

# Análise de Séries Temporais

## 0.2 - Aula Prática

Prof. Dr. Eder Angelo Milani

30/05/2025

### Algumas séries temporais

#### Série temporal do número de casos de tuberculose

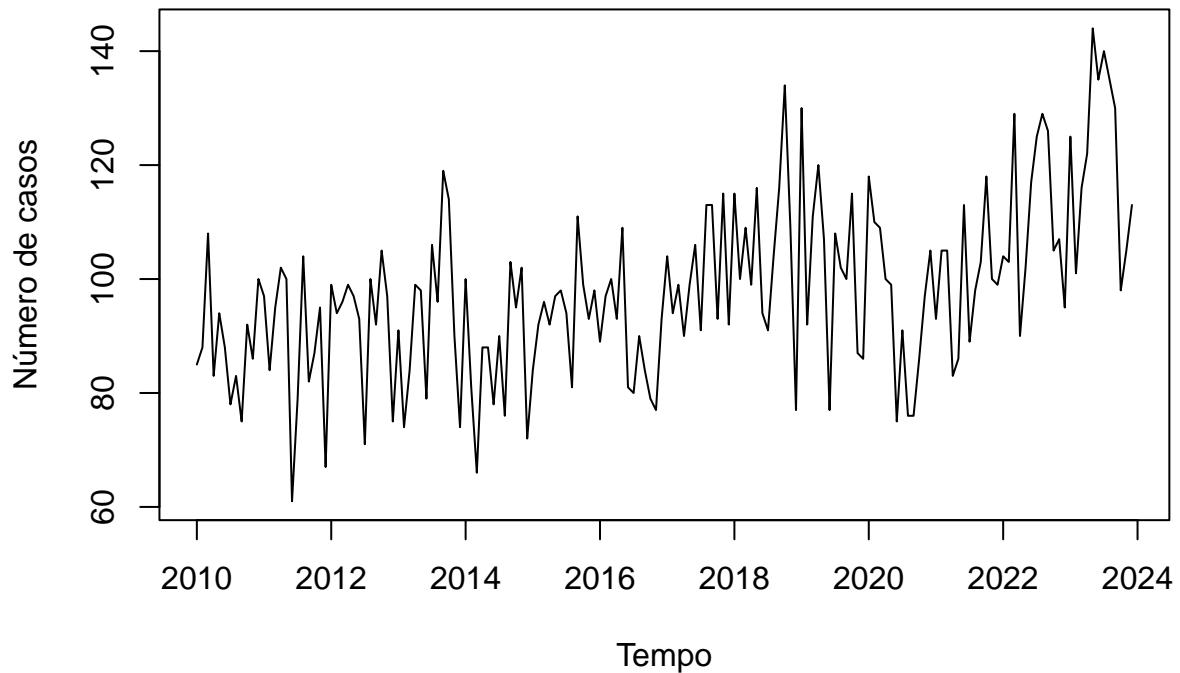
A seguir é apresentada a série temporal do número de casos de tuberculose no Estado de Goiás, compreendendo o número de casos mensal de janeiro de 2010 a dezembro de 2023. Os valores foram obtidos a partir do Sistema de Informação de Agravos de Notificação - Sinan Net, acessado em 20/09/2024, no link aqui.

```
# serie temporal do numero de casos de tuberculose em Goias
```

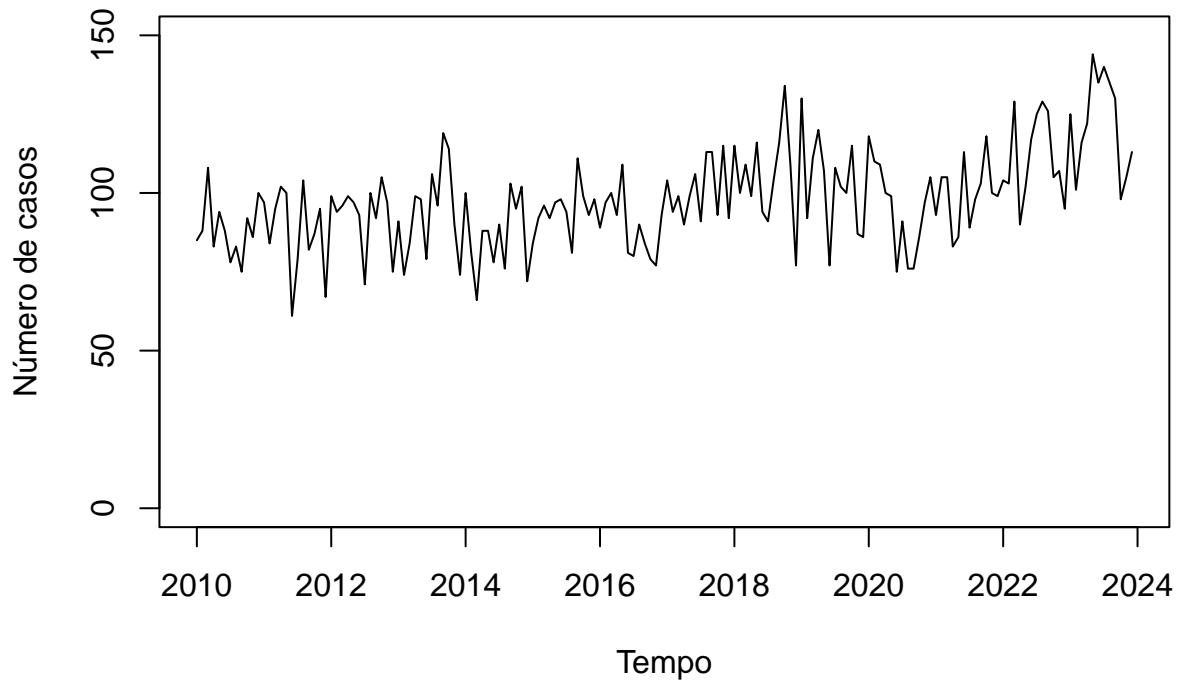
```
serie_tuberculose <- c(85,88,108,83,94,88,78,83,75,92,86,100,
  97,84,95,102,100,61,79,104,82,87,95,67,
  99,94,96,99,97,93,71,100,92,105,97,75,
  91,74,84,99,98,79,106,96,119,114,90,74,
  100,81,66,88,88,78,90,76,103,95,102,72,
  84,92,96,92,97,98,94,81,111,99,93,98,
  89,97,100,93,109,81,80,90,84,79,77,93,
  104,94,99,90,99,106,91,113,113,93,115,92,
  115,100,109,99,116,94,91,104,116,134,109,77,
  130,92,111,120,107,77,108,102,100,115,87,86,
  118,110,109,100,99,75,91,76,76,86,97,105,
  93,105,105,83,86,113,89,98,103,118,100,99,
  104,103,129,90,102,117,125,129,126,105,107,95,
  125,101,116,122,144,135,140,135,130,98,105,113)

tuberculose <- ts(serie_tuberculose, start= c(2010, 1), frequency = 12)

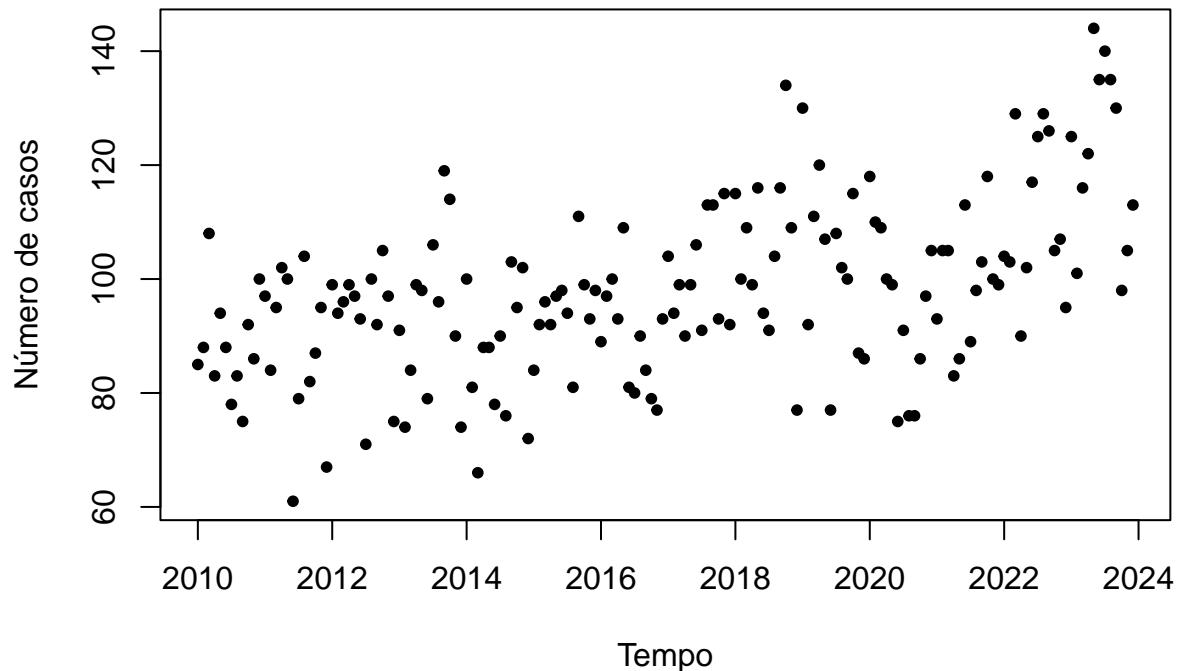
plot(tuberculose, ylab="Número de casos", xlab="Tempo")
```



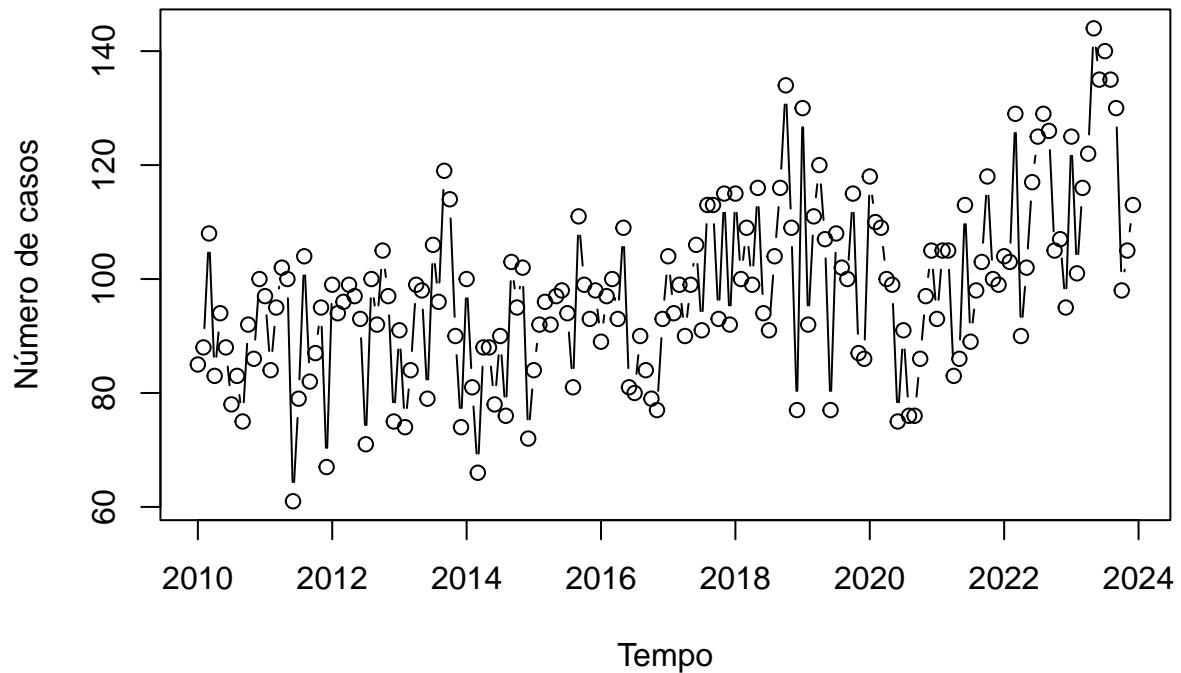
```
plot(tuberculosis, ylab="Número de casos", xlab="Tempo", ylim=c(0, 150))
```



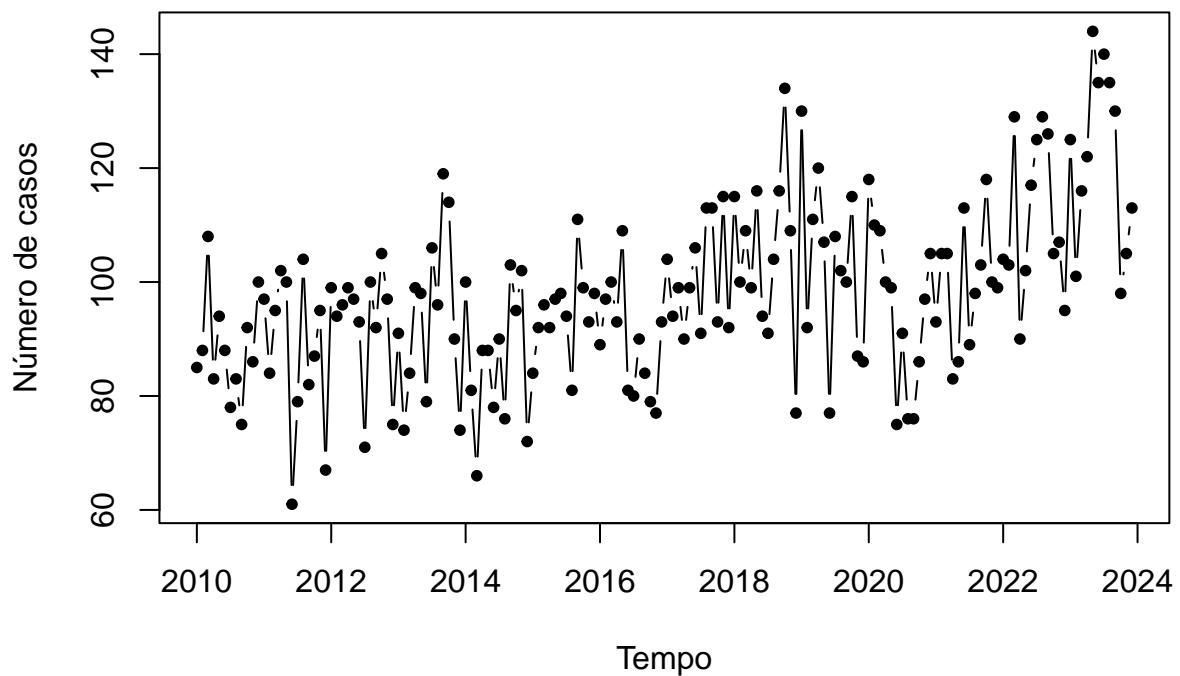
```
plot(tuberculosis, type="p", pch=20 , ylab="Número de casos", xlab="Tempo")
```



```
plot(tuberculosis, type="b", ylab="Número de casos", xlab="Tempo")
```



```
plot(tuberculosis, type="b", pch=20 , ylab="Número de casos", xlab="Tempo")
```



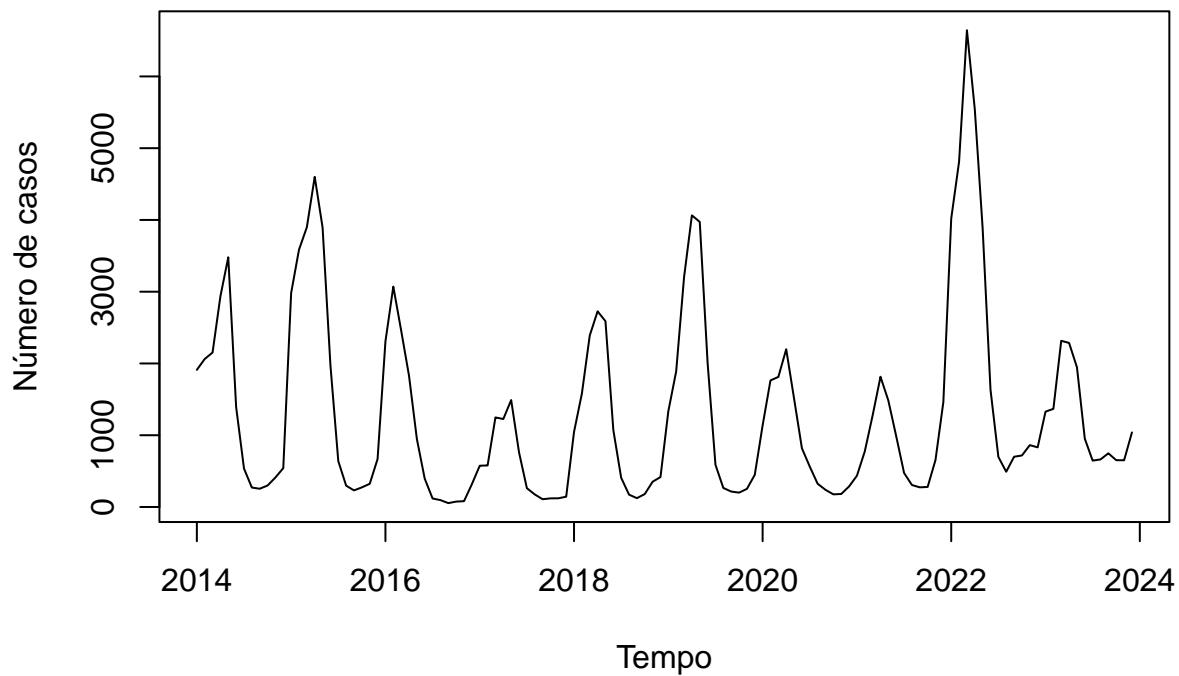
## Série temporal do número de casos de dengue

A seguir é apresentada a série temporal do número de casos de dengue no Estado de Goiás (UF da residência é Goiás) com exame sorológico (IgM) positivo, que compreende o número de casos mensal de janeiro de 2014 a dezembro de 2023. Os valores foram obtidos a partir do Sistema de Informação de Agravos de Notificação - Sinan Net - acessado em 20/09/2024, no link aqui.

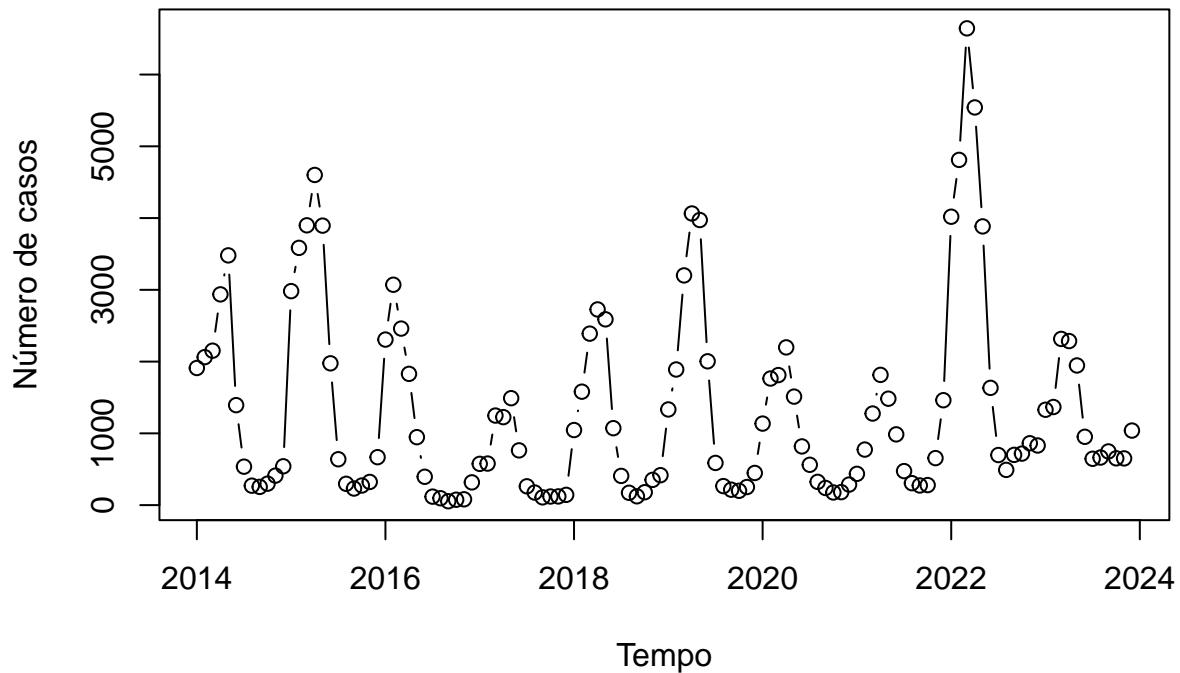
```
# serie do numero de casos mensal
serie_dengue <- c(1911, 2062, 2152, 2937, 3480, 1393, 536, 271, 254, 299, 411, 541,
2982, 3585, 3899, 4601, 3895, 1976, 639, 297, 231, 273, 323, 668,
2307, 3071, 2460, 1829, 947, 395, 118, 96, 54, 75, 81, 317,
575, 579, 1248, 1226, 1490, 763, 262, 175, 108, 120, 121, 143,
1046, 1581, 2390, 2727, 2589, 1072, 407, 172, 123, 180, 352, 417,
1333, 1890, 3201, 4064, 3973, 2004, 588, 265, 215, 200, 252, 448,
1136, 1764, 1812, 2198, 1512, 818, 562, 324, 239, 176, 181, 286,
436, 774, 1276, 1815, 1483, 986, 474, 306, 274, 279, 655, 1462,
4019, 4811, 6644, 5542, 3884, 1634, 698, 491, 700, 718, 862, 831,
1329, 1367, 2316, 2286, 1946, 952, 646, 663, 748, 652, 651, 1039)

dengue <- ts(serie_dengue, start= c(2014, 1), frequency = 12)

plot(dengue, ylab="Número de casos", xlab="Tempo")
```



```
plot(dengue, type="b", ylab="Número de casos", xlab="Tempo")
```



## Série temporal do número de queimadas no bioma cerrado

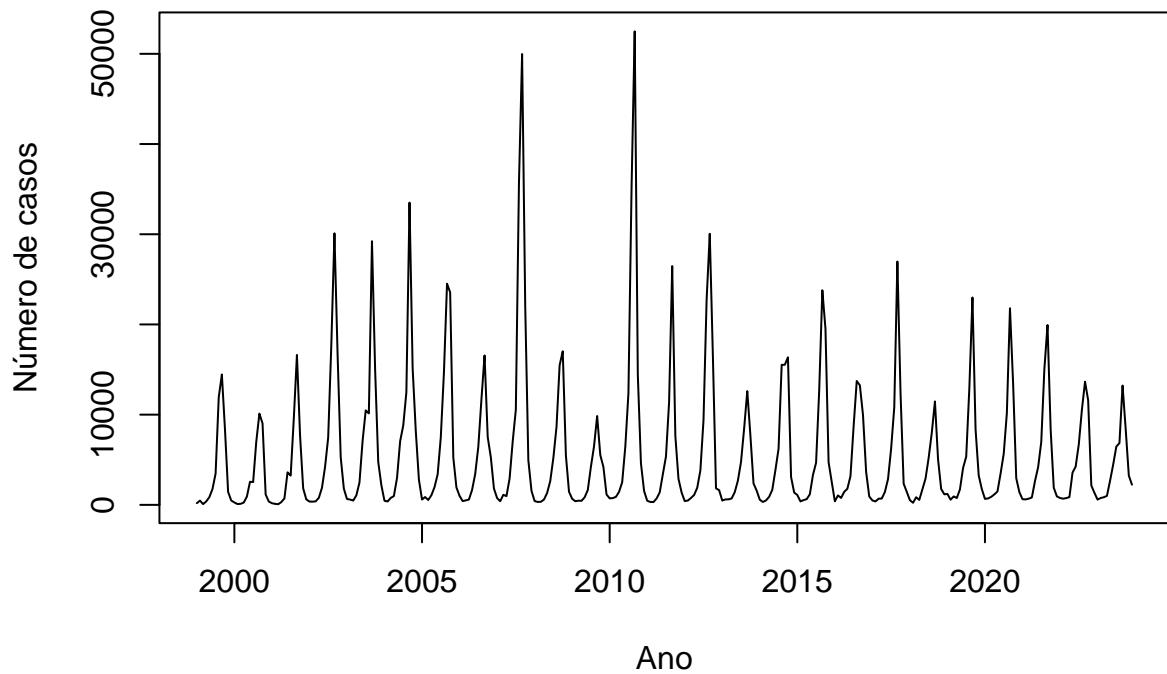
A série temporal em estudo é do número mensal de casos de queimadas no bioma cerrado no período de janeiro de 1999 até dezembro de 2023. Os valores foram obtidos no site do INPE, para acessar click aqui.

```
# numero mensal de casos de queimada no bioma cerrado
```

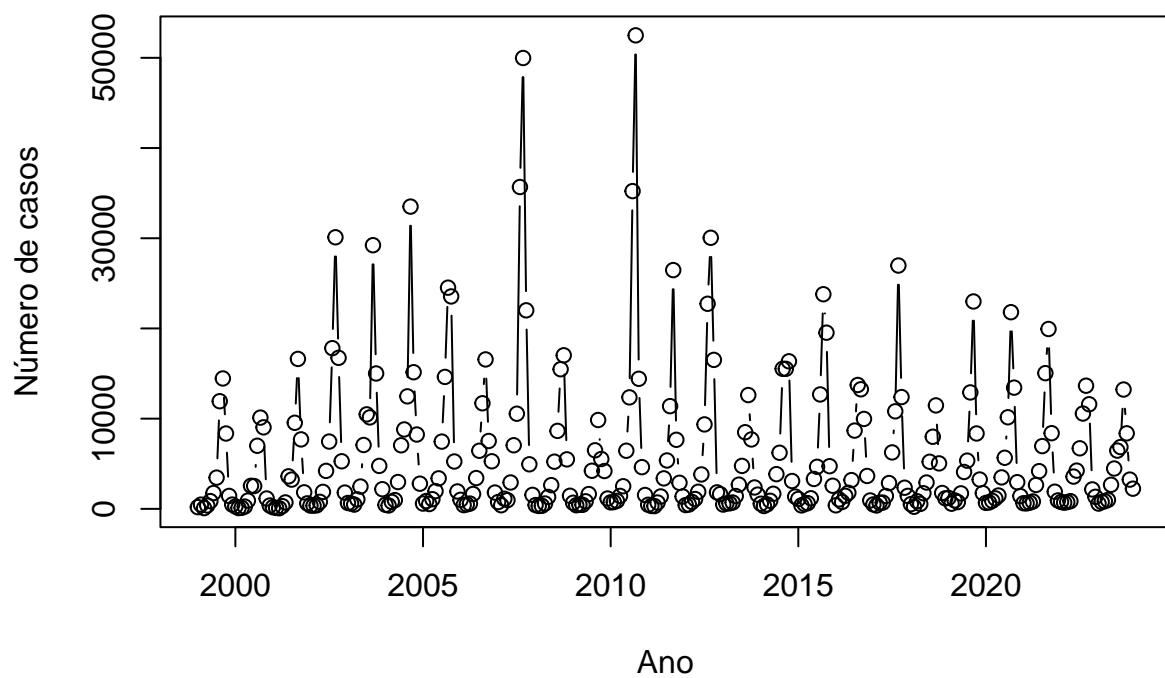
```
serie_queimada <-  
c(188, 469, 82, 382, 856, 1757, 3465, 11933, 14455, 8354, 1434, 475,  
266, 89, 114, 253, 914, 2556, 2530, 6995, 10117, 9038, 1149, 372,  
182, 109, 69, 304, 713, 3606, 3242, 9541, 16623, 7702, 1834, 609,  
368, 355, 391, 777, 1884, 4210, 7462, 17815, 30105, 16744, 5273, 1802,  
654, 587, 472, 1052, 2475, 7079, 10477, 10140, 29227, 15008, 4763, 2179,  
452, 376, 773, 970, 2975, 7047, 8804, 12473, 33509, 15148, 8243, 2792,  
587, 869, 536, 1071, 1968, 3383, 7454, 14627, 24521, 23550, 5251, 1930,  
1001, 413, 498, 584, 1650, 3415, 6442, 11705, 16566, 7524, 5274, 1808,  
755, 402, 1118, 964, 2924, 7051, 10548, 35678, 49980, 22008, 4939, 1551,  
421, 321, 339, 590, 1320, 2632, 5229, 8643, 15477, 17024, 5486, 1452,  
676, 403, 461, 448, 836, 1641, 4224, 6492, 9851, 5526, 4195, 1146,  
717, 751, 883, 1438, 2508, 6443, 12359, 35226, 52491, 14419, 4623, 1536,  
456, 321, 308, 703, 1376, 3378, 5366, 11387, 26468, 7656, 2889, 1374,  
421, 478, 764, 1092, 1896, 3817, 9362, 22737, 30053, 16515, 1830, 1635,  
475, 604, 613, 706, 1418, 2684, 4761, 8496, 12615, 7696, 2370, 1579,  
567, 315, 497, 897, 1673, 3849, 6220, 15525, 15523, 16357, 3085, 1363,  
1096, 383, 528, 634, 1174, 3313, 4662, 12684, 23795, 19531, 4731, 2563,  
392, 1048, 767, 1449, 1782, 3187, 8675, 13730, 13256, 9968, 3647, 932,  
491, 376, 687, 696, 1428, 2858, 6258, 10815, 26975, 12393, 2345, 1440,  
521, 235, 842, 549, 1729, 2922, 5220, 7992, 11467, 5041, 1763, 1168,  
1213, 574, 936, 753, 1719, 4088, 5346, 12906, 22989, 8356, 3251, 1743,  
662, 709, 890, 1158, 1481, 3487, 5663, 10155, 21802, 13440, 2957, 1415,  
625, 607, 686, 820, 2649, 4181, 6955, 15043, 19939, 8389, 1901, 933,  
763, 685, 751, 853, 3578, 4239, 6713, 10567, 13651, 11594, 2154, 1337,  
585, 759, 845, 993, 2668, 4472, 6459, 6850, 13230, 8371, 3242, 2239)
```

```
queimada <- ts(serie_queimada, start= c(1999, 1), frequency = 12)
```

```
plot(queimada, ylab="Número de casos", xlab="Ano")
```



```
plot(queimada, type="b", ylab="Número de casos", xlab="Ano")
```



## Série temporal do número de nascidos vivos em Goiás

A série temporal em estudo é do número mensal de nascidos vivos em Goiás no período de janeiro de 2011 até dezembro de 2023. Os valores foram obtidos no site do DATASUS, para acessar click aqui. Os dados foram obtidos em 20/09/2024.

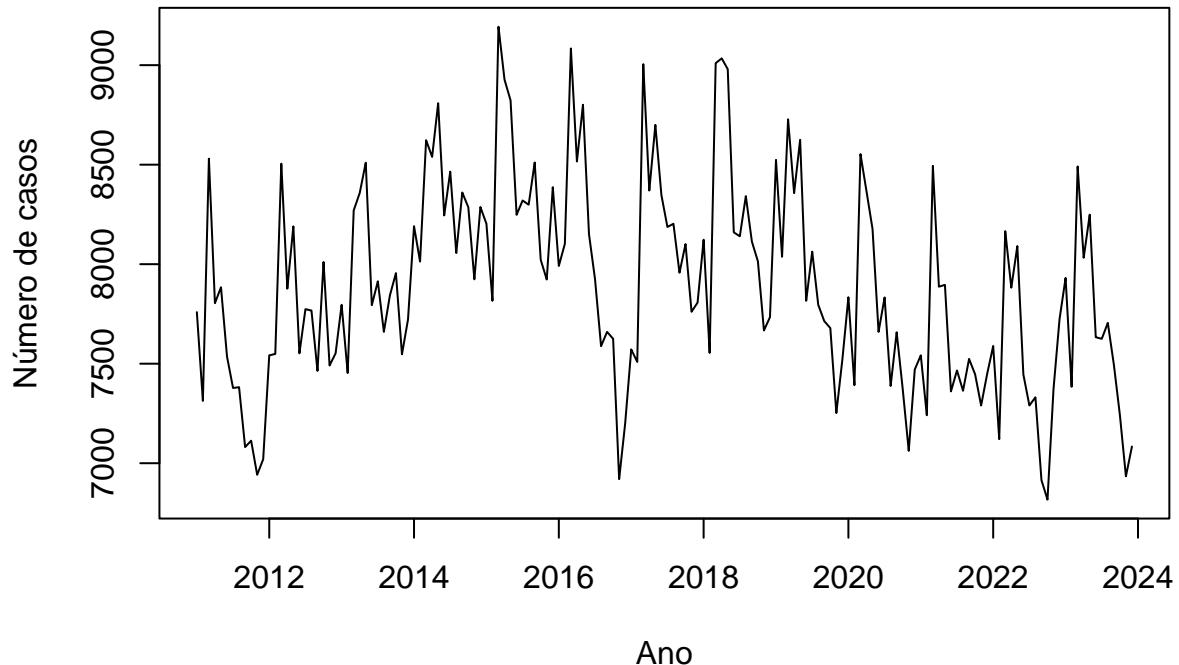
```
#serie temporal do numero mensal de nascidos vivos em Goias
```

```
serie_nascido <- c(7759, 7313, 8530, 7804, 7884, 7536, 7378, 7382, 7081, 7113, 6942, 7020,
  7542, 7549, 8505, 7877, 8190, 7552, 7774, 7767, 7464, 8011, 7491, 7552,
  7796, 7453, 8270, 8358, 8510, 7794, 7914, 7660, 7843, 7955, 7547, 7722,
  8191, 8013, 8623, 8539, 8809, 8245, 8465, 8056, 8360, 8286, 7924, 8287,
  8203, 7816, 9193, 8927, 8823, 8248, 8320, 8299, 8511, 8022, 7923, 8387,
  7991, 8102, 9084, 8516, 8801, 8151, 7925, 7588, 7660, 7625, 6920, 7200,
  7572, 7509, 9005, 8370, 8700, 8348, 8187, 8203, 7957, 8101, 7761, 7807,
  8123, 7554, 9010, 9034, 8980, 8160, 8140, 8342, 8114, 8014, 7667, 7734,
  8524, 8037, 8728, 8357, 8625, 7816, 8063, 7796, 7714, 7679, 7252, 7521,
  7834, 7392, 8553, 8366, 8177, 7660, 7833, 7388, 7658, 7374, 7062, 7471,
  7542, 7241, 8494, 7887, 7896, 7361, 7466, 7364, 7524, 7447, 7290, 7449,
  7589, 7121, 8166, 7882, 8091, 7446, 7290, 7331, 6915, 6817, 7373, 7724,
  7931, 7384, 8491, 8032, 8249, 7633, 7625, 7705, 7495, 7242, 6934, 7084)
```

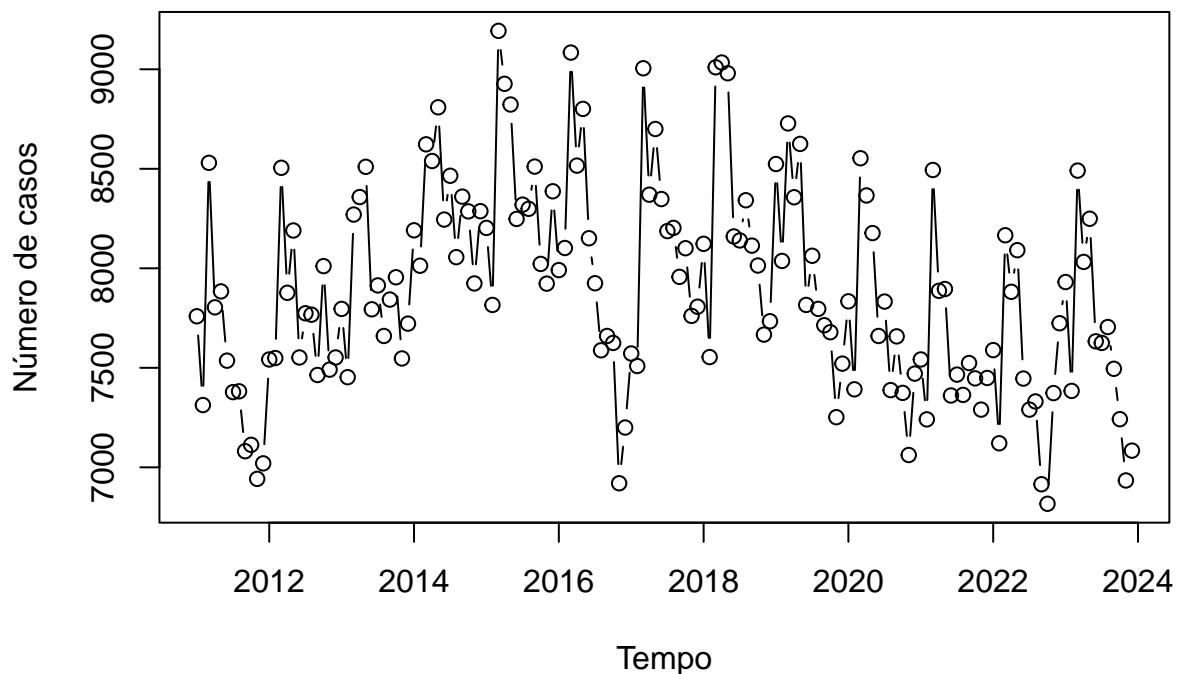
  

```
nascido <- ts(serie_nascido, start= c(2011, 1), frequency = 12)

plot(nascido, ylab="Número de casos", xlab="Ano")
```



```
plot(nascido, type="b", ylab="Número de casos", xlab="Tempo")
```



## Série temporal do IPCA mensal

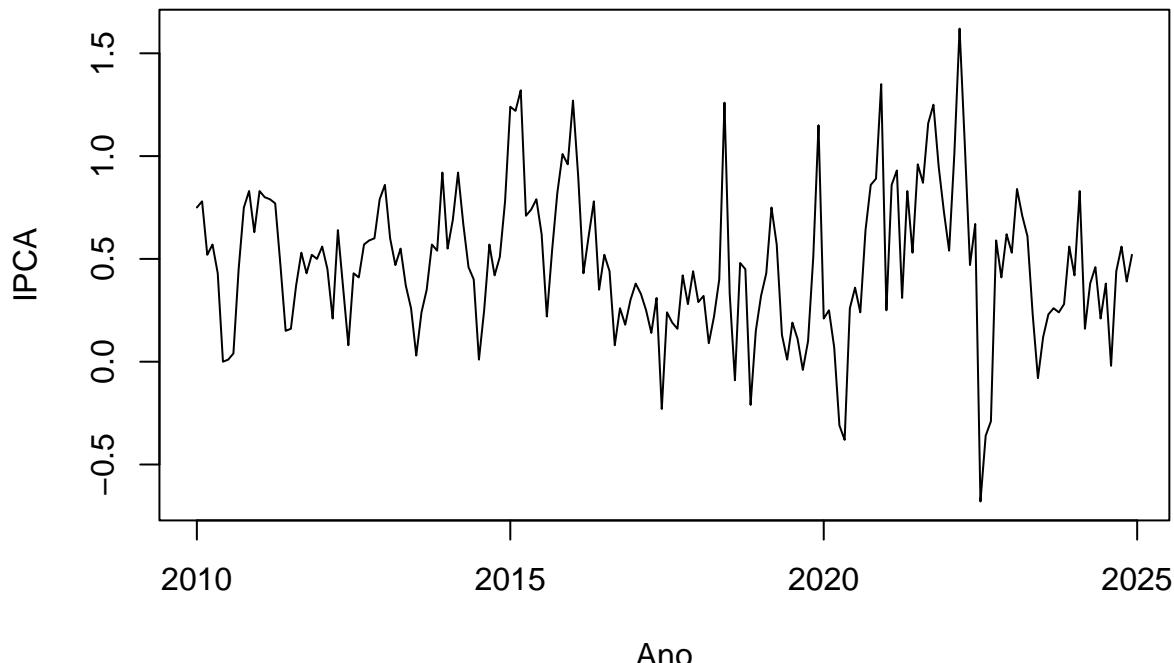
A série temporal a seguir é do IPCA (índice nacional de preços ao consumidor amplo) do período de julho de 2010 até dezembro de 2024. Os valores foram obtidos no site do IBGE, para acessar click aqui. Os dados foram acessados em 01/05/2024.

```
# serie temporal do IPCA
```

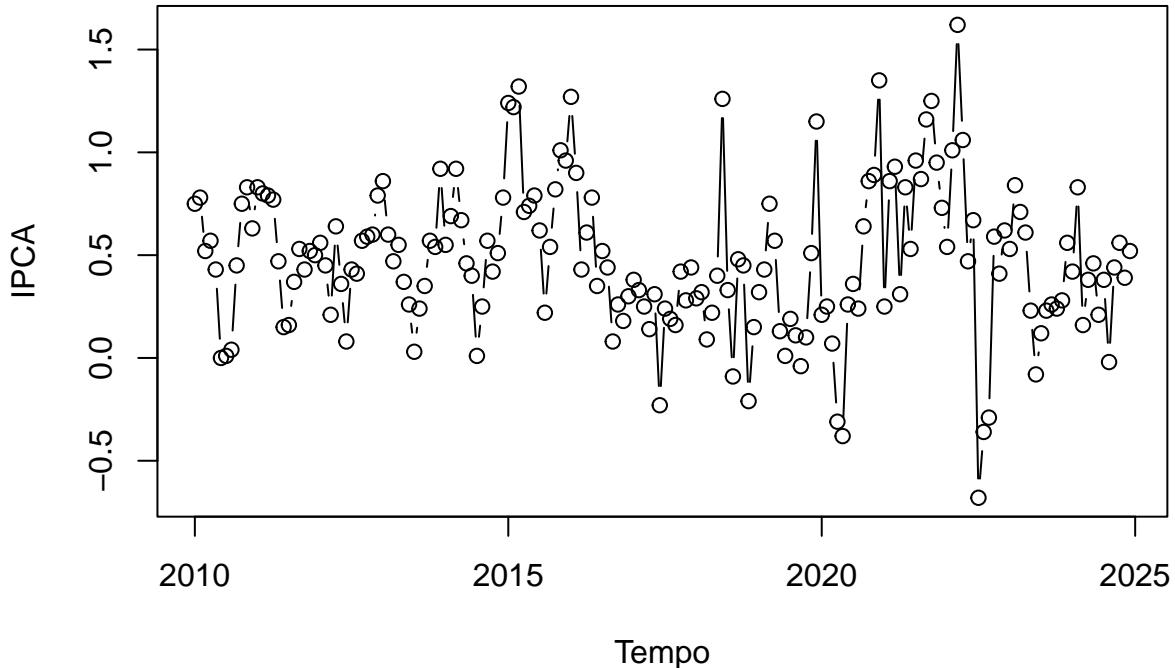
```
serie_ipca <- c(0.75, 0.78, 0.52, 0.57, 0.43, 0.00, 0.01, 0.04, 0.45, 0.75, 0.83, 0.63,
 0.83, 0.80, 0.79, 0.77, 0.47, 0.15, 0.16, 0.37, 0.53, 0.43, 0.52, 0.50,
 0.56, 0.45, 0.21, 0.64, 0.36, 0.08, 0.43, 0.41, 0.57, 0.59, 0.60, 0.79,
 0.86, 0.60, 0.47, 0.55, 0.37, 0.26, 0.03, 0.24, 0.35, 0.57, 0.54, 0.92,
 0.55, 0.69, 0.92, 0.67, 0.46, 0.40, 0.01, 0.25, 0.57, 0.42, 0.51, 0.78,
 1.24, 1.22, 1.32, 0.71, 0.74, 0.79, 0.62, 0.22, 0.54, 0.82, 1.01, 0.96,
 1.27, 0.90, 0.43, 0.61, 0.78, 0.35, 0.52, 0.44, 0.08, 0.26, 0.18, 0.30,
 0.38, 0.33, 0.25, 0.14, 0.31,-0.23, 0.24, 0.19, 0.16, 0.42, 0.28, 0.44,
 0.29, 0.32, 0.09, 0.22, 0.40, 1.26, 0.33,-0.09, 0.48, 0.45,-0.21, 0.15,
 0.32, 0.43, 0.75, 0.57, 0.13, 0.01, 0.19, 0.11,-0.04, 0.10, 0.51, 1.15,
 0.21, 0.25, 0.07,-0.31,-0.38, 0.26, 0.36, 0.24, 0.64, 0.86, 0.89, 1.35,
 0.25, 0.86, 0.93, 0.31, 0.83, 0.53, 0.96, 0.87, 1.16, 1.25, 0.95, 0.73,
 0.54, 1.01, 1.62, 1.06, 0.47, 0.67,-0.68,-0.36,-0.29, 0.59, 0.41, 0.62,
 0.53, 0.84, 0.71, 0.61, 0.23,-0.08, 0.12, 0.23, 0.26, 0.24, 0.28, 0.56,
 0.42, 0.83, 0.16, 0.38, 0.46, 0.21, 0.38,-0.02, 0.44, 0.56, 0.39, 0.52)
```

```
ipca <- ts(serie_ipca, start= c(2010, 1), frequency = 12)
```

```
plot(ipca, ylab="IPCA", xlab="Ano")
```



```
plot(ipca, type="b", ylab="IPCA", xlab="Tempo")
```



## Cálculo de diferenças

Exemplo do cálculo de diferenças de uma série temporal.

```
z <- c(10, 9, 11, 8, 10)
diff(z)
```

```
## [1] -1  2 -3  2
diff(z, differences = 2)
```

```
## [1]  3 -5  5
```

Primeira diferença da série do número de nascidos vivo em Goiás.

```
# nascidos vivos
```

```
diff_nascido <- diff(nascido)
summary(diff_nascido)
```

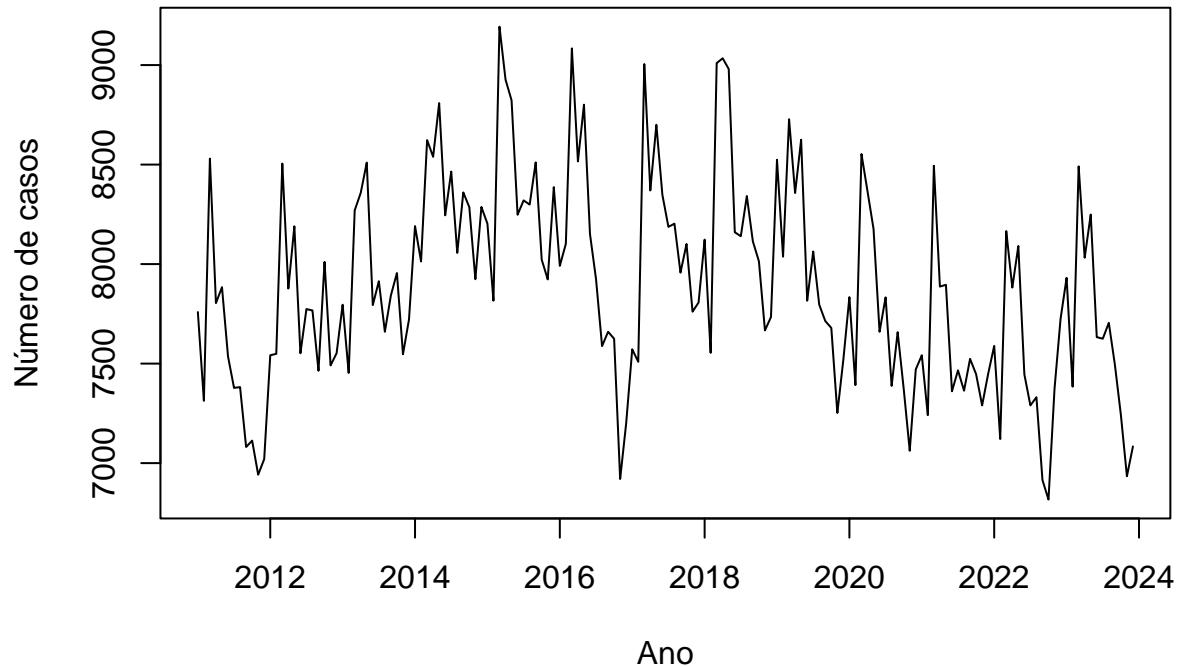
```
##      Min.    1st Qu.     Median      Mean    3rd Qu.      Max.
```

```
## -820.000 -341.500  -35.000   -4.355  214.500 1496.000
```

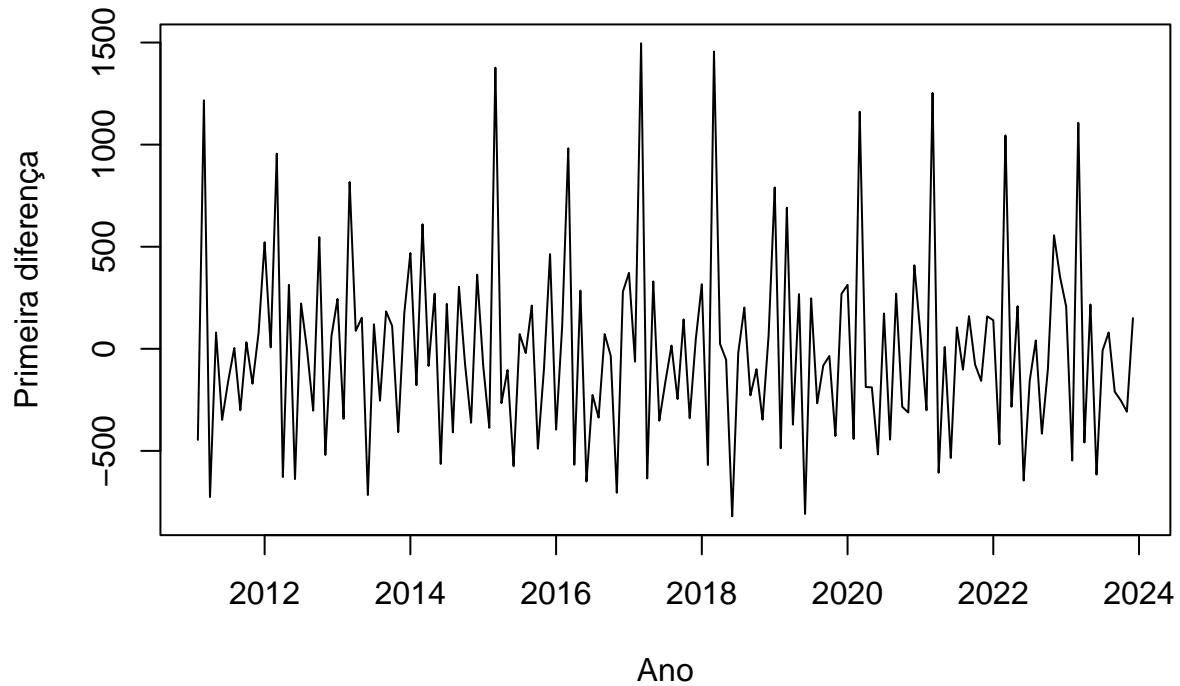
```
length(nascido)
```

```
## [1] 156
```

```
length(diff_nascido)  
## [1] 155  
#par(mfrow = c(2, 1))  
plot(nascido, ylab="Número de casos", xlab="Ano")
```



```
plot(diff_nascido, ylab="Primeira diferença", xlab="Ano")
```

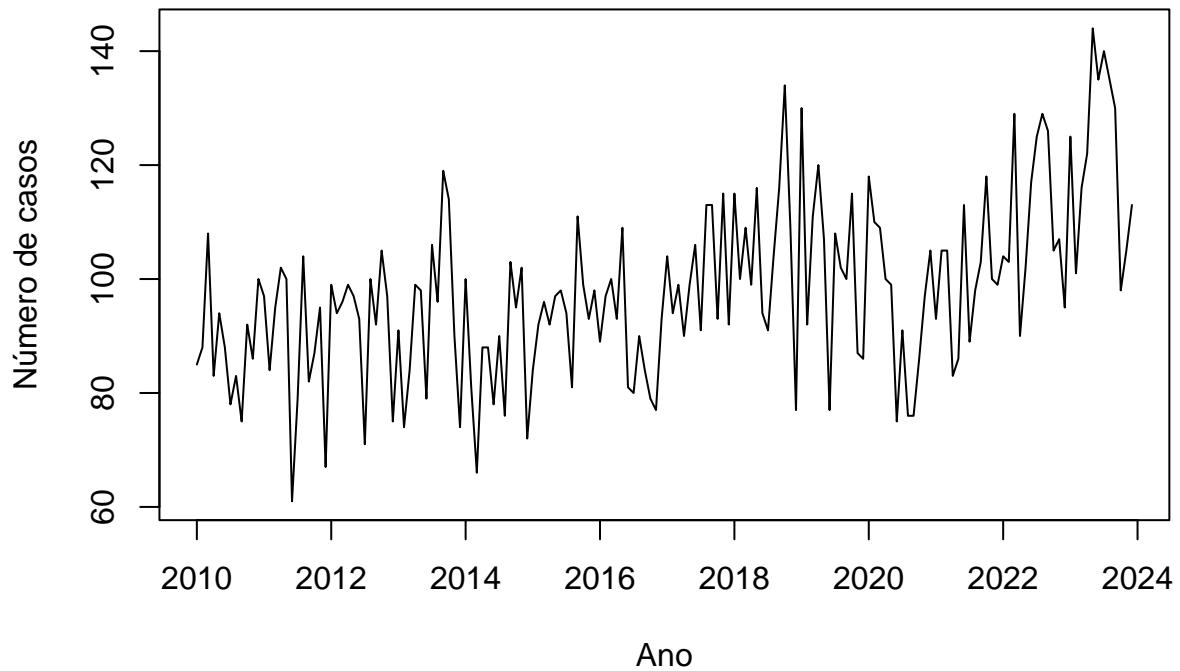


```
# casos de tuberculose

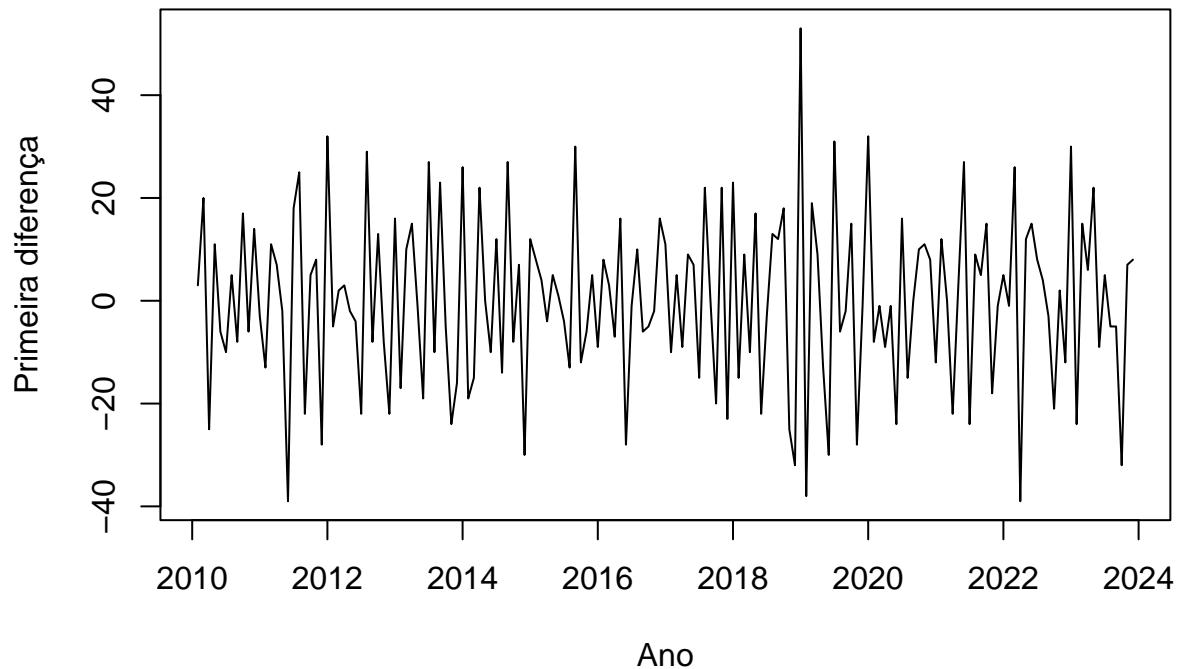
diff_tuberculose <- diff(tuberculose)
summary(diff_tuberculose)

##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -39.0000 -10.0000    0.0000   0.1677  11.5000  53.0000

#par(mfrow = c(2,1))
plot(tuberculose, ylab="Número de casos", xlab="Ano")
```



```
plot(diff_tuberculosis, ylab="Primeira diferença", xlab="Ano")
```

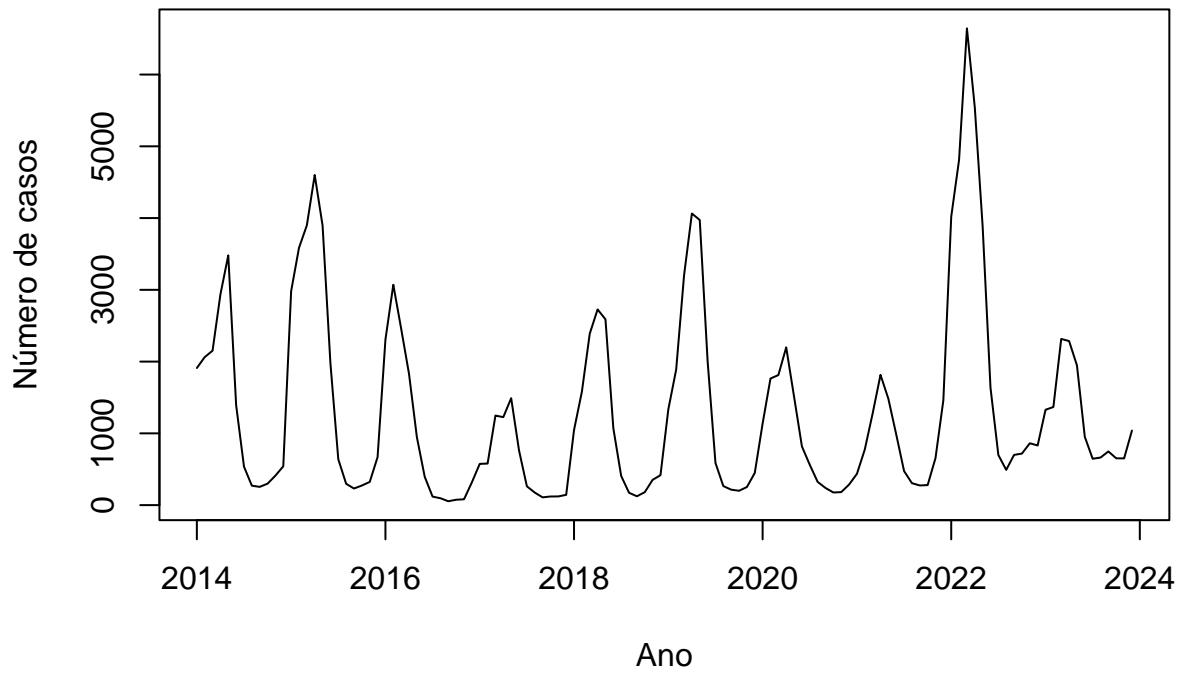


```
# casos de dengue

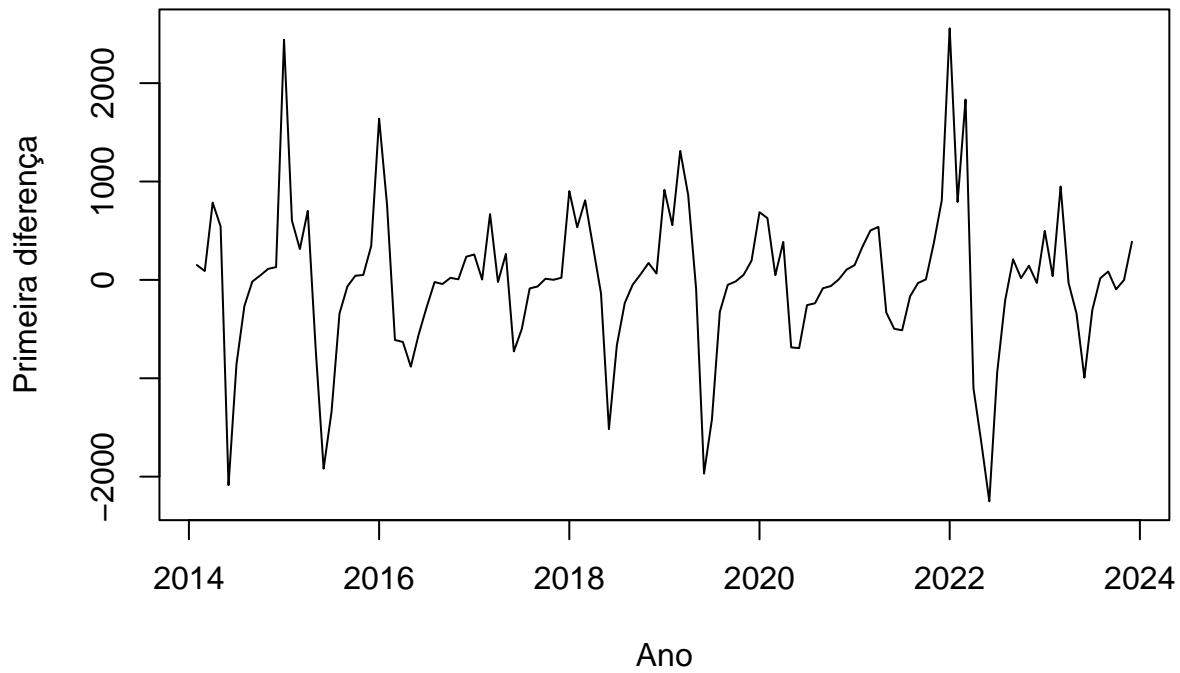
diff_dengue <- diff(dengue)
summary(diff_dengue)

##      Min.   1st Qu.    Median     Mean   3rd Qu.   Max.
## -2250.000 -271.000     6.000   -7.328  337.500 2557.000

#par(mfrow = c(2,1))
plot(dengue, ylab="Número de casos", xlab="Ano")
```



```
plot(diff_dengue, ylab="Primeira diferença", xlab="Ano")
```



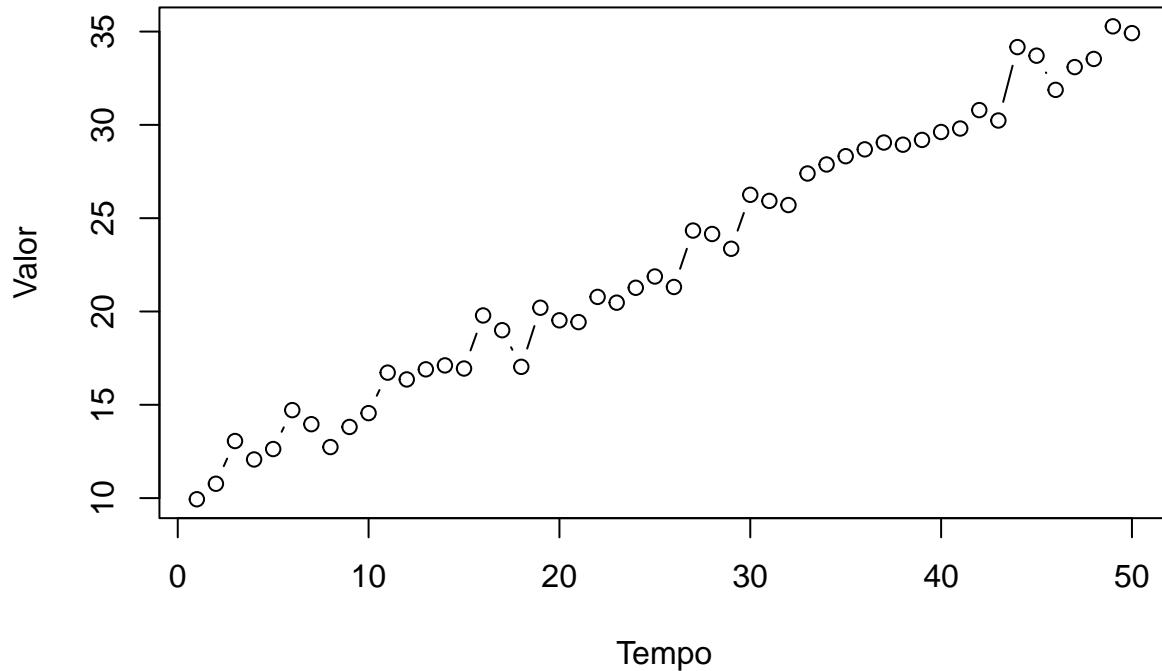
## Lidando com valores ausentes

```
# Pacotes necessários
library(zoo)

## Warning: pacote 'zoo' foi compilado no R versão 4.4.3
##
## Anexando pacote: 'zoo'
##
## Os seguintes objetos são mascarados por 'package:base':
## 
##     as.Date, as.Date.numeric

# Simulando uma série temporal com tendência
set.seed(123)
tempo <- 1:50
serie <- ts(10 + 0.5 * tempo + rnorm(50, 0, 1))

plot(serie, type="b", xlab="Tempo", ylab="Valor")
```



```
# inserindo NAs em posicoes especificas
serie_na <- serie
serie_na[c(10, 15, 25, 35)] <- NA
serie_na

## Time Series:
## Start = 1
## End = 50
## Frequency = 1
## [1] 9.939524 10.769823 13.058708 12.070508 12.629288 14.715065 13.960916
## [8] 12.734939 13.813147      NA 16.724082 16.359814 16.900771 17.110683
## [15]        NA 19.786913 18.997850 17.033383 20.201356 19.527209 19.432176
## [22] 20.782025 20.473996 21.271109      NA 21.313307 24.337787 24.153373
## [29] 23.361863 26.253815 25.926464 25.704929 27.395126 27.878133      NA
## [36] 28.688640 29.053918 28.938088 29.194037 29.619529 29.805293 30.792083
## [43] 30.234604 34.168956 33.707962 31.876891 33.097115 33.533345 35.279965
## [50] 34.916631

# forward fill
# funcao na.locf substitui cada NA pelo nao-NA mais recente anterior a ele
serie_locf <- na.locf(serie_na)
serie_locf

## Time Series:
## Start = 1
## End = 50
## Frequency = 1
```

```

## [1] 9.939524 10.769823 13.058708 12.070508 12.629288 14.715065 13.960916
## [8] 12.734939 13.813147 13.813147 16.724082 16.359814 16.900771 17.110683
## [15] 17.110683 19.786913 18.997850 17.033383 20.201356 19.527209 19.432176
## [22] 20.782025 20.473996 21.271109 21.271109 21.313307 24.337787 24.153373
## [29] 23.361863 26.253815 25.926464 25.704929 27.395126 27.878133 27.878133
## [36] 28.688640 29.053918 28.938088 29.194037 29.619529 29.805293 30.792083
## [43] 30.234604 34.168956 33.707962 31.876891 33.097115 33.533345 35.279965
## [50] 34.916631

# media movel de 2 valores
#
k <- 3 # para pegar dois valores o valor do k tem que ser 3
# janela = (Valor, NA, Valor)
media_movel <- rollapply(serie_na, width = k,
                         FUN = function(y) mean(y, na.rm = TRUE),
                         fill = NA, align = "center")
media_movel

## Time Series:
## Start = 1
## End = 50
## Frequency = 1
## [1] NA 11.25602 11.96635 12.58617 13.13829 13.76842 13.80364 13.50300
## [9] 13.27404 15.26861 16.54195 16.66156 16.79042 17.00573 18.44880 19.39238
## [17] 18.60605 18.74420 18.92065 19.72025 19.91380 20.22940 20.84238 20.87255
## [25] 21.29221 22.82555 23.26816 23.95101 24.58968 25.18071 25.96174 26.34217
## [33] 26.99273 27.63663 28.28339 28.87128 28.89355 29.06201 29.25055 29.53962
## [41] 30.07230 30.27733 31.73188 32.70384 33.25127 32.89399 32.83578 33.97014
## [49] 34.57665 NA

serie_media <- ifelse(is.na(serie_na), media_movel, serie_na)

# cbind(serie_na, serie_media, serie_na - serie_media)

# media movel de 2 valores antes do NA
k <- 3 # para pegar dois valores o valor do k tem que ser 3
# janela = (Valor, Valor, NA)
media_movel2 <- rollapply(serie_na, width = k,
                           FUN = function(y) mean(y, na.rm = TRUE), fill = NA, align = "right")

media_movel2

## Time Series:
## Start = 1
## End = 50
## Frequency = 1
## [1] NA 11.25602 11.96635 12.58617 13.13829 13.76842 13.80364
## [9] 13.50300 13.27404 15.26861 16.54195 16.66156 16.79042 17.00573 18.44880
## [17] 19.39238 18.60605 18.74420 18.92065 19.72025 19.91380 20.22940 20.84238
## [25] 20.87255 21.29221 22.82555 23.26816 23.95101 24.58968 25.18071 25.96174
## [33] 26.34217 26.99273 27.63663 28.28339 28.87128 28.89355 29.06201 29.25055
## [41] 29.53962 30.07230 30.27733 31.73188 32.70384 33.25127 32.89399 32.83578
## [49] 33.97014 34.57665

```

```

serie_media2 <- ifelse(is.na(serie_na), media_movel2, serie_na)

# cbind(serie_na, serie_media2, serie_na - serie_media2)

# interpolação linear
# função an.approx - os NAs são preenchidos com valores em linha reta
# entre o valor anterior e o próximo
# exemplo da função
# x <- c(1, 2, NA, NA, 5)
# pegando os valores 2 e 5, nas posições 2 e 5,
# a reta é  $y = x$ 
# na.approx(x)

serie_interpolacao <- na.approx(serie_na)

#round(cbind(serie, serie_na, serie_locf, serie_media, serie_media2, serie_interpolacao), 2)

```

## Função de autocorrelação - FAC

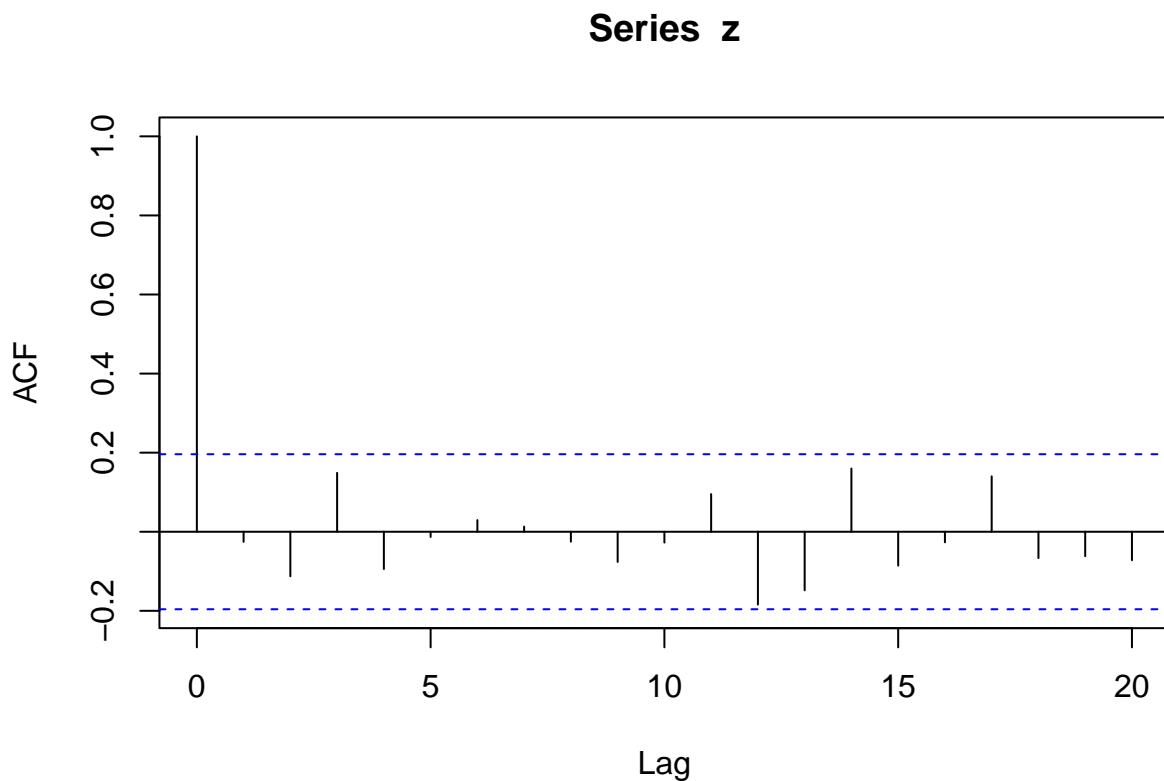
Inicialmente, vamos simular um ruído branco e na sequência obter o gráfico da FAC. O intuito aqui é comparar o resultado teórico com o prático.

```

set.seed(123)
# vamos gerar n valores de uma v.a. normal com média 0 e variância 10
n <- 100
z <- rnorm(n, 0, 1)

acf(z)

```



Detalhes importantes:

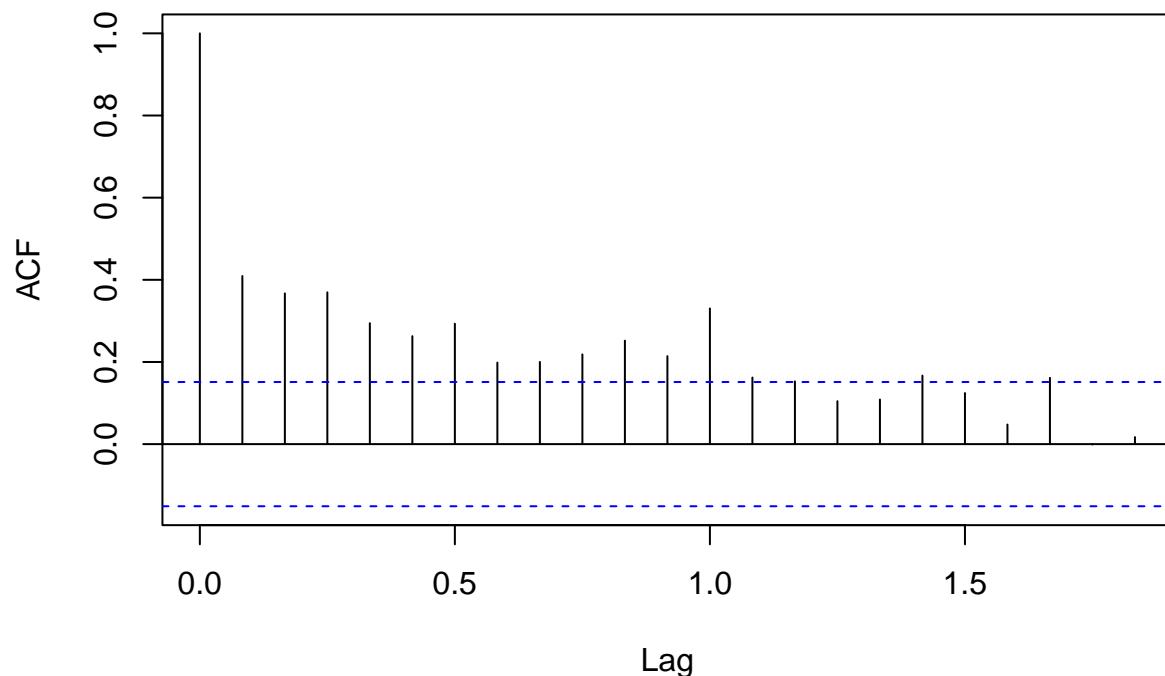
- Altura das barras: indica a força da autocorrelação em cada defasagem.
- Linhas tracejadas: valores fora do limite indicam autocorrelação estatisticamente diferente de zero.
- Padrão das barras: queda lenta sugere não estacionariedade, ou presença de tendência/ciclo.

Comparando com o gráfico teórico, vemos que ambas as FACs no lag zero assumem valor 1, enquanto que para os demais lags todos podem ser considerados zeros na FAC estimada, pois os valores estão dentro dos limites, sendo assim, igual ao teórico.

### FAC das séries em estudo

```
# FAC para a serie de tuberculose
acf(tuberculose, main="ACF da série temporal do número de casos de tuberculose")
```

## ACF da série temporal do número de casos de tuberculose

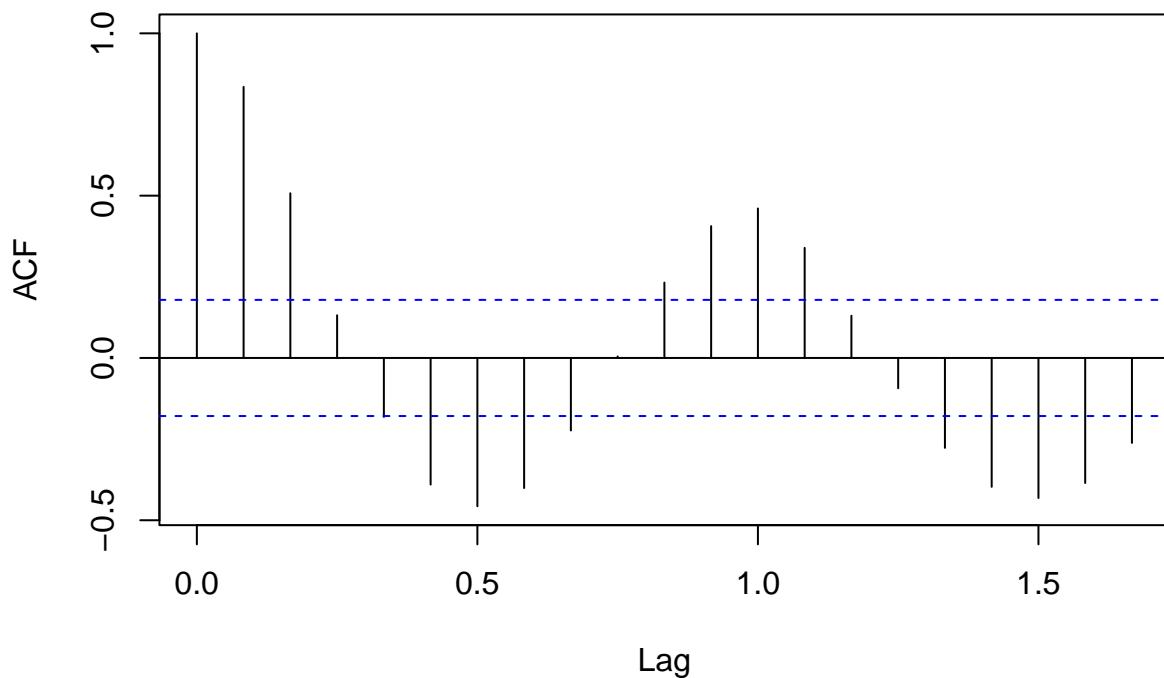


```
#acf(tuberculose, main="ACF da série temporal do número de casos de tuberculose", lag.max=36)

#acf(serie_tuberculose, main="ACF da série temporal do número de casos de tuberculose")
# note a diferença nos lags

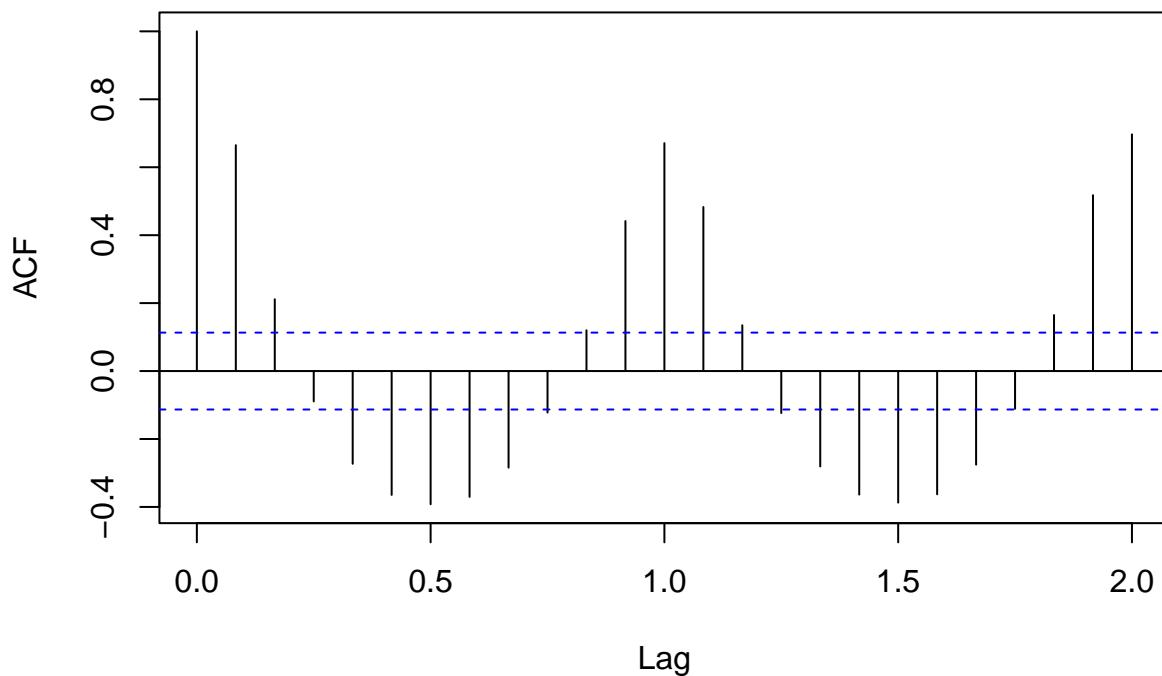
# acf para a série de dengue
acf(dengue, main="ACF da série temporal do número de casos de dengue")
```

## ACF da série temporal do número de casos de dengue



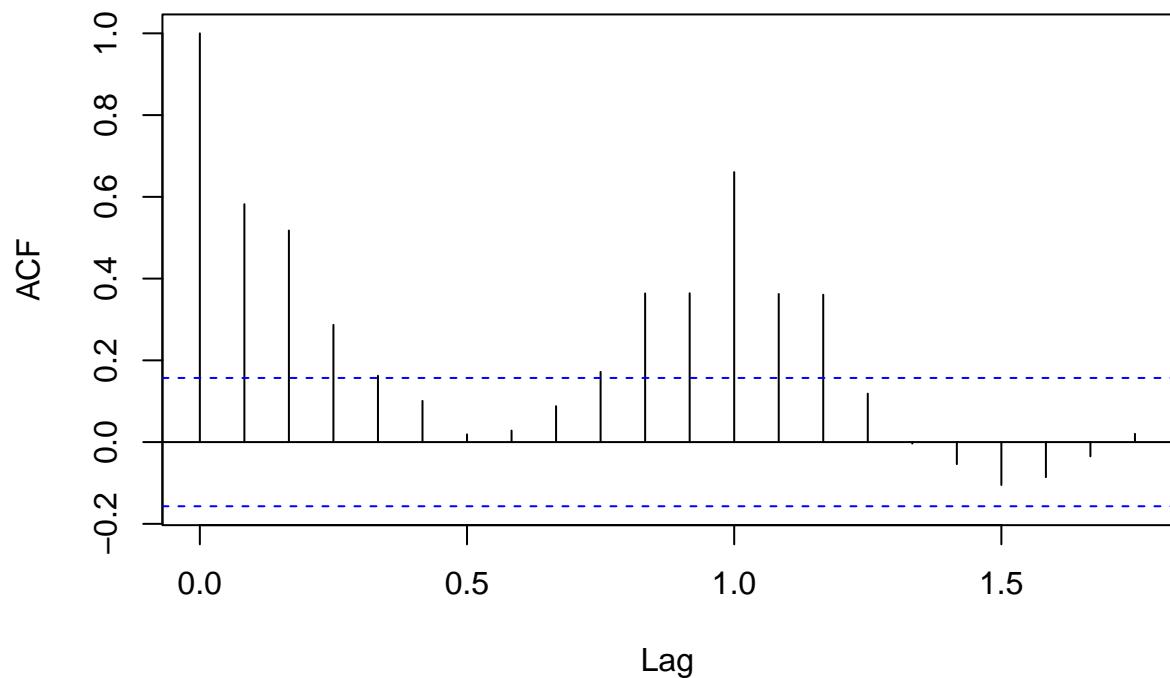
```
#acf(dengue, main="ACF da série temporal do número de casos de dengue", lag.max = 36)  
# acf para a serie de queimada  
acf(queimada, main="ACF da série temporal do número de casos de queimada")
```

## ACF da série temporal do número de casos de queimada



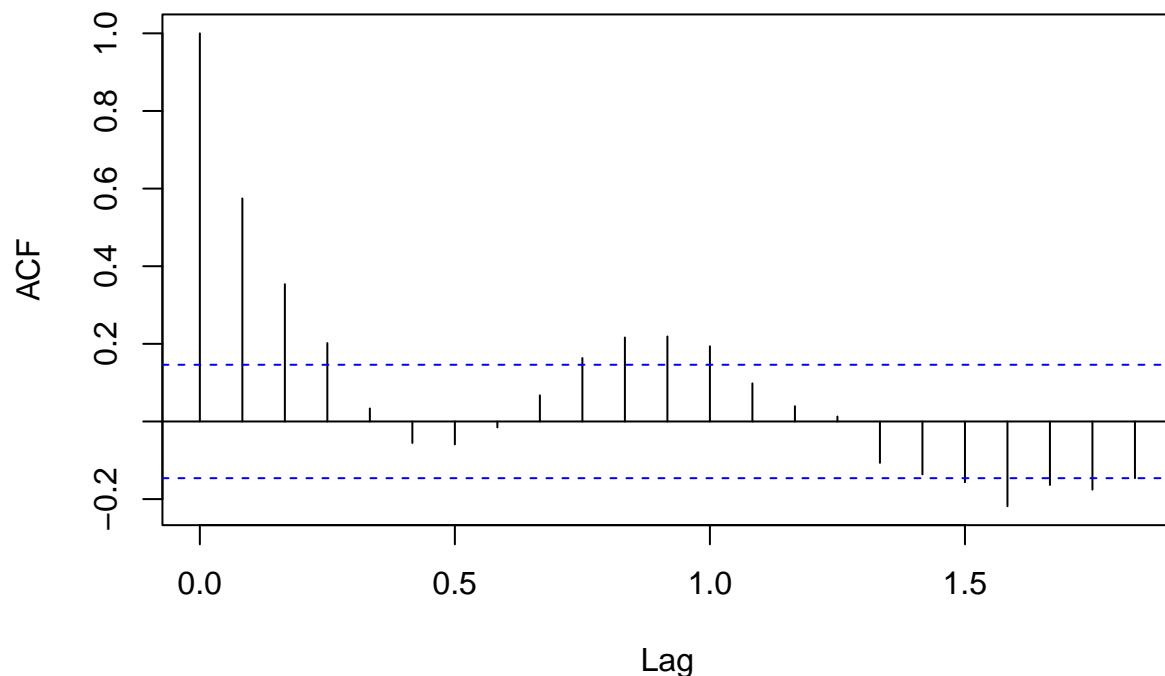
```
#acf(queimada, main="ACF da série temporal do número de casos de queimada", lag.max=36)  
# acf para a serie nascidos vivo  
acf(nascido, main="ACF da série temporal do número de casos de nascidos vivos")
```

## ACF da série temporal do número de casos de nascidos vivos



```
#acf(nascido, main="ACF da série temporal do número de casos de nascidos vivos", lag.max=36)  
# acf para a serie ipca  
acf(ipca, main="ACF da série temporal do IPCA")
```

## ACF da série temporal do IPCA



```
#acf(ipca, main="ACF da série temporal do IPCA", lag.max=36)
```

Observação: a FAC de uma função periódica tem a mesma periodicidade do processo original.