

Introdução

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.

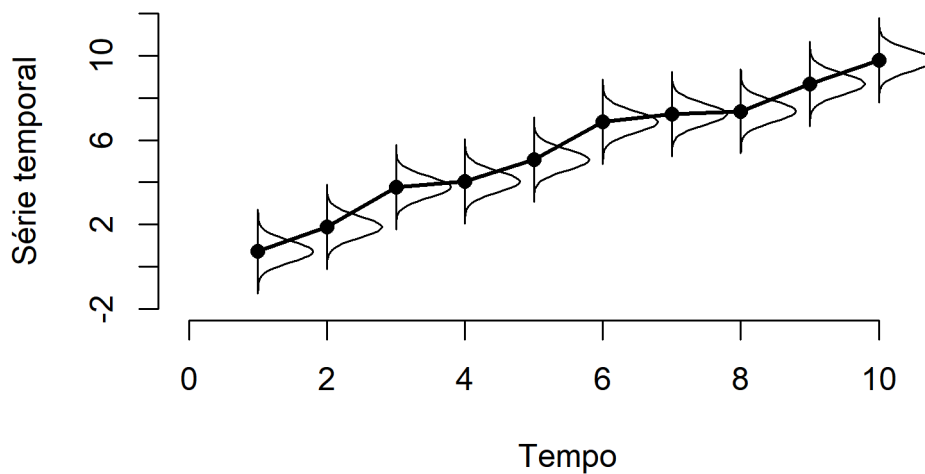


Figure 1: Figure 1 - Ilustração de uma série temporal

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.

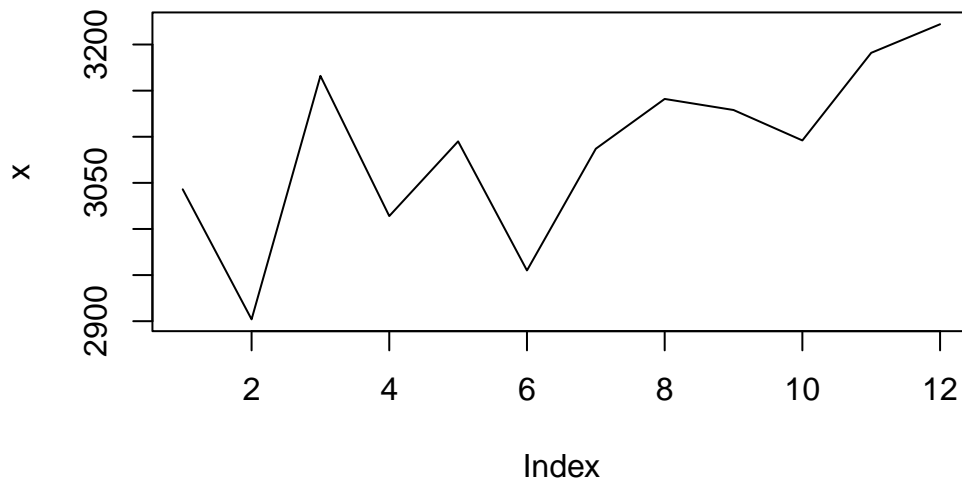
A classe `ts`

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')
```



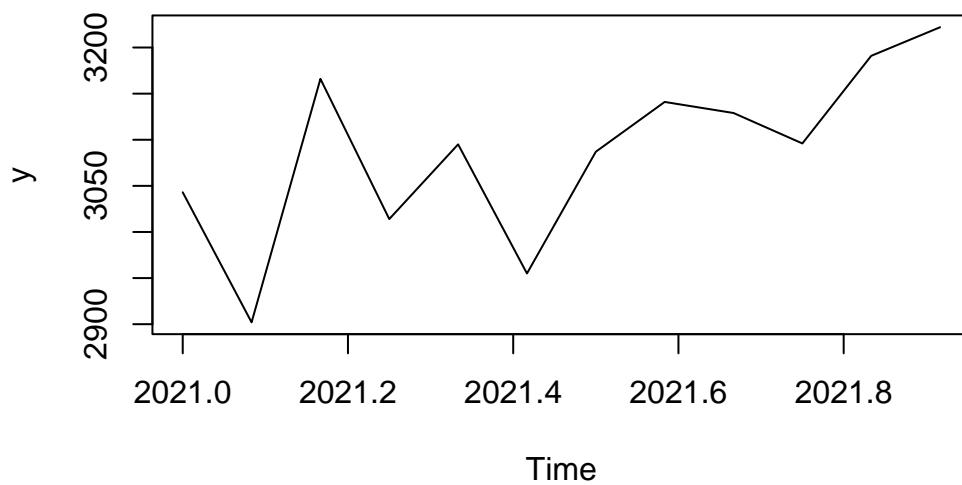
Contudo, é útil construir a série temporal como um objeto da classe `ts`. Tal função possui dois argumentos importantes:

- **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

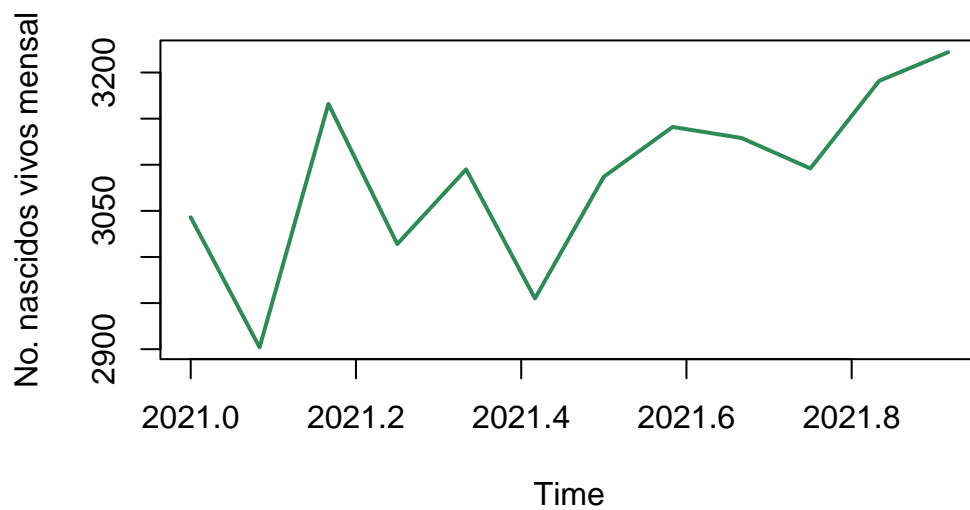
```
ts.plot(y)
```



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).

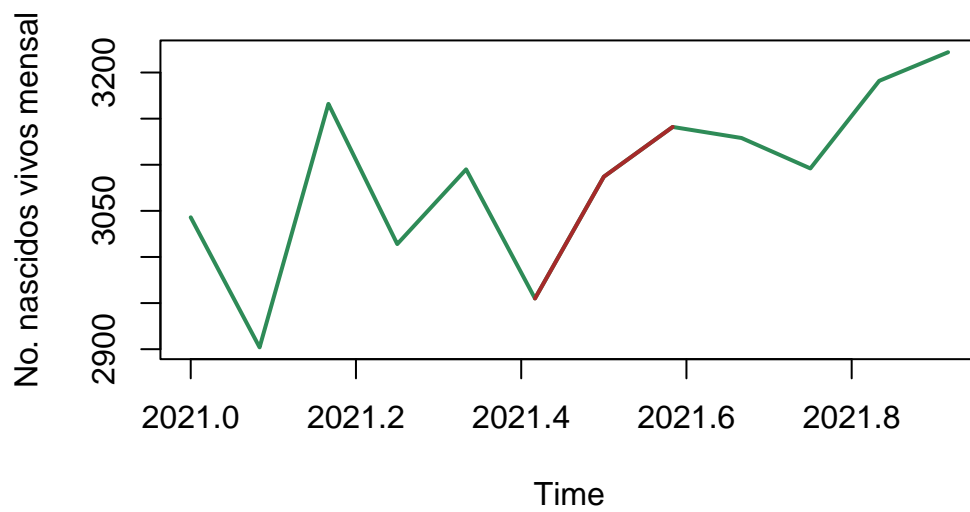
Também podemos customizar o gráfico.

```
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.

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
```

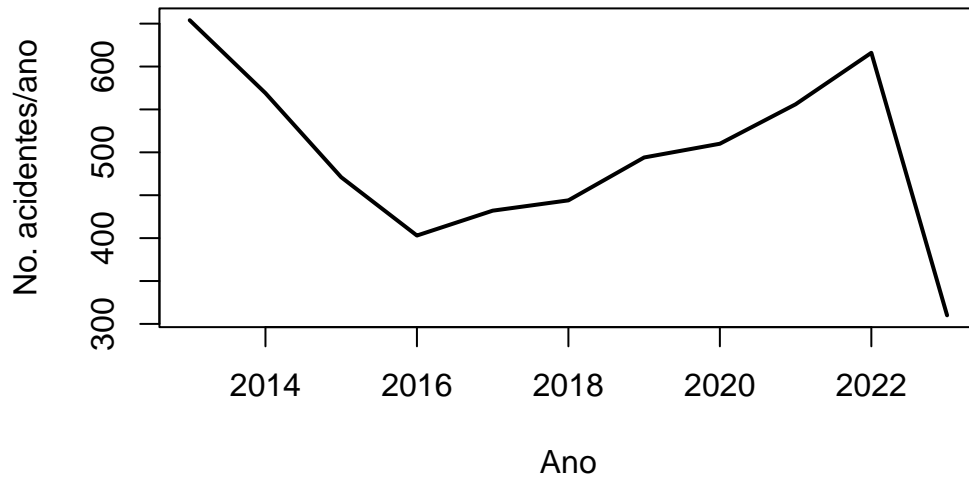
Ao utilizar o operador `.N` em `[, .N, c]`, é retornado o número de linhas que possuem o agrupamento em `c`. Vamos agrupar as datas do nosso banco por ano.

```
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')
```



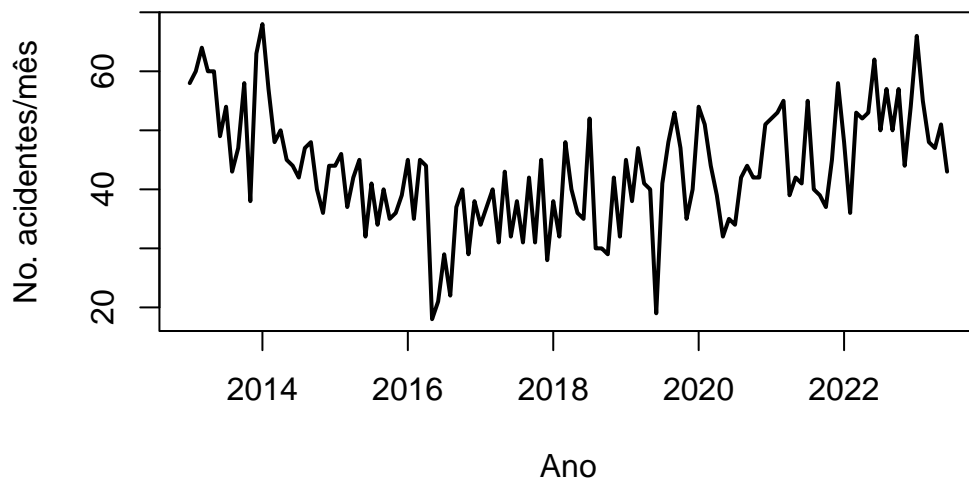
Também podemos fazer uma série mensal:

```
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.