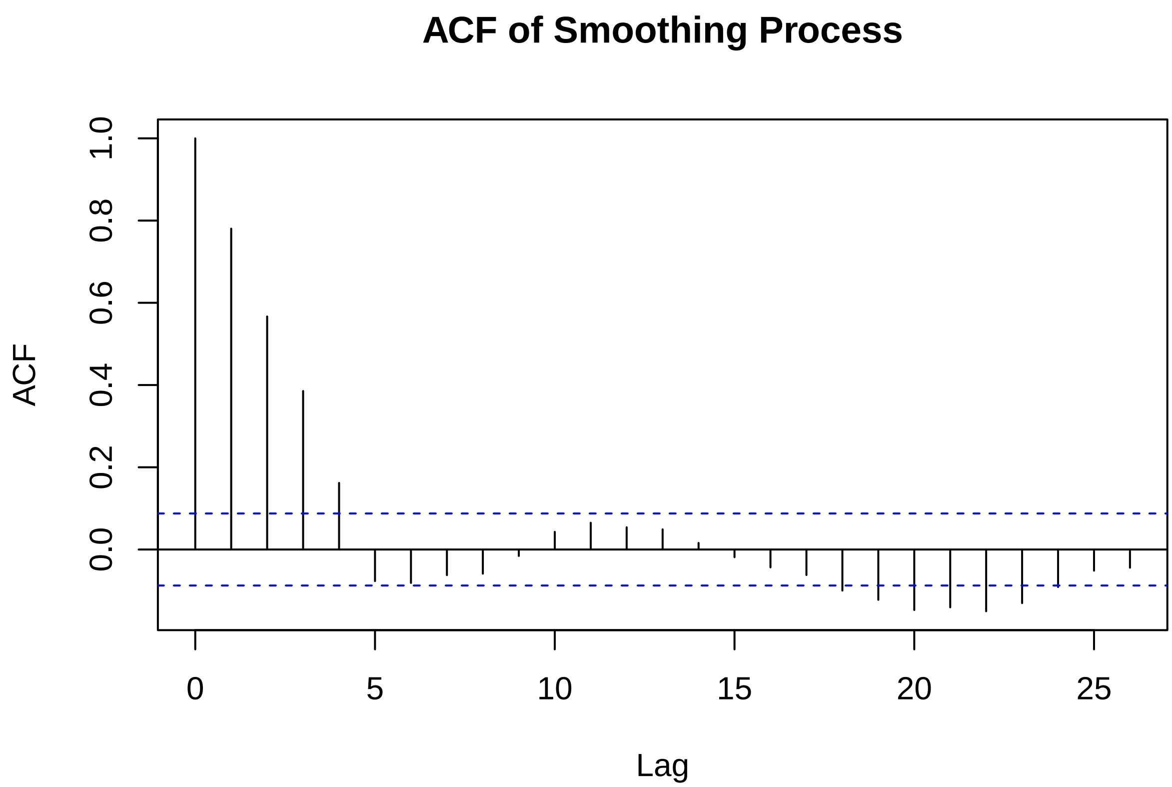


Observation: there is a obvious difference between z_t and x_t. Clearly x_t showcases more smoothness than the z_t does. Thsi could be result from taking average of previous, current, and future data value of z_t. (b). Plot sample acfs of Zt and Xt and compare the two plots.

```
acf(z_t, main="ACF of White Noise")
```




```r
acf(x_t, na.action = na.pass, main="ACF of Smoothing Process")
```



Comparison: The values in the (ACF) of z_t are consistently close to 0 at all lags except lag 0. This behavior aligns with the characteristic of uncorrelated data and is indicative of a Gaussian white noise process.

In contrast, the ACF of x_t exhibits 5 significant values exceeding 0, particularly at lags close to 0. Additionally, the ACF pattern suggests that x_t follows an MA(4) model.