

# **Introdução aos modelos lineares dinâmicos**

James D Santos

2025-04-09

## **Table of contents**

# Prefácio

# 1 Summary

In summary, this book has no content whatsoever.

$1 + 1$

[1] 2

## 2 Introdução

### 2.1 Definição de séries temporais

Considere um fenômeno aleatório que será observado ao longo do tempo. A coleção de variáveis aleatórias indexadas no tempo associadas a este fenômeno será denominada série temporal. Neste caso, para cada instante de tempo  $t$ , há uma variável  $X(t)$  associada.

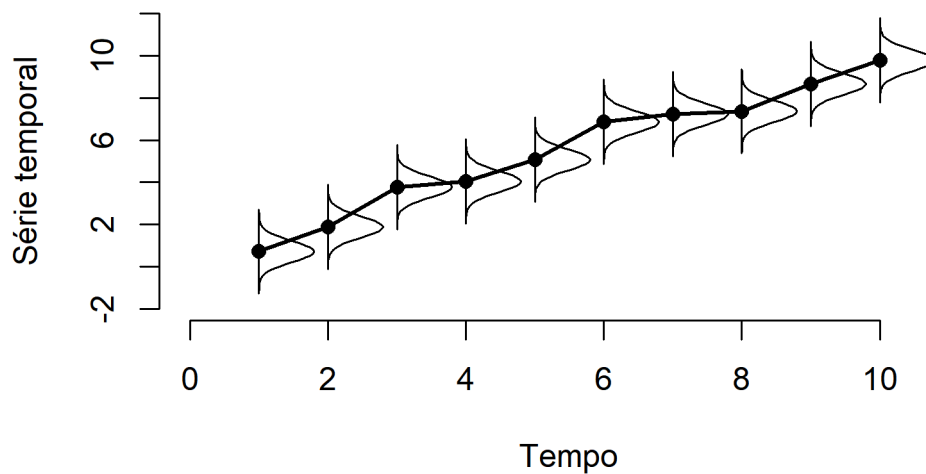


Figure 2.1: Figure 1 - Ilustração de uma série temporal

Alguns autores definem séries temporais simplesmente como uma os valores de uma variável registrados ao longo do tempo. Essa definição não é útil para nós, uma vez que o tempo não necessariamente possui influência na variável, ou seja, é possível que a distribuição de  $X(t)$  não dependa de  $t$ .

Importante: estamos interessados apenas em séries temporais nas quais o modelo de probabilidades depende do tempo  $t$ .

A partir deste momento,  $X(t)$  será escrita como  $X_t$  e representará a variável aleatória associada ao tempo  $t$  e a versão minúscula  $x_t$  representará o valor observado.

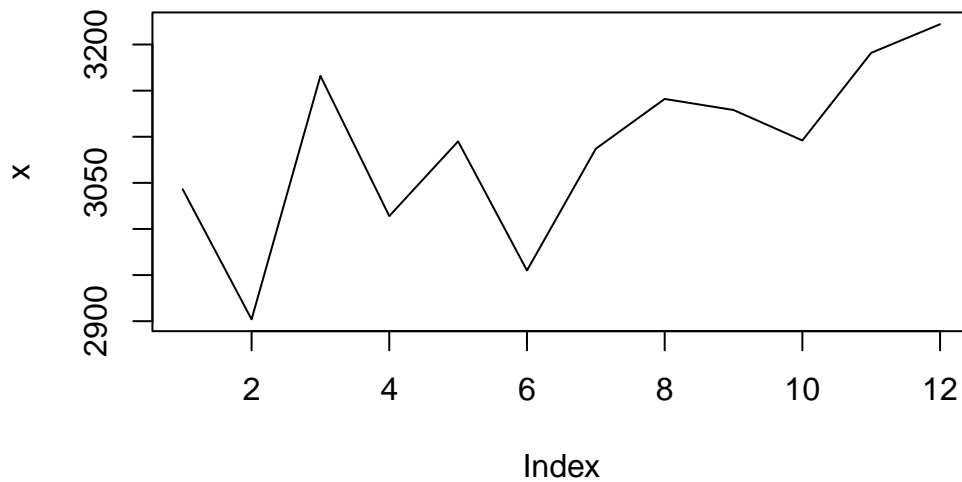
## 2.2 A classe `ts`

A tabela abaixo apresenta o número de nascidos vivos por mês na cidade de Manaus em 2021.

Mês	No. nascidos vivos
Janeiro	3043
Fevereiro	2902
Março	3166
Abril	3014
Maio	3095
Junho	2955
Julho	3087
Agosto	3141
Setembro	3129
Outubro	3096
Novembro	3191
Dezembro	3222

Para todos os efeitos, uma série temporal é um vetor numérico.

```
x <- c(  
  3043, 2902, 3166, 3014,  
  3095, 2955, 3087, 3141,  
  3129, 3096, 3191, 3222  
  
)  
  
plot(x, type = 'l')
```



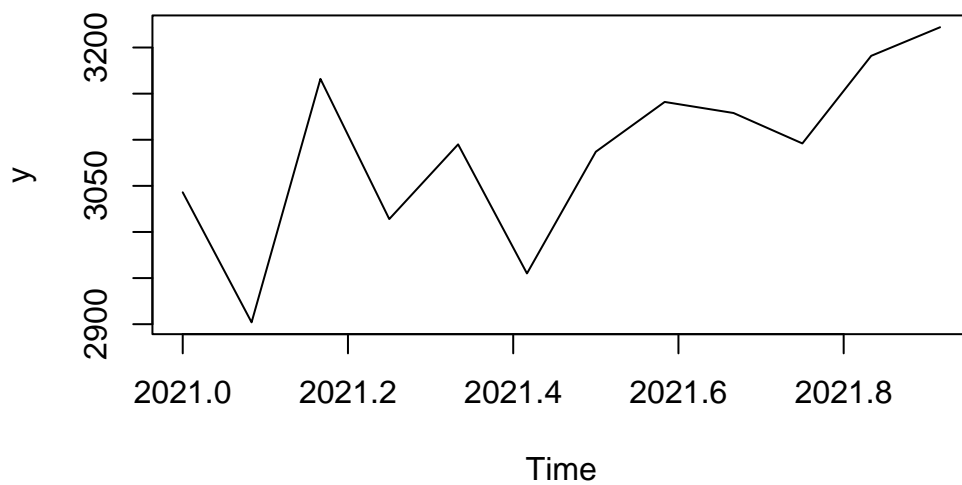
Contudo, é útil construir a série temporal como um objeto da classe **ts**. Tal função possui dois argumentos importantes:

- **frequency**: representa o número de observações por unidade de tempo. Por exemplo, se o tempo está sendo registrado em anos, mas os dados são mensais, então **frequency=12**.
- **start**: representa o momento que a série começa. Pode ser representado um único número ou por um vetor de dois números, com o segundo representando o momento dentro do período. Por exemplo **start=c(1996,2)** implica que a primeira observação data de fevereiro de 1996.

```
y <- ts( x, start = c(2021,1), frequency = 12)
y
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2021	3043	2902	3166	3014	3095	2955	3087	3141	3129	3096	3191	3222

```
ts.plot(y)
```

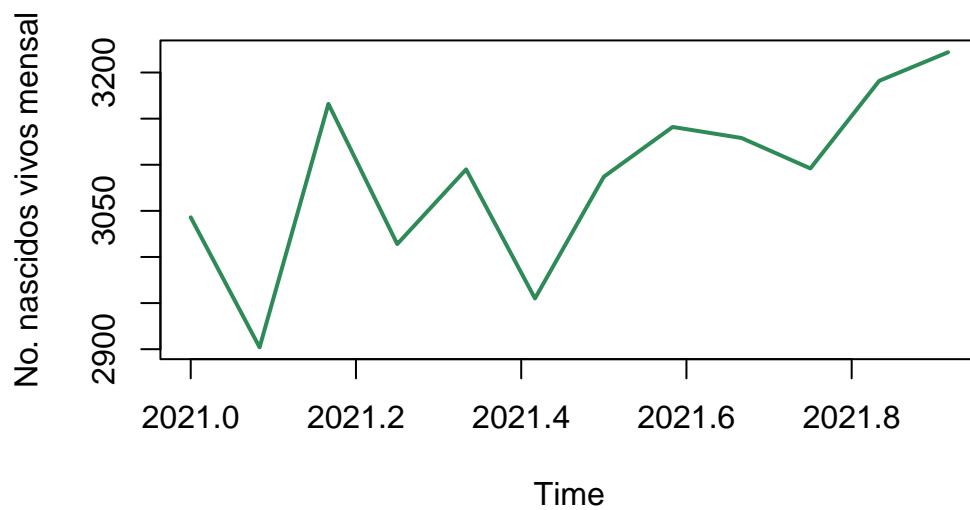


No gráfico acima, a parte decimal no eixo  $x$  representa a fração do tempo entre de um ano (começando em 0 e acumulando  $1/12$  para cada mês subsequente).

Também podemos customizar o gráfico.

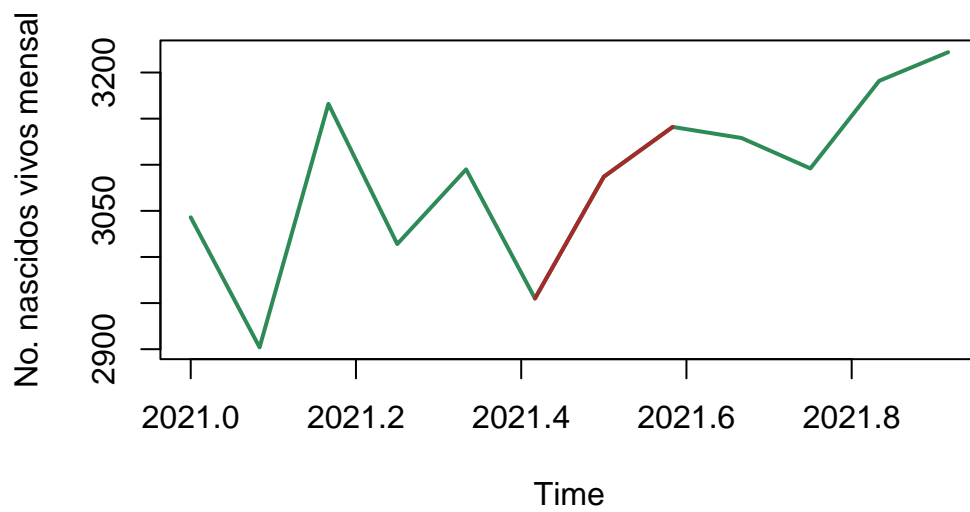
```
plot(y, ylab = 'No. nascidos vivos mensal', lwd = 2, col = 'seagreen')
```





A função `window` seleciona um subconjunto da série temporal. Abaixo selecionamos apenas os nascimentos entre Junho e Agosto e marcamos estes valores no gráfico.

```
z <- window(y, start=c(2021,6), end = c(2021,8))  
  
plot(y, ylab = 'No. nascidos vivos mensal', lwd = 2, col = 'seagreen')  
lines(z, col = 'brown', lwd= 2)
```



Exercício 1 Considere o total mensal de nascidos vivos na cidade de Manaus entre 2019 e 2020:

Ano	Mês	No. nascidos vivos
2019	1	3.199
	2	2.963
	3	3.146
	4	2.966
	5	3.074
	6	2.919
	7	3.129
	8	3.230
	9	3.456
	10	3.486
	11	3.220
	12	3.151
2020	1	3.185
	2	3.131
	3	3.256
	4	3.008
	5	3.080
	6	2.919

Ano	Mês	No. nascidos vivos
	7	3.208
	8	3.126
	9	3.126
	10	3.210
	11	2.957
	12	3.068

1. Construa um único vetor com os três anos apresentados
2. A partir do vetor criado, construa um objeto do tipo `ts`
3. Faça um gráfico da série.
4. Crie um janela para ver apenas o ano de 2020.
5. Represente a janela acima no gráfico anterior.

## 2.3 O pacote `data.table`

Assim como números e textos possuem classes específicas, as datas no ambiente `R` também possuem sua `Date`.

```
# 3 de agosto de 1998 (formato americano)
x <- '1998/8/3'
as.Date(x)
```

```
[1] "1998-08-03"
```

Para que o `R` entenda uma data no formato nacional, é necessário mudar o formato:

```
# 3 de agosto de 1998 (formato nacional)
x <- '3/8/1998'
as.Date(x, format = '%d/%m/%Y')
```

```
[1] "1998-08-03"
```

Lidamos com datas quando temos uma fonte de dados bruta, mas em geral nosso objetivo é determinar a quantidade de eventos dentro de dias, semanas, meses ou anos. O pacote `data.table` permite lidar com esse problema de modo rápido. Podemos criar um objeto deste

tipo utilizando a função `fread`. A seguir, vamos baixar uma base de dados de acidentes com aeronaves, mantida pela Força Aérea Brasileira e transformar a data de formato nacional para a classe `Date`.

```
library(data.table)
url <- 'https://drive.google.com/uc?authuser=0&id=1iYrnwXgmLK07x8b330aD73sc0VruZEuz&export=csv'

aereo <- fread(url, encoding = 'Latin-1')
aereo$ocorrencia_dia <- as.Date(aereo$ocorrencia_dia, '%d/%m/%Y')
```

Um objeto do tipo `data.table` permite uma série de consultas. Em geral, pode-se fazer `aereo[a,b,c]`, onde `a` é uma consulta/função nas linhas, `b` nas colunas e `c` é um agrupador. Uma excelente introdução pode ser vista em [Introduction to data.table](#).

Abaixo, selecionamos apenas a coluna de interesse.

```
fab_dia <- aereo[, 'ocorrencia_dia', ]
head(fab_dia)
```

```
      ocorrencia_dia
1:      2023-04-05
2:      2023-06-24
3:      2023-06-27
4:      2023-06-30
5:      2023-06-25
6:      2023-06-23
```

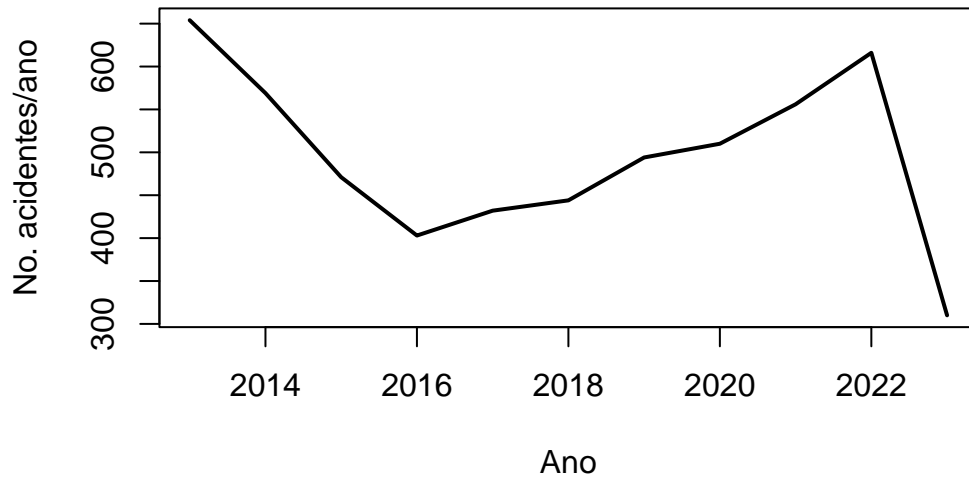
Ao utilizar o operador `.N` em `[, .N, c]`, é retornado o número de linhas que possuem o agrupamento em `c`. Vamos agrupar as datas do nosso banco por ano.

```
fab_ano <- fab_dia[, .N, by=.(year(ocorrencia_dia))]
fab_ano <-fab_ano[ order(year) ]
head(fab_ano)
```

```
      year    N
1: 2013  654
2: 2014  569
3: 2015  471
4: 2016  403
5: 2017  432
6: 2018  444
```

Agora, podemos fazer o gráfico da série

```
fab_ano <- ts( fab_ano, start = 2013)
plot(fab_ano[,2], lwd = 2, ylab = 'No. acidentes/ano', xlab = 'Ano')
```



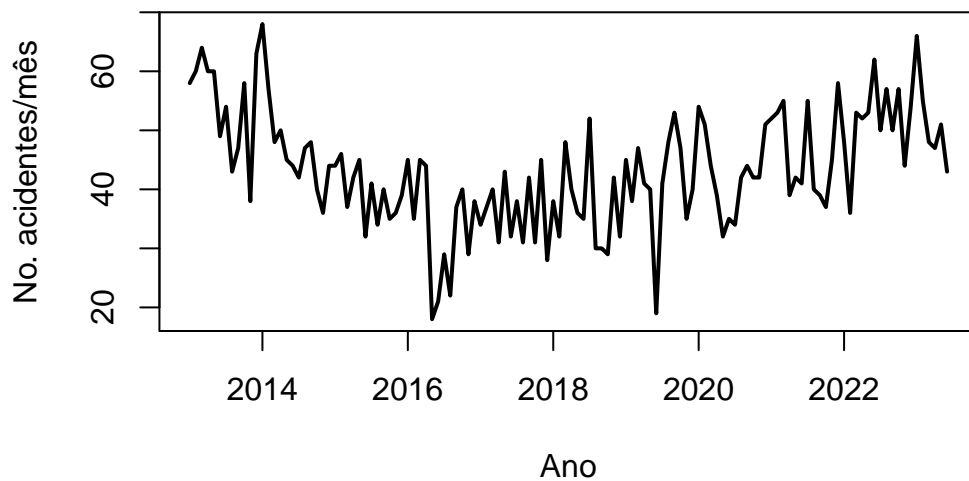
Também podemos fazer uma série mensal:

```
fab_mes <- fab_dia[, .N, by=.(year(ocorrencia_dia), month(ocorrencia_dia))]  
  
fab_mes <-fab_mes[ order(year, month ) ]  
head(fab_mes)
```

	year	month	N
1:	2013	1	58
2:	2013	2	60
3:	2013	3	64
4:	2013	4	60
5:	2013	5	60
6:	2013	6	49

O gráfico dessa nova série é:

```
fab_mes <- ts( fab_mes[,3], start = c(2013, 1), frequency = 12)
plot(fab_mes, lwd = 2, ylab = 'No. acidentes/mês', xlab = 'Ano')
```



### Exercício 1

A série abaixo contém a data dos óbitos maternos no Brasil a partir de 2010.

```
url <- 'https://drive.google.com/uc?authuser=0&id=1tYFFT9L2iopKmBDUI3P8qNIRaOnMYj7d&export=csv'
```

Crie uma série temporal com o número de óbitos mensal e faça um gráfico. Crie uma janela para colocar no gráfico o período da pandemia de COVID-19.

## 3 Tipos de sinal

Em geral, a série temporal possui componentes de dois tipos: sinal e ruído. O primeiro é uma função do tempo geralmente relacionado com a média da série, enquanto que o segundo está relacionado com a variância. Podemos assumir que essa relação é aditiva:

$$X_t = \text{sinal}(t) + \varepsilon_t$$

onde  $\varepsilon_t$  é o ruído. Em alguns casos essa relação é multiplicativa, ou seja,

$$X_t = \exp\{\text{sinal}(t) + \varepsilon_t\},$$

e, nesses casos, aplicamos o logaritmo na série para que as componentes se tornem aditivas.

Os sinais mais importantes são:

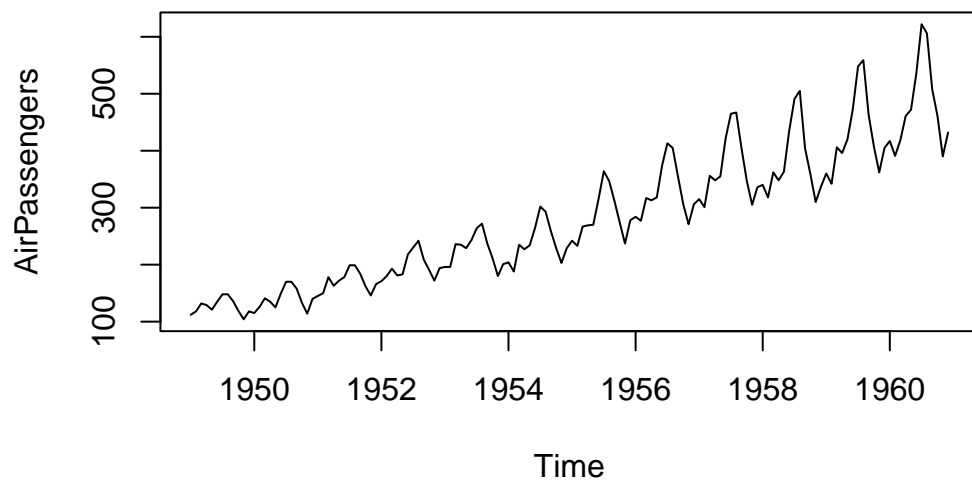
- Tendência: um comportamento de subida ou descida que pode ser observado no médio/longo prazo
- Sazonalidade: são componentes que surgem sistematicamente ao longo do tempo, como por exemplo: flutuações de temperatura entre estações, início e fim do semestre letivo, Natal, dias úteis, feriados flutuantes como a Páscoa e o Carnaval.

O ruídos mais importantes são:

- Branco: possuem variância constante e não correlacionados.
- Média móvel de ordem  $q$ : possuem variância constante e são correlacionados com até  $q$  ruídos anteriores.

A série a seguir representa o número de vendas de passagens aéreas nos EUA. Note o comportamento da tendência e da sazonalidade.

```
plot(AirPassengers)
```





## 4 A função de autocorrelação

Considere inicialmente uma amostra aleatória  $X_1, \dots, X_n$  (ou seja, todas as variáveis são independentes e possuem a mesma distribuição). Sejam

$$A_h = \{X_1, \dots, X_{n-h}\}$$

e

$$B_h = \{X_h, \dots, X_n\}.$$

Então, a correlação entre  $A_h$  e  $B_h$  é nula.

Deste modo, um meio de verificar se a coleção observada é uma série temporal é observar a correlação amostral entre

$$a_h = \{x_1, \dots, x_{n-h}\}$$

e

$$b_h = \{x_h, \dots, x_n\},$$

para diferentes valores de  $h$ .

A função  $r(h)$  que representa a correlação amostral entre  $a_h$  e  $b_h$  é denominada **autocorrelação**. O valor  $h$  é denominado **defasagem** (do inglês, *lag*).

Propriedades

- $r(0) = 1$
- $-1 \leq r(h) \leq 1$

Correlograma O gráfico  $(h, r(h))$  é denominado correlograma, ou gráfico da função de autocorrelação.

### 4.1 O correlograma de uma amostra aleatória

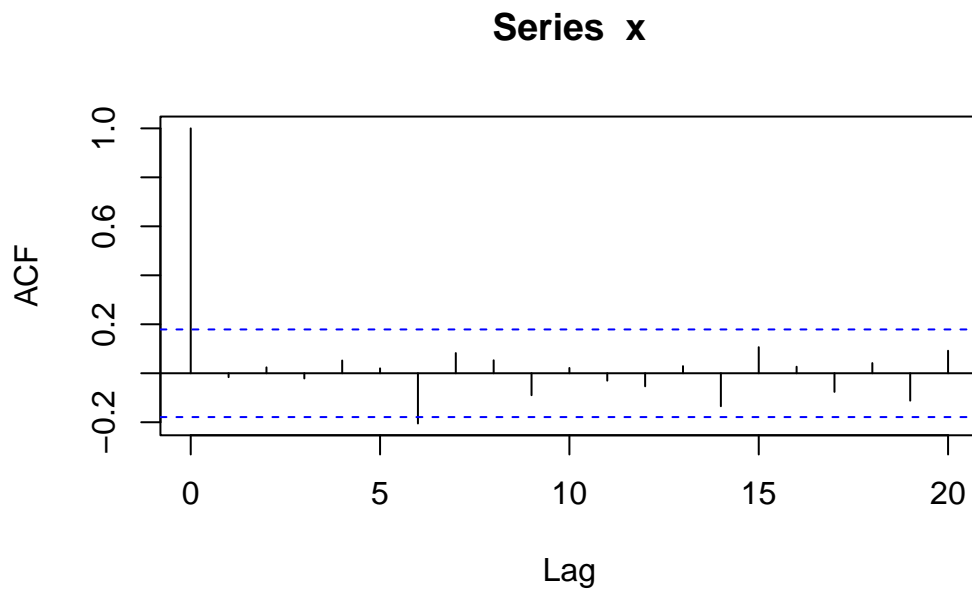
Quando a amostra é aleatória, a função de autocorrelação é nula para qualquer defasagem diferente de 0. Deste modo, o correlograma deve apresentar valores próximos de zero.

Para entender o que próximo de zero significa, os limites do intervalo de confiança para o coeficiente de correlação sobre a hipótese de que esta é nula são colocados no gráfico.

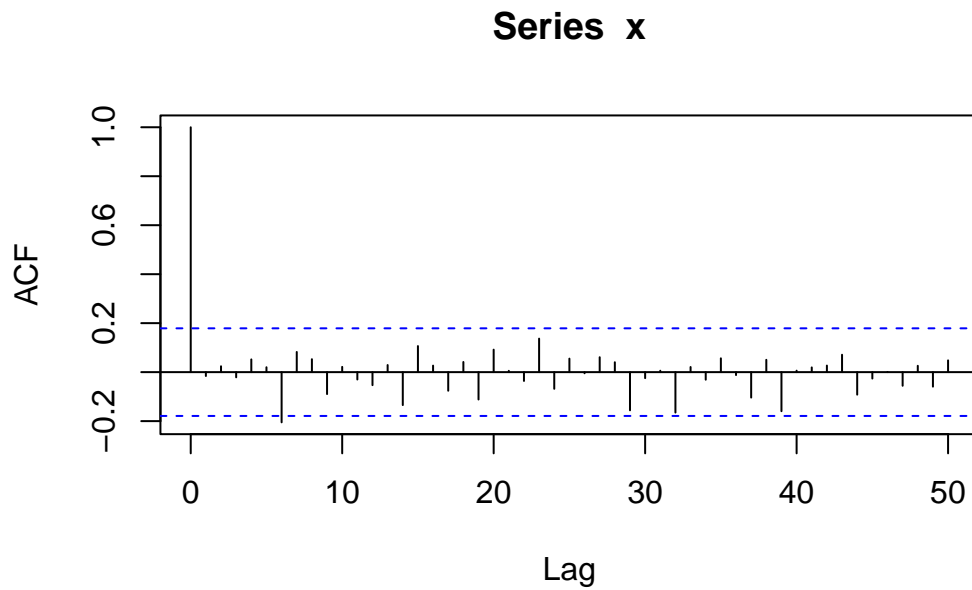
Abaixo ilustramos um correlograma para uma amostra de variáveis aleatórias independentes com distribuição normal padrão.

```
x <- rnorm(120)

# correlograma
acf(x)
```



```
# o mesmo correlograma com uma defasagem maior
acf(x, lag = 50)
```



## 4.2 O correlograma com a componente de tendência

Quando uma série exibe tendência, o correlograma exibe um decaimento lento e persistente.

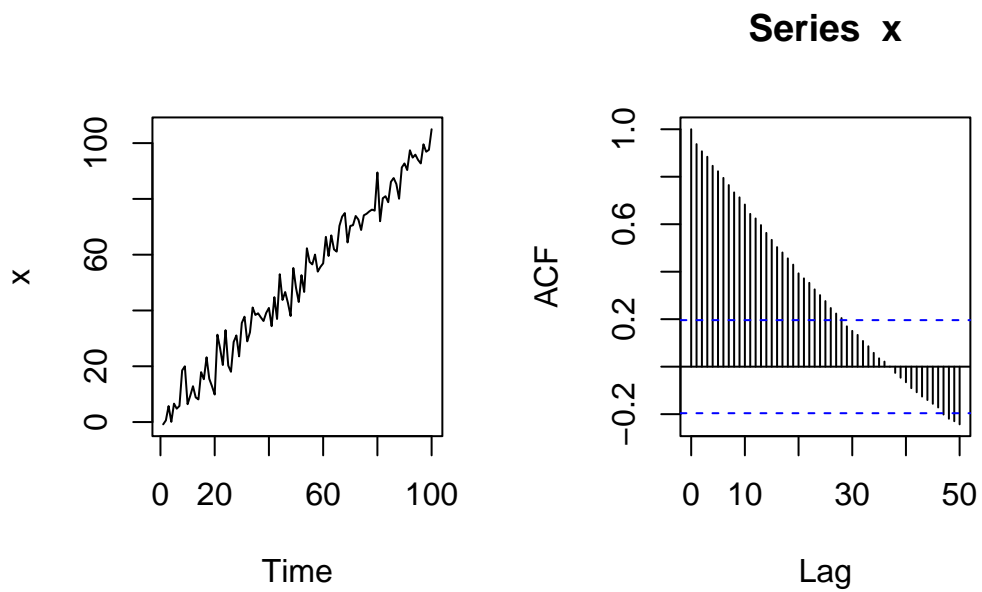
Considere, por exemplo, a série

$$x_t = t + \varepsilon_t,$$

onde  $\varepsilon_t \sim \text{Normal}(0, 5^2)$ . Abaixo simulamos essa série e apresentamos o respectivo correlograma

```
x <- rnorm(100, 1:100, 5)

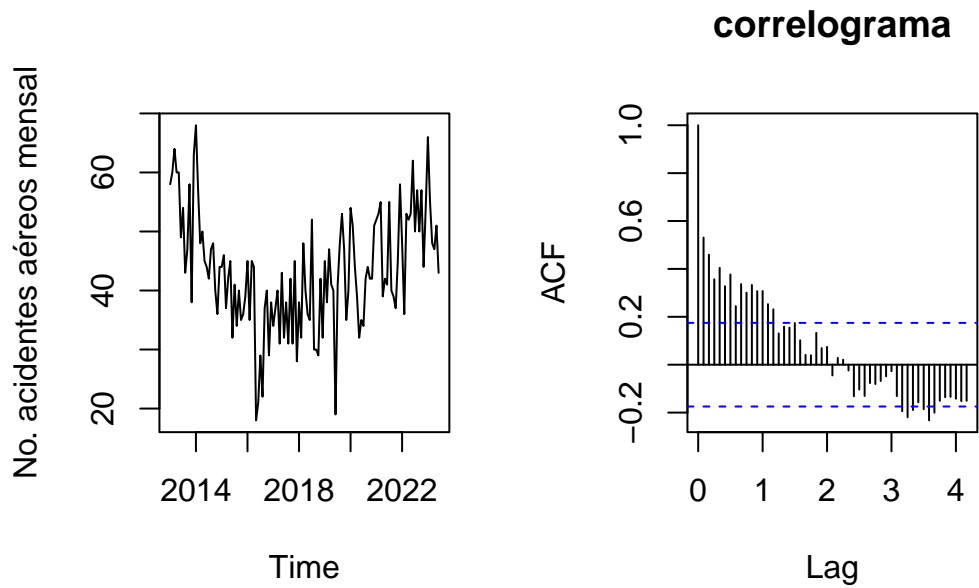
oo <- par( mfrow=c(1,2))
ts.plot(x)
acf(x, lag = 50)
```



```
par(oo)
```

Observe as similaridades do correlograma acima com o observado para a série de acidentes aéreos mensais vista anteriormente.

```
oo <- par( mfrow=c(1,2))
ts.plot( fab_mes , ylab = 'No. acidentes aéreos mensal' )
acf(fab_mes , lag = 50, main ='correlograma')
```



`par(oo)`

### 4.3 O correlograma com a componente de sazonalidade - sinal harmônico

O sinal sazonal é caracterizado por um comportamento periódico. Existem dois comportamentos sazonais típicos. O primeiro é baseado na função harmônica:

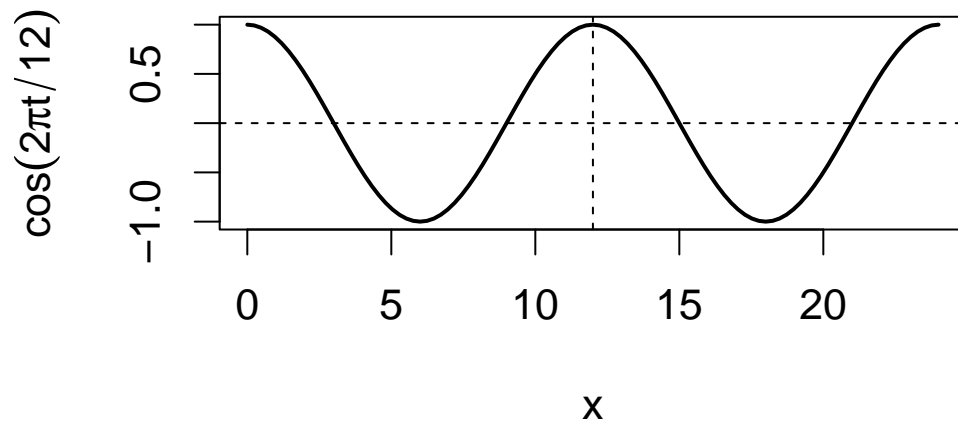
$$\text{sinal}(t) = A \cos\left(\frac{2\pi}{p}t + \phi\right)$$

Neste tipo de sinal, há um comportamento em forma de onda já estabelecido. Eis algumas informações importantes:

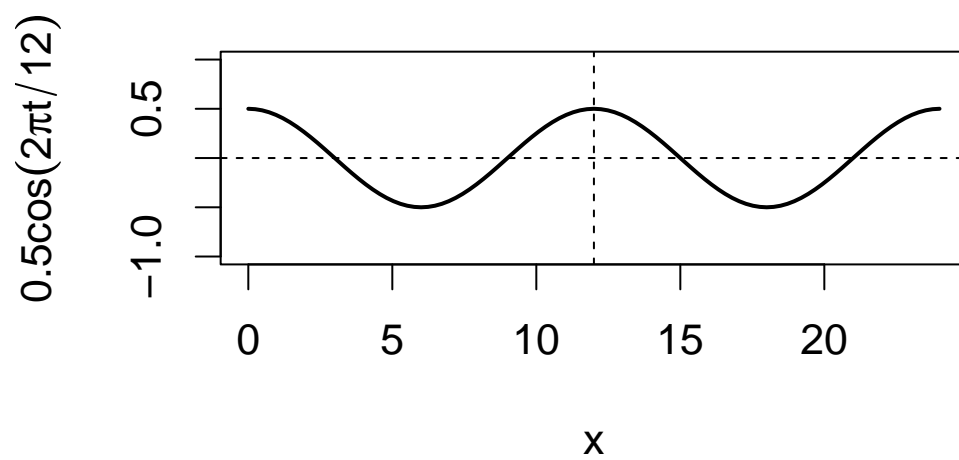
- O valor  $p$ , denominado período, equivale ao tempo que demora para o padrão se repetir.
- $A$  é denominado amplitude e representa o maior/menor valor que este sinal pode atingir.
- Por último,  $\phi$  é denominado fase, e serve basicamente para deslocar a onda.

Abaixo seguem alguns exemplos de harmônicos, todos com período 12:

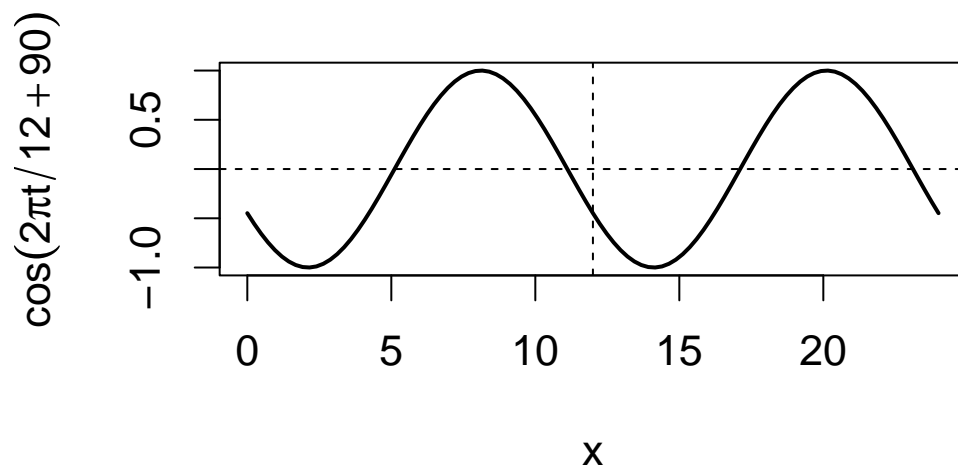
```
oo <- par( cex = 1.3)
curve( cos( x* 2*pi/12), 0,24, lwd = 2, ylab = expression( cos( 2*pi*t/12 )))
abline(h = 0, lty = 2 )
abline(v=12, lty = 2)
```



```
curve( .5*cos( x* 2*pi/12), 0,24, lwd = 2, ylab = expression( .5*cos( 2*pi*t/12 )), ylim =
abline(h = 0, lty = 2 )
abline(v=12, lty = 2)
```



```
curve( cos( x* 2*pi/12+90), 0,24, lwd = 2, ylab = expression( cos( 2*pi*t/12 +90)), ylim =
abline(h = 0, lty = 2 )
abline(v=12, lty = 2)
```

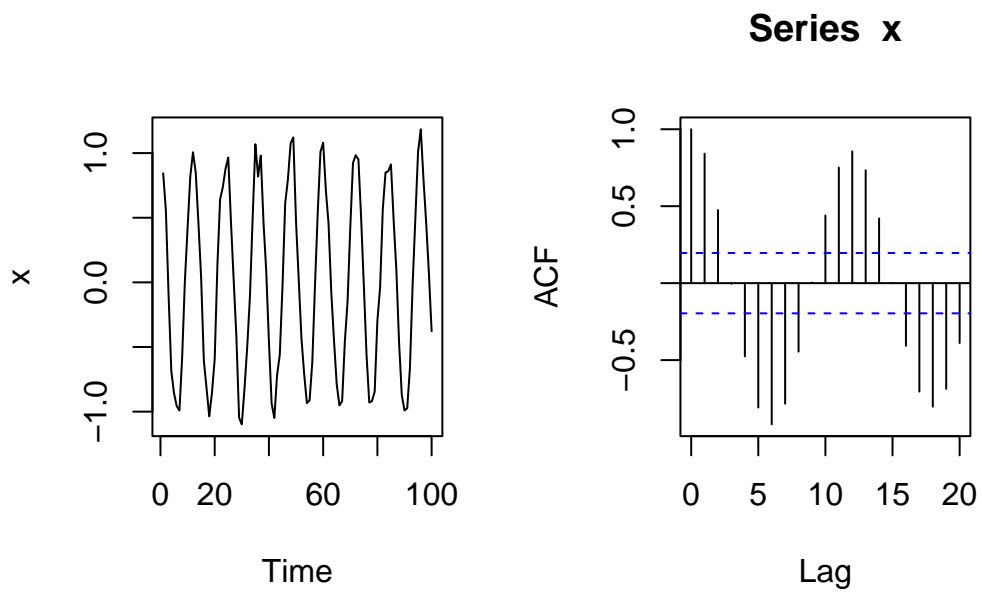


Abaixo simulamos uma série temporal com um sinal do tipo harmônico. Observe que o comportamento em forma de onda é aparente na função de autocorrelação.

```
x <- cos( 2*pi/12 * 1:100) + rnorm(100,0,.1)

oo <- par( mfrow = c(1,2))
ts.plot(x)
acf(x)
```

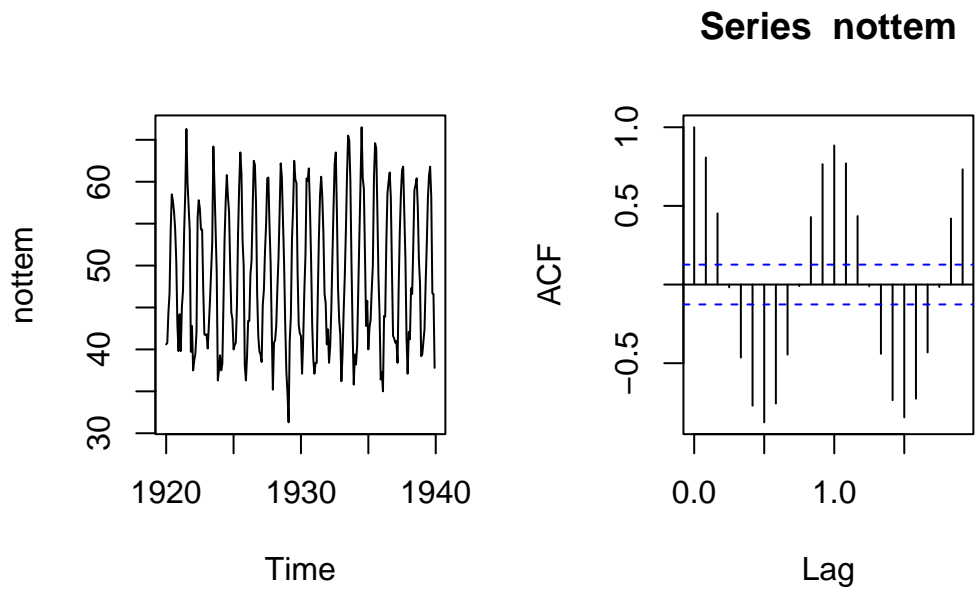




```
par(oo)
```

Abaixo, apresentamos a temperatura mensal observada no Castelo de Nottingham, entre 1920-1939. Compare os resultados com os gráficos acima.

```
oo <- par( mfrow = c(1,2))
plot(nottem)
acf(nottem)
```



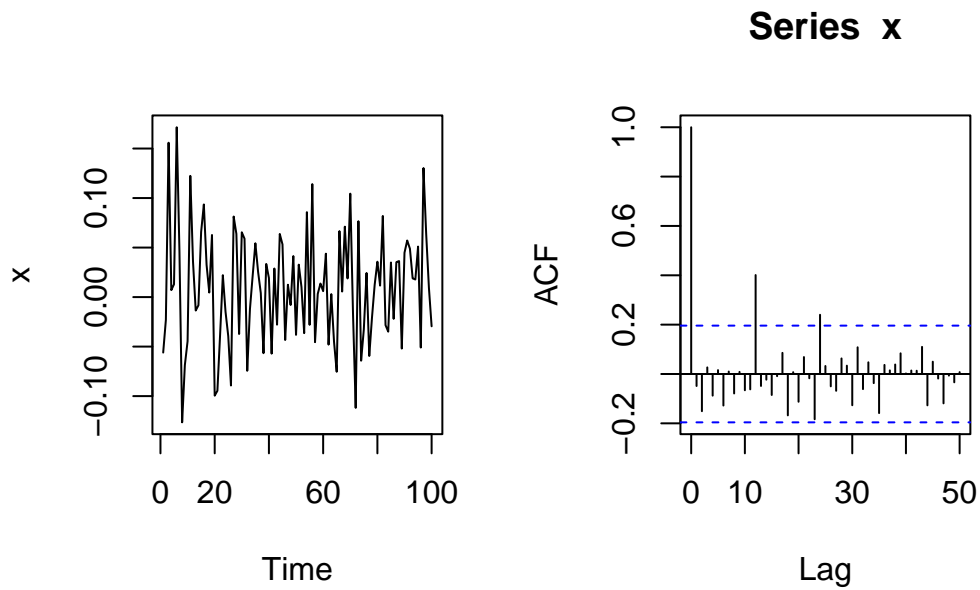
```
par(oo)
```

#### 4.4 O correlograma com a componente de sazonalidade - sinal autorregressivo

Nesse tipo de sazonalidade, ainda há um período  $p$ , mas não há um sinal harmônico. O valor da série no tempo  $t$  é baseado no valor observado no tempo  $t - p$ .

Quando a sazonalidade possui essa característica, há uma autocorrelação marcante nos múltiplos de  $p$ . Observe a série simulada abaixo, com um período  $p = 12$

```
set.seed(123)
oo <- par( mfrow = c(1,2))
x <- rnorm(12,0,.1)
for(i in 13:100) x[i] <- .6*x[i-12] + rnorm(1,0,.05)
ts.plot(x)
acf(x, lag = 50)
```



```
par(oo)
```

## 4.5 Exercícios

Exercício 1 Estude o comportamento da série `ldeaths`, que conta o número mensal de óbitos por doenças pulmonares no Reino Unido.

Exercício 2 Estude o comportamento da série do número de óbitos maternos mensais.

Exercício 3 Em 2017, um epidemiologista estava interessado na série de suicídios no Mato Grosso do Sul. O banco de dados utilizado é dado a seguir. Construa uma série mensal e estude seu comportamento

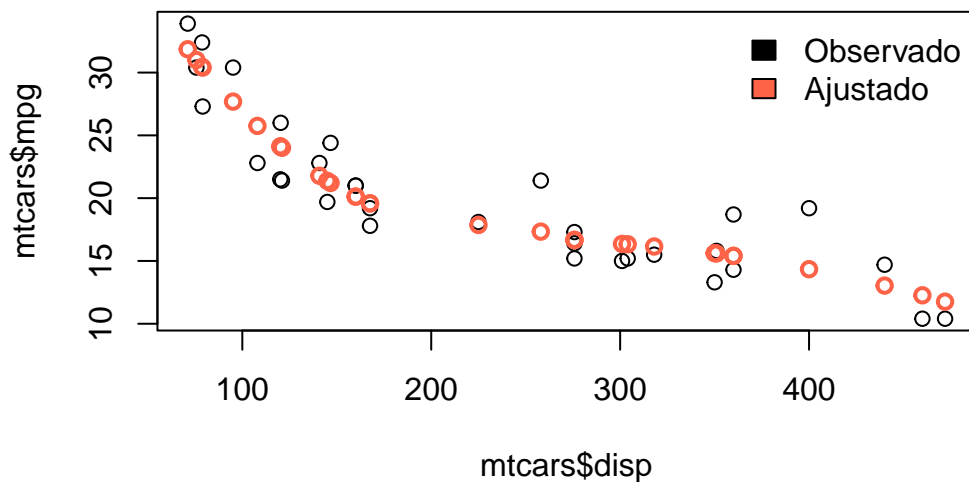
```
url <- 'https://drive.google.com/uc?authuser=0&id=1DMSgrQDl0636Lw0Y0MYJHJrgw_2uXntM&export=csv'
```

## 5 Mais ferramentas exploratórias

### 5.1 Estudo da tendência utilizando o loess

*Loes* é um modelo de regressão não linear não paramétrico. Abaixo mostramos como utilizá-lo considerando um banco de dados com diversas marcas de veículos e a relação entre a variável milhas por galão (mpg) e deslocamento (disp)

```
plot( mtcars$disp, mtcars$mpg )  
lw <- loess( mpg ~ disp, data = mtcars )  
points(mtcars$disp, lw$fitted, col = 'tomato', lwd = 2)  
legend('topright', c('Observado', 'Ajustado'), fill=c(1, 'tomato'), bty='n')
```



Podemos estimar a tendência utilizando o loess, imaginando uma regressão do tipo:

$$E(X_t) = g(t),$$

ou seja, utilizando o tempo como regressora. Vamos ilustrar a ideia utilizando a série de acidentes aéreos mensais da FAB.

```
# criando a variável regressora
tempo <- 1 : length(fab_mes)

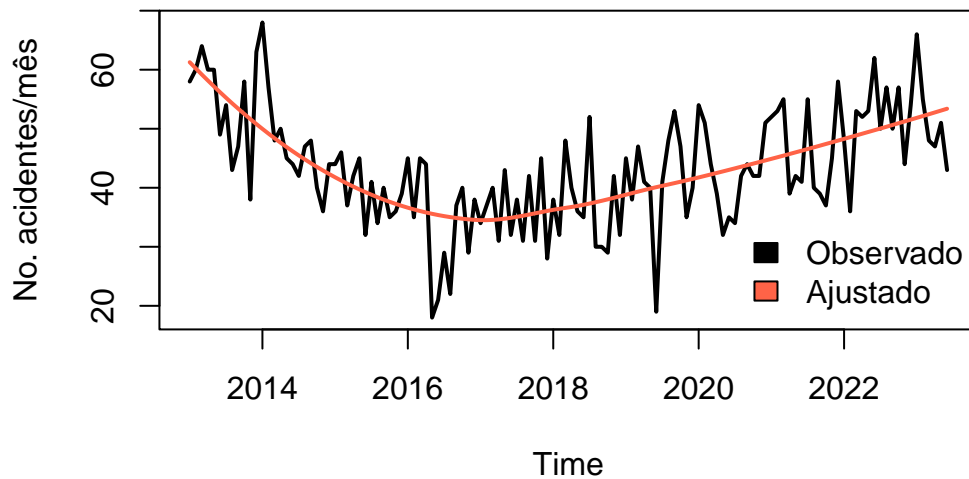
# aplicando o loess
lw <- loess( fab_mes ~ tempo)

# transformando o valor predito em uma série temporal

fit <- ts(lw$fitted, start = start(fab_mes), frequency = frequency(fab_mes) )

# gráfico da tendência estimada

ts.plot( fab_mes, ylab = 'No. acidentes/mês' , lwd = 2)
lines(fit, lwd = 2, col = 'tomato')
legend('bottomright', c('Observado','Ajustado'),fill=c(1,'tomato'), bty='n')
```



Acima estimamos a tendência. Denomine este sinal estimado por  $\hat{g}(t)$ . Agora, considere a

série

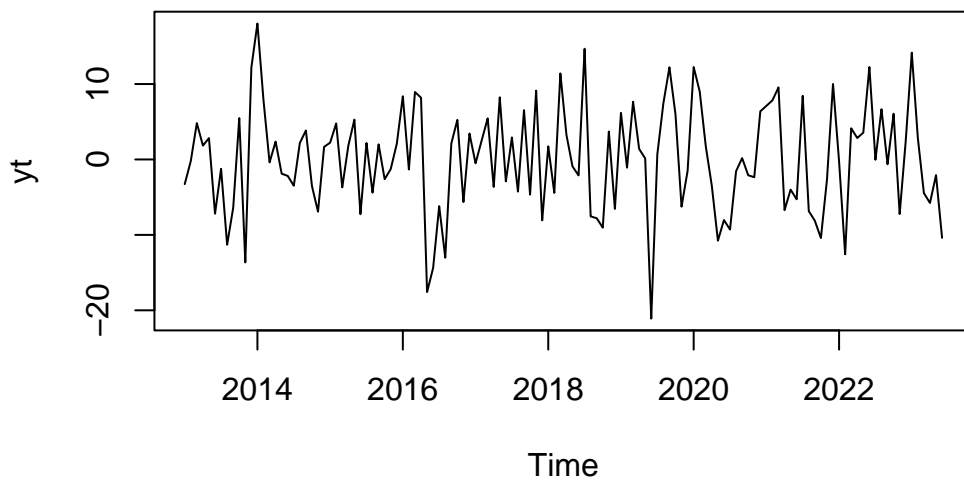
$$y_t = x_t - \hat{g}(t).$$

Ao analisar esta série, duas coisas podem acontecer:

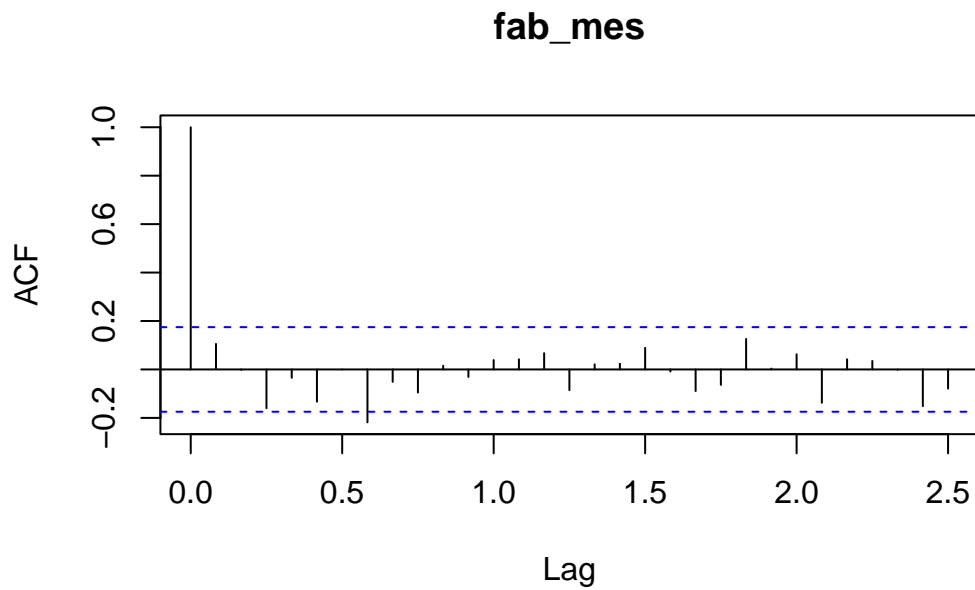
- Vamos encontrar um comportamento semelhante a um ruído (correlograma com barras praticamente nulas)
- Vamos encontrar algum outro sinal ainda não ajustado.

Os gráficos abaixo mostram que não há mais sinais para procurar

```
yt <- fab_mes - fit  
  
ts.plot(yt)
```



```
acf(yt, lag = 30)
```



Portanto, esta série pode ser escrita como

$$x_t = \text{tendência}_t + \varepsilon_t,$$

onde  $\varepsilon_t$  é um ruído branco.

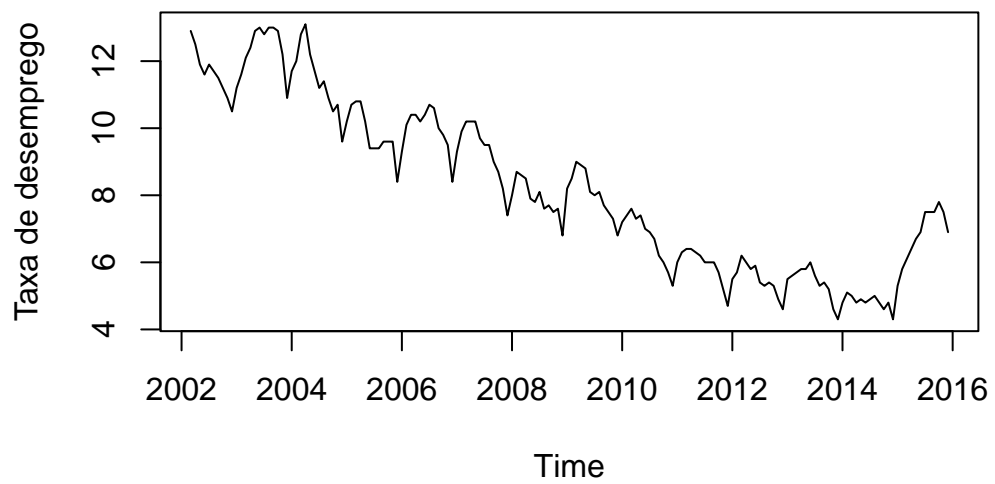
Abaixo, vamos analisar a série de taxa de desemprego mensal, entre março de 2002 e dezembro de 2015.

```
url <- 'https://www.dropbox.com/s/rmgymzsic99qawd/desemprego.csv?dl=1'

banco <- fread(url)

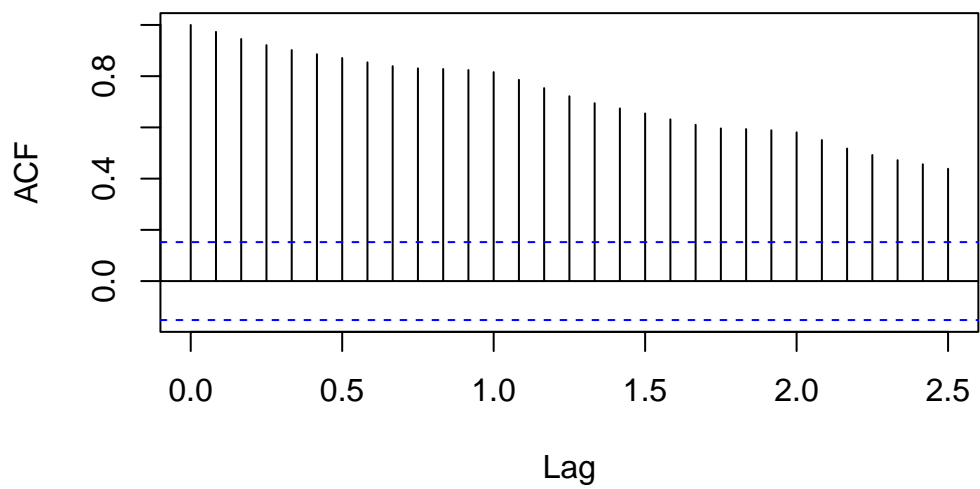
desemprego<- ts( banco[, 'V2', ], start = c(2002,3), frequency=12)

ts.plot(desemprego, ylab = 'Taxa de desemprego')
```



```
acf(desemprego, lag = 30)
```

**V2**





É possível verificar que há tendência e sazonalidade na série. Vamos estimar a componente de tendência primeiro.

```
# criando a variável regressora
tempo <- 1 : length(desemprego)

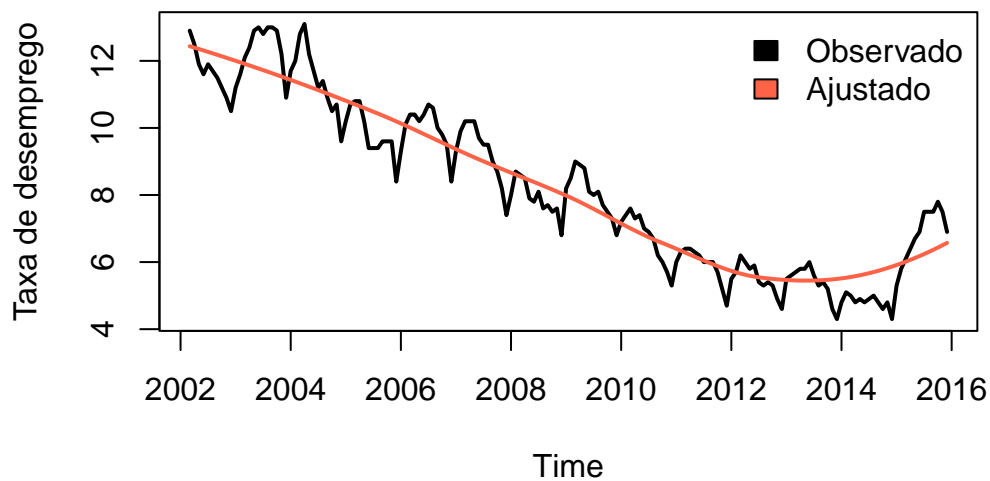
# aplicando o loess
lw <- loess( desemprego ~ tempo)

# transformando o valor predito em uma série temporal

fit <- ts(lw$fitted, start = start(desemprego), frequency = frequency(desemprego) )

# gráfico da tendência estimada

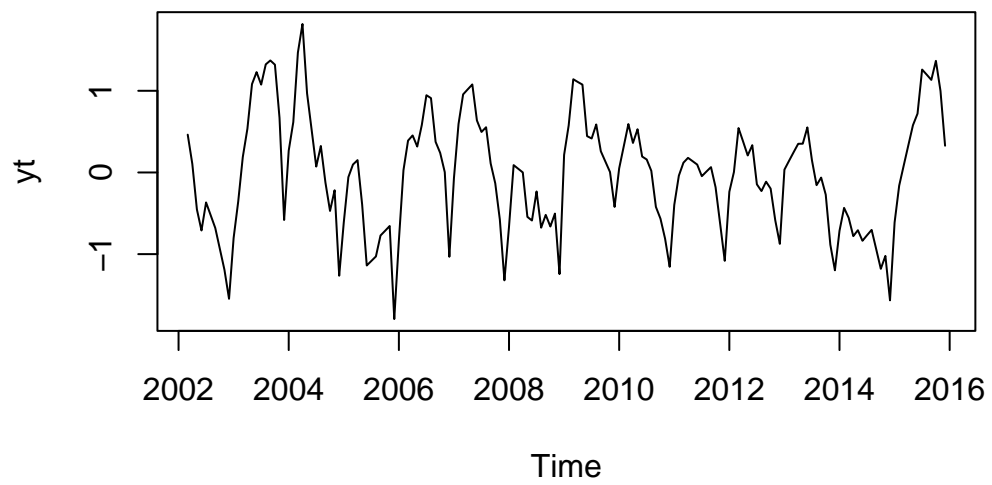
ts.plot( desemprego, ylab = 'Taxa de desemprego' , lwd = 2)
lines(fit, lwd = 2, col = 'tomato')
legend('topright', c('Observado','Ajustado'),fill=c(1,'tomato'), bty='n')
```



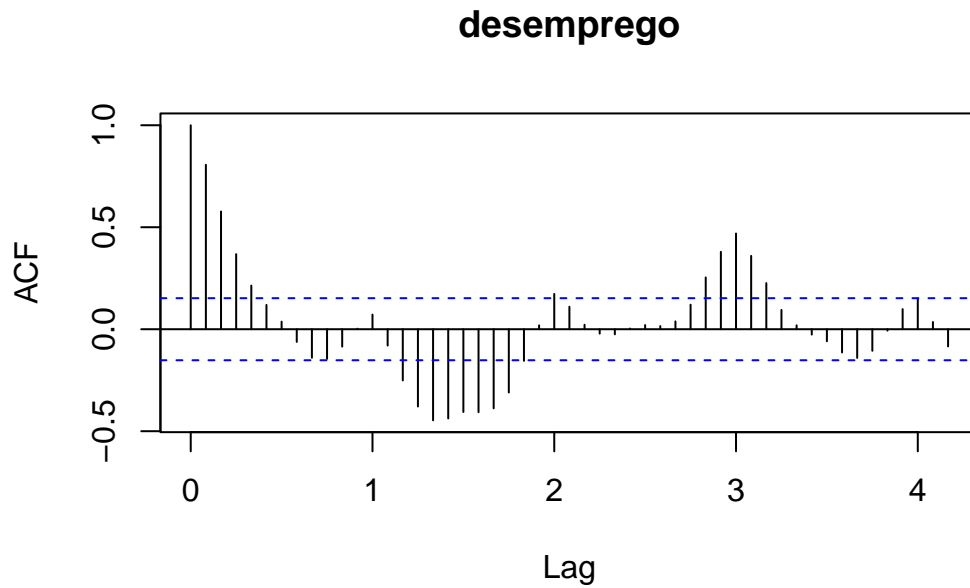
Vamos eliminar a tendência estimada e avaliar o restante.

```
yt <- desemplego - fit
```

```
ts.plot(yt)
```



```
acf(yt, lag = 50)
```



Fica claro o comportamento sazonal. Note que o período não parece ser anual, mas sim de 3 em 3 anos. Vamos avaliar esse aspecto com mais detalhes na próxima seção.

### 5.1.1 Exercícios

Exercício 1 Considerando o banco de dados sobre suicídios no Mato Grosso do Sul:

1. Estime a tendência
2. Remova a tendência estimada e verifique se o resultado é um ruído branco

Exercício 2 Verifique se há tendência na série `ldeaths`.

Exercício 3 Verifique se há tendência na série de óbitos maternos, cuja url é

```
url <- 'https://drive.google.com/uc?authuser=0&id=1tYFFT9L2iopKmBDUI3P8qNIRaOnMYj7d&export=pdf'
```

Faça duas análises, uma com a série inteira e outra eliminando os dados a partir de 2020.

Exercício 4. O banco de dados abaixo apresenta algumas séries temporais mensais com o número de nascidos vivos em Manaus

```
url <- 'https://drive.google.com/uc?authuser=0&id=139h6x2g7PkAHNTzsbQKU15G2MqoXYk6Y&export=pdf'
```

1. Estime a tendência dos nascimentos considerando duas séries: partos vaginais e cesários. Coloque as duas informações no mesmo gráfico.
2. Elimine a tendência de cada série e verifique se há outro sinal a ser estimado.

Exercício 1

## 5.2 O periodograma

### 5.2.1 Introdução

Todo padrão sazonal possui um período - a quantidade de tempo necessária para que o padrão se repita. O inverso desse período é denominado frequência fundamental, que é a fração de um ciclo por unidade de tempo.

**Exemplo:** Considere o período de 12 meses. Então, a frequência fundamental é  $1/12$  (ou seja, cada mês representa um doze ávos do período de 1 ano).

Lembremos que o sinal harmônico é igual a

$$\text{sinal}(t) = A \sin(2\pi\omega t + \phi),$$

onde  $\omega = 1/p$  é a frequência. Com um pouco de trigonometria, podemos mostrar que

$$\text{sinal}(t) = \beta_1 \cos(2\pi\omega t) + \beta_2 \sin(2\pi\omega t)$$

onde  $\beta_1 = A \cos(\phi)$  e  $\beta_2 = -A \sin(\phi)$ . É possível mostrar também que  $A = \sqrt{\beta_1^2 + \beta_2^2}$  e  $\phi = \cos^{-1}(\beta_1/A)$ . A vantagem dessa nova forma é que o sinal pode ser escrito como um modelo linear e pode ser estimado facilmente. A soma de quadrados explicada pela regressão é proporcional à

$$I(\omega) = \hat{A}(\omega)^2$$

e podemos mostrar que a estimativa de máxima verossimilhança para  $\omega$  é o valor que maximiza  $I(\omega)$ .

Periodograma O gráfico de  $I(\omega)$  é denominado periodograma.

Vamos criar a função periodograma

```
Iw <- function(y,w){
  # matriz de planetamento
  n <- length(y)
  t <- 1:n
```

```

x <- cbind(cos(2*pi*w*t), sin(2*pi*w*t))

# coeficientes do modelo linear
beta <- coefficients(lm(y ~x-1))

# amplitude
A <- sqrt(sum(beta^2))

# Iw
.5*n*A^2
}

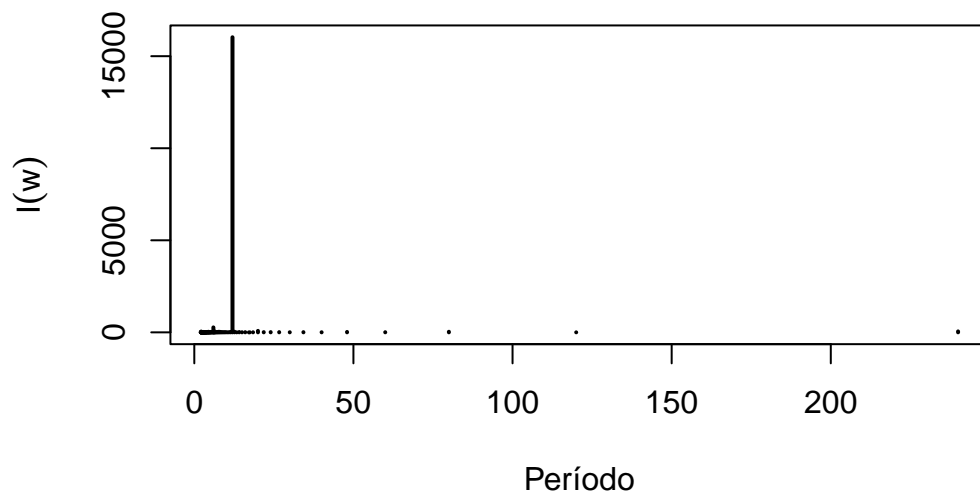
periodograma <- function(y){
  # gráfico
  n <- length(y)
  w_detec <- ( 1 : floor( (n-1)/ 2) ) / n
  I_w <- sapply( w_detec, function(w) Iw(y , w) )
  plot( 1 / w_detec , I_w, xlab = 'Período', ylab = expression(I(w)), type = 'h' , lwd

  # encontrando o periodo
  fund <- w_detec[ which( I_w == max(I_w))]
  cat('Período: ',1/fund,'\n')
}

```

Na função acima  $y$  é a série temporal. Vamos aplicar essa função para a série de temperaturas no Castelo de Nottingham.

```
periodograma(nottem)
```

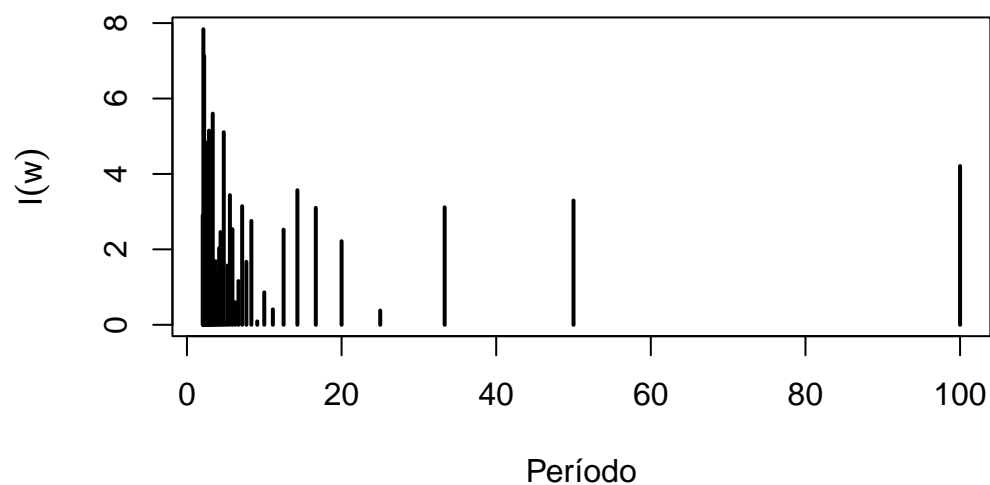


Período: 12

### 5.2.2 Periodograma para amostras aleatórias

É importante notar que a frequência fundamental é um pico expressivo em relação aos demais. Abaixo mostramos o periodograma para uma amostra aleatória - note como há vários picos, evidenciando a falta de uma frequência fundamental.

```
periodograma(rnorm(100))
```

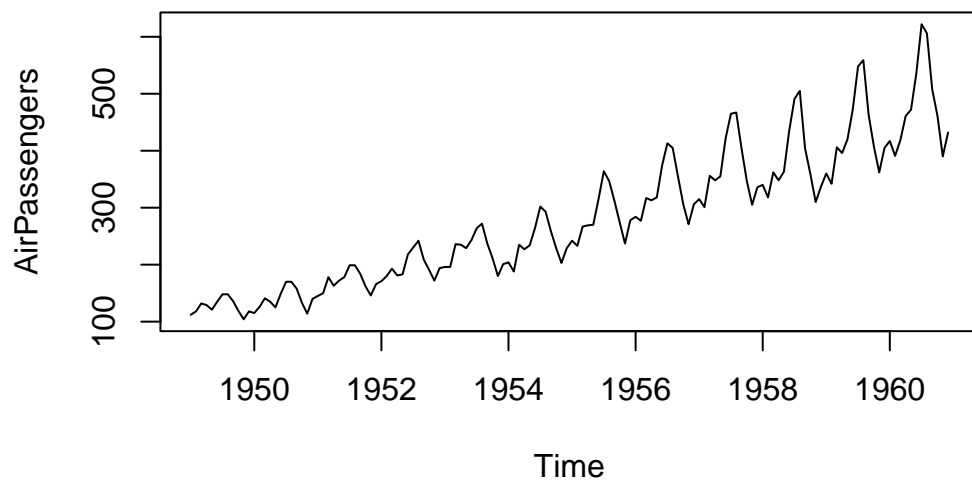


Período: 2.12766

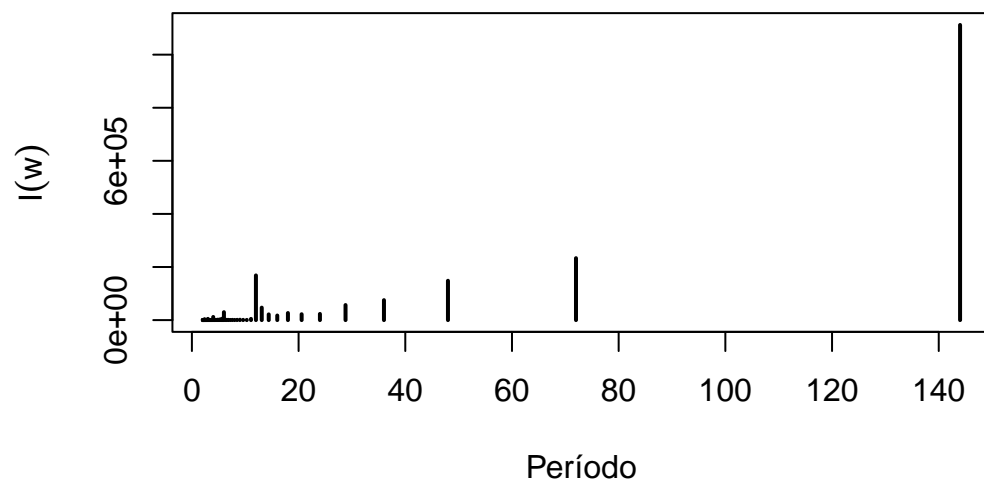
### 5.2.3 Periodograma na presença de tendência

É importante remover a tendência antes de aplicar o período. Por exemplo, considere novamente a série `AirPassengers`

```
ts.plot(AirPassengers)
```



```
periodograma(AirPassengers)
```





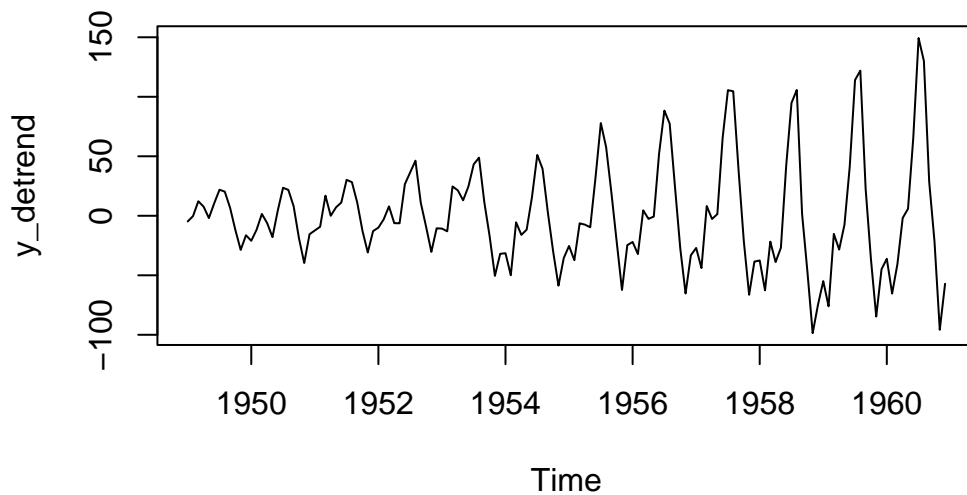
Período: 144

Embora exista uma sazonalidade clara, o periodograma retorna um período de 12 anos, algo irreal. Vamos remover a tendência.

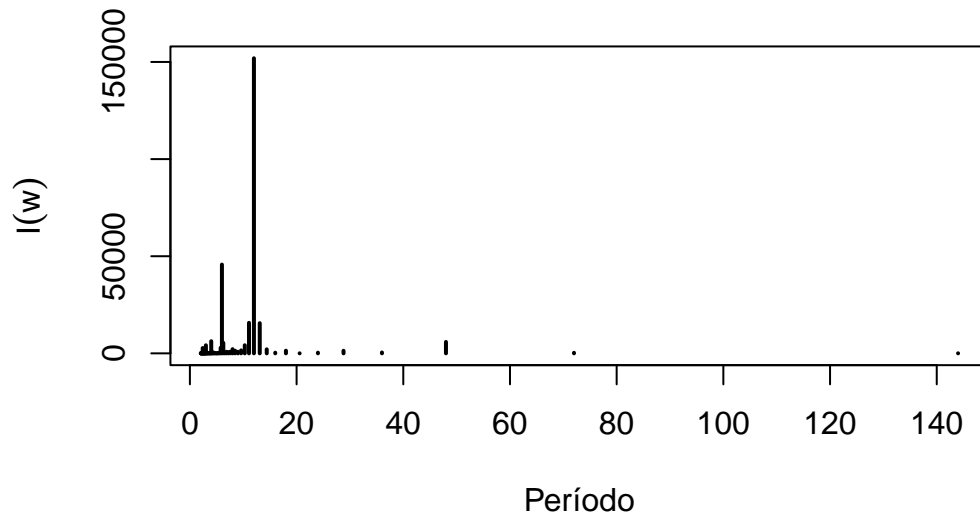
```
# criando o loess
y <- AirPassengers
tempo <- 1:length(y)
lw <- loess( y ~ tempo )
fit <- lw$fitted

# criando a série livre de tendência
y_detrend <- ts( y-fit, start = start(y), frequency = frequency(y))

ts.plot( y_detrend )
```



```
periodograma( y_detrend )
```



Período: 12

### 5.2.4 Exercícios

Exercício 1. Determine o período da série `ldeaths` - número de óbito mensais por doenças pulmonares no Reino Unido.

Exercício 2. Determine o período da série de óbitos maternos, cuja url é:

```
url <- 'https://drive.google.com/uc?authuser=0&id=1tYFFT9L2iopKmBDUI3P8qNIRaOnMYj7d&export=csv'
```

Exercício 3. A série `co2` representa a concentração de  $\text{CO}_2$  na atmosfera medida em Mauna Loa. Analise a tendência da série e estime o período.

Exercício 4. Determine a tendência e o período da série mensal do número de nascidos vivos em Manaus, independente do tipo de parto.

## 5.3 Ajuste por fatores sazonais

Considere novamente o harmônico

$$\text{sinal}(t) = A \cos\left(\frac{2\pi}{p}t + \phi\right)$$

O sinal no tempo  $t = 1 + p$  é equivalente ao sinal no tempo 1:

$$\text{sinal}(1 + p) = A \cos\left(\frac{2\pi}{p} + 2\pi + \phi\right) = A \cos\left(\frac{2\pi}{p} + \phi\right) = \text{sinal}(1)$$

Isso é verdade para todo  $t = 1 + kp$ , onde  $k = 1, 2, \dots$ . Isto implica que, na prática, a imagem de  $\text{sinal}(t)$  só pode ser  $p$  valores:

$$\text{sinal}(t) = \begin{cases} \text{sinal}(1), & t = 1, 1 + p, 1 + 2p, \dots, \\ \text{sinal}(2), & t = 2, 2 + p, 2 + 2p, \dots, \\ \vdots, & \vdots \\ \text{sinal}(p), & t = p, 2p, 3p, \dots, \end{cases}$$

Então

$$x_t = \text{sinal}(t) + \varepsilon_t$$

pode ser considerado uma ANOVA com um fator.

Considere a série `ldeaths`, que já sabemos ter uma leve tendência decrescente e sazonalidade com período 12. Primeiro, vamos encontrar a série livre de tendência:

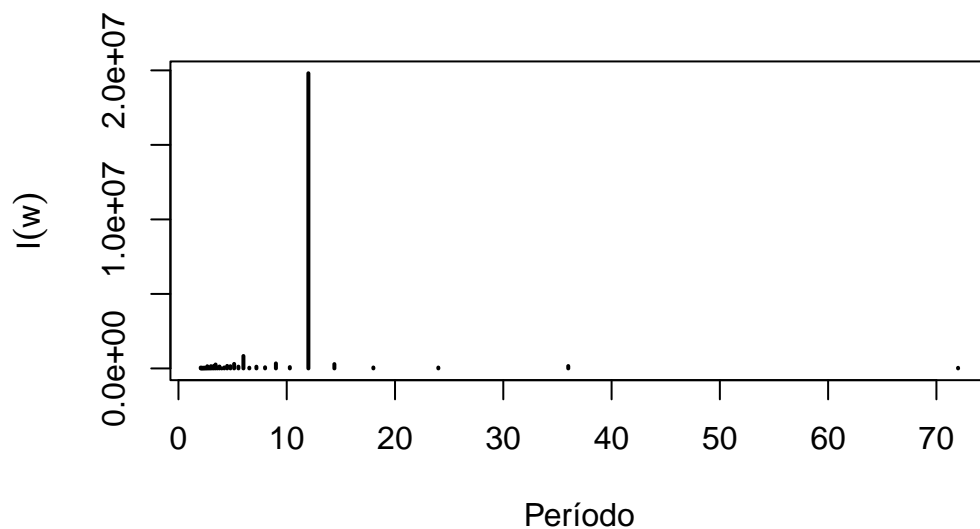
```
tempo <- 1:length(ldeaths)
lw <- loess( ldeaths ~ tempo)
tend <- lw$fitted

tend <- ts( tend, start = start(ldeaths), frequency = frequency(ldeaths))

# série sem tendência
d_ldeaths <- ldeaths - tend
```

Em seguida, encontramos o período.

```
periodograma(d_ldeaths)
```



Período: 12

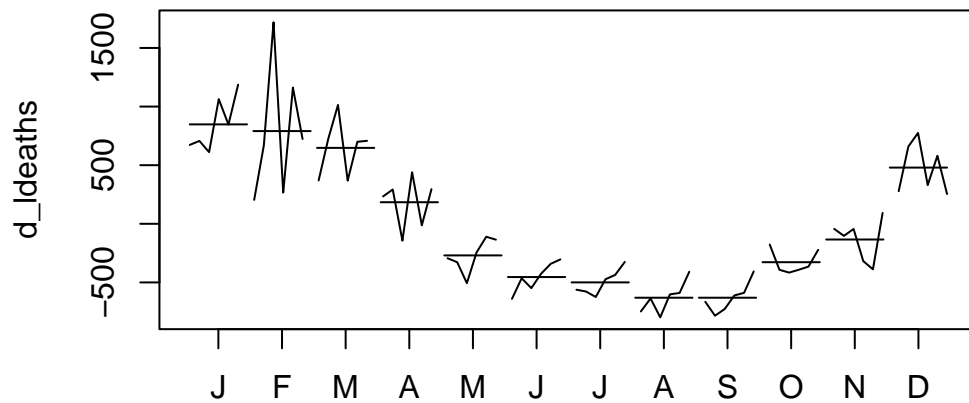
Nesse momento, devemos atribuir o período para o objeto `ts`, fazendo

```
d_ldeaths <- ts(d_ldeaths, frequency= 12)
```

Note que o código acima foi inócuo porque o objeto `ldeaths` já tinha o período 12.

Antes de fazer o ajuste sazonal, é possível fazer um gráfico com doze séries temporais, uma para cada fator sazonal (no nosso exemplo, uma série só de janeiros, de fevereiro, etc). Mostramos isso a seguir:

```
monthplot(d_ldeaths)
```



No gráfico acima, a linha horizontal em cada mês representa a média.

Agora, vamos estimar os 12 fatores sazonais. Para isso, vamos utilizar a função `cycle`, que mostra a posição de cada observação dentro do ciclo sazonal. Eis um exemplo de seu funcionamento:

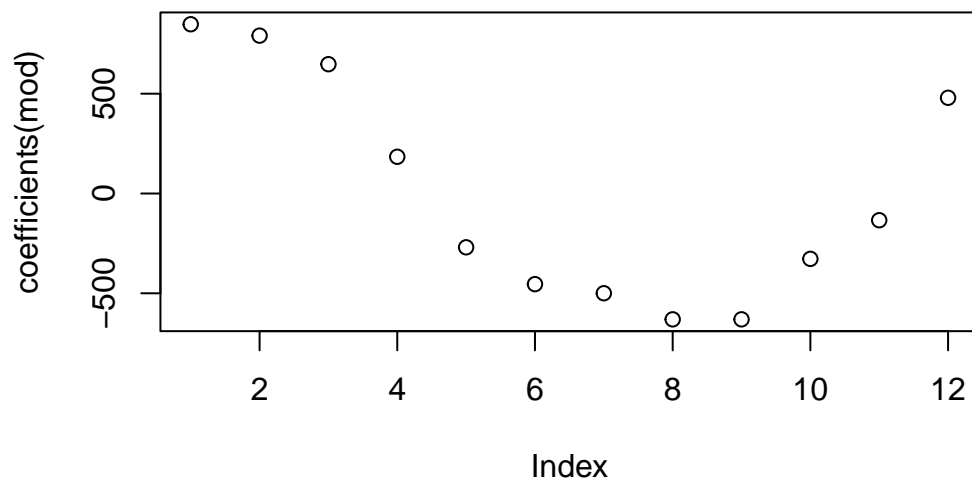
```
head(cycle(d_ldeaths), 14)
```

```
[1] 1 2 3 4 5 6 7 8 9 10 11 12 1 2
```

Vamos transformar o resultado do `cycle` em um fator e ajustar uma ANOVA:

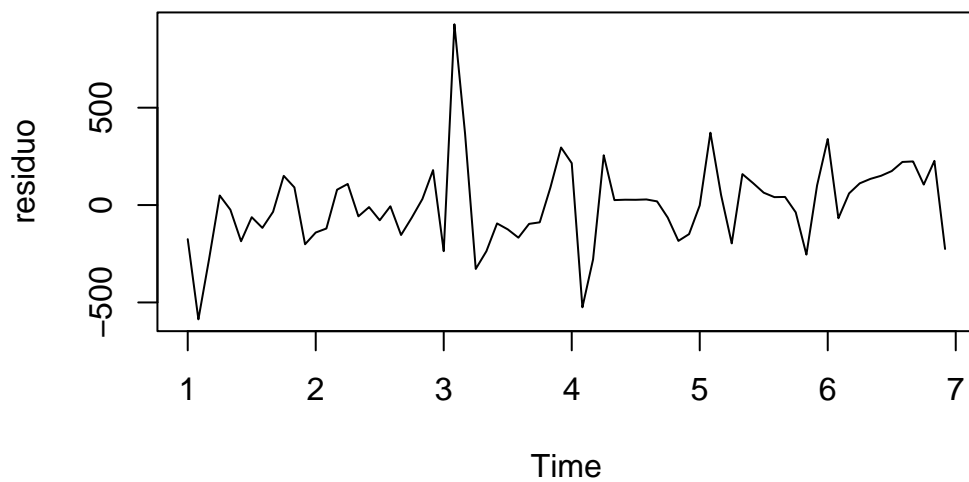
```
ciclo <- as.factor(cycle(d_ldeaths))
mod <- lm( d_ldeaths ~ciclo-1)

plot(coefficients(mod))
```



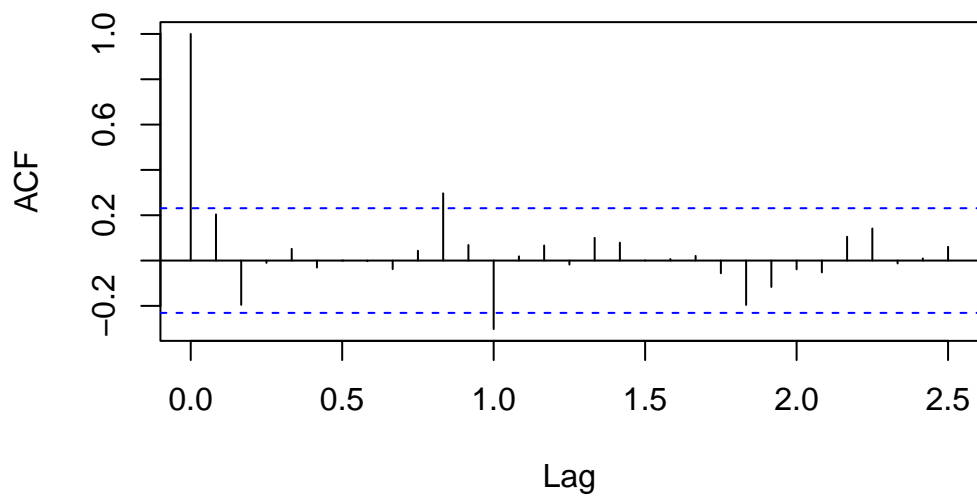
Por último, vamos dessazonalizar a série:

```
saz_fit <- mod$fitted.values  
residuo <- d_ldeaths - saz_fit  
  
plot(residuo)
```



```
acf(residuo, lag = 30)
```

### Series residuo



### 5.3.1 Exercícios