



**Cursos Integrados
em Vigilância em Saúde**

Curso —

Construção de diagramas de controle na vigilância em saúde

UNIVERSIDADE FEDERAL DE SANTA CATARINA

Reitor Irineu Manoel de Souza

Vice-Reitora Joana Célia dos Passos

Pró-Reitora de Pós-graduação Werner Kraus

Pró-Reitor de Pesquisa e Inovação Jacques Mick

Pró-Reitor de Extensão Olga Regina Zigelli Garcia

CENTRO DE CIÊNCIAS DA SAÚDE

Diretor Fabrício de Souza Neves

Vice-Diretora Ricardo de Souza Magini

DEPARTAMENTO DE SAÚDE PÚBLICA

Chefe do Departamento Rodrigo Otávio Moretti Pires

Subchefe do Departamento Sheila Rúbia Lindner

Coordenadora do Curso Alexandra Crispim Boing

INSTITUTO TODOS PELA SAÚDE (ITPS)

Diretor presidente Jorge Kalil (Professor titular da Faculdade de Medicina da Universidade de São Paulo; Diretor do Laboratório de Imunologia do Incor)

ASSOCIAÇÃO BRASILEIRA DE SAÚDE COLETIVA (ABRASCO)

Presidente Rosana Teresa Onocko Campos

EQUIPE DE PRODUÇÃO

Denis de Oliveira Rodrigues

Kamila de Oliveira Belo

Marcelo Eduardo Borges

Oswaldo Gonçalves Cruz

Alexandra Crispim Boing

Antonio Fernando Boing



Curso

Construção de diagramas de controle na vigilância em saúde

Dados Internacionais de Catalogação-na-Publicação (CIP)

C758 Construção de diagramas de controle na vigilância em saúde/ Oswaldo Gonçalves Cruz, Kamila de Oliveira Belo, Denis de Oliveira Rodrigues, Marcelo Eduardo Borges . Santa Catarina ; São Paulo ; Rio de Janeiro : UFSC ; ITPS ; Abrasco; 2022. 53p. (Cursos Integrados em Vigilância em Saúde).

Publicação Online
[10.52582/curso-analise-dados-vigilancia-modulo8](https://repositorio.ufsc.br/handle/10.52582/curso-analise-dados-vigilancia-modulo8)

1. Vigilância em saúde 2. Análise de dados I. Título



Sumário

Analisando epidemias e endemias	06
1. Introdução ao diagrama de controle	07
2. Construindo o diagrama de controle de dengue	10
2.1 Obtendo os dados.....	11
2.2 Montando o diagrama de controle.....	19
3. Construindo o diagrama de controle de Hepatite A.....	35

Analisando epidemias e endemias

Neste curso aprofundaremos as análises para a vigilância em saúde construindo **diagramas de controle**. Em epidemiologia este termo se refere ao conjunto de métodos quantitativos utilizados para estabelecer a faixa de *endemicidade* de um determinado agravo ou doença, permitindo avaliar se em um determinado período do tempo há uma *epidemia* ou há um comportamento *endêmico*. Em poucas palavras, construir um diagrama de controle é realizar uma avaliação gráfica da faixa de oscilação no número de casos esperados e definir se os valores apresentados no gráfico estão acima de uma ocorrência esperada, ou se indicam a probabilidade de epidemia ou surto.

Ao final deste curso, você será capaz de:

1. realizar análise exploratória para determinar anos epidêmicos e não epidêmicos;
2. calcular a média e o desvio padrão de semanas epidemiológicas (ou meses) para períodos não epidêmicos (padrão sazonal);
3. construir a visualização de diagramas de controle.

Atenção



Para seguir com este curso você deve conhecer as ferramentas básicas para uso da linguagem R e do RStudio. Lembre-se que você pode acessar a qualquer momento o curso “**Análise de dados para a vigilância em saúde – curso básico**” e o curso “**Visualização de dados de interesse para a Vigilância em Saúde**” obtendo os códigos desejados para a confecção do seu diagrama. Caso não tenha feito os cursos, sugerimos fortemente que se inscreva neles. Maiores informações em <https://www.abrasco.org.br/site/analise-de-dados-para-a-vigilancia-em-saude/>

1. Introdução ao diagrama de controle

Na vigilância em saúde, quando necessitamos definir quais são os limites esperados para uma doença ao longo do tempo, construímos um **diagrama de controle**. Trata-se de uma visualização gráfica que identifica a ocorrência de uma epidemia ou surto, ou seja, permite que visualizemos de forma rápida o aumento dos casos de uma doença conhecida para que as equipes de saúde possam aplicar medidas de controle precocemente.

Muitas vezes, no dia a dia, monitorar algumas doenças como a dengue é desafiador. Automatizar esta análise incluindo o diagrama de controle qualificará sua rotina de detecção precoce de epidemias. Para confeccionar um diagrama de controle existem diferentes métodos. O mais conhecido e aplicado atualmente é o que utiliza o cálculo da mediana por semana epidemiológica dos casos notificados em um período de dez anos.

Durante o intervalo de tempo que será escolhido para se estimar a faixa endêmica (em geral 10 anos), e sobre a qual colocaremos os valores do ano em que queremos saber se há uma epidemia, é importante que não tenha havido grandes alterações no número de casos. Ou seja, **os anos com registro de epidemias (epidêmicos) devem ser excluídos** desse período que nos informará o canal endêmico. Isto porque para o uso da metodologia escolhida neste curso os aumentos bruscos na análise podem enviesar os resultados gerando interpretações inadequadas.

A partir do número de casos ao longo dos dez anos, se faz a determinação do chamado **canal endêmico**, ou seja, o padrão regular da frequência dos casos em três componentes do diagrama de controle:

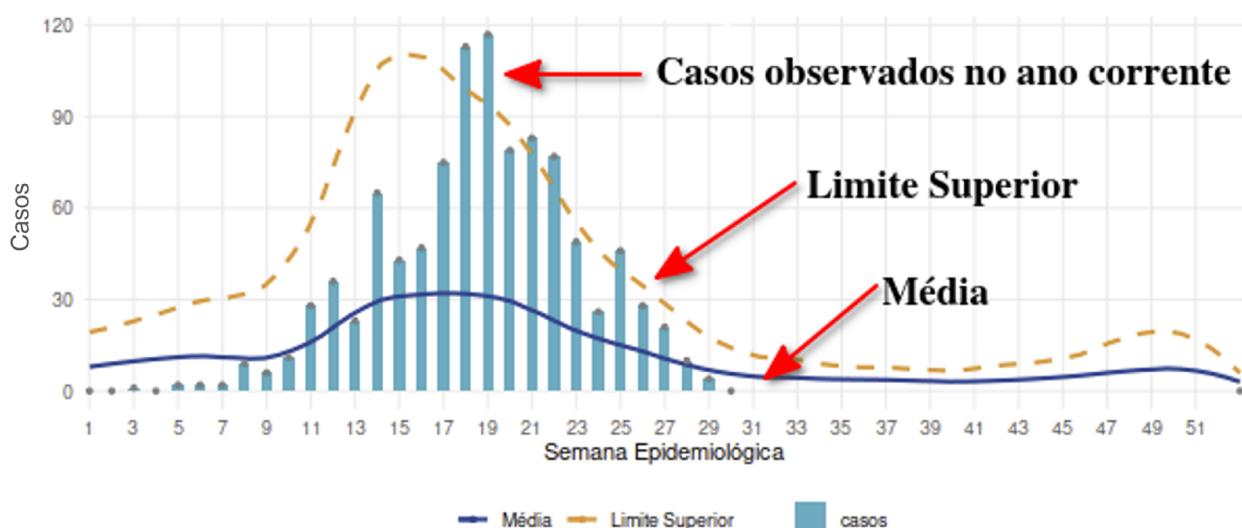
- 1. O limite superior:** limiar epidêmico, indica a frequência esperada máxima de casos em cada semana epidemiológica. É calculado com base no quartil 75 ou ± 2 desvios padrões, correspondendo à dispersão do número de casos nos últimos dez anos em cada semana.
- 2. A média de casos:** nível endêmico, representa a frequência média esperada de casos em cada semana epidemiológica, baseada na medida de tendência central da distribuição de casos nos últimos dez anos, em cada semana epidemiológica.
- 3. E a distribuição dos casos observados:** indica a frequência de casos em cada semana epidemiológica do ano que está sendo estudado. Pode ser utilizada neste componente o número absoluto de casos ou o cálculo da média de casos ou até mesmo a incidência de casos.

Dessa forma, representam-se esses componentes em um gráfico como na Figura 1. Nele os componentes 1 e 2 são representados em uma linha e o 3 é representado por barras. O *eixo x* do gráfico é a unidade temporal escolhida (a semana epidemiológica) e o *eixo y* a frequência dos dados estudados.

Fique tranquilo, veremos todas estas etapas com passo a passo neste curso!

Observe a seguir na Figura 1 a representação gráfica de um diagrama de casos de uma doença organizado por semana epidemiológica:

Figura 1: Exemplo de um diagrama de controle.



Assim, concluímos que nosso diagrama de controle é composto por:

- Linha laranja tracejada: limite superior de casos esperados em cada semana epidemiológica considerando-se os dados referentes aos anos anteriores não epidêmicos.
- Linha azul escura: média de casos calculada com base nessa série histórica pregressa.
- Barras em azul claro: número de casos observados por semana epidemiológica no ano estudado.

A representação gráfica é um recurso poderoso! Você aprenderá neste curso que a sua confecção é relativamente simples, permitindo aos profissionais da vigilância em saúde o monitoramento dos casos e a análise de sua distribuição ao longo do tempo revelando se há uma epidemia ou uma endemia em curso.

Lembre-se que:



- **Endemia:** é quando um agravo ocorre com regularidade previsível (incidência habitual) em uma população, com pequenas flutuações ao longo do tempo.
- **Epidemia:** é quando a ocorrência da doença em um dado intervalo de tempo está acima do esperado, configurando um excesso de casos em relação à média dos últimos anos (incidência habitual) para um lugar e população em período determinado.

2. Construindo o diagrama de controle de dengue

Agora, vamos praticar?

Utilizaremos dados de dengue como exemplo para demonstrar o método mais aplicados para monitoramento: o diagrama de controle semanal. As arboviroses são doenças constantemente monitoradas e de relevância epidemiológica em vários municípios do país, principalmente dengue, zika e chikungunya. O Brasil é o país que mais contribui com suas magnitudes nas Américas. Assim acompanhar suas trajetórias é uma atividade realizada com frequência pelas vigilâncias e pode se tornar muito difícil em anos epidêmicos.

Para essa prática, vamos demonstrar a coleta de dados de dengue e a elaboração do diagrama de controle com apoio da linguagem R, de forma a automatizar este processo de avaliação que nos é rotineiro. Vamos lá?

Atenção



Uma série temporal que possua, por exemplo, apenas três anos não é indicada para ser monitorada via diagrama de controle por possuir poucos dados. O mesmo se aplica a uma série temporal de quatro anos seguidos com uma epidemia, pois não é possível estabelecer uma avaliação segura com poucos anos não epidêmicos!

Lembre-se que o diagrama de controle pode se tornar muito sensível a pequenas variações. Assim, cálculos inadequados podem levar o profissional de vigilância a realizar conclusões inadequadas da realidade como, por exemplo, indicar endemia quando há epidemia. Isso pode ocorrer se o canal epidêmico ficar com limite superior elevado pela incorporação de série histórica com anos epidêmicos.

2.1 Obtendo os dados

O primeiro passo é obter os dados que serão analisados. Utilizaremos aqui como fonte de dados o sistema [InfoDengue](#). O **InfoDengue** é um projeto criado pela Fiocruz e pela Fundação Getúlio Vargas que emite alertas e faz o monitoramento de arboviroses para todo o Brasil. No seu *site* é possível acessar boletins epidemiológicos de alerta e obter séries temporais de casos de dengue, chikungunya ou zika notificados por grande parte dos municípios pelo Brasil.

O InfoDengue é uma fonte de dados bastante interessante pois além da incidência de casos oferece dados de temperatura e umidade dos municípios.



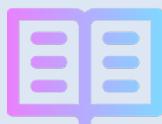
O InfoDengue é um sistema de alerta para arboviroses baseado em dados híbridos gerados por meio da análise integrada de dados minerados a partir da web social e de dados climáticos e epidemiológicos. Implementado em 2015, o sistema foi desenvolvido por pesquisadores do Programa de Computação Científica (Fundação Oswaldo Cruz, RJ) e da Escola de Matemática Aplicada (Fundação Getúlio Vargas) com a forte colaboração da Secretaria Municipal de Saúde do Rio de Janeiro e o Observatório da Dengue/UFMG.

Em 2021 o sistema ganhou amplitude nacional com o apoio do Ministério da Saúde, realizando análises complexas em nível estadual.

O InfoDengue disponibiliza relatórios semanais, sendo também possível baixar dados via **API** com séries históricas das arboviroses, variáveis climatológicas e correções de atraso de notificação (os chamados *nowcast*).

Clique [AQUI](#) para obter os boletins e diversos dados diretamente do *site*.

Para a nossa análise vamos utilizar a série histórica de dengue da cidade de Foz do Iguaçu, no Paraná, com dados entre o período de 2012 até 2022, organizados por semana epidemiológica. Para acessar esses dados do InfoDengue utilizaremos a sua **API** e, assim, além de automatizar nossa análise automatizaremos também a coleta de dados.



API se refere ao termo em inglês *Application Programming Interface*, que pode ser traduzido como “Interface de Programação de Aplicativos”. Ela funciona como uma espécie de porta que garante em tempo real que haja conexão com diversas aplicações e softwares, como o R. API's são muito importantes na área da saúde, e são amplamente utilizadas na atualidade por permitir a integração de diferentes funcionalidades entre sistemas em um único local.

A API torna o acesso às informações que queremos muito mais práticas, rápidas e é claro automatizadas. Os sistemas de informação do Ministério da Saúde estão em constantes transformações de digitais e alguns já é possível o acesso através de uma API como o e-SUS AB PEC e o e-SUS Notifica.

Por meio da API vamos conectar o seu R à base de dados do InfoDengue e, assim, efetuar o *download* dos dados que utilizaremos. Siga os passos abaixo para aprender a consultar a API e obter os dados necessários em seu computador:

1. Primeiro vamos instalar e carregar os pacotes necessários para importar a API e criar o diagrama de controle de dengue. Serão o `tidyverse`, o `lubridate` e o `xts`. Replique o código abaixo em seu RStudio:

```
# Instalando os pacotes necessários
if(!require(tidyverse)) install.packages("tidyverse"); library(tidyverse)
if(!require(lubridate)) install.packages("lubridate"); library(lubridate)
if(!require(foreign)) install.packages("foreign"); library(foreign)
if(!require(xts)) install.packages("xts"); library(xts)
```

2. Em seguida, vamos lá no RStudio definir os objetos necessários para gerar uma consulta à API do InfoDengue:

- `url`: o endereço eletrônico (URL) da API,
- `geocode`: o código do município segundo IBGE de Foz do Iguaçu, que é `4108304`,
- `format`: o formato dos dados; utilizaremos o arquivo do tipo `.csv`,
- `disease`: a arbovirose de interesse, que será a dengue,
- `ew_start` e `ew_end`: o período que será estudado, a semana epidemiológica de início (1) e fim (53); aqui, em geral, pode-se selecionar parte de um ano, mas quando lidamos com séries anuais basta deixar 52 ou 53, e
- `ey_start` e `ey_end`: os anos de início e fim de nossa série histórica: 2012 e 2022 no nosso exemplo.

Veja abaixo como construímos o `script` no R com essas informações necessárias. Replique essas linhas de código em seu RStudio.

```
# Atribuição de dados para consultar a API do InfoDengue
url <- "https://info.dengue.mat.br/api/alertcity?" # inserimos o endereço do InfoDengue
geocode <- 4108304 # indicamos código do IBGE de Foz do Iguaçu
disease <- "dengue" # selecionamos a doença
format <- "csv" # indicamos o formato de arquivo que será baixado
ew_start <- 1 # indicamos o início da semana epidemiológica
ew_end <- 53 # indicamos o final da semana epidemiológica
ey_start <- 2012 # indicamos o início do ano a ser exportado
ey_end <- 2022 # indicamos o final do ano a ser exportado
```

3. O terceiro passo é concatenar, ou seja, unir em uma única *string* os objetos escritos no passo **2**. Para concatenar todos os objetos, vamos utilizar a função `paste0()`. Dessa forma, criamos uma *URL* (um endereço como um *link* para acessar sites) de consulta à **API** que acessará o repositório e retornará os dados solicitados no formato *.csv*. O nosso objetivo aqui será programar o próprio **R** para que ele se encarregue de trazer os dados usando a internet, de forma automática para o objeto que chamaremos de `{consulta}`.

Replique o código abaixo no seu **RStudio** e visualize o objeto `consulta` que armazenará o *link* como de um site qualquer, mas, no nosso caso, é também um caminho para fazermos o *download* dos dados de dengue.

```
# programando o R para ele trazer os dados diretamente da internet
consulta <- paste0(url,
                  "geocode=", geocode,
                  "&disease=", disease,
                  "&format=", format,
                  "&ew_start=", ew_start,
                  "&ew_end=", ew_end,
                  "&ey_start=", ey_start,
                  "&ey_end=", ey_end)

# visualizando o link armazenado no objeto {`consulta`}
consulta
```

```
#> [1] "https://info.dengue.mat.br/api/alertcity?geocode=4108304&disease=dengue&
format=csv&ew_start=1&ew_end=53&ey_start=2012&ey_end=2022"
```

Ao executar o código acima, é possível visualizar um *link*, ou seja, a *URL* que criamos para consultar a API do InfoDengue. Você poderá inspecionar esse caminho de acesso *web* clicando [AQUI](#). OU, se preferir, copie e cole o *link* (exibido em seu output do **RStudio**) em seu navegador de internet e visualize dos dados que foram solicitados. Aproveite e perceba como os dados estão armazenados no InfoDengue.



Você também poderá obter esses dados manualmente. Veja na Figura 2 como é a tela para a seleção e *download* dos dados de forma manual. Clique [AQUI](#) para acessá-los do seu navegador de internet.

Figura 2: Tela da API do InfoDengue para *download* manual dos dados.

The screenshot shows a web browser window with the following details:

- Header:** A green bar at the top contains the text "Situação Epidemiológica" with a location pin icon, a close button ("x"), and a plus sign ("+"). Below it is a navigation bar with left and right arrows, a refresh icon, and a URL bar showing "info.dengue.mat.br/services/api".
- Logo:** On the left, there is a large, stylized lightbulb icon with purple and blue gradients and radiating lines.
- Page Header:** The title "INFO DENGUE" is displayed above a blue navigation bar. The bar includes links for "Início", "Sobre nós", "Equipe", "Participe", "Dados", "Relatórios", a search bar ("Encontre um município ..."), and a "Login" button.
- Sponsor Logos:** Logos for "FGV EMAP ESCOLA DE MATEMÁTICA APLICADA" and "FIOCRUZ" are visible on the right side of the header.
- Main Content Area:** The main content area has a grey header bar with "Dados / API /". Below it, the word "API" is centered. A sub-instruction "Para mais informações, acesse a documentação do serviço da API." is present.
- Filter Section:** A section titled "Filtro Município (código IBGE, nome do município ou U.F.):" contains a text input field labeled "Digite aqui o filtro desejado para a consulta de municípios" and a placeholder "Digite o código IBGE, nome do município ou U.F. para buscar o município desejado."
- Municipio Selection:** A dropdown menu titled "Município:" is shown with the placeholder "Selecione o município desejado para a consulta".
- Date Selection:** Two date inputs for epidemiological weeks are provided: "Data da Semana Epidemiológica de inicio da consulta" (set to "01/08/2022") and "Data da Semana Epidemiológica de término da consulta" (set to "29/08/2022").
- Arbovirose Selection:** A dropdown menu titled "Arbovirose" is set to "Dengue" with the placeholder "Selecione a arbovirose para a consulta".
- Format Selection:** A dropdown menu titled "Data da Semana Epidemiológica de término da consulta" is set to "CSV" with the placeholder "Selecione o formato do arquivo de saída".
- Action Buttons:** At the bottom are two buttons: "Download" (in blue) and "Mostrar Dicionário de Dados".

4. Agora que armazenamos os dados no objeto `{consulta}`, precisaremos executar a função `read_csv()` para que o R carregue o arquivo que é do tipo `.csv` com os dados da dengue. Nesta etapa iremos armazenar estes dados em um novo objeto que nomearemos de `{dengue_foz}` para que possamos iniciar nossa análise sem alterar o banco de dados original. Lembre-se que esta é uma boa prática para quem utiliza linguagem de programação!

Vamos lá. Observe o *script* abaixo e replique-o em seu RStudio:

```
# Armazenando o banco de dados do InfoDengue {consulta} no objeto {dengue_foz} para analisá-lo
dengue_foz <- read_csv(consulta)
```

```
#> Rows: 562 Columns: 25
#> — Column specification —
#> Delimiter: ","
#> dbl (23): SE, casos_est, casos_est_min, casos_est_max, casos, p_rt1, p_inc1...
#> date (2): data_iniSE, versao_modelo
#>
#> 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.
```

Atenção



Pode ser que demore alguns segundos para que a conexão seja estabelecida e o *download* seja realizado. Verifique o sinal de internet de sua rede local e, caso encontre problemas, procure o serviço de apoio à informática da sua localidade.

Caso não consiga acessar a API do Infodengue você poderá utilizar o arquivo localmente em seu computador. Para isso, acesse o menu lateral “Arquivos” do curso e faça *download* do banco de dados de nome **consulta.csv**.

Agora, replique o script abaixo em seu **RStudio** e importe o arquivo para o R:

```
# Esta primeira linha verifica se o pacote está instalado.  
# Caso não esteja, o código abaixo irá prosseguir com a instalação do pacote readr  
if(!require(readr)) install.packages("readr")  
  
#Carregando o pacote readr  
library(readr)  
  
# Utilize a função read_csv para importar os dados  
# do arquivo consulta.csv para um objeto chamado "dengue_foz"  
dengue_foz <- read_csv(file = "Dados/consulta.csv")  
  
# visualizando o arquivo `consulta.csv` no objeto {`dengue_foz`}  
dengue_foz
```

```
#> # A tibble: 562 × 25
#>   data_iniSE    SE casos...¹ casos...² casos...³ casos    p_rt1 p_inc...⁴ Local...⁵ nivel
#>   <date>      <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
#> 1 2022-10-02 202240    163    115    633    111 6.74e-1   63.1    0     1
#> 2 2022-09-25 202239    161    148    236    146 6.30e-1   62.3    0     1
#> 3 2022-09-18 202238    156    150    185    149 3.47e-1   60.4    0     1
#> 4 2022-09-11 202237    145    141    157    141 4.49e-2   56.1    0     1
#> 5 2022-09-04 202236    169    166    179    166 1.41e-1   65.4    0     1
#> 6 2022-08-28 202235    167    165    175    166 5.38e-2   64.7    0     1
#> 7 2022-08-21 202234    199    197    206    197 4.08e-1   77.1    0     2
#> 8 2022-08-14 202233    202    201    209    201 1.65e-1   78.2    0     2
#> 9 2022-08-07 202232    196    195    201    196 1.05e-3   75.9    0     2
#> 10 2022-07-31 202231   210    210    215    210 1.96e-6   81.3    0     2
#> # ... with 552 more rows, 15 more variables: id <dbl>, versao_modelo <date>,
#> # tweet <dbl>, Rt <dbl>, pop <dbl>, tempmin <dbl>, umidmax <dbl>,
#> # receptivo <dbl>, transmissao <dbl>, nivel_inc <dbl>, umidmed <dbl>,
#> # umidmin <dbl>, tempmed <dbl>, tempmax <dbl>, notif_accum_year <dbl>, and
#> # abbreviated variable names ¹casos_est, ²casos_est_min, ³casos_est_max,
#> # "p_inc100k, ⁵Localidade_id
#> # i Use `print(n = ...)` to see more rows, and `colnames()` to see all variable names
```

Siga em frente no curso.

Pronto, agora já importamos para o R um *data.frame* com dados de dengue da cidade de Foz do Iguaçu/PR no objeto `{dengue_foz}`. Vamos agora desenhar nosso diagrama de controle.



Atenção

Como o número de notificações podem sofrer inclusões e alterações constantemente, é possível, que os valores podem divergir de quando você importar. Isso é normal.

2.2 Montando o diagrama de controle

Até aqui já possuímos os dados que precisamos para a construção do nosso diagrama de controle do agravo dengue para estudar se há uma epidemia ou endemia município de Foz do Iguaçu/PR em 2022. Nesse momento agora precisamos escolher as variáveis necessárias para nossa análise. Esta etapa é fundamental!

Para esta escolha você necessitará estudar o dicionário de dados do InfoDengue, que se encontra disponível no site clicando [AQUI](#). Observe a Tabela 1 abaixo e escolha as variáveis necessárias para a construção do diagrama de controle:

Tabela 1: Dicionário de dados da API do InfoDengue.

Variável do InfoDengue	Descrição
data_ini_SE	Semana epidemiológica
casos_est	Número estimado de casos por semana usando o modelo de nowcasting
cases_est_min e cases_est_max	Intervalo de credibilidade de 95% do número estimado de casos
casos	Número de casos notificados por semana (Os valores são atualizados retrospectivamente todas as semanas)
p_rt1	Probabilidade de ($R_t > 1$). Para emitir o alerta laranja, usamos o critério $p_{rt1} > 0,95$ por 3 semanas ou mais
p_inc100k	Taxa de incidência estimada por 100.000
Localidade_id	Divisão submunicipal (atualmente implementada apenas no Rio de Janeiro)
nivel	Nível de alerta (1 = verde, 2 = amarelo, 3 = laranja, 4 = vermelho)
id	Identificação única
versao_modelo	Versão do modelo (uso interno)
tweet	Número de mensagens mencionando sintomas de dengue em tweets geolocalizados
Rt	Estimativa pontual do número reprodutivo de casos
pop	População estimada (IBGE)
tempmin	Média das temperaturas mínimas diárias ao longo da semana
umidmax	Média da umidade relativa máxima diária do ar ao longo da semana
receptivo	Indica receptividade climática, ou seja, condições para alta capacidade vetorial. 0 = desfavorável, 1 = favorável, 2 = favorável nesta semana e na semana passada, 3 = favorável por pelo menos três semanas (suficiente para completar um ciclo de transmissão)
transmissao	Evidência de transmissão sustentada
nivel_inc	Incidência estimada abaixo do limiar pré-epidemia, 1 = acima do limiar pré-epidemia, mas abaixo do limiar epidêmico, 2 = acima do limiar epidêmico
notif_accum_year	Número acumulado de casos no ano

Agora que já conhecemos todas as variáveis disponíveis é possível verificar que utilizaremos nesse exemplo apenas **três variáveis** da tabela que fizemos *download*. Abaixo listamos as colunas necessárias:

- `data_iniSE`: data de início da semana epidemiológica (SE),
- `casos`: número de casos de dengue na semana epidemiológica, e
- `p_inc100k`: incidência de dengue por 100 mil habitantes na semana epidemiológica.

Existem ainda várias outras variáveis que podem ser bastante interessantes para avaliação da situação de dengue fornecidas pelo InfoDengue tais como: temperatura mínima, umidade máxima e número de `twittes` realizados sobre o tema no município (monitoramento de rumores). Você pode visualizá-las e incluí-las para ampliar sua análise.

Para o próximo passo criaremos um novo objeto, o `{foz_ts}`. Ele será do tipo *série temporal*. Para isso, utilizaremos a função `xts()` do pacote `xts`. Para utilização da função será necessário definir dois argumentos:

- `x`: argumento que deverá indicar os dados a serem plotados. Ou seja, colocaremos no `x` a coluna da base de dados `{dengue_foz}` com os valores da incidência de casos (`p_inc100k`).
- `order.by`: argumento que deverá indicar a data correspondente aos dados indicados no argumento `x`. Aqui, utilizaremos a coluna com os valores da data de início das semanas epidemiológicas (`data_iniSE`).

Perceba que utilizaremos o operador `$` (*cífrão*) para indicar a seleção das variáveis dentro da base de dados. Observe o *script* abaixo e replique-o em seu `RStudio`:

```
# criando o objeto `foz_ts`
foz_ts <- xts(
  # selecionando a variável com os dados da incidência de dengue
  x = dengue_foz$p_inc100k,
  # selecionando a variável que contém as datas correspondentes
  order.by = dengue_foz$data_iniSE)
```

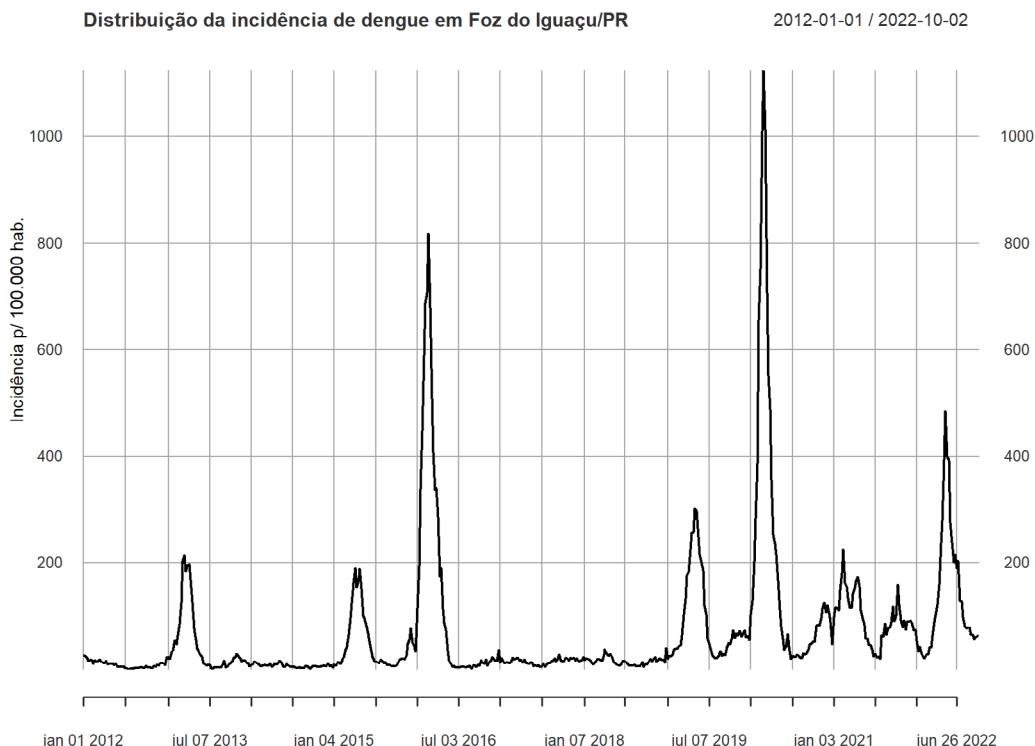
Agora que criamos uma tabela apenas com as variáveis `p_inc100k` e `data_iniSE`, vamos transformá-las em uma visualização gráfica utilizando a função `plot()`. Assim, criaremos a estrutura gráfica para a distribuição da incidência do agravo dengue por 100 mil habitantes no período entre 2012 e 2022. Para isso precisaremos utilizar os seguintes argumentos da função `plot()`:

- `x`: indicar qual o objeto que está armazenando a série temporal. No nosso exemplo será a tabela `{foz_ts}`.
- `main`: para incluir o título do gráfico. Vamos utilizar “Distribuição da incidência de dengue em Foz do Iguaçu/PR”.
- `yLab`: para incluir título do *eixo y* do gráfico. Vamos colocar “Incidência p/ 100.000 hab.”.

Observe o *script* abaixo a estrutura do código com a parametrização acima e replique-o em seu RStudio:

```
# Plotando o diagrama de controle
plot(
  # indicando a série temporal
  x = foz_ts,
  
  # colocando o título do gráfico
  main = 'Distribuição da incidência de dengue em Foz do Iguaçu/PR',
  
  # escrevendo o título do eixo y
  ylab = 'Incidência p/ 100.000 hab.'
)
```

Figura 3: Gráfico de linha da distribuição da incidência de dengue por 100 mil habitantes, no período entre 2012 e 2022, em Foz do Iguaçu/PR.



Você deverá obter um gráfico como o indicado na Figura 3 acima.

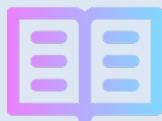
Precisamos analisar a distribuição dos valores de incidência do agravo dengue em toda a série histórica escolhida (anos entre 2012 e 2022), observando-se especialmente se há anos epidêmicos desta série temporal. Utilizaremos para isso a função `summary()`.

Observe o *script* abaixo e replique-o em seu RStudio:

```
# Analisando as estatísticas básicas do banco de dengue importado
summary(dengue_foz$p_inc100k)
```

#>	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
#>	0.000	9.777	21.104	71.871	76.864	1124.500

Observe o *output!* A média de incidência (`mean`) é um pouco maior que 70 casos por 100 mil habitantes e a mediana é um pouco maior que 20 casos por 100 mil habitantes. Essa diferença entre média e mediana é um bom indicador de que há valores muito altos e extremos puxando a média para cima, ou seja, *outliers* que estão demonstrando uma distribuição assimétrica dos dados. Na verdade, podemos observar isso também na Figura 3.



Os *outliers* são valores que estão fugindo do que se espera da normalidade dos dados e que podem, e provavelmente irão, causar distorções às suas análises.

Outliers são também comumente chamados em análise de dados de:

- valores que aparecem fora da curva,
- valores anômalos,
- valores discrepantes ou
- valores atípicos.

No nosso exemplo eles podem indicar surtos ou epidemias.

Para continuar a avaliar a distribuição dos dados que estamos analisando observaremos o comportamento da incidência de dengue ao longo dos dez anos (2012-2022). Para conseguir esta avaliação faremos (*plotaremos*) um gráfico do tipo *boxplot*.

Com o *boxplot* será possível visualizar a taxa de incidência em cada um dos anos exportados observando quais os anos se comportaram como *outliers* nesta série. Para isso, definiremos como parâmetro a classificação utilizada como [critério pelo Programa Nacional de Controle de Dengue](#) para avaliação da incidência de casos:

- Baixa incidência: até 100 casos por 100 mil habitantes;
- Média incidência: entre 101 e 299 casos por 100 mil habitantes, aqui utilizaremos como referência 200 casos;
- Alta incidência: 300 casos ou mais por 100 mil habitantes ou mais.

Representaremos estes parâmetros sobrepondo algumas linhas no *boxplot*:

1. uma linha tracejada azul (*no gráfico chamaremos de blue*) para representar a baixa incidência,
2. uma linha tracejada laranja (*orange*) para representar a média incidência, e
3. uma linha tracejada vermelha (*red*) para representar a alta incidência.



Atenção

Apesar de serem parâmetros muito utilizados no âmbito da vigilância e monitoramento de dengue, a definição de outros valores é válida a partir da realidade do seu município ou estado.

Agora executaremos a função `boxplot()` para plotar o gráfico. Vamos utilizar a variável `p_inc100k` da base de dados `{dengue_foz}` estratificada pelo ano da data de início da semana epidemiológica (`data_iniSE`). Fique atento ao observar o *script*, pois esta união é indicada na linguagem R pelo uso do símbolo `~`(til). Aqui também definiremos os títulos dos eixos x e y utilizando os argumentos `xlab` e `ylab`, respectivamente.

Para plotar as linhas vamos utilizar também a função `abline()` definindo os argumentos:

- `h`: os valores que as linhas devem cruzar no eixo y . No nosso exemplo vamos colocar linhas naqueles valores que definem, baixa, média e alta incidência de dengue,
- `lty`: tipo da linha. Aqui o número dois (2) representa linhas tracejadas,
- `lwd`: largura da linha, e
- `col`: cores das linhas. Como serão três linhas, utiliza-se a função `c()` para definir as três cores.



Acompanhe o *script* abaixo e reproduza-o em seu RStudio:

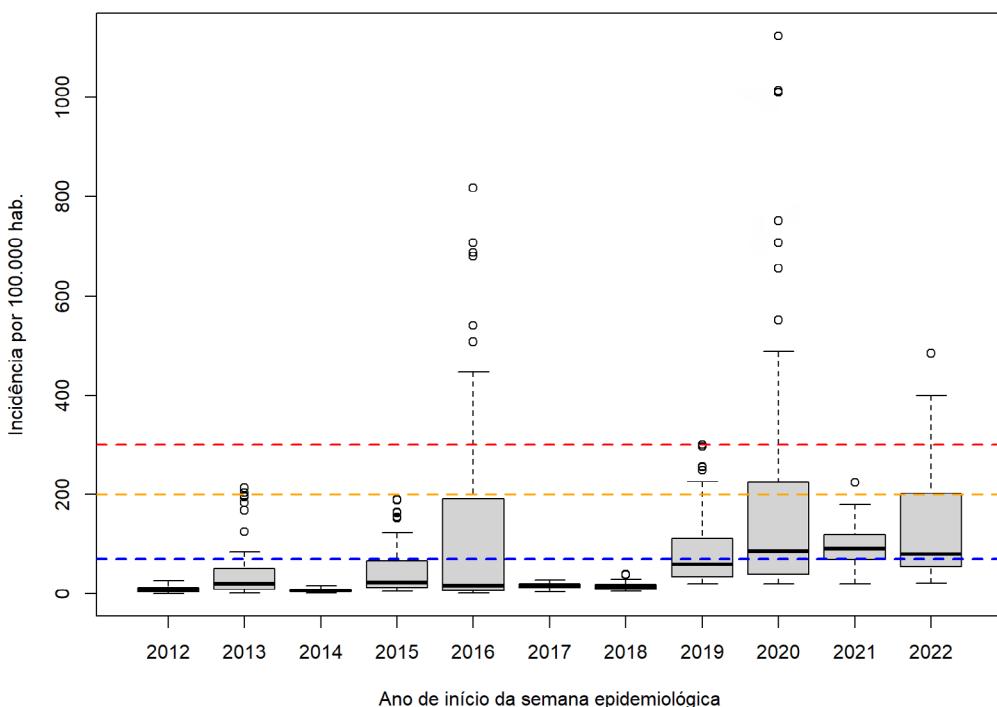
```
# Plotando gráfico boxplot com incidência por (`~`) ano
boxplot(dengue_foz$p_inc100k ~ year(dengue_foz$data_iniSE),
        ylab = 'Incidência por 100.000 hab.',
        xlab = "Ano de início da semana epidemiológica",

# Inserindo o título do boxplot
main = "Distribuição da incidência anual de dengue em Foz do Iguaçu-PR entre 2012-2022.")

# Criando linhas de análise a partir dos parâmetros que definimos
abline(
  h = c(300, 200, 70),
  lty = 2,
  lwd = 2,
  col = c('red', 'orange', 'blue')
)
```

Figura 4: Gráfico boxplot da distribuição da incidência de dengue por 100 mil habitantes, por ano, em Foz do Iguaçu/PR.

Distribuição da incidência anual de dengue em Foz do Iguaçu-PR entre 2012-2022.



Visualize o gráfico que retornou em seu RStudio. Ele deve ser semelhante ao observado aqui acima na Figura 4.

Precisaremos analisá-lo com atenção para indicar quais dos anos plotados são anos epidêmicos. Observe que, embora tenhamos em 2013 taxas mais altas que os demais anos, apenas seus *outliers* tocam a linha de cor laranja (média incidência). Ou seja, 2013 possui magnitude menor que os demais anos e o seu valor do terceiro quartil é inferior a 50%, por isso o incluiremos como ano não epidêmico. Assim, selecionaremos para a construção do diagrama de controle os anos de 2012, 2013, 2014, 2017 e 2018, pois são anos não epidêmicos - estão abaixo da baixa incidência de casos.

Agora que já escolhemos os anos **não epidêmicos** que levaremos em conta na criação do diagrama de controle, precisamos criar o objeto `{nao_epidemic}` que armazenará (receberá) esses anos.



Muitas séries históricas não possuem anos epidêmicos e você não precisará excluir qualquer valor. Essa decisão irá variar de acordo com a doença/agravo, o local e o período considerados na análise.

Vamos lá! Acompanhe o *script* abaixo e replique-o em seu RStudio:

```
# Criando o objeto {`nao_epidemic`} com anos não epidêmicos
nao_epidemic <- c(2012, 2013, 2014, 2017, 2018)
```

Pronto! Dados armazenados no objeto `{nao_epidemic}`. Agora podemos utilizá-los para criar mais um passo da nossa análise. Lembre-se que estamos analisando o comportamento de dengue em Foz do Iguaçu/PR em 2022 e, para isso, estamos comparando os dados deste ano com os demais anos não epidêmicos escolhidos (2012, 2013, 2014, 2017 e 2018) em um diagrama de controle. Continue sua análise seguindo os passos abaixo:

1. criaremos o objeto `{dengue_2022}` que armazenará os dados,
2. utilizaremos a função `filter()` do pacote `dplyr` e a função `year()` do pacote `lubridate` para a seleção do ano de 2022, e
3. criaremos uma coluna contendo as semanas epidemiológicas com a função `mutate()`.

É bem simples. Observe o *script* abaixo e replique-o em seu RStudio:

```
# Criando o objeto `dengue_2022` com ano de 2022
dengue_2022 <- dengue_foz |>

# Filtrando o ano para 2022
filter(year(data_iniSE) == 2022) |>

# Criando uma nova coluna chamada 'sem_epi', referente à semana epidemiológica
mutate(sem_epi = epiweek(data_iniSE))
```

Pronto! Agora utilizaremos o objeto `{dengue_2022}` para outra etapa da criação do diagrama. Acompanhe o passo a passo abaixo escrito no *script*. Com ele iremos construir a estrutura básica do diagrama de controle.

Esta etapa é importante, pois criaremos a tabela `{dengue_stat}` com cinco variáveis calculadas: a média, desvio padrão e limites superior e inferior da taxa de incidência por semana epidemiológica para *plotar* no gráfico. Para isso será necessário:

1. Filtrar utilizando as funções `filter()` e `year()`, os anos não epidêmicos utilizando o operador `%in%` que retorna TRUE quando o ano está contido no vetor `nao_epidemic`.
2. Utilizar a função `mutate()` para extrair da data de início da semana epidemiológica o número da semana e atribuir seu valor a `sem_epi`.
3. Agrupar com a função `group_by()` os dados por semana, ou seja, teremos os dados por semana para cada ano selecionado.
4. Utilizar a função `summarise()` e criar a média, desvio padrão e os limites máximo e mínimo para cada uma das semanas epidemiológicas dos períodos não epidêmicos.

Vamos lá, observe o *script* abaixo e replique o código em seu RStudio:

```
# Criando o gráfico com o diagrama de controle
dengue_stat <- dengue_foz |>

# Filtrando os dados da série em que o ano não é epidêmico
filter(year(data_iniSE) %in% nao_epidemic) |>

# Criando uma nova coluna chamada 'sem_epi', referente à semana epidemiológica
mutate(sem_epi = epiweek(data_iniSE)) |>

# Agrupando os dados pela semana epidemiológica
group_by(sem_epi) |>

# Criando medidas-resumo e limites superior e inferior
summarise(
  n = n(),
  media = mean(p_inc100k, na.rm = TRUE),
  desvio = sd(p_inc100k, na.rm = TRUE) ,
  sup = media + 2 * desvio,
  inf = media - 2 * desvio
)
```

Visualize agora as primeiras linhas da tabela `{dengue_stat}` utilizando a função `head()`. Observe o *script* abaixo e replique-o em seu RStudio:

```
# Visualizando a tabela `dengue_stat`  
head(dengue_stat)
```

```
#> # A tibble: 6 × 6  
#>   sem_epi     n  media  desvio    sup    inf  
#>   <dbl> <int> <dbl>  <dbl> <dbl>  <dbl>  
#> 1     1      6  21.4   11.8  45.0  -2.24  
#> 2     2      5  18.5   6.51  31.5   5.49  
#> 3     3      5  17.3   5.07  27.4   7.13  
#> 4     4      5  18.7   7.90  34.5   2.86  
#> 5     5      5  19.4   8.94  37.2   1.49  
#> 6     6      5  20.9   18.1  57.2  -15.4
```

Pronto, agora já temos as medidas necessárias para a construção gráfica do diagrama de controle de dengue. Assim, já poderemos fazer seu gráfico e incluir os valores de incidência da dengue às semanas epidemiológicas do ano de 2022.

Será necessário utilizar as funções do pacote `ggplot2`, um pacote construído para visualização de dados. Com o `ggplot2` os gráficos podem ser construídos de camada em camada, detalhando a visualização dos dados que queremos analisar.

Acompanhe o *script* abaixo com atenção e replique-o em seu RStudio. Nesse curso não teremos a oportunidade de detalhar os comandos para gerar os gráficos, assim, se tiver interesse, clique [aqui](#) para saber mais sobre o `ggplot2`. Aliás, vale muito a pena conhecer melhor o `ggplot2` para fazer gráficos muito interessantes. Mas para fazer os seus diagramas de controle basta seguir o *script* abaixo e adaptar as informações às suas necessidades.

```

# Definindo a base a ser utilizada
ggplot(data = dengue_stat) +

  # Definindo argumentos estéticos com as variáveis usadas em x e em y
  aes(x = sem_epi, y = media) +

  # Adicionando a linha referente à incidência média de dengue.
  # Adicionando uma geometria de linha na cor azul e largura de 1.2 pixel
  geom_line(aes(color = 'cor_media_casos'), size = 1.2) +

  # Adicionando uma geometria de linha na cor laranja. Além disso, inserindo
  # um argumento estético para o eixo y que, no caso, é a variável de limite
  # superior
  geom_line(aes(y = sup, color = 'cor_limite'), size = 1.2) +

  # Adicionando uma geometria de colunas utilizando a base de dados
  # `dengue_2022` e y como a incidência de dengue em 2022. O argumento
  # `fill` refere-se à cor das barras e `alpha` à transparência.
  geom_col(data = dengue_2022,
            aes(y = p_inc100k, fill = 'cor_incidencia'), alpha = 0.4) +

  # Arrumando o eixo x, definindo o intervalo de tempo que será utilizado (`breaks`)
  # uma sequência de semanas epidemiológicas de 1 a 53
  # o argumento `expand` ajuda nesse processo.
  scale_x_continuous(breaks = 1:53, expand = c(0, 0)) +

  # Definindo os títulos dos eixos x e y
  labs(x = 'Semana Epidemiológica',
       y = 'Incidência por 100 mil hab.',
       title = 'Diagrama de controle de dengue em Foz do Iguaçu/PR no ano de 2022.') +

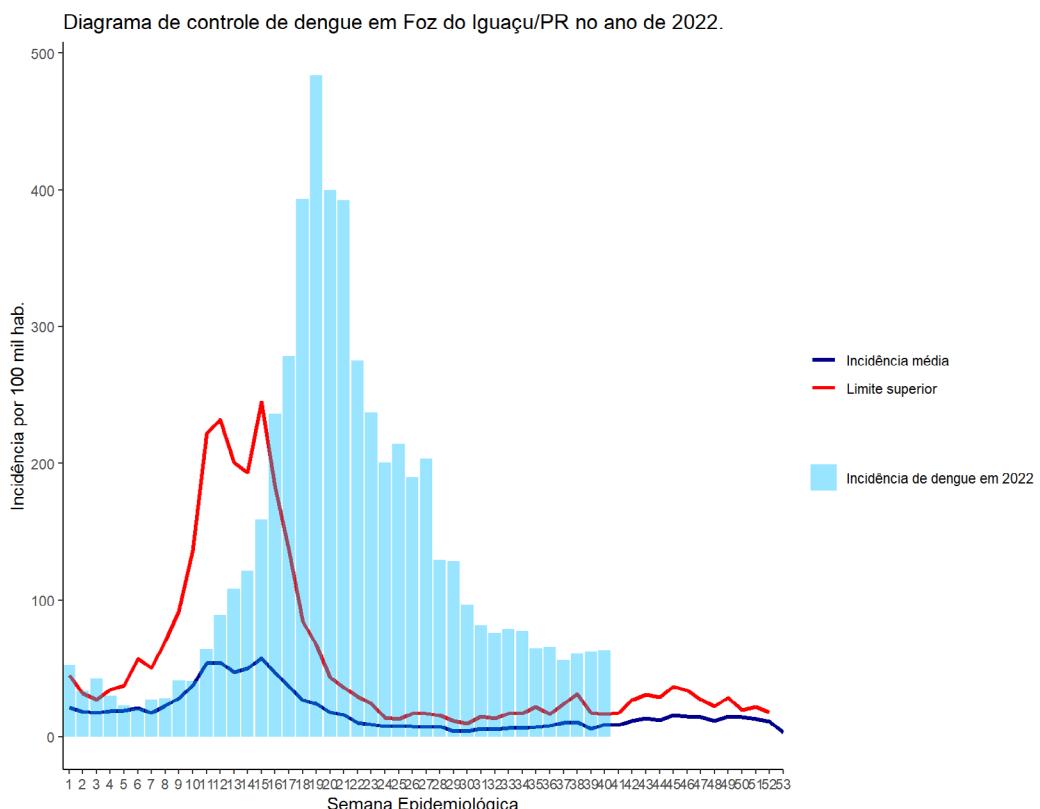
  # Definindo o tema do gráfico
  theme_classic() +

  # Criando a legenda das linhas
  scale_color_manual(
    name = "",
    values = c('cor_media_casos' = 'darkblue', 'cor_limite' = 'red'),
    labels = c("Incidência média", "Limite superior")
  ) +

  # Criando a legenda das barras
  scale_fill_manual(
    name = "",
    values = c('cor_incidencia' = 'deepskyblue'),
    labels = "Incidência de dengue em 2022"
)

```

Figura 5: Gráfico do Diagrama de Controle de dengue em Foz do Iguaçu/PR.



Ao observar a Figura 5 acima verificamos que 2022 foi um ano que ultrapassou o limite superior do diagrama de controle. Portanto, concluímos que Foz do Iguaçu pode estar vivendo um ano epidêmico. Ao observarmos a distribuição dos dados verificamos também que **o padrão sazonal foi tardio em relação aos anos não epidêmicos**. Isso significa que as semanas de maior incidência estavam entre as semanas epidemiológicas 10 e 16 (entre o mês de março e primeira quinzena de abril). Já em 2022 as semanas com maior incidência ficaram entre 17 e 23 (última semana de abril a primeira semana de junho), com o pico na semana 19 (entre 8 e 14 de maio).

Outro ponto a ser destacado na análise é que o limite superior (linha de cor vermelha) possui um padrão que oscila um pouco quando temos taxas de incidência maiores de 200 por 100 mil habitantes. Algumas vezes precisamos alisar estas oscilações e minimizar esse efeito utilizando a função `stat_smooth()` para suavizar as médias e o limite superior. Para isto, basta substituir a função `geom_line()` pela função `stat_smooth()` no script utilizado.

A função `stat_smooth()` suavizará a linha do gráfico, sendo o argumento `span` responsável em diminuir ou aumentar a suavização (de 0 a 1, sendo que quanto mais próximo a 1 mais suavidade se obtém). O argumento `se` definido como `FALSE` desabilita o intervalo de confiança ao redor das linhas. Os demais argumentos possuem a mesma função.

Acompanhe o *script* a seguir com atenção e veja a diferença do diagrama ao suavizarmos os ruídos. Replique os códigos em seu RStudio:

```
# Definindo a base a ser utilizada
ggplot(data = dengue_stat) +

# Definindo argumentos para as variáveis do eixo x e y do gráfico
aes(x = sem_epi, y = media) +

# Suavizando a linha referente à incidência média de dengue
# o argumento `size` para definir largura da linha = 1.2 pixel
# o argumento `se` = FALSE desabilita o intervalo de confiança
# e o argumento `span` definindo o valor da suavização
stat_smooth(
  aes(color = 'cor_incidencia_media'),
  size = 1.2,
  se = FALSE,
  span = 0.2
) +

# Suavizando a linha referente ao limite superior
stat_smooth(
  aes(y = sup, color = 'cor_limite'),
  size = 1.2,
  se = FALSE,
  span = 0.2
) +

# Adicionando uma geometria de colunas utilizando a base de dados
# {`dengue_2022`} e y como a incidência de dengue em 2022.
geom_col(data = dengue_2022,
          aes(y = p_inc100k, fill = 'cor_incidencia'),
          alpha = 0.4) +

# Arrumando o eixo x, definindo o intervalo de tempo que será utilizado (`breaks`)
# uma sequência de semanas epidemiológicas de 1 a 53
# o argumento `expand` ajuda nesse processo.
scale_x_continuous(breaks = 1:53, expand = c(0, 0)) +

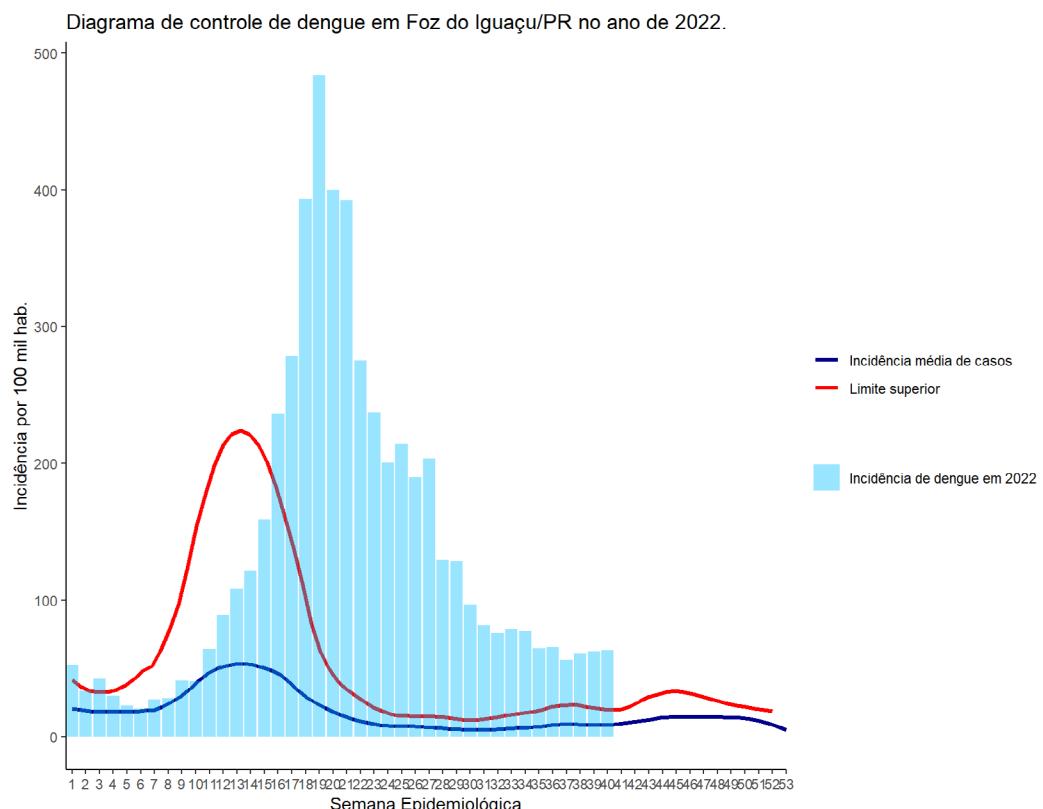
# Definindo os títulos dos eixos x, y e também do gráfico
labs(x = 'Semana Epidemiológica',
      y = 'Incidência por 100 mil hab.',
      title = 'Diagrama de controle de dengue em Foz do Iguaçu/PR no ano de 2022.') +

# Definindo o tema do gráfico
theme_classic() +

# Criando a legenda das linhas
scale_color_manual(
  name = "",
  values = c('cor_incidencia_media' = 'darkblue', 'cor_limite' = 'red'),
  labels = c("Incidência média de casos", "Limite superior")
) +

# Criando a legenda das barras
scale_fill_manual(
  name = "",
  values = c('cor_incidencia' = 'deepskyblue'),
  labels = "Incidência de dengue em 2022"
)
```

Figura 6: Gráfico alisado do diagrama de controle de dengue em Foz do Iguaçu/PR.



Veja a Figura 6. Ficou bacana, não é mesmo? Com o uso da função `stat_smooth()` foi possível visualizar de uma maneira mais clara que a partir da semana epidemiológica de número 16, em 2022, a taxa de incidência de dengue ultrapassa o limite superior do diagrama de controle, exibindo valores acima do esperado até depois da semana 28.

Vale também ressaltar o fato de que não foi usado o limite inferior do diagrama de controle, pois, neste caso, para os anos endêmicos o intervalo inferior seria menor que zero na maior parte das semanas. Isto pode variar conforme o agravo e podemos utilizar essa técnica para monitorar questões operacionais como, por exemplo, atraso de notificação.



Existem diversas tabelas e ferramentas para ajudar a selecionar as cores para a personalização de seu gráfico. Você pode consultar a página da [w3 color picker](#) e visualizar algumas cores.

Lembre-se que no R o símbolo (#) indica um código hexadecimal. Assim, o código #000000 representa a cor preta e #FFFFFF representa a cor branca. Combinando as três cores básicas podemos especificar mais de 16 milhões de cores e tons de cinza!

Para saber mais sobre a visualização gráfica de dados faça o curso “Visualização de dados de interesse para a vigilância em saúde” disponível na nossa coleção de cursos para a vigilância em saúde.

Pronto. Você já é capaz de avaliar a situação de dengue em qualquer município do Brasil. Mas como seria produzir um diagrama de controle para outras doenças? Acompanhe na seção a seguir o desenvolvimento de mais um diagrama de controle utilizando dados de Hepatite A.

3. Construindo o diagrama de controle de Hepatite A

É possível construir um diagrama de controle para qualquer doença que tenha seus casos suspeitos notificados com registro do início dos sintomas. Nesta subseção vamos analisar a situação da Hepatite A no município São Paulo (SP), utilizando como referência a avaliação do ano de 2017.

A vigilância das Hepatites Virais visa monitorar os casos da doença no território, recomendando ações que controlem e previnam em tempo oportuno as infecções. Dessa forma, analisar a ocorrência de casos ou surtos é estratégico para mitigar a doença. Aqui utilizaremos um diagrama de controle para avaliar um dos agravos de rotina para a vigilância em saúde.

Vamos lá! Considere que a coordenadoria de vigilância epidemiológica em saúde do Estado de São Paulo recebeu um alerta de aumento de casos de Hepatite A no início de março de 2017 no município de SP. Rapidamente, a equipe realizou a exportação dos dados do Sistema de Informação de Agravos de Notificação (Sinan Net) para avaliação e enviou os dados para seu e-mail.

Aqui neste exemplo vamos supor que você é membro da equipe técnica municipal de vigilância e agravos de doenças transmissíveis.

Você recebeu dois arquivos do tipo `.dbf`:

1. o primeiro contendo os casos confirmados de Hepatite A entre os anos de 2007 e 2016, e
2. e uma segunda exportação do Sinan Net apenas com os casos confirmados de Hepatite A de janeiro e fevereiro de 2017.

Agora como profissional da vigilância epidemiológica você iniciará a avaliação utilizando a linguagem de programação R!



Todos os bancos de dados utilizados para análises neste módulo se encontram no menu lateral “Arquivos” do curso. Lembre-se de fazer o download do material do curso diretamente do Ambiente Virtual de Aprendizagem do curso e arquivá-los em um diretório que deverá indicar para que o R o localize.

Primeiro, importaremos os dois arquivos para o ambiente do RStudio. Observe o código abaixo e replique-o em seu computador:

```
# criando objeto do tipo dataframe (tabela) {sinan_hep_sp_2007_2016} com o banco de
# dados {sinan_hep_sp_2007_2016.dbf}
sinan_hep_sp_2007_2016 <- read.dbf("Dados/sinan_hep_sp_2007_2016.dbf", as.is = TRUE)

# criando objeto do tipo dataframe (tabela) {sinan_hep_sp_fev_2017} com o banco de
# dados {sinan_hep_sp_fev_2017.dbf}
sinan_hep_sp_fev_2017 <- read.dbf("Dados/sinan_hep_sp_fev_2017.dbf", as.is = TRUE)
```

Pronto, seus dados já podem ser utilizados pelo R! Agora será necessário calcular a frequência de casos por semana epidemiológica e ano dos primeiros sintomas. Para isso, considere criar as variáveis correspondentes utilizando as funções do pacote `lubridate` para transformar datas: `epiweek()` e `year()`. Ambas as funções possuem apenas um argumento obrigatório, que é a variável no formato `date` (data). Você precisará utilizar a função `mutate()` do pacote `dplyr` para realizar as transformações.

Acompanhe o script abaixo e replique-o em seu RStudio:

```
# Criando a tabela `sinan_hep_sp_cont_07_16`
sinan_hep_sp_cont_07_16 <- sinan_hep_sp_2007_2016 |>

# Utilizando a função `mutate()` para criar as novas colunas
mutate(

  # Criando uma nova coluna chamada 'sem_epi', referente à semana
  # epidemiológica dos primeiros sintomas
  sem_epi = epiweek(x = DT_SIN_PRI),

  # Criando uma nova coluna chamada 'ano', referente ao ano dos primeiros sintomas
  ano = year(x = DT_SIN_PRI)) |>

  # Contando a frequência de notificações por ano e semana epidemiológica
  count(ano, sem_epi)
```

Agora que você já criou as colunas, visualize as primeiras linhas desta nova tabela utilizando a função `head()`. A nova tabela `{sinan_hep_sp_cont_07_16}` deverá possuir três colunas: `ano`, `sem_epi` e `n`. Reproduza o *script* abaixo em seu computador:

```
head(sinan_hep_sp_cont_07_16)
```

```
#>   ano sem_epi n
#> 1 2007      1 2
#> 2 2007      2 2
#> 3 2007      3 7
#> 4 2007      4 2
#> 5 2007      5 7
#> 6 2007      6 3
```

Pronto. Agora você deverá aplicar os mesmos passos à base de dados `{sinan_hep_sp_fev_2017}`. Não se preocupe, o código abaixo é praticamente o mesmo do mostrado anteriormente, apenas com a modificação da base de dados. Acompanhe abaixo o *script* e repita em seu RStudio:

```
# Criando o objeto `sinan_hep_sp_cont_fev17` e
# realizando a contagem dos casos segundo a
# semana epidemiológica e ano dos primeiros sintomas

sinan_hep_sp_cont_fev17 <- sinan_hep_sp_fev_2017 |>
  mutate(sem_epi = epiweek(DT_SIN_PRI),
        ano = year(DT_SIN_PRI)) |>
  count(ano, sem_epi)
```

Pronto! Já temos duas tabelas com as mesmas variáveis: `{sinan_hep_sp_cont_07_16}` e `{sinan_hep_sp_cont_fev17}`. Agora, vamos visualizar a variação dos casos confirmados de Hepatite A do Estado de São Paulo entre os anos 2007 e 2016. O objetivo dessa análise é identificar em quais anos ocorreram surtos ou epidemias. Queremos, portanto, identificar os anos epidêmicos para, eventualmente, retirá-los da estatística necessária para o diagrama de controle.

Para avaliação da distribuição de dados e análise dos anos epidêmicos vamos plotar um gráfico do tipo *boxplot* utilizando a função `boxplot()`. Com ela realizaremos o cruzamento da variável `n` (o número de casos confirmados de Hepatite A entre 2007 e 2016) com a variável `ano` (anos do início dos sintomas).

Para ter algum parâmetro de análise vamos utilizar a **mediana geral dos casos** (linha laranja) para visualizar aqueles anos que se apresentaram com número maior de casos.

Acompanhe o *script* abaixo:

```
# Definindo a mediana geral de casos confirmados entre 2007 e 2016
mediana_geral <- median(sinan_hep_sp_cont_07_16$n)

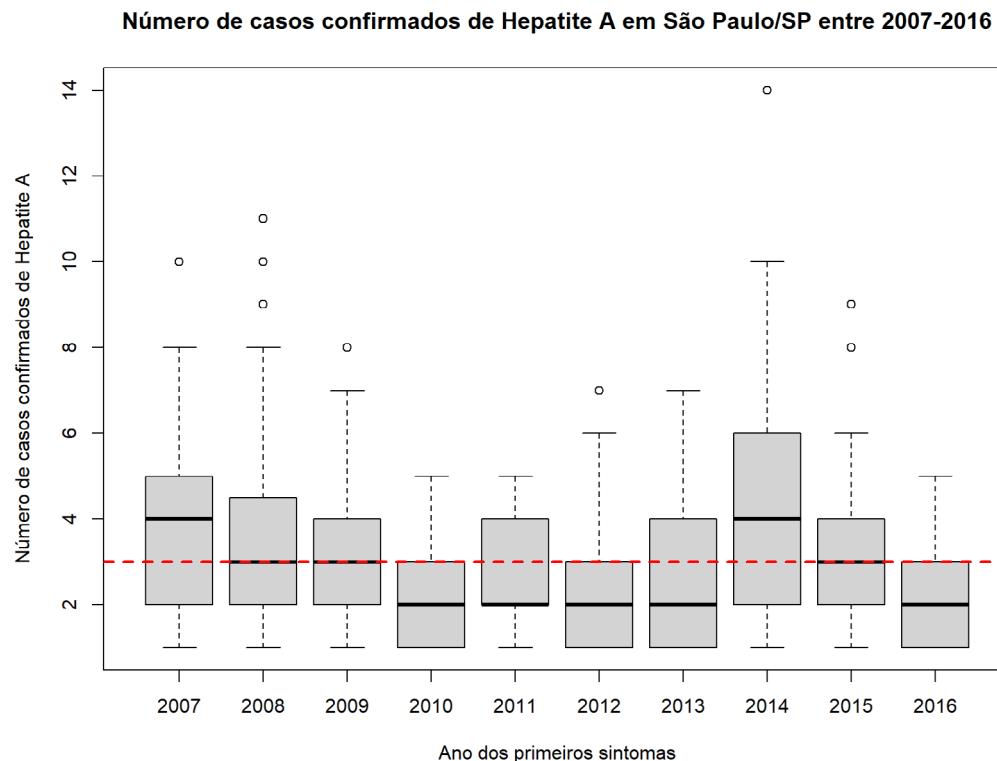
# utilizando a função `boxplot()` para criar o gráfico
boxplot(
  # Definindo o cruzamento número de casos por ano
  # Aqui utilizamos o símbolo "~" para sinalizar o cruzamento das variáveis
  sinan_hep_sp_cont_07_16$n ~ sinan_hep_sp_cont_07_16$ano,

  # Definindo os títulos dos eixos x e y
  ylab = 'Número de casos confirmados de Hepatite A',
  xlab = 'Ano dos primeiros sintomas',

  # Definindo o título do boxplot
  main = 'Número de casos confirmados de Hepatite A em São Paulo/SP entre 2007-2016'
)

# Criando linhas de análise
abline(
  h = mediana_geral,
  lty = 2,
  lwd = 2,
  col = "red"
)
```

**Figura 7: Gráfico boxplot da distribuição de casos
de Hepatite A, por ano, em São Paulo/SP.**



Após plotar o gráfico da Figura 7, você visualizará o *boxplot*. Agora vamos analisá-lo juntos:

- **cada retângulo (box)** representa a variação do número de casos dentro de cada ano,
- **a linha mais grossa dentro do retângulo (box)** representa a mediana de casos de cada ano,
- **a linha reta tracejada na cor preta acima do retângulo (box)** representa o limite superior de cada ano,
- **os círculos ao alto (outliers)** representam valores muito altos, além dos limites em cada ano, e
- **a linha tracejada na cor vermelha** representa a mediana geral de todos os anos (2007 a 2016).

Percebemos que 2007 e 2014 registraram número de casos de Hepatite A maior, pois suas medianas (linha mais grossa dentro do retângulo) são mais altas que os demais boxes. Além disso, é possível observar que 2014 registrou um pico de casos, se destacando na série. Já sabemos, portanto, que 2007 e 2014 apresentaram aumento nos casos de Hepatite A e por isso serão considerados para análise anos epidêmicos.

Agora, criamos um objeto que armazenará **os anos que definimos como não epidêmicos** para Hepatite A: 2008, 2009, 2010, 2011, 2012, 2013, 2015 e 2016. Perceba no script abaixo que utilizamos uma sequência de 2006 até 2013 com o auxílio do símbolo: (dois pontos) e adicionamos os anos de 2015 e de 2016 separados por , (vírgula). Com isso, escreveremos o código de forma mais curta.

Replique o código abaixo e execute os comandos em seu RStudio:

```
# criando o objeto {nao_epidemic} para armazenar os anos não epidêmicos
nao_epidemic <- c(2008:2013, 2015, 2016)
```

Pronto, já temos armazenados os anos não epidêmicos no objeto `{nao_epidemic}` para nossa avaliação. Vamos agora filtrar os casos confirmados de Hepatite A cujo ano dos primeiros sintomas estão contidos nestes anos. Execute os passos a seguir:

1. crie um novo objeto do tipo `data.frame` que armazenará os dados: `{hep_stat}`,
2. utilize a função `filter()` para os anos não epidêmicos,
3. agrupe os dados pela semana epidemiológica com a função `group_by()`, e
4. realize o cálculo das métricas necessárias para construção gráfica do diagrama de controle: média, desvio padrão e limites superior e inferior. Utilize para isso a função `summarise()` e as funções `mean()`, `sd()` e as fórmulas dos limites.

Acompanhe abaixo o *script* e reproduza-o em seu RStudio:

```
# Criando o objeto `hep_stat`
hep_stat <- sinan_hep_sp_cont_07_16 |>

# Filtrando os anos contidos no grupo de anos não epidêmicos
filter(ano %in% nao_epidemic) |>

# Agrupando os dados pela semana epidemiológica
group_by(sem_epi) |>

# Criando medidas-resumo e limites superior e inferior
summarise(
  media = mean(n, na.rm = TRUE),
  desvio = sd(n, na.rm = TRUE) ,
  sup = media + 2 * desvio,
  inf = media - 2 * desvio
)
```

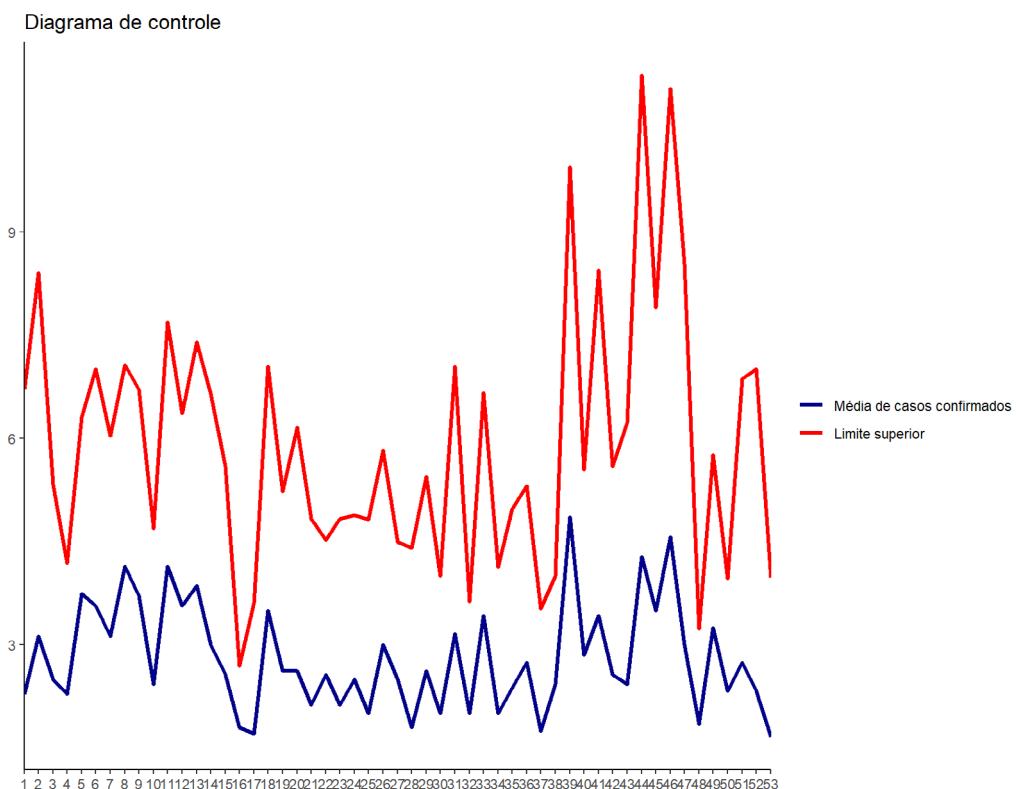
Feito! Agora chegou a hora de criar a visualização gráfica para avaliar o número de casos de Hepatite A confirmados até fevereiro de 2017, comparando-o aos demais anos. Essa avaliação é muito similar à realizada para construir o diagrama de controle de dengue. Assim será utilizado novamente o pacote `ggplot2`. Vamos lá. Siga o passo a passo abaixo:

1. Salve o gráfico inicial em um novo objeto: `{grafico_base}`. Necessitaremos dele como base estrutural para toda a análise, evitando assim retrabalho.
2. Adicione uma legenda ao gráfico tornando-o autoexplicativo. Para isso, use a função `aes()` aninhada à função `geom_line()` e o seu argumento `color`.
3. Personalize ainda mais o gráfico com a função `scale_color_manual()` e use os seguintes argumentos:
 - `Values`, que define os valores e suas respectivas cores.
 - `Labels`, que define o rótulo de dados para a legenda.
 - `Name`, que define o título da legenda que, nesse caso, ficará em branco (" ").

Colocaremos em prática essas três etapas e construiremos o nosso diagrama de controle. Observe o *script* abaixo e replique-o com muita atenção em seu RStudio:

```
# Criando um novo objeto `grafico_base`  
grafico_base <- ggplot(data = hep_stat) +  
  
# Definindo as variáveis usadas no eixo x e em y do gráfico  
aes(x = sem_epi, y = media) +  
  
# Adicionando uma geometria de linha com largura de 1.2 pixel  
# referente ao número médio de casos confirmados.  
# Além disso, inserindo um argumento estético para a cor,  
# que será convertida na legenda.  
geom_line(aes(color = "cor_media_casos"), size = 1.2) +  
  
# Adicionando uma geometria de linha referente ao limite superior.  
# Além disso, inserindo um argumento estético para o eixo y e  
# para a cor, que será convertida na legenda.  
geom_line(aes(y = sup, color = 'cor_limite'), size = 1.2) +  
  
# Arrumando o eixo x, definindo o intervalo de tempo que será utilizado (`breaks`)  
# uma sequência de semanas epidemiológicas de 1 a 53  
# o argumento `expand` ajuda nesse processo.  
scale_x_continuous(breaks = 1:53, expand = c(0, 0)) +  
  
# Definindo os títulos dos eixos x e y  
labs(x = '',  
     y = '',  
     title = 'Diagrama de controle') +  
  
# Definindo o tema do gráfico  
theme_classic() +  
  
# Criando a legenda das linhas  
scale_color_manual(  
  name = "",  
  values = c('cor_media_casos' = 'darkblue', 'cor_limite' = 'red'),  
  labels = c("Média de casos confirmados", "Limite superior"))  
)  
  
#visualizando o gráfico criado  
grafico_base
```

Figura 8: Gráfico da distribuição de casos de Hepatite A, por ano, em São Paulo/SP.



Observe que na Figura 8 mais uma vez existe variação aleatória das medidas calculadas, chamadas em séries temporais de **ruídos**. Vimos no exemplo anterior que a variação aleatória manifesta-se visualmente na forma de rugosidade nas linhas dos gráficos de séries temporais. O termo 'ruído' é efetivamente empregado para referenciar a variação aleatória na análise de séries temporais.

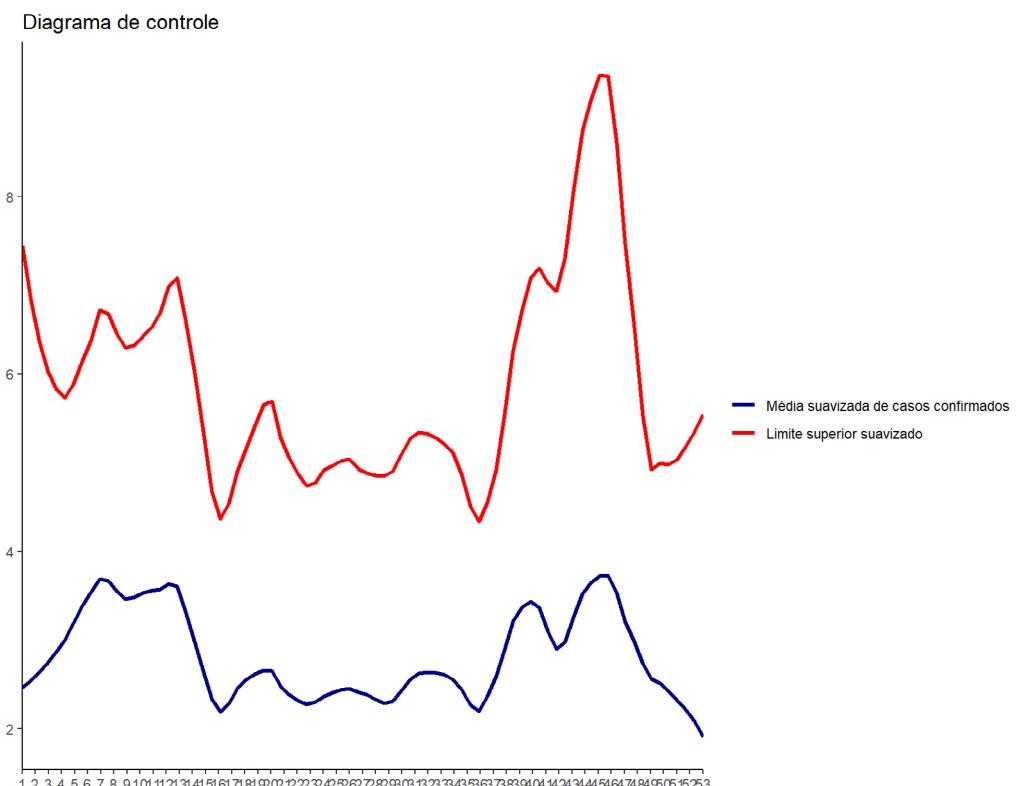
Lembre-se que uma forma de melhorar a visualização gráfica e evidenciar seus componentes de variação é suavizando as curvas, como fizemos na avaliação da dengue em Foz do Iguaçu/PR. Assim, aplicaremos ao gráfico feito acima a função `stat_smooth()` para avaliar se há a evidência da variação.

Observe o script abaixo e replique-o em seu RStudio:

```
grafico_base_suavizado <- ggplot(data = hep_stat) +  
  
  # Definindo as variáveis usadas no eixo x e em y do gráfico  
  aes(x = sem_epi, y = media) +  
  
  # Suavizando a linha referente ao número médio de casos.  
  # o argumento `size` para definir largura da linha = 1.2 pixel  
  # O argumento `se` = FALSE desabilita o intervalo de confiança  
  # e o argumento `span` definindo o valor da suavização  
  stat_smooth(  
    aes(color = 'cor_media_casos'),  
    size = 1.2,  
    se = FALSE,  
    span = 0.2  
  ) +  
  
  # Suavizando a linha referente ao limite superior  
  stat_smooth(  
    aes(y = sup, color = 'cor_limite'),  
    size = 1.2,  
    se = FALSE,  
    span = 0.2  
  ) +  
  
  # Arrumando o eixo x, definindo o intervalo de tempo que será utilizado (`breaks`)  
  # uma sequência de semanas epidemiológicas de 1 a 53  
  # o argumento `expand` ajuda nesse processo.  
  scale_x_continuous(breaks = 1:53, expand = c(0, 0)) +  
  
  # Definindo os títulos dos eixos x e y  
  labs(x = '',  
       y = '',  
       title = 'Diagrama de controle') +  
  
  # Definindo o tema do gráfico  
  theme_classic() +  
  
  # Criando a legenda das linhas  
  scale_color_manual(  
    name = "",  
    values = c('cor_media_casos' = 'darkblue', 'cor_limite' = 'red'),  
    labels = c("Média suavizada de casos confirmados", "Limite superior suavizado"))  
)  
  
#visualizando o gráfico criado  
grafico_base_suavizado
```

```
#> `geom_smooth()` using method = 'loess' and formula 'y ~ x'  
#> `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Figura 9: Gráfico alisado da distribuição de casos de Hepatite A, por ano, em São Paulo/SP.



Pronto. Percebeu a diferença na Figura 9? Ao optar pela suavização de curvas é preciso ficar atendo às análises. Em nosso exemplo, além de excluir anos com alto número de casos, também incluímos o desvio padrão para construção da média anual, o que pode mascarar uma epidemia. Não é raro que os surtos epidêmicos ocorram de forma brusca e inesperada.



Ao se analisar uma série temporal, o que diferenciaria um surto ou epidemia de uma variação aleatória? Fique atento, pois epidemias não ocorrem ao acaso e o esforço de analisar as séries temporais vem justamente no sentido de antecipar os problemas e preveni-los.

Por sua magnitude, os surtos epidêmicos não podem ser confundidos com as variações irregulares. Todas e quaisquer mudanças de padrão no acompanhamento de uma doença são importantes e devem ser investigados!

Perceba que o gráfico ainda não contém os dados de 2017, pois é apenas o gráfico com linhas básicas de análise que você poderá a qualquer momento incluir ou excluir anos a ele. Agora precisaremos somar a ele os dados de casos de Hepatite A de 2017. Para isso utilizaremos apenas uma ação de somar incluindo o operador `+`(mais) e a função `geom_col` do pacote `ggplot2`.

Acompanhe o passo a passo a seguir para fazer a inclusão de dados de 2017:

- 1.** calcularemos a frequência de casos e plotaremos no gráfico como sendo as barras, ou seja, eixo y (`y = n`),
- 2.** incluiremos uma legenda para o número de casos (`fill = 'cor_n_casos'`),
- 3.** modificaremos a opacidade das cores do gráfico para 0.4 (`alpha = 0.4`), e
- 4.** incluiremos a legenda das barras utilizando a função `scale_fill_manual()`.

Observe o *script* abaixo com atenção e acrescente as modificações em seu código no RStudio:

```
# Gráfico criado anteriormente
grafico_base_suavizado + 

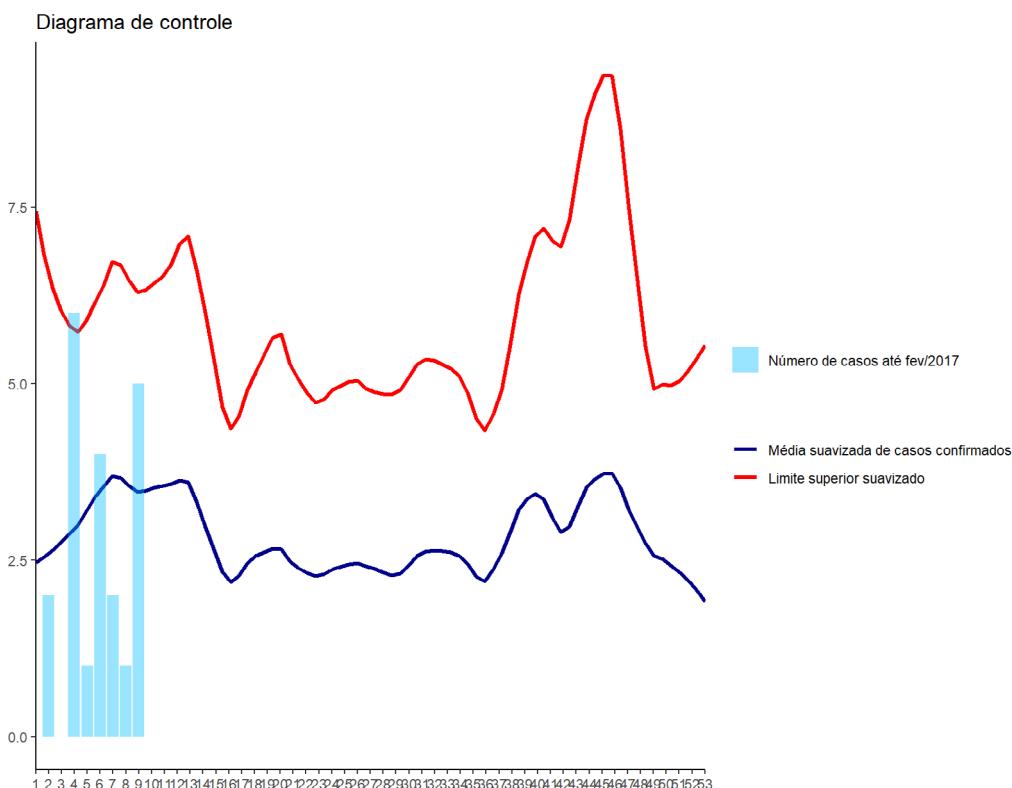
# Adicionando (+) uma geometria de colunas utilizando a base de dados
# `sinan_hep_sp_cont_fev17`
geom_col(data = sinan_hep_sp_cont_fev17,

          # O eixo y é definido como a frequência de casos (n).
          # As barras serão preenchidas com valor textual para definição da legenda.
          aes(y = n, fill = 'cor_n_casos'), alpha = 0.4) + 

# Definindo a legenda das barras, convertendo o valor textual em um nome de cor
# e definindo o rótulo
scale_fill_manual(
  name = "",
  values = c('cor_n_casos' = 'deepskyblue'),
  labels = "Número de casos até fev/2017"
)
```

```
#> `geom_smooth()` using method = 'loess' and formula 'y ~ x'
#> `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Figura 10: Gráfico do diagrama de controle de Hepatite A, em São Paulo/SP até fevereiro de 2017.



Observe a Figura 10 e perceba que o número de casos nas primeiras semanas do ano de 2017 tem ultrapassado a média dos últimos anos estudados. A semana epidemiológica 4 (entre os dias 22/01/2017 e 28/01/2017) apresentou uma magnitude que ultrapassou o limite esperado. Além disso, outras semanas epidemiológicas estão com o número de casos confirmados acima da média (semanas 6 e 9), mas abaixo do limite superior.



Lembre-se que para uma boa avaliação de situação de saúde será preciso descrever de forma detalhada todos os casos: caracterizando-o em tempo, lugar e pessoa. Será preciso exaurir todas as razões que estão levando ou podem levar a um aumento repentino dos casos de Hepatite A.

Considere agora que você acabou de receber por e-mail os dados atualizados de Hepatite A incluindo também o mês de março de 2017 no formato **.dbf**. Você precisará atualizar o seu diagrama de controle e enviá-lo novamente às áreas afins! Para isso, você irá aproveitar as análises realizadas até aqui e apenas incluir dados mais atualizados. Vamos lá, siga o passo a passo:

1. Primeiro, importe os dados para o utilizarmos no R.

Digite o *script* abaixo em seu computador:

```
# Importando o banco de dados {`sinan_hep_sp_mar_2017.dbf`} para o `R`
sinan_hep_sp_mar_2017 <- read.dbf("Dados/sinan_hep_sp_mar_2017.dbf", as.is = TRUE)
```

2. Segundo, você deve realiza o cálculo de frequência dos casos de Hepatite A.

Replique o *script* abaixo em seu RStudio:

```
# Criando o objeto {`sinan_hep_sp_cont_mar17`}
# realizando a contagem dos casos segundo a
# semana epidemiológica e ano dos primeiros sintomas
sinan_hep_sp_cont_mar17 <- sinan_hep_sp_mar_2017 |>
  mutate(
    sem_epi = epiweek(DT_SIN_PRI),
    ano = year(DT_SIN_PRI)
  ) |>
  count(ano, sem_epi)
```

3. Por fim, sobreponha o gráfico base com os dados atualizados até março de 2017.

Acompanhe os códigos abaixo e reproduza-os em seu RStudio:

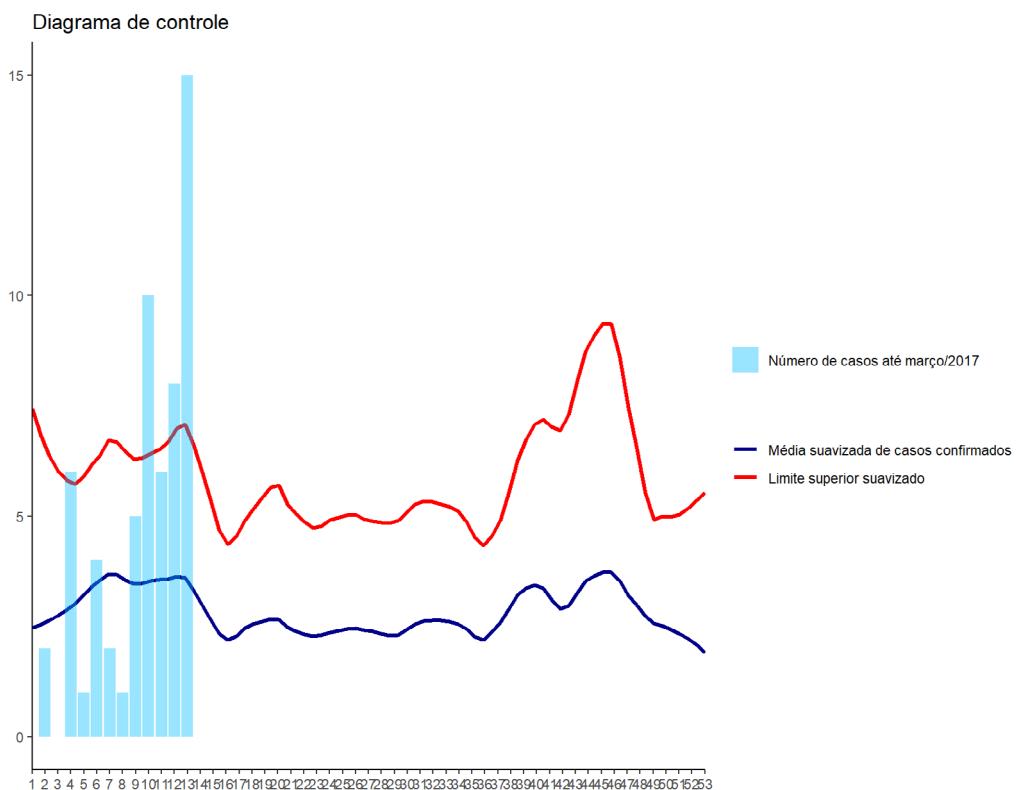
```
# Gráfico criado anteriormente
grafico_base_suavizado + 

# Adicionando uma geometria de colunas utilizando a base de dados
# {`sinan_hep_sp_cont_mar17`}
geom_col(data = sinan_hep_sp_cont_mar17,
         aes(y = n, fill = 'cor_n_casos'), alpha = 0.4) + 

# Definindo a legenda das barras, convertendo o valor textual em um nome de cor
# e definindo o rótulo
scale_fill_manual(
  name = "",
  values = c('cor_n_casos' = 'deepskyblue'),
  labels = "Número de casos até março/2017"
)
```

```
#> `geom_smooth()` using method = 'loess' and formula 'y ~ x'
#> `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Figura 11: Gráfico do diagrama de controle de Hepatite A, em São Paulo/SP até março de 2017.



Pronto. Plotando o novo gráfico, conforme Figura 11, foi possível notar que o município de São Paulo apresenta um surto epidêmico. Perceba que há um comportamento irregular do número de casos confirmados de Hepatite A nos três primeiros meses de 2017, apresentando um grave problema que deverá ser enfrentado. O número de casos confirmados de Hepatite A foi, durante todo o mês de março, acima da média e, quase sempre, acima do limite superior esperado.

A linguagem de programação R lhe apoiará como uma ferramenta poderosa para avaliação de mudanças de cenários epidemiológicos, oferecendo uma visão rápida da magnitude do problema. Em São Paulo os casos continuaram aumentando, deflagrando a posteriori como uma grande epidemia de Hepatite A no estado.

Observe abaixo o *script* para importação dos dados de 2017 completos: de janeiro até dezembro. Replique os códigos abaixo em seu RStudio:

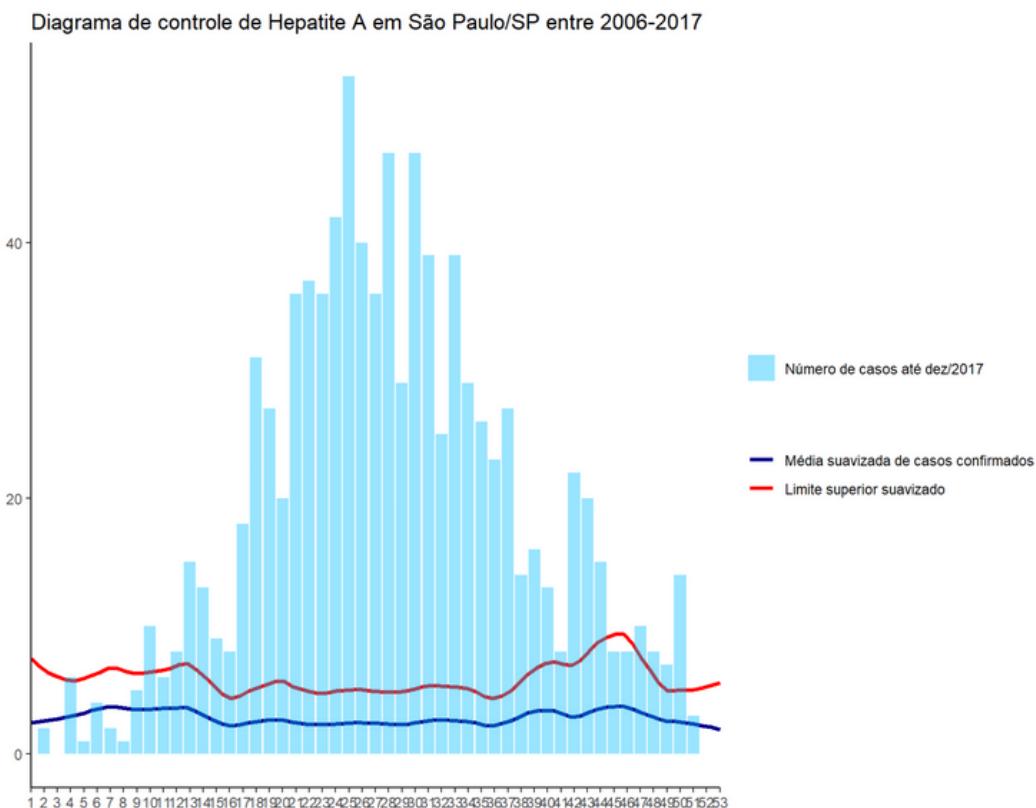
```
# Importando o banco de dados {`sinan_hep_sp_dez_2017.dbf`} para o `R`
sinan_hep_sp_dez_2017<- read.dbf("Dados/sinan_hep_sp_dez_2017.dbf", as.is = TRUE)

sinan_hep_sp_cont_17 <- sinan_hep_sp_dez_2017 |>
  mutate(
    sem_epi = epiweek(DT_SIN_PRI),
    ano = year(DT_SIN_PRI)
  ) |>
  count(ano, sem_epi)

grafico_base_suavizado +
  geom_col(data = sinan_hep_sp_cont_17,
           aes(y = n, fill = 'cor_n_casos'), alpha = 0.4) +
  scale_fill_manual(
    name = "",
    values = c('cor_n_casos' = 'deepskyblue'),
    labels = "Número de casos até dez/2017"
  ) +
  ggtitle("Diagrama de controle de Hepatite A em São Paulo/SP entre 2006-2017")
```

```
#> `geom_smooth()` using method = 'loess' and formula 'y ~ x'
#> `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Figura 12: Gráfico do diagrama de controle de Hepatite A, em São Paulo/SP até dezembro de 2017.



Observe na Figura 12 a existência de uma epidemia no Município de São Paulo, com número de casos maior que o limite superior.

Apesar da típica transmissão da infecção de Hepatite A por via fecal-oral, geralmente por meio de água e alimentos contaminados, nos últimos anos a disseminação tem sido associada às práticas sexuais, observada mais frequentemente entre homens que fazem sexo com homens (HSH).

Ocorreu que, em 2016, mais de 20 países da União Europeia foram acometidos com aumento na notificação de casos de Hepatites A e logo surgiram casos na Américas. Naquele momento a predominância era na população HSH. A cidade de São Paulo agiu à época com uma forte campanha para uso de preservativos e a prática do sexo seguro, além de investir na divulgação de informação para evitar a exposição fecal-oral durante a atividade sexual e cuidados de higiene. Outra medida importante foi reforçar principalmente a vacinação contra Hepatite A em pessoas com critérios já definidos para vacinação (pessoas vivendo com HIV/Aids, portadores crônicos de VHB e VHC e outras hepatopatias crônicas). Para saber mais sobre o surto em São Paulo clique [aqui](#)

Estas medidas foram determinantes para a redução da doença nos anos seguintes. A vacina contra a Hepatite A é fornecida pelo SUS de forma gratuita e é obrigatória no calendário vacinal do Programa Nacional de Imunização (PNI) para crianças de 15 meses a 4 anos, 11 meses e 29 dias.

Agora utilize seus novos conhecimentos na sua prática na vigilância. Será excelente, temos certeza!



Existem outros algoritmos para o cálculo do diagrama de controle. Veja um exemplo de uso no [boletim InfoDengue](#) para a semana 29 de 2022 em Foz do Iguaçu/PR.

Além disso, o R possui alguns pacotes que podem ajudar na aplicação de outras metodologias para o acompanhamento de doenças e a detecção de surtos. Alguns deles são MEM, qcc e surveillance.



Nossos cursos

Pronto, chegamos ao final deste curso! Agora você já conhece as principais ações para automatizar o diagrama de controle de doenças com o apoio da linguagem de programação R. Quer seguir a diante no aprendizado? Você encontrará outras etapas para aprofundamento das análises de dados em vigilância em saúde nos outros cursos. Aproveite e já faça sua inscrição nos cursos abaixo clicando nos *links*:

- [Análises de dados para Vigilância em Saúde - curso básico.](#)
- [Visualização de dados de interesse para a vigilância em saúde.](#)
- [Produção automatizada de relatórios na vigilância em saúde.](#)
- [Linkage de bases de dados de saúde.](#)
- [Análise espacial de dados para a vigilância em saúde.](#)
- [Construção de painéis \(dashboards\) para monitoramento de indicadores de saúde.](#)

Aproveite!

