Name: Your name here
Due: 2024/09/25

Day 4 - Lab 1: Decomposition of time series

Introduction

The purpose of today's lecture is to understand how to further explore how to decompose a time series into its constituent components in R. To guide our exploration, we will again return to the Vermont temperatures data set.

> ❗ Important
>
> Be sure to set each code chunk to `eval = T` after completing the skeleton code provided for you.

```r
# packages
library(tidyverse)
library(lubridate)

# load data
vt_temps <- readr::read_csv("vt_temps.csv")
```

---

0. [1 pt] With whom are you working today?

1. [14 pt] One (final?) time, we return to the Vermont temperatures data set. The purpose of this question is to reconstruct the decomposition of an additive time series by hand.

   a) [1 pt] Once again, create a `ts` object, called `vt_ts`, of the Vermont temperatures time series spanning 1900-01 to 2000-12.

   b) [2 pt] Plot the decomposition of the `vt_ts` object, first storing the decomposition as `vt_decompose`. Is there clear evidence of a trend or seasonal component?

c) [2 pt] Calculate the trend component for 1900-07 to 1901-07 **by hand**. Verify you have correctly calculated the trend component by printing out a table of the trend component calculated by hand and via the decompose function.

> Hint
>
> To calculate this moving average by hand, consider using a **for loop**. Additionally, it is helpful to know that 1900-07 corresponds to $t = 7$ and 1901-07 corresponds to $t = 18$. Skeleton code to calculate the moving average is provided below.

```r
# create a storage vector of length length(7:18)
ma_byhand <- matrix(NA, nrow = 12, ncol = 1)

# loop through the 12 values of t: 7, 8, 9, ..., 16, 17, 18
# and calculate the moving average, storing it in matrix
for(t in 7:18){
  ma_byhand[t-6,] <-
}

knitr::kable(tibble(byhand = ma_byhand, withr = vt_decompose$trend[7:18]))
```

d) [1 pt] Repeat this process for the entire time series. Plot the hand-calculated trend by the trend calculated by `decompose`. You should see a perfect 1-1 relationship.

```
# now create a storage vector for the entire data set
## excluding the first 6 and last 6 obs (why?)
vt_trend <- matrix(NA, nrow = length(vt_ts) - 12, ncol = 1)
for(t in 7:(length(vt_ts)-6)){
  vt_trend[t-6,] <-
}

# append the NAs we skipped
vt_trend_full <- c(rep(NA, 6), vt_trend, rep(NA, 6))

# plot
```

e) [2 pt] Next, calculate the seasonal component associated with each time point t for the entire time series **by hand**. The skeleton code below should help. Print the first 6 rows of the resulting matrix of seasonal effects.

```
# calculate the difference between the raw ts and the calculated trend
seasonal_differences <-

# coerce into a matrix and print
seasonal_differences_mat <- matrix(seasonal_differences, ncol = 12, byrow = T)
round(seasonal_differences_mat [1:6,], 2)
```

f) [2 pt] Calculate the average seasonal effect associated with each month, and compare these 12 values to those obtained from the `decompose` function. Are they the same?

```
seasonal_means <-

knitr::kable(tibble(byhand = seasonal_means, withr = vt_decompose$figure))
```

g) [1 pt] What is the mean of the current estimated seasonal effects?

> Note
>
> It is typical to force the mean of the seasonal effects to equal exactly 0, a process called **centering**.
>
> For additive models, seasonal effects are centered by subtracting the mean of all the average seasonal effects from each average seasonal effect. For multiplicative models, seasonal effects are centered by dividing each seasonal mean by the mean of all the seasonal means. (You may need to read this paragraph a few times)

h) [1 pt] Center the seasonal effects. Compare the newly centered seasonal means to the values obtained from `decompose`. Are they the same now?

```
seasonal_means_centered <-
knitr::kable(tibble(byhand = seasonal_means_centered, withr = vt_decompose$figure))
```

i) [2 pt] We have nearly reconstructed the output from `decompose`! All that remains is the residual error series. Calculate the residual error series from the previously created objects representing the trend and seasonal effects. Compare the by-hand calculation to the results from `decompose`. Are they the same?

Looking ahead

The goal of this lab was to elucidate how additive time series are decomposed, and how the data may be used to estimate each of the terms in a decomposition. We will continue to develop our time series toolkit next week as we consider serial correlation.