

Tarea - 2 Loess

Rodrigo Alan García Pérez

Responde esta tarea, para evaluar el código de las secciones grises presiona 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 --
## v dplyr      1.1.2      v readr      2.1.4
## v forcats    1.0.0      v stringr   1.5.0
## v ggplot2    3.4.2      v tibble    3.2.1
## v lubridate  1.9.2      v tidyr     1.3.0
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

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.

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

```
## Rows: 260 Columns: 2
## -- Column specification -----
## Delimiter: ","
## dbf (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 x 2
##   period sales.kg
##   <dbl>   <dbl>
## 1     1     686.
## 2     2     768.
## 3     3     895.
```

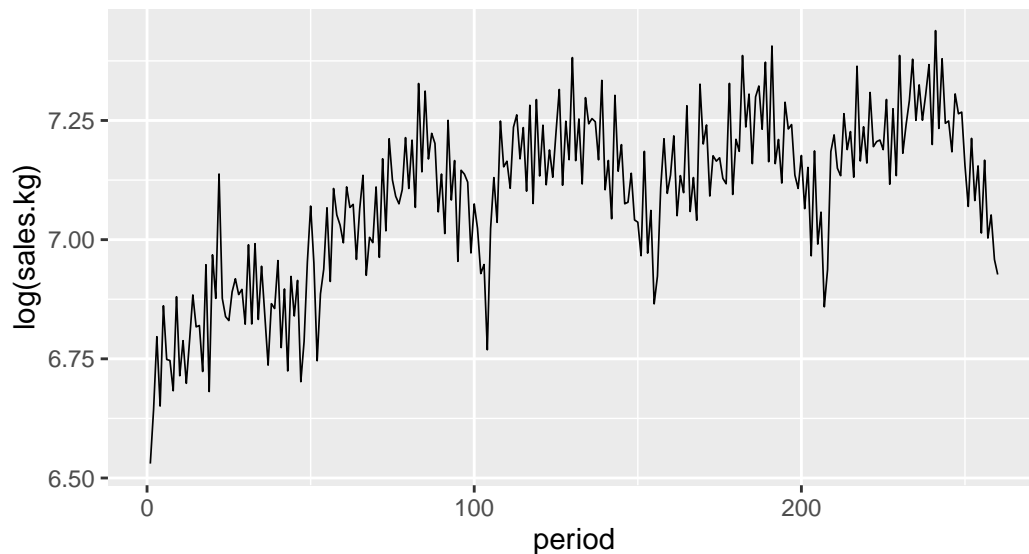
```
## 4      4      774.
## 5      5      955.
## 6      6      853.
```

```
glimpse(ventas)
```

```
## Rows: 260
## Columns: 2
## $ period   <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18~
## $ sales.kg <dbl> 685.537, 768.234, 894.643, 773.501, 954.600, 852.853, 851.082~
```

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

```
## 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.
```



La gráfica muestra que las ventas de un producto a lo largo de 5 años tienen una tendencia ascendente en los primeros 100 períodos, seguida de una fase de estabilización con mayor volatilidad, caracterizada por picos y caídas pronunciadas, que pueden ser causadas por estacionalidad, periodos de baja demanda o problemas puntuales con el producto. Hacia el final del período, se observa un descenso en las ventas, lo que podría indicar una desaceleración reciente.

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)
```

```
ajuste_trend_1
```

```
## Call:
## loess(formula = log(sales.kg) ~ period, data = ventas, span = 0.5,
##       degree = 2)
##
## Number of Observations: 260
```

```
## Equivalent Number of Parameters: 6.2
```

```
## Residual Standard Error: 0.1092
```

```
ventas <- ventas |>
  mutate(trend_1 = ajuste_trend_1$fitted,
         res_trend_1 = log(sales.kg) - trend_1 )
ventas
```

```
## # A tibble: 260 x 4
```

```
##   period sales.kg trend_1 res_trend_1
##   <dbl>   <dbl>   <dbl>   <dbl>
```

```
## 1     1     686.    6.72    -0.188
```

```
## 2     2     768.    6.72   -0.0795
```

```
## 3     3     895.    6.73    0.0675
```

```
## 4     4     774.    6.73   -0.0833
```

```
## 5     5     955.    6.74    0.122
```

```
## 6     6     853.    6.74    0.00373
```

```
## 7     7     851.    6.75   -0.00361
```

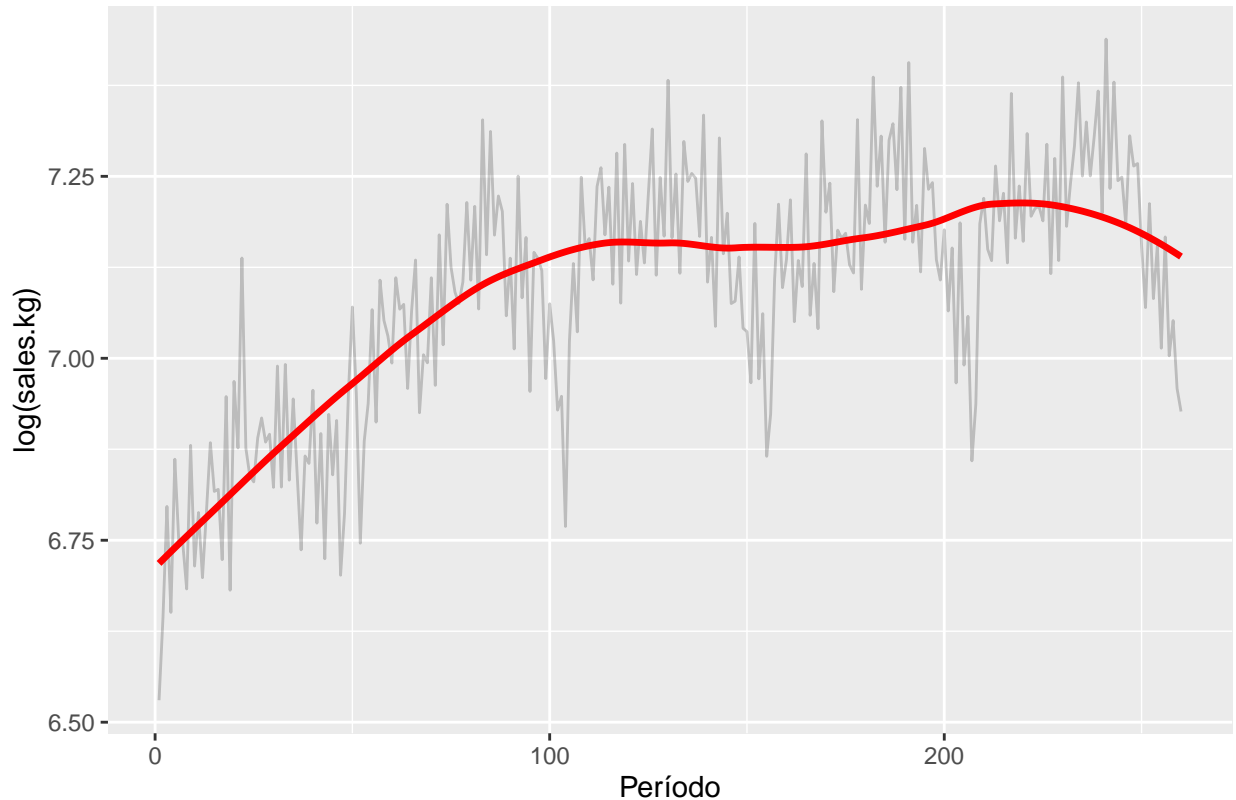
```
## 8     8     799.    6.76   -0.0723
```

```
## 9     9     973.    6.76    0.120
```

```
## 10    10     824.    6.77   -0.0513
```

```
## # i 250 more rows
```

```
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("Período") +
  labs(caption = "Suavizamiento apropiado")
```



En la grafica anterior podemos observar una tendencia en ascenso en las ventas a lo largo de los más de 200 periodos. Con un ascenso más pronunciado en los primeros 100 periodos y una ligera tendencia a la baja del periodo 200 en adelante.

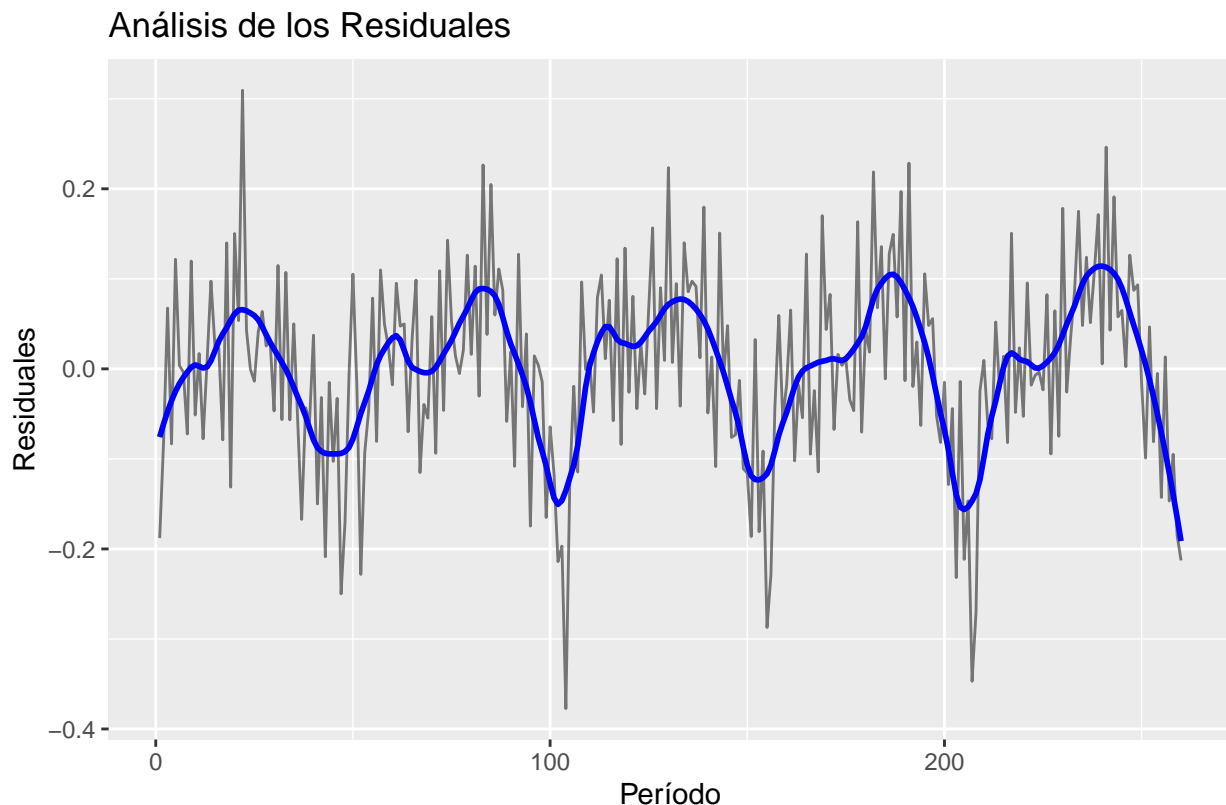
3. Ahora calcula los residuales de este ajuste y descríbelos mediante un suavizamiento más fino. Verifica que se ha estimado la mayor parte de la tendencia, e intenta capturar la variación estacional de los residuales.

Ya hemos calculado los residuales como parte del código anterior. Ahora solo es necesario graficar estos residuales.

```
ajuste_trend_2 <- loess(res_trend_1 ~ period, ventas, span = 0.1, degree = 2)

ventas <- ventas |>
  mutate(trend_2 = ajuste_trend_2$fitted,
         res_trend_2 = res_trend_1 - trend_2 )

ggplot(ventas, aes(x = period, y = res_trend_1)) +
  geom_line(alpha = 0.5) +
  geom_line(aes(y = trend_2), colour = "blue", size = 1) +
  xlab("Período") +
  ylab("Residuales") +
  labs(title = "Análisis de los Residuales",
       caption = "Suavizamiento fino de los residuales")
```



Suavizamiento fino de los residuales

En la grafica anterior observamos que se elimino la tendencia pero aun se puede apreciar la variacion estacional de una manera muy clara. Tenemos presente el mismo patron 5 veces, lo que corresponde a los 5 años de datos de ventas.

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

```
ajuste_trend_3 <- loess(res_trend_2 ~ period, ventas, span = 0.05, degree = 2)

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : k-d tree limited by memory. ncmx= 260

ventas <- ventas |>
  mutate(trend_3 = ajuste_trend_3$fitted,
         res_trend_3 = res_trend_2 - trend_3 )

ggplot(ventas, aes(x = period, y = res_trend_2)) +
  geom_line(alpha = 0.5) +
  geom_line(aes(y = trend_3, colour = "purple", size = 1) +
  xlab("Período") +
  ylab("Residuales Ajustados") +
  labs(title = "Análisis de la Componente de Mayor Frecuencia",
       caption = "Grafico de los residuales después de ajustar la estacionalidad")
```

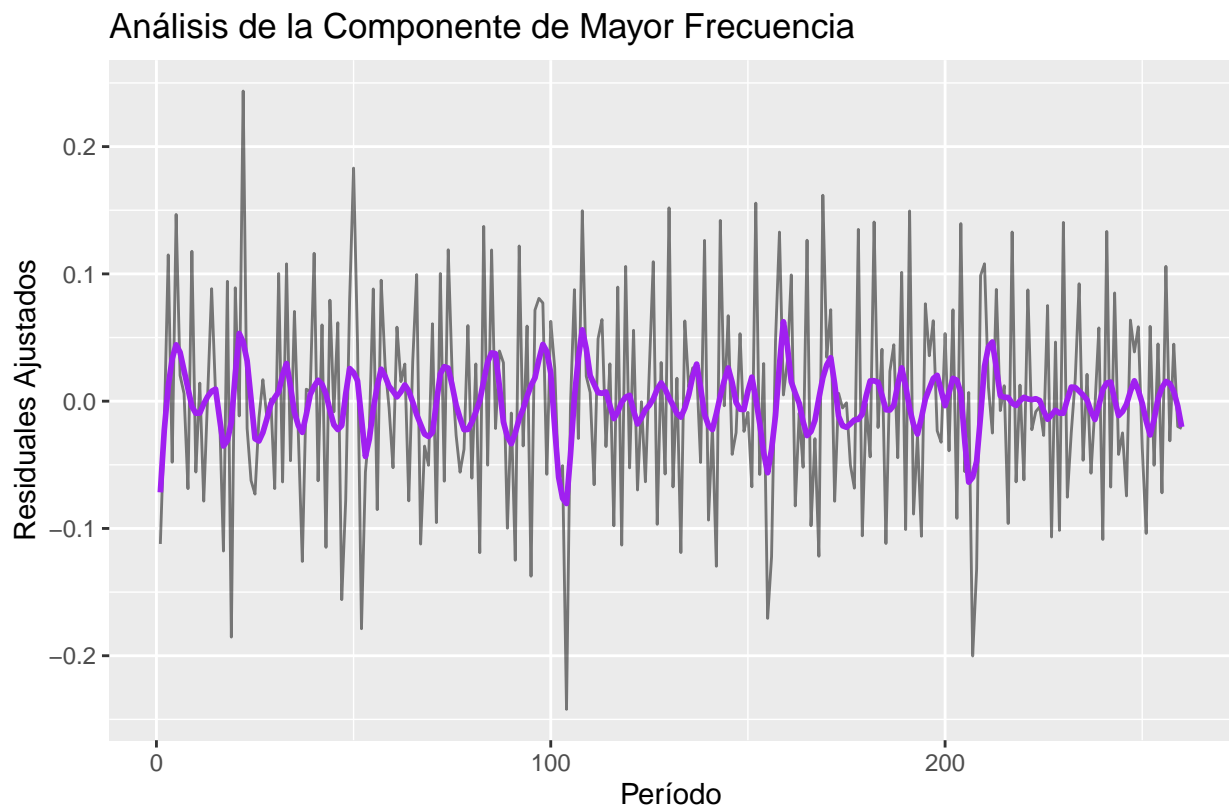


Grafico de los residuales después de ajustar la estacionalidad

Después de eliminar la tendencia y la estacionalidad, no podemos observar en la gráfica un patrón claro en las ventas. Esto sugiere que lo que queda son principalmente fluctuaciones aleatorias y no hay una estructura significativa de largo plazo restante.

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

```
# Librerías necesarias
library(ggplot2)
```

```
library(patchwork)

# Gráfico de la serie original
p1 <- ggplot(ventas, aes(x = period, y = log(sales.kg))) +
  geom_line(color = "black")

# Gráfico de la tendencia
p2 <- ggplot(ventas, aes(x = period, y = trend_1)) +
  geom_line(color = "red")

# Gráfico del componente estacional
p3 <- ggplot(ventas, aes(x = period, y = trend_2)) +
  geom_line(color = "blue")

# Gráfico de los residuales ajustados
p4 <- ggplot(ventas, aes(x = period, y = trend_3)) +
  geom_line(color = "purple")

p5 <- ggplot(ventas, aes(x = period, y = res_trend_3)) +
  geom_point(color = "darkgreen", alpha = 0.6)

# Combinación de los gráficos en un panel
p1 + p2 + p3 + p4 + p5 + plot_layout(ncol = 1, heights=c(20,20,20,20,20))
```

