

Tarea - 2

Thomas M. Rudolf

Responde esta tarea, para evaluar el código de las secciones grises persona Cmd/Ctrl enter sobre la línea de código a evaluar.

Cuando termines la tarea puedes presionar el botón de Knit que está en la parte superior (abajo de donde dice tarea-eda-2.Rmd) para crear un documento html, como alternativa puedes pegar tus resultados en un documento externo.

Envía tu tarea como documento (pdf o html) con código y texto describiendo lo que observas.

Recuerda cargar los paquetes que vas a usar:

```
library(tidyverse)
```

```
## — Attaching core tidyverse packages — tidyverse 2.0.0 —
## ✓ dplyr      1.1.2      ✓ readr      2.1.4
## ✓ forcats    1.0.0      ✓ stringr    1.5.0
## ✓ ggplot2     3.4.2      ✓ tibble     3.2.1
## ✓ lubridate  1.9.2      ✓ tidyr      1.3.0
## ✓ purrr      1.0.2
## — Conflicts — tidyverse_conflicts() —
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(readr)
```

Series de tiempo

Consideramos la ventas semanales de un producto a lo largo de 5 años, transformaremos la variable de ventas utilizando el logaritmo.

1. Describe que observas en la gráfica. Se puede observar que las ventas tienen una tendencia creciente. Tal vez no línea, sino cuadrático con un posible máximo entre 200 y 300. Adicionalmente, se nota que hay un comportamiento periódico en intervalos de 50 (0-50, 50-100, etc.) (ciclicidad) con un pico en la mitad del intervalo (+25 o +75) o un poco más después (estacionalidad). Este podría ser un senos entre 0 y pi o una polinomio de orden 2 o más con un máximo en la posición indicado.

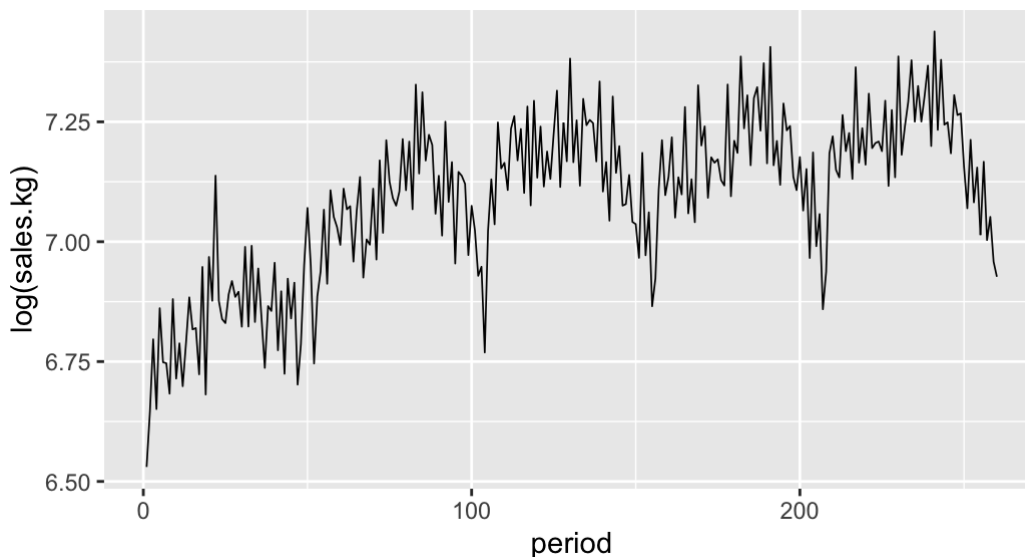
```
ventas <- read_csv("ventas_semanal.csv")
```

```
## Rows: 260 Columns: 2
## — Column specification —————
## Delimiter: ","
## dbl (2): period, sales.kg
##
## 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.
```

```
head(ventas)
```

```
## # A tibble: 6 × 2
##   period sales.kg
##   <dbl>   <dbl>
## 1     1     686.
## 2     2     768.
## 3     3     895.
## 4     4     774.
## 5     5     955.
## 6     6     853.
```

```
ggplot(ventas, aes(x = period, y = log(sales.kg))) +
  geom_line(linewidth = 0.3)
```



Usaremos suavizamiento con curvas loess para capturar los distintos tipos de variación que observamos en la serie.

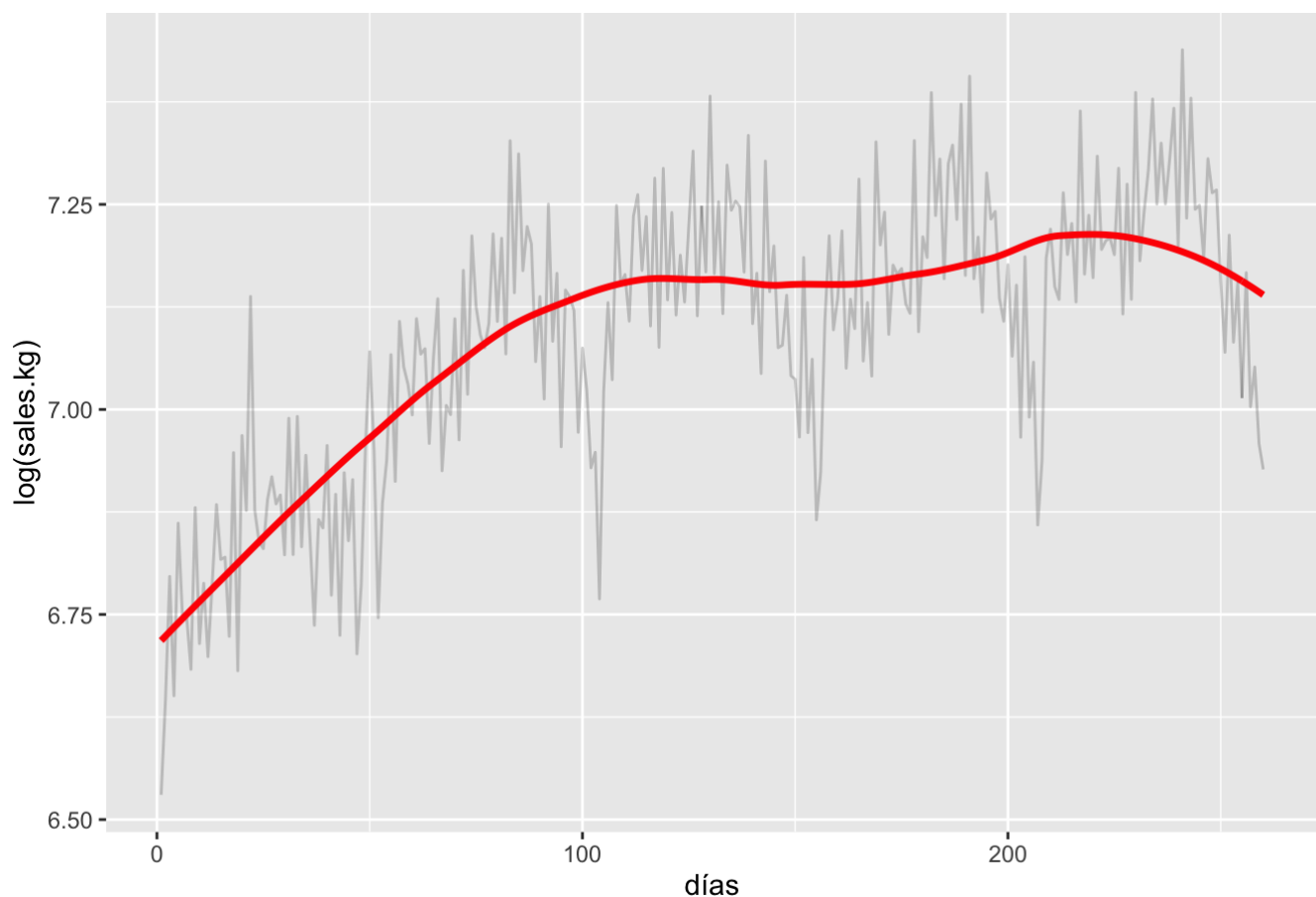
2. Utiliza un suavizador *loess* para capturar la tendencia de la serie.

```
ajuste_trend_1 <- loess(log(sales.kg) ~ period, ventas, span = 0.5,
                        degree = 2)

ventas <- ventas |>
  add_column(trend_1 = ajuste_trend_1$fitted,
            res_trend_1 = ajuste_trend_1$residuals)

ggplot(ventas, aes(x=period, y=log(sales.kg))) +
  geom_line(alpha = 0.2) +
  geom_line(aes(y = trend_1), colour = "red", size = 1.2) + xlab("días") +
  labs(caption = "Suavizamiento apropiado")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Suavizamiento apropiado

3. Ahora calcula los residuales de este ajuste y descríbelos mediante un suavizamiento más fino. Verifica que se ha estimado la mayor parte de la tendencia, e intenta capturar la variación estacional de los residuales.

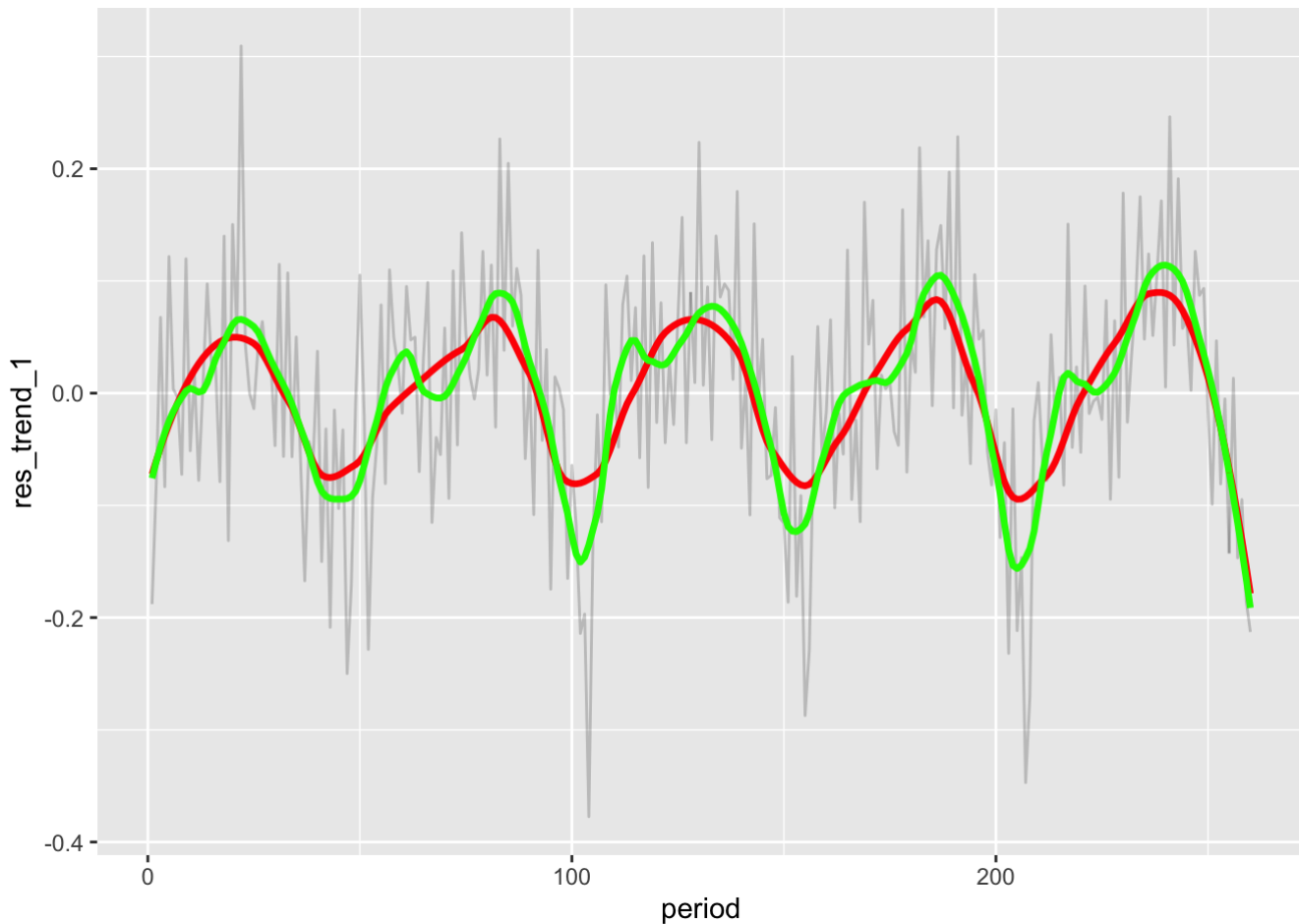
Como se puede observar, la variación estacional parece un seno. El valor $\text{span} = 0.2$ da un buen resultado usando una relación cuadrática (red line). Usando el valor de 0.1 para span (green line) todavía más detallado. Un valor menos a $\text{span} = 0.1$ no llega a mejores resultados, y mantiene más ruido.

```

periodic_res2_02 <- loess(res_trend_1 ~ period, ventas, span=0.2, degree = 2)
periodic_res2_01 <- loess(res_trend_1 ~ period, ventas, span=0.1, degree = 2)

g1 <- ggplot(ventas, aes(x=period, y=res_trend_1)) +
  geom_line(alpha = 0.2) +
  geom_line(aes(y = fitted(periodic_res2_02)), colour = "red", size = 1.2) +
  geom_line(aes(y = fitted(periodic_res2_01)), colour = "green", size = 1.2)
periodic_res2 <- periodic_res2_01
g1

```

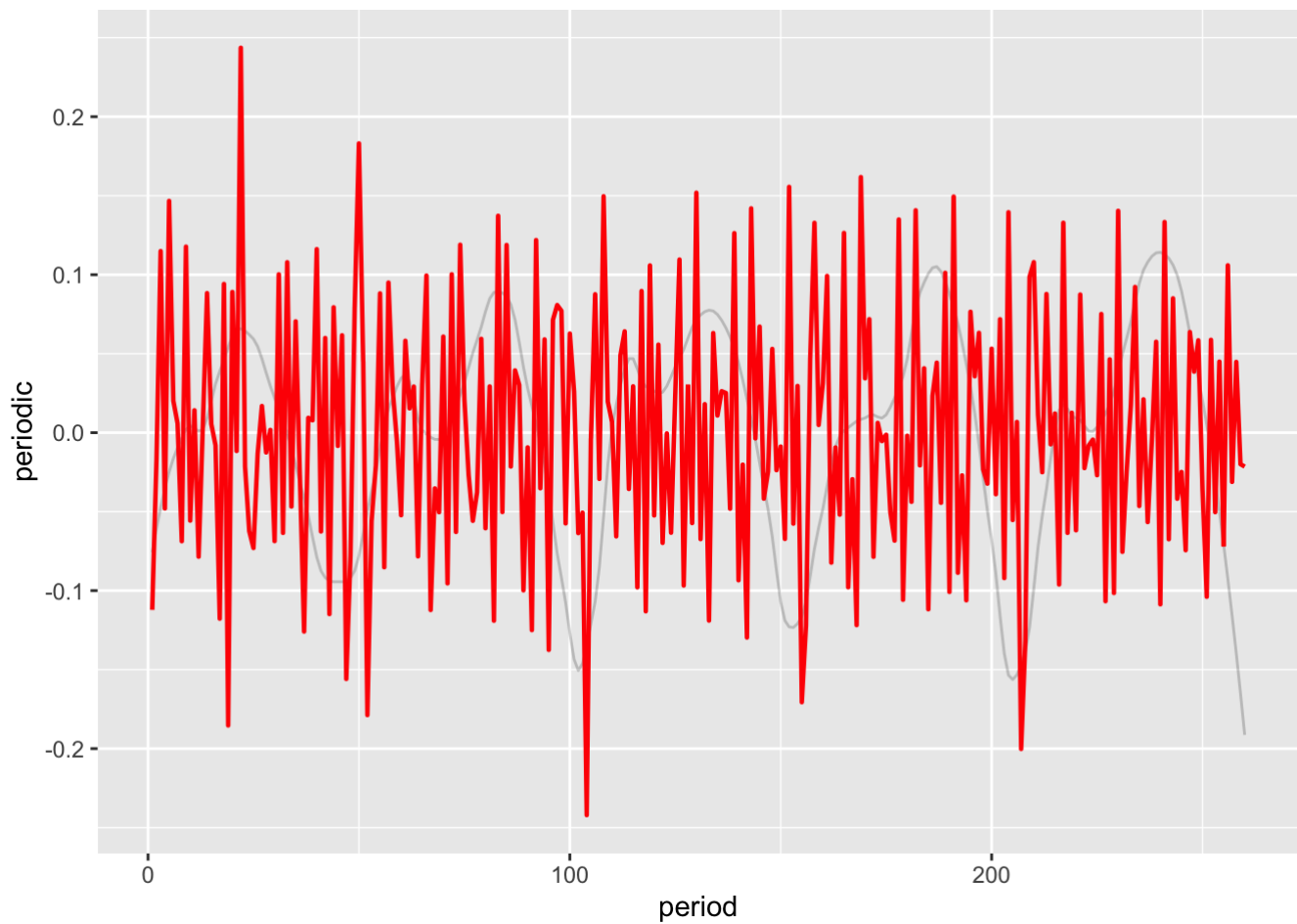


4. Grafica los residuales obtenidos después de ajustar el componente estacional para estudiar la componente de mayor frecuencia.

```

ventas <-ventas |>
  add_column(periodic = periodic_res2$fitted,
             periodic_res = periodic_res2$residuals)
g2 <- ggplot(ventas, aes(x=period, y=periodic), colour = "green", size = 1.0) +
  geom_line(alpha = 0.2) +
  geom_line(aes(y = periodic_res), colour = "red", size = 0.8)
g2

```



5. (opcional) Visualiza el ajuste, genera una gráfica de paneles, en cada uno muestra una componente de la serie de tiempo y los residuales.