

Análises do dados de vacinação

Gestantes e puérperas

11/setembro/2023

Contents

1	Sobre a base de dados e pacotes do R utilizados	1
2	Tratamento dos dados	9
2.1	Variável que indica se gestante ou puérpera	9
2.2	Raça	10
2.3	Idade	12
2.4	UF de aplicação	12
2.5	Município de aplicação	14
2.6	UF de residência	14
2.7	Município de residência	17
2.8	Qual vacina aplicada	17
2.9	Qual dose	20
2.10	Sobre tempo entre aplicação e importação no RNDS	23
3	Exclusão dos casos inconsistentes	28
3.1	Dados com status “entered-in-error”	28
3.2	ID do paciente que se repete para um mesmo número de dose e para o mesmo estabelecimento de saúde	29
3.3	Ano de aplicação da vacina	30

1 Sobre a base de dados e pacotes do R utilizados

A seguir são carregados os pacotes do R (<https://www.r-project.org>) utilizados para filtragem e tratamento dos dados considerados no dashboard <https://observatorioobstetrico.shinyapps.io/vacinacao-covid19>.

```
#carregar pacotes
loadlibrary <- function(x) {
  if (!require(x, character.only = TRUE)) {
    install.packages(x, dependencies = T)
    if (!require(x, character.only = TRUE))
      stop("Package not found")
  }
}
```

```

packages <-
  c(
    "readr",
    "readxl",
    "janitor",
    "dplyr",
    "forcats",
    "stringr",
    "lubridate",
    "summarytools",
    "magrittr",
    "questionr",
    "knitr",
    "data.table",
    "janitor",
    "modelsummary"
  )
lapply(packages, loadlibrary)

options(scipen=999)

```

```

#funções que vamos usar para as medidas descritivas
media <- function(x)
  mean(x, na.rm = TRUE)
mediana <- function(x)
  median(x, na.rm = TRUE)
DP <- function(x)
  sd(x, na.rm = TRUE)
minimo <- function(x)
  base::min(x, na.rm = TRUE)
maximo <- function(x)
  base::max(x, na.rm = TRUE)
q25 <- function(x)
  stats::quantile(x, p = 0.25, na.rm = TRUE)
q75 <- function(x)
  stats::quantile(x, p = 0.75, na.rm = TRUE)

```

```

IQR <- function(x)
  round(q75(x) - q25(x), 2)
n <- function(x)
  sum(!is.na(x))
faltantes <- function(x)
  round(sum(is.na(x)), digits = 0)

#dados <- read_csv("vacinacao-total-2023-04-04.csv")
dados1 <- read.csv("vacinacao_covid_001801.csv")
dados2 <- read.csv("vacinacao_covid_001901.csv")

dados <- rbind(dados1, dados2)
remove(dados1)
remove(dados2)

```

Dados obtidos em <https://opendatasus.saude.gov.br/dataset/covid-19-vacinacao> na data de 08/setembro/2023. A extração dos dados é realizada por meio de consultas à API ElasticSearch disponibilizada em <https://opendatasus.saude.gov.br/dataset/covid-19-vacinacao>.

Além disso, visando diminuir o volume de dados retornado pelas consultas, é realizada uma pré-filtragem dos dados diretamente na API, sendo retornados apenas registros cujo campo `vacina_grupoAtendimento_codigo` seja igual a 1901 (código para puerpera) ou 1801 (código para gestante). Essa etapa de extração foi realizada utilizando a linguagem de programação python.

1.0.1 Código para extração

```

import requests
import json
import time
import http.server
import socketserver
import csv

from apscheduler.schedulers.background import BackgroundScheduler
from datetime import datetime

def get_dados_vacinacao(grupoAtendimento = '001801'):
    start_time = time.time()
    print(f'Iniciando exportação de registros do grupo de atendimento {grupoAtendimento}')

```

```

        as {datetime.fromtimestamp(start_time).strftime("%H:%M:%S")}')
total = 0
response = requests.post("https://imunizacao-es.saude.gov.br/_search?scroll=1m",
    json={
        'size': 10000,
        'query': {
            'bool': {
                'filter': [{
                    'term': {
                        'vacina_grupoAtendimento_codigo': grupoAtendimento
                    }
                }]
            }
        }
    },
    auth=('imunizacao_public', 'qlto5t&7r_@+##Tlstigi'),
    headers={'Content-type': 'application/json'})
json_response = json.loads(response.text)
print(f'Total de registros a serem exportados: {json_response["hits"]["total"]["value"]}')
hits = [ dict(sorted(hit['_source'].items())) for hit in json_response['hits']['hits']]
csv_header = hits[0].keys()

with open(f'vacinacao_covid_{grupoAtendimento}.csv', 'w', encoding="utf-8") as f:
    writer = csv.DictWriter(f, fieldnames=csv_header)
    writer.writeheader()
    writer.writerows(hits)

total += len(hits)
print(f'Registros exportados: {total}')

while (hits):
    response = requests.post("https://imunizacao-es.saude.gov.br/_search/scroll",
        json={ 'scroll': '1m', 'scroll_id': json.loads(response.text)['_scroll_id']},
        auth=('imunizacao_public', 'qlto5t&7r_@+##Tlstigi'),
        headers={'Content-type': 'application/json'})
    hits = [ dict(sorted(hit['_source'].items())) for hit in json.loads(response.text)['hits']['hits']]

```

```

with open(f'vacinacao_covid_{grupoAtendimento}.csv', 'a', encoding="utf-8") as f:
    writer = csv.DictWriter(f, fieldnames=csv_header)
    writer.writerows(hits)

total += len(hits)
print(f'Registros exportados: {total}')

print(f'Finalizando exportação de registros do grupo de atendimento {grupoAtendimento} as {time.strftime("%H:%M:%S", time.localtime())}')
print(f'Importado {total} de registros em {"{:.2f}".format((time.time() - start_time)/60)} minutos')
return hits

get_dados_vacinacao('001901')
get_dados_vacinacao('001801')

```

#Criação das variáveis de data, mês e ano da aplicação

```

dados <- dados %>%
  dplyr::mutate(
    data_nascimento = str_sub(paciente_dataNascimento, start = 1, end = 10),
    data_aplic = str_sub(vacina_dataAplicacao, start = 1, end = 10),
    data_import = str_sub(data_importacao_rnds, start = 1, end = 10),
    dt_nasc = as.Date(data_nascimento),
    dt_aplic = as.Date(data_aplic),
    dt_import = as.Date(data_import),
    ano_aplic = year(dt_aplic),
    mes_aplic = month(dt_aplic)
  )

summary(dados$dt_aplic)

```

```

##           Min.          1st Qu.          Median            Mean          3rd Qu.           Max.
## "1989-01-20" "2021-07-07" "2021-09-13" "2021-11-16" "2022-02-01" "2023-09-08"

```

```
summary(dados$dt_import)
```

```

##           Min.          1st Qu.          Median            Mean          3rd Qu.           Max.
## "1989-01-20" "2021-07-13" "2021-09-21" "2021-09-06" "2022-02-27" "2023-09-09"

```

A data máxima de aplicação é 2023-09-08 e a data máxima de importação é 2023-09-09.

Os dados já estão filtrados vacina_grupoAtendimento_codigo igual a 1801 (código de gestante) ou 1901 (código de puerpera).

```
dados$vacina_grupoAtendimento_codigo <- as.character(dados$vacina_grupoAtendimento_codigo)
#tabela de frequência para grupo de atendimento
questionr::freq(
  dados$vacina_grupoAtendimento_codigo,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%
  kable(caption = "Tabela de frequências para grupo de atendimento",
        digits = 2, format = "markdown")
```

Table 1: Tabela de frequências para grupo de atendimento

	n	%	val%
1801	2268948	77.5	77.5
1901	657013	22.5	22.5
Total	2925961	100.0	100.0

Há atualmente 2925961 observações na base de dados e são as variáveis:

```
names(dados)
```

```
## [1] "X.timestamp"
## [2] "X.version"
## [3] "co_condicao_maternal"
## [4] "data_importacao_datalake"
## [5] "data_importacao_rnds"
## [6] "document_id"
## [7] "ds_condicao_maternal"
## [8] "dt_deleted"
## [9] "estabelecimento_municipio_codigo"
## [10] "estabelecimento_municipio_nome"
## [11] "estabelecimento_razaoSocial"
## [12] "estabelecimento_uf"
```

```
## [13] "estabelecimento_valor"
## [14] "estalecimento_noFantasia"
## [15] "id_sistema_origem"
## [16] "paciente_dataNascimento"
## [17] "paciente_endereco_cep"
## [18] "paciente_endereco_coIbgeMunicipio"
## [19] "paciente_endereco_coPais"
## [20] "paciente_endereco_nmMunicipio"
## [21] "paciente_endereco_nmPais"
## [22] "paciente_endereco_uf"
## [23] "paciente_enumSexoBiologico"
## [24] "paciente_id"
## [25] "paciente_idade"
## [26] "paciente_nacionalidade_enumNacionalidade"
## [27] "paciente_racaCor_codigo"
## [28] "paciente_racaCor_valor"
## [29] "sistema_origem"
## [30] "status"
## [31] "vacina_categoria_codigo"
## [32] "vacina_categoria_nome"
## [33] "vacina_codigo"
## [34] "vacina_dataAplicacao"
## [35] "vacina_descricao_dose"
## [36] "vacina_fabricante_nome"
## [37] "vacina_fabricante_referencia"
## [38] "vacina_grupoAtendimento_codigo"
## [39] "vacina_grupoAtendimento_nome"
## [40] "vacina_lote"
## [41] "vacina_nome"
## [42] "vacina_numDose"
## [43] "data_nascimento"
## [44] "data_aplic"
## [45] "data_import"
## [46] "dt_nasc"
## [47] "dt_aplic"
## [48] "dt_import"
```

```
## [49] "ano_aplic"
```

```
## [50] "mes_aplic"
```

Veja que há também as variáveis `vacina_grupoAtendimento_nome` e `vacina_categoria_codigo` e `vacina_categoria_nome`.

```
with(  
  dados,  
  ctable(  
    vacina_grupoAtendimento_codigo,  
    vacina_grupoAtendimento_nome,  
    prop = "n"  
  )  
)
```

```
## Cross-Tabulation
```

```
## vacina_grupoAtendimento_codigo * vacina_grupoAtendimento_nome
```

```
## Data Frame: dados
```

```
##
```

```
## -----  
##              vacina_grupoAtendimento_nome  Gestante  Puérpera    Total  
## vacina_grupoAtendimento_codigo  
##              1801                2268948           0  2268948  
##              1901                   0       657013   657013  
##              Total                2268948       657013  2925961  
## -----
```

```
dados$vacina_categoria_codigo <-  
  as.character(dados$vacina_categoria_codigo)  
with(dados,  
  ctable(  
    vacina_grupoAtendimento_codigo,  
    vacina_categoria_codigo,  
    prop = "n"  
  ))
```

```
## Cross-Tabulation
```

```
## vacina_grupoAtendimento_codigo * vacina_categoria_codigo
```

```
## Data Frame: dados
```



```
##
## -----
##          vacina_categoria_codigo      21      25      Total
##  vacina_grupoAtendimento_codigo
##          1801      2268948      0      2268948
##          1901      0      657013      657013
##          Total      2268948      657013      2925961
## -----
```

```
with(dados,
      ctable(vacina_grupoAtendimento_codigo, vacina_categoria_nome, prop = "n"))
```

```
## Cross-Tabulation
```

```
## vacina_grupoAtendimento_codigo * vacina_categoria_nome
```

```
## Data Frame: dados
```

```
##
```

```
## -----
##          vacina_categoria_nome  Gestantes  Puérperas  Puérperas  Tot
##  vacina_grupoAtendimento_codigo
##          1801      2268948      0      0      22689
##          1901      0      436477      220536      6570
##          Total      2268948      436477      220536      29259
## -----
```

2 Tratamento dos dados

2.1 Variável que indica se gestante ou puérpera

```
dados <- dados %>%
  mutate(gest_puerp = ifelse(vacina_grupoAtendimento_codigo == "1801",
                             "gesta",
                             "puerp"))
```

```
#tabela de frequência para gest_puerp
```

```
questionr::freq(
  dados$gest_puerp,
  cum = FALSE,
  total = TRUE,
```

```

na.last = TRUE,
valid = TRUE
) %>%
kable(caption = "Tabela de frequências para gestante ou puérpera",
      digits = 2, format = "markdown")

```

Table 2: Tabela de frequências para gestante ou puérpera

	n	%	val%
gesta	2268948	77.5	77.5
puerp	657013	22.5	22.5
Total	2925961	100.0	100.0

2.2 Raça

```

#tabela de frequência raca/cor original
questionr::freq(
  dados$paciente_racaCor_codigo,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%
kable(caption = "Tabela de frequências para raça/cor original",
      digits = 2, format = "markdown")

```

Table 3: Tabela de frequências para raça/cor original

	n	%	val%
1	1038108	35.5	35.5
2	140890	4.8	4.8
3	859039	29.4	29.4
4	460966	15.8	15.8
5	3766	0.1	0.1
99	422963	14.5	14.5
NA	229	0.0	NA

	n	%	val%
Total	2925961	100.0	100.0

#Criando a variável de raça/cor

```
dados <- dados %>%
```

```
  mutate(
    raca = case_when(
      paciente_racaCor_codigo == "1" ~ "branca",
      paciente_racaCor_codigo == "2" ~ "preta",
      paciente_racaCor_codigo == "3" ~ "parda",
      paciente_racaCor_codigo == "4" ~ "amarela",
      paciente_racaCor_codigo == "5" ~ "indigena",
      TRUE ~ NA_character_
    )
  )
```

#tabela de frequência raça/cor

```
questionr::freq(
  dados$raca,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%
  kable(caption = "Tabela de frequências para raça/cor",
        digits = 2, format = "markdown")
```

Table 4: Tabela de frequências para raça/cor

	n	%	val%
amarela	460966	15.8	18.4
branca	1038108	35.5	41.5
indigena	3766	0.1	0.2
parda	859039	29.4	34.3
preta	140890	4.8	5.6
NA	423192	14.5	NA

	n	%	val%
Total	2925961	100.0	100.0

2.3 Idade

#Criação da variável de idade como diferença entre data de nascimento e de aplicação

```
dados <- dados %>%
```

```
  mutate(
    idade_anos = as.period(interval(start = dt_nasc, end = dt_aplic))$year,
    dif_idade = idade_anos - paciente_idade
  )
```

*#Descritiva da diferença da idade (base de dados) e criada pela diferença de
#datas de nascimento e aplicação*

```
datasummary((dif_idade) ~ 1*(n+faltantes+media+DP+mediana+minimo+maximo),
  data = dados, output = 'markdown')
```

	n	faltantes	media	DP	mediana	minimo	maximo
dif_idade	2925732.00	229.00	0.00	0.00	0.00	0.00	0.00

Podemos observar que há 0 observações com a variável original de idade (`paciente_idade`) diferente da variável criada como a diferença entre data de nascimento e de aplicação da dose.

2.4 UF de aplicação

#Criando a variável de UF de aplicação

```
dados <- dados %>%
```

```
  mutate(
    aplic_uf = case_when(
      estabelecimento_uf == "" ~ NA_character_,
      estabelecimento_uf == "XX" ~ NA_character_,
      TRUE ~ estabelecimento_uf
    )
  )
```

```

#tabela de frequência para UF aplicação
questionr::freq(
  dados$aplic_uf,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%
  kable(caption = "Tabela de frequências para UF de aplicação",
        digits = 2, format = "markdown")

```

Table 6: Tabela de frequências para UF de aplicação

	n	%	val%
AC	7265	0.2	0.2
AL	34183	1.2	1.2
AM	60325	2.1	2.1
AP	6777	0.2	0.2
BA	202120	6.9	6.9
CE	105830	3.6	3.6
DF	44790	1.5	1.5
ES	90091	3.1	3.1
GO	101027	3.5	3.5
MA	83800	2.9	2.9
MG	367513	12.6	12.6
MS	21090	0.7	0.7
MT	60656	2.1	2.1
PA	95585	3.3	3.3
PB	74660	2.6	2.6
PE	202299	6.9	6.9
PI	54164	1.9	1.9
PR	244806	8.4	8.4
RJ	212332	7.3	7.3
RN	67990	2.3	2.3
RO	29151	1.0	1.0
RR	1685	0.1	0.1

	n	%	val%
RS	165323	5.7	5.7
SC	96671	3.3	3.3
SE	36057	1.2	1.2
SP	444245	15.2	15.2
TO	15526	0.5	0.5
Total	2925961	100.0	100.0

2.5 Município de aplicação

#Criando a variável de município (código IBGE) da aplicação

```
dados <- dados %>%
```

```
  mutate(
    aplic_muni = ifelse(
      is.na(estabelecimento_municipio_codigo) |
        estabelecimento_municipio_codigo == 999999,
      NA,
      estabelecimento_municipio_codigo
    )
  )
```

2.6 UF de residência

#Criando a variável de UF da residência

```
dados <- dados %>%
```

```
  mutate(
    resid_uf = case_when(
      paciente_endereco_uf == "" ~ NA_character_,
      paciente_endereco_uf == "XX" ~ NA_character_,
      paciente_endereco_uf == "None" ~ NA_character_,
      TRUE ~ paciente_endereco_uf
    )
  )
```

#tabela de frequência para UF residência

```
questionr::freq(
```

```

dados$resid_uf,
cum = FALSE,
total = TRUE,
na.last = TRUE,
valid = TRUE
) %>%
kable(caption = "Tabela de frequências para UF de residência",
      digits = 2, format = "markdown")

```

Table 7: Tabela de frequências para UF de residência

	n	%	val%
AC	7726	0.3	0.3
AL	36456	1.2	1.3
AM	59526	2.0	2.0
AP	7030	0.2	0.2
BA	207663	7.1	7.1
CE	108109	3.7	3.7
DF	45816	1.6	1.6
ES	87768	3.0	3.0
GO	99298	3.4	3.4
MA	81361	2.8	2.8
MG	366474	12.5	12.6
MS	22547	0.8	0.8
MT	61372	2.1	2.1
PA	96277	3.3	3.3
PB	77240	2.6	2.6
PE	197156	6.7	6.8
PI	53850	1.8	1.8
PR	237485	8.1	8.1
RJ	209996	7.2	7.2
RN	67739	2.3	2.3
RO	29424	1.0	1.0
RR	2433	0.1	0.1
RS	166127	5.7	5.7
SC	100380	3.4	3.4

	n	%	val%
SE	35806	1.2	1.2
SP	434318	14.8	14.9
TO	16773	0.6	0.6
NA	9811	0.3	NA
Total	2925961	100.0	100.0

Podemos observar que há 9811 NA (dados faltantes) para UF de residência. Vamos verificar a distribuição de dados faltantes da UF de residência por UF de aplicação:

```
#criação do indicador de missing para UF de residência
dados <- dados %>%
  mutate(indic_na_uf_resid = ifelse(is.na(resid_uf), "sim", "não"))

with(dados, ctable(aplic_uf, indic_na_uf_resid, prop = "r"))

## Cross-Tabulation, Row Proportions
## aplic_uf * indic_na_uf_resid
## Data Frame: dados
##
## -----
##      indic_na_uf_resid      não      sim      Total
## aplic_uf
## AC      7252 ( 99.8%)      13 (0.2%)      7265 (100.0%)
## AL     34112 ( 99.8%)      71 (0.2%)     34183 (100.0%)
## AM     60011 ( 99.5%)     314 (0.5%)     60325 (100.0%)
## AP      6751 ( 99.6%)      26 (0.4%)      6777 (100.0%)
## BA    201545 ( 99.7%)     575 (0.3%)    202120 (100.0%)
## CE    105137 ( 99.3%)     693 (0.7%)    105830 (100.0%)
## DF     44530 ( 99.4%)     260 (0.6%)     44790 (100.0%)
## ES     89855 ( 99.7%)     236 (0.3%)     90091 (100.0%)
## GO    100715 ( 99.7%)     312 (0.3%)    101027 (100.0%)
## MA     83412 ( 99.5%)     388 (0.5%)     83800 (100.0%)
## MG    366324 ( 99.7%)    1189 (0.3%)   367513 (100.0%)
## MS     21048 ( 99.8%)      42 (0.2%)     21090 (100.0%)
## MT     60544 ( 99.8%)     112 (0.2%)     60656 (100.0%)
```


##	PA	95245 (99.6%)	340 (0.4%)	95585 (100.0%)
##	PB	74425 (99.7%)	235 (0.3%)	74660 (100.0%)
##	PE	201450 (99.6%)	849 (0.4%)	202299 (100.0%)
##	PI	53889 (99.5%)	275 (0.5%)	54164 (100.0%)
##	PR	244168 (99.7%)	638 (0.3%)	244806 (100.0%)
##	RJ	211598 (99.7%)	734 (0.3%)	212332 (100.0%)
##	RN	67833 (99.8%)	157 (0.2%)	67990 (100.0%)
##	RO	29052 (99.7%)	99 (0.3%)	29151 (100.0%)
##	RR	1673 (99.3%)	12 (0.7%)	1685 (100.0%)
##	RS	164912 (99.8%)	411 (0.2%)	165323 (100.0%)
##	SC	96502 (99.8%)	169 (0.2%)	96671 (100.0%)
##	SE	35953 (99.7%)	104 (0.3%)	36057 (100.0%)
##	SP	442752 (99.7%)	1493 (0.3%)	444245 (100.0%)
##	TO	15462 (99.6%)	64 (0.4%)	15526 (100.0%)
##	Total	2916150 (99.7%)	9811 (0.3%)	2925961 (100.0%)
##	-----			

2.7 Município de residência

```
#Criando a variável de município (código IBGE) da residência
dados <- dados %>%
  mutate(
    resid_muni = ifelse(
      is.na(paciente_endereco_coIbgeMunicipio) |
        paciente_endereco_coIbgeMunicipio == 999999,
      NA,
      paciente_endereco_coIbgeMunicipio
    )
  )
```

2.8 Qual vacina aplicada

```
#tabela de frequência para vacina_nome
questionr::freq(
  dados$vacina_nome,
  cum = FALSE,
  total = TRUE,
```

```

na.last = TRUE,
valid = TRUE
) %>%
kable(caption = "Tabela de frequências para vacina nome (original)",
      digits = 2, format = "markdown")

```

Table 8: Tabela de frequências para vacina nome (original)

	n	%	val%
COVID-19 ASTRAZENECA - ChAdOx1-S	6074	0.2	0.2
COVID-19 ASTRAZENECA/FIOCRUZ - COVISHIELD	250309	8.6	8.6
COVID-19 JANSSEN - Ad26.COV2.S	76264	2.6	2.6
COVID-19 PEDIÁTRICA - PFIZER COMIRNATY	435	0.0	0.0
COVID-19 PFIZER - COMIRNATY	1840705	62.9	62.9
COVID-19 PFIZER - COMIRNATY BIVALENTE	106687	3.6	3.6
COVID-19 PFIZER - COMIRNATY PEDIÁTRICA	180	0.0	0.0
COVID-19 PFIZER - COMIRNATY PEDIÁTRICA MENOR DE 5 ANOS	120	0.0	0.0
COVID-19 SINOVAC - CORONAVAC	2431	0.1	0.1
COVID-19 SINOVAC/BUTANTAN - CORONAVAC	642756	22.0	22.0
Total	2925961	100.0	100.0

```

dados$id <- 1:dim(dados)[1]
#Criando a variável de qual vacina aplicada
dados <- dados %>%
  mutate(
    qual_vacina = case_when(
      vacina_nome == "COVID-19 PFIZER - COMIRNATY BIVALENTE" ~ "Bivalente",
      id %in% str_which(vacina_nome, "AstraZeneca") ~ "AstraZeneca",
      id %in% str_which(vacina_nome, "ASTRA") ~ "AstraZeneca",
      id %in% str_which(vacina_nome, "Coronavac") ~ "Coronavac",
      id %in% str_which(vacina_nome, "CORONAVAC") ~ "Coronavac",
      id %in% str_which(vacina_nome, "JANSSEN") ~ "Janssen",
      id %in% str_which(vacina_nome, "PFIZER") ~ "Pfizer",
      id %in% str_which(vacina_nome, "Pfizer") ~ "Pfizer",
      id %in% str_which(vacina_nome, "Covishield") ~ "Covishield",

```

```

vacina_nome == "" ~ NA_character_,
vacina_nome == "Pendente Identificação" ~ NA_character_,
vacina_nome == "Pendente Identificação" ~ NA_character_,
vacina_nome == "INF3" ~ NA_character_,
TRUE ~ vacina_nome

# vacina_nome == "Vacina covid-19 - Ad26.COV2.S - Janssen-Cilag" ~ "Janssen",
# vacina_nome == "Covid-19-AstraZeneca" ~ "AstraZeneca",
# vacina_nome == "Covid-19-Coronavac-Sinovac/Butantan" ~ "Coronavac",
# vacina_nome == "Vacina covid-19 - BNT162b2 - BioNTech/Fosun Pharma/Pfizer" ~ "Pfizer",
# vacina_nome == "Vacina Covid-19 - Covishield" ~ "Covishield",
# vacina_nome == "" ~ NA_character_,
# vacina_nome == "Pendente Identificação" ~ NA_character_,
# vacina_nome == "Pendente Identificação" ~ NA_character_,
# TRUE ~ vacina_nome

)
)

```

```

#tabela de frequência para qual_vacina
questionr::freq(
  dados$qual_vacina,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%
  kable(caption = "Tabela de frequências para qual vacina",
        digits = 2, format = "markdown")

```

Table 9: Tabela de frequências para qual vacina

	n	%	val%
AstraZeneca	256383	8.8	8.8
Bivalente	106687	3.6	3.6
Coronavac	645187	22.1	22.1
Janssen	76264	2.6	2.6
Pfizer	1841440	62.9	62.9
Total	2925961	100.0	100.0

	n	%	val%
--	---	---	------

2.9 Qual dose

```
#tabela de frequência para vacina_descricao_dose
questionr::freq(
  dados$vacina_descricao_dose,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%
  kable(caption = "Tabela de frequências para qual dose",
        digits = 2, format = "markdown")
```

Table 10: Tabela de frequências para qual dose

	n	%	val%
	4	0.0	0.0
1ª Dose	1068927	36.5	36.5
1ª Dose Revacinação	1521	0.1	0.1
1º Reforço	4046	0.1	0.1
2ª Dose	985170	33.7	33.7
2ª Dose Revacinação	1885	0.1	0.1
2º Reforço	140974	4.8	4.8
3ª Dose	5599	0.2	0.2
3ª Dose Revacinação	1052	0.0	0.0
3º Reforço	1590	0.1	0.1
4ª Dose	3365	0.1	0.1
4ª Dose Revacinação	1784	0.1	0.1
5ª Dose	113	0.0	0.0
5ª Dose Revacinação	15	0.0	0.0
Dose	2053	0.1	0.1
Dose Adicional	29254	1.0	1.0
Reforço	677760	23.2	23.2

	n	%	val%
Revacinação	632	0.0	0.0
Única	217	0.0	0.0
Total	2925961	100.0	100.0

Vamos analisar a seguir o cruzamento dos dados de qual vacina aplicada com a descrição da dose.

```
#tabela cruzada de qual dose e qual vacina
```

```
with(dados, table(qual_vacina,vacina_descricao_dose))
```

```
##          vacina_descricao_dose
## qual_vacina    1ª Dose 1ª Dose Revacinação 1º Reforço 2ª Dose
##  AstraZeneca      3   63940                295         93   42067
##  Bivalente        0      7                  0        834      1
##  Coronavac        0  311746                48        343  297370
##  Janssen          0     54                 121        44     49
##  Pfizer           1  693180               1057       2732  645683
##          vacina_descricao_dose
## qual_vacina    2ª Dose Revacinação 2º Reforço 3ª Dose 3ª Dose Revacinação
##  AstraZeneca                489     43199     253                241
##  Bivalente                  0        12      11                  0
##  Coronavac                  25     8493     179                37
##  Janssen                    127    11334      22                143
##  Pfizer                     1244    77936    5134                631
##          vacina_descricao_dose
## qual_vacina    3º Reforço 4ª Dose 4ª Dose Revacinação 5ª Dose
##  AstraZeneca      333     431                494         0
##  Bivalente         0       0                  0         0
##  Coronavac        50      34                  38         0
##  Janssen          274     268                150         0
##  Pfizer           933     2632               1102     113
##          vacina_descricao_dose
## qual_vacina    5ª Dose Revacinação Dose Dose Adicional Reforço Revacinação
##  AstraZeneca                1      0                5333   99093     118
##  Bivalente                  0      0                  6  105806         0
##  Coronavac                  1      0                1142   25651     20
```

```
##      Janssen                0  2053                2965  58424                43
##      Pfizer                 13    0                19808 388786                451
##                               vacina_descricao_dose
## qual_vacina      Única
## AstraZeneca      0
## Bivalente        10
## Coronavac        10
## Janssen           193
## Pfizer            4
```

Vamos fazer agora a junção de informações que se referem a mesma dose de vacina.

```
dados <- dados %>%
  mutate(
    num_dose = case_when(
      vacina_descricao_dose == "1ª Dose" ~ "1a dose",
      vacina_descricao_dose == "2ª Dose" ~ "2a dose",
      vacina_descricao_dose == "3ª Dose" ~ "3a dose",
      vacina_descricao_dose == "4ª Dose" ~ "4a dose",
      (vacina_descricao_dose == "1º Reforço" |
       vacina_descricao_dose == "Dose Adicional") |
      vacina_descricao_dose == "Reforço" ~ "Dose adicional + 1º reforço + reforço",
      vacina_descricao_dose == "Dose" |
      vacina_descricao_dose == "Única" ~ "Dose+Única",
      TRUE ~ as.character(vacina_descricao_dose)
    )
  )
```

```
#tabela cruzada de qual dose e qual vacina
with(dados, table(qual_vacina, num_dose))
```

```
##                               num_dose
## qual_vacina      1a dose 1ª Dose Revacinação 2a dose 2ª Dose Revacinação
## AstraZeneca      3  63940                295  42067                489
## Bivalente        0    7                0    1                0
## Coronavac        0 311746                48 297370                25
## Janssen          0    54                121    49                127
## Pfizer           1 693180                1057 645683                1244
##                               num_dose
```

## qual_vacina	2º Reforço	3a dose	3ª Dose	Revacinação	3º Reforço	4a dose
## AstraZeneca	43199	253		241	333	431
## Bivalente	12	11		0	0	0
## Coronavac	8493	179		37	50	34
## Janssen	11334	22		143	274	268
## Pfizer	77936	5134		631	933	2632

##	num_dose
## qual_vacina	4ª Dose Revacinação
## AstraZeneca	494
## Bivalente	0
## Coronavac	38
## Janssen	150
## Pfizer	1102

##	num_dose
## qual_vacina	5ª Dose
## AstraZeneca	0
## Bivalente	0
## Coronavac	0
## Janssen	0
## Pfizer	113

##	Dose adicional + 1º reforço + reforço	Dose+Única	Revacinação
## AstraZeneca	104519	0	118
## Bivalente	106646	10	0
## Coronavac	27136	10	20
## Janssen	61433	2246	43
## Pfizer	411326	4	451

```
dados_bruto <- dados %>%
  select(paciente_id, resid_muni = paciente_endereco_nmMunicipio,
         aplicacao_muni = estabelecimento_municipio_nome,
         dt_nasc, dt_aplic, dt_import, ano_aplic,
         mes_aplic, aplic_uf, gest_puerp, idade_anos,
         raca, resid_uf, qual_vacina, num_dose,
         sexo = paciente_enumSexoBiologico
  )

saveRDS(dados_bruto, "dados_vac_select_bruto.rds")
```

Há na base de dados 2925961 observações.

2.10 Sobre tempo entre aplicação e importação no RNDS

A variável `vacina_dataAplicacao` indica a data de aplicação da vacina e a variável `data_importacao_rnds` não está no dicionário, mas estamos entendendo como a data da importação na RNDS (Rede Nacional de

Dados em Saúde). Vamos analisar o tempo entre a data da aplicação e a data da importação no sistema.

```
# tempo_rnds_aplic: tempo entre aplicação e importação no RNDs
dados <- dados %>%
  mutate(
    tempo_rnds_aplic = as.numeric(dt_import - dt_aplic),
  )
```

```
#tempos negativos (data a importação anterior à data da aplicação)
sum(dados$tempo_rnds_aplic<0)
```

```
## [1] 50168
```

Podemos observar que há 50168 casos inconsistentes, em que a data da importação é anterior à data da aplicação.

Segue a distribuição por estado de aplicação com relação ao indicador de tempo negativo:

```
dados <- dados %>%
  mutate(indic_tempo_neg = ifelse(
    is.na(tempo_rnds_aplic),
    NA,
    ifelse(tempo_rnds_aplic < 0, "sim", "não")
  ))

with(dados, ctable(aplic_uf, indic_tempo_neg, prop = "n"))
```

```
## Cross-Tabulation
```

```
## aplic_uf * indic_tempo_neg
```

```
## Data Frame: dados
```

```
##
```

```
## -----
```

	indic_tempo_neg	não	sim	Total
aplic_uf				
AC		7265	0	7265
AL		34183	0	34183
AM		60325	0	60325
AP		6777	0	6777
BA		202120	0	202120
CE		67818	38012	105830

##	DF	44790	0	44790
##	ES	90091	0	90091
##	GO	101027	0	101027
##	MA	83800	0	83800
##	MG	357321	10192	367513
##	MS	21090	0	21090
##	MT	60333	323	60656
##	PA	95585	0	95585
##	PB	74660	0	74660
##	PE	202299	0	202299
##	PI	54164	0	54164
##	PR	243426	1380	244806
##	RJ	212332	0	212332
##	RN	67990	0	67990
##	RO	29151	0	29151
##	RR	1685	0	1685
##	RS	165063	260	165323
##	SC	96670	1	96671
##	SE	36057	0	36057
##	SP	444245	0	444245
##	TO	15526	0	15526
##	Total	2875793	50168	2925961
##	-----	-----	-----	-----

E por UF de residência:

```
with(dados, ctable(resid_uf, indic_tempo_neg, prop="n"))
```

```
## Cross-Tabulation
## resid_uf * indic_tempo_neg
## Data Frame: dados
##
## -----
```

##	indic_tempo_neg	não	sim	Total
## resid_uf				
## AC		7726	0	7726
## AL		36438	18	36456
## AM		59506	20	59526

##	AP	7028	2	7030
##	BA	207595	68	207663
##	CE	71489	36620	108109
##	DF	45792	24	45816
##	ES	87744	24	87768
##	GO	99249	49	99298
##	MA	81331	30	81361
##	MG	356470	10004	366474
##	MS	22536	11	22547
##	MT	61034	338	61372
##	PA	96235	42	96277
##	PB	77186	54	77240
##	PE	197068	88	197156
##	PI	53790	60	53850
##	PR	236090	1395	237485
##	RJ	209820	176	209996
##	RN	67651	88	67739
##	RO	29411	13	29424
##	RR	2431	2	2433
##	RS	165839	288	166127
##	SC	100339	41	100380
##	SE	35795	11	35806
##	SP	433922	396	434318
##	TO	16763	10	16773
##	<NA>	9515	296	9811
##	Total	2875793	50168	2925961
##	-----	-----	-----	-----

Se filtrarmos apenas os casos com tempo positivo, temos as seguintes medidas descritivas do tempo de atraso de importação por UF de aplicação:

```
dados_tempo_pos <- dados %>%
  filter(tempo_rnds_aplic >= 0)

datasummary((aplic_uf) ~ tempo_rnds_aplic*(n+faltantes+media+DP+mediana+minimo+maximo),
  data = dados_tempo_pos, output = 'markdown')
```

aplic_uf	n	faltantes	media	DP	mediana	minimo	maximo
AC	7265.00	0.00	14.41	26.85	6.00	0.00	459.00
AL	34183.00	0.00	11.76	46.47	0.00	0.00	728.00
AM	60325.00	0.00	21.17	56.86	4.00	0.00	806.00
AP	6777.00	0.00	31.82	56.45	12.00	0.00	689.00
BA	202120.00	0.00	27.26	58.38	6.00	0.00	835.00
CE	67818.00	0.00	169.61	253.79	28.00	0.00	846.00
DF	44790.00	0.00	9.44	34.89	0.00	0.00	815.00
ES	90091.00	0.00	8.21	41.04	1.00	0.00	765.00
GO	101027.00	0.00	8.89	32.45	1.00	0.00	732.00
MA	83800.00	0.00	12.20	35.14	1.00	0.00	742.00
MG	357321.00	0.00	17.46	48.15	2.00	0.00	844.00
MS	21090.00	0.00	14.71	46.50	2.00	0.00	771.00
MT	60333.00	0.00	8.10	38.17	0.00	0.00	3295.00
PA	95585.00	0.00	21.96	50.69	4.00	0.00	825.00
PB	74660.00	0.00	18.65	59.57	1.00	0.00	773.00
PE	202299.00	0.00	17.68	39.76	4.00	0.00	756.00
PI	54164.00	0.00	13.84	41.57	1.00	0.00	736.00
PR	243426.00	0.00	8.05	38.48	0.00	0.00	833.00
RJ	212332.00	0.00	18.31	47.10	0.00	0.00	852.00
RN	67990.00	0.00	9.17	35.55	1.00	0.00	751.00
RO	29151.00	0.00	7.75	34.19	0.00	0.00	702.00
RR	1685.00	0.00	99.55	92.43	92.00	0.00	398.00
RS	165063.00	0.00	7.92	29.30	0.00	0.00	675.00
SC	96670.00	0.00	11.32	48.00	0.00	0.00	720.00
SE	36057.00	0.00	31.23	69.30	2.00	0.00	685.00
SP	444245.00	0.00	17.68	59.03	0.00	0.00	913.00
TO	15526.00	0.00	6.81	32.32	0.00	0.00	762.00

E por UF de residência:

```
datasummary((resid_uf) ~ tempo_rnds_aplic*(n+faltantes+media+DP+mediana+minimo+maximo),
             data = dados_tempo_pos, output = 'markdown')
```

resid_uf	n	faltantes	media	DP	mediana	minimo	maximo
AC	7726.00	0.00	14.84	31.67	6.00	0.00	759.00
AL	36438.00	0.00	12.25	47.12	0.00	0.00	733.00
AM	59506.00	0.00	21.30	57.36	4.00	0.00	806.00
AP	7028.00	0.00	31.16	58.06	11.00	0.00	745.00
BA	207595.00	0.00	26.33	57.84	5.00	0.00	835.00
CE	71489.00	0.00	156.62	247.14	21.00	0.00	846.00
DF	45792.00	0.00	10.17	36.88	0.00	0.00	815.00
ES	87744.00	0.00	8.64	41.53	1.00	0.00	765.00
GO	99249.00	0.00	9.56	35.09	1.00	0.00	755.00
MA	81331.00	0.00	12.25	35.92	1.00	0.00	742.00
MG	356470.00	0.00	17.52	48.81	2.00	0.00	844.00
MS	22536.00	0.00	14.77	46.98	1.00	0.00	771.00
MT	61034.00	0.00	8.39	39.32	0.00	0.00	3295.00
PA	96235.00	0.00	21.48	50.38	4.00	0.00	825.00
PB	77186.00	0.00	18.50	59.17	1.00	0.00	773.00
PE	197068.00	0.00	17.84	41.04	4.00	0.00	766.00
PI	53790.00	0.00	14.19	43.40	1.00	0.00	744.00
PR	236090.00	0.00	8.34	39.30	0.00	0.00	833.00
RJ	209820.00	0.00	18.32	48.35	0.00	0.00	852.00
RN	67651.00	0.00	9.50	37.20	1.00	0.00	768.00
RO	29411.00	0.00	7.92	34.24	0.00	0.00	702.00
RR	2431.00	0.00	68.71	90.07	8.00	0.00	706.00
RS	165839.00	0.00	8.05	30.19	0.00	0.00	766.00
SC	100339.00	0.00	11.67	49.04	0.00	0.00	766.00
SE	35795.00	0.00	31.07	69.17	2.00	0.00	705.00
SP	433922.00	0.00	17.69	59.02	0.00	0.00	913.00
TO	16763.00	0.00	7.46	35.43	0.00	0.00	762.00

3 Exclusão dos casos inconsistentes

3.1 Dados com status “entered-in-error”

Há na base de dados a variável **status** que tem as categorias “entered-in-error” e “final”. Quando **status** é “entered-in-error”, a variável **dt_deleted** informa uma data (caso contrário, o campo é vazio).

```
#tabela de frequência para status
questionr::freq(
  dados$status,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%
  kable(caption = "Tabela de frequências para status",
        digits = 2, format = "markdown")
```

Table 13: Tabela de frequências para status

	n	%	val%
entered-in-error	66370	2.3	2.3
final	2859591	97.7	97.7
Total	2925961	100.0	100.0

Vamos excluir os casos “entered-in-error” por entender que esses casos foram identificados com algum erro e não deveriam ser considerados na base de dados.

```
dados1 <- dados %>%
  filter(status != "entered-in-error")
```

3.2 ID do paciente que se repete para um mesmo número de dose e para o mesmo estabelecimento de saúde

Há na bases de dados a coluna `paciente_id`. Vamos agora identificar os casos duplicados com relação à combinação `paciente_id`, data de nascimento, `estabelecimento_valor` (pensando na hipótese de poder repetir o `paciente_id` entre os estabelecimentos) e `vacina_descricao_dose` (número da dose).

```
dados_dup_id <- janitor::get_dupes(dados1, paciente_id, dt_nasc,
                                   estabelecimento_valor, vacina_descricao_dose)
dim(dados_dup_id)[1]
```

```
## [1] 49467
```

Podemos observar que há 49467 observações duplicadas (observações que não são únicas). Vamos filtrar então só os casos únicos de combinação de `paciente_id`, data de nascimento, `estabelecimento_valor` e

vacina_descricao_dose (número da dose).

```
dados2 <-  
  dplyr::distinct(dados1,  
                  paciente_id,  
                  dt_nasc,  
                  estabelecimento_valor,  
                  vacina_descricao_dose,  
                  .keep_all = TRUE)
```

Ficamos agora com 2834476 observações.

3.3 Ano de aplicação da vacina

Agora vamos analisar a data da aplicação da vacina.

```
# with(dados2, ctable(mes_aplic, ano_aplic, round.digits = 0))
```

Veja que há 53 casos com data de aplicação em 2020 e 1 em 2019.

No que segue vamos excluir esses casos de vacinação em 2019 e 2020.

```
# vamos excluir os casos de vacinação em 2020 e em 2019  
dados3 <- dados2 %>%  
  filter(ano_aplic %in% c(2021, 2022, 2023))
```

Ficamos agora com 2834420 observações.

A variável que indica o sexo biológico é `paciente_enumSexoBiologico`. Vamos avaliar a frequência desta variável.

```
#tabela de frequência para sexo  
questionr::freq(  
  dados3$paciente_enumSexoBiologico,  
  cum = FALSE,  
  total = TRUE,  
  na.last = TRUE,  
  valid = TRUE  
) %>%  
  kable(caption = "Tabela de frequências para variável  
          sexo", digits = 2, format = "markdown")
```

Table 14: Tabela de frequências para variável sexo

	n	%	val%
	184	0	0
F	2806065	99	99
M	28171	1	1
Total	2834420	100	100

Note que há 28355 observações que não são sexo biológico feminino.

No que segue vamos excluir esses casos.

```
# vamos excluir os casos de sexo biológico não feminino
dados4 <- dados3 %>%
  filter(paciente_enumSexoBiologico == "F")
```

Ficamos agora com 2806065 observações.

A próxima seleção é avaliar a idade das gestantes.

```
#medidas descritiva da idade
datasummary((idade_anos) ~ 1 * (n + faltantes + media + DP + mediana + minimo +
                                maximo),
  data = dados4,
  output = 'markdown'
)
```

	n	faltantes	media	DP	mediana	minimo	maximo
idade_anos	2806065.00	0.00	28.50	6.71	28.00	0.00	114.00

Além da idade apresentada de forma quantitativa, podemos criar uma idade em categorias: até 9 anos, de 10 até 55 e maior que 55 anos.

```
dados4 <- dados4 %>%
  mutate(
    idade_cat = case_when(
      idade_anos < 10 ~ "< 10 anos",
      idade_anos >= 10 & idade_anos <= 55 ~ "de 10 a 55 anos",
      TRUE ~ "> 55 anos"
```

```

    )
  )

#tabela de frequência para categoria de idade
questionr::freq(
  dados4$idade_cat,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%
  kable(caption = "Tabela de frequências para categoria de idade",
        digits = 2, format = "markdown")

```

Table 16: Tabela de frequências para categoria de idade

	n	%	val%
< 10 anos	422	0.0	0.0
> 55 anos	4147	0.1	0.1
de 10 a 55 anos	2801496	99.8	99.8
Total	2806065	100.0	100.0

Vamos filtrar de 10 a 55 anos no que segue.

```

# vamos excluir os casos <10 e >55 anos
dados5 <- dados4 %>%
  filter(idade_anos >= 10 & idade_anos <= 55)

```

Com essas seleções ficamos com 2801496 observações.

```

#tabela de frequência para grupo de atendimento
questionr::freq(
  dados5$gest_puerp,
  cum = FALSE,
  total = TRUE,
  na.last = TRUE,
  valid = TRUE
) %>%

```



```
kable(caption = "Tabela de frequências para grupo de atendimento",
      digits = 2, format = "markdown")
```

Table 17: Tabela de frequências para grupo de atendimento

	n	%	val%
gesta	2170577	77.5	77.5
puerp	630919	22.5	22.5
Total	2801496	100.0	100.0

Por fim, no total, excluimos 124465 observações que identificamos todas as inconsistências anteriormente descritas.

Apresentamos agora a distribuição das vacinas por UF de aplicação:

```
with(dados5, ctable(aplic_uf, num_dose, prop="n"))
```

```
## Cross-Tabulation
```

```
## aplic_uf * num_dose
```

```
## Data Frame: dados5
```

```
##
```

```
## -----
```

	num_dose	1a dose	1ª Dose Revacinação	2a dose	2ª Dose Revacinação	2º Reforço
## aplic_uf						
## AC	0	2629	11	2458	4	3
## AL	0	13250	1	11616	5	10
## AM	0	24394	25	17082	24	20
## AP	0	2459	6	2075	6	2
## BA	3	68565	168	62254	343	155
## CE	0	40754	20	34221	44	34
## DF	0	15283	20	14889	15	16
## ES	0	33370	0	30604	0	39
## GO	0	38424	44	35014	43	34
## MA	0	32106	32	28711	23	44
## MG	0	136583	214	127135	153	79
## MS	0	8057	14	7328	7	4
## MT	0	24906	61	22073	48	12
## PA	0	35904	51	32459	62	42

##	PB	0	25899	39	24180	46	28
##	PE	1	79750	144	71423	66	26
##	PI	0	17286	31	16119	33	56
##	PR	0	86610	254	81629	400	69
##	RJ	0	73230	71	66993	107	138
##	RN	0	24608	0	20131	0	35
##	RO	0	12046	11	10655	18	9
##	RR	0	623	2	553	1	4
##	RS	0	60467	69	55531	178	75
##	SC	0	37151	42	34835	42	23
##	SE	0	13677	12	11422	6	23
##	SP	0	124883	0	124492	0	358
##	TO	0	6135	15	5601	2	14
##	Total	4	1039049	1357	951483	1676	1344
##	-----	-----	-----	-----	-----	-----	-----

Apresentamos agora a distribuição das vacinas por UF de residência:

```
with(dados5, ctable(resid_uf, num_dose, prop="n"))
```

```
## Cross-Tabulation
```

```
## resid_uf * num_dose
```

```
## Data Frame: dados5
```

```
##
```

```
## -----
```

##	num_dose	1a dose	1ª Dose Revacinação	2a dose	2ª Dose Revacinação	2º Reforço
----	----------	---------	---------------------	---------	---------------------	------------

```
## resid_uf
```

##	AC	0	2819	11	2580	3	38
----	----	---	------	----	------	---	----

##	AL	0	14043	3	12355	7	10
----	----	---	-------	---	-------	---	----

##	AM	0	23931	25	16886	24	20
----	----	---	-------	----	-------	----	----

##	AP	0	2541	6	2168	6	2
----	----	---	------	---	------	---	---

##	BA	3	70088	164	63877	327	158
----	----	---	-------	-----	-------	-----	-----

##	CE	0	41444	22	34874	48	36
----	----	---	-------	----	-------	----	----

##	DF	0	16167	20	15216	15	16
----	----	---	-------	----	-------	----	----

##	ES	0	32437	1	29955	1	37
----	----	---	-------	---	-------	---	----

##	GO	0	37332	45	34447	48	33
----	----	---	-------	----	-------	----	----

##	MA	0	31011	33	27715	22	43
----	----	---	-------	----	-------	----	----

##	MG	0	135620	208	126437	154	84
----	----	---	--------	-----	--------	-----	----

##	MS	0	8572	15	7790	7	5
##	MT	0	25045	64	22341	45	12
##	PA	0	36251	51	32548	60	42
##	PB	0	26744	40	25068	49	30
##	PE	1	77163	131	69334	64	29
##	PI	0	17226	28	16004	27	54
##	PR	0	83970	253	79205	391	69
##	RJ	0	72531	71	66307	101	135
##	RN	0	24503	1	20090	1	34
##	RO	0	12081	11	10738	18	9
##	RR	0	940	1	809	1	4
##	RS	0	60537	66	55810	178	75
##	SC	0	38681	45	35975	49	24
##	SE	0	13502	12	11337	7	23
##	SP	0	123826	12	122673	13	342
##	TO	0	6664	15	5991	4	2
##	<NA>	0	3380	3	2953	6	4
##	Total	4	1039049	1357	951483	1676	1344
##	-----	-----	-----	-----	-----	-----	-----

```
dados6 <- dados5 %>%
```

```
select(
  paciente_id,
  dt_nasc,
  dt_aplic,
  dt_import,
  ano_aplic,
  mes_aplic,
  gest_puerp,
  idade_anos,
  raca,
  resid_uf,
  resid_muni,
  paciente_endereco_nmMunicipio,
  qual_vacina,
  num_dose,
  aplic_uf,
```

```
    aplic_muni,  
    aplicacao_muni = estabelecimento_municipio_nome  
  )  
  
saveRDS(dados6, file = "dados_vac_select.rds")
```