

Análises do SIVEP-Gripe

Gestantes e puérperas

09/fevereiro/2022

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/covid_gesta_puerp_br.

```
#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",
    "writexl",
    "modelsummary"
  )
lapply(packages, loadlibrary)
```

A base de dados SIVEP-Gripe (Sistema de Informação da Vigilância Epidemiológica da Gripe) tem os registros dos casos e óbitos de SRAG (Síndrome Respiratória Aguda Grave). A notificação é compulsória para síndrome gripal (caracterizado por pelo menos dois dos seguintes sinais e sintomas: febre, mesmo que referida, calafrios, dor de garganta, dor de cabeça, tosse, coriza, distúrbios olfatórios ou de paladar) e que tem dispneia / desconforto respiratório ou pressão persistente no peito ou Saturação de O₂ menor que 95% no ar ambiente ou cor azulada dos lábios ou rosto. Indivíduos assintomáticos com confirmação laboratorial por biologia molecular ou exame imunológico para infecção por COVID-19 também são relatados.

Para notificações no Sivep-Gripe, os casos hospitalizados em hospitais públicos e privados e todas as mortes devido a infecções respiratórias agudas graves, independentemente da hospitalização, devem ser considerados.

A vigilância da SRAG no Brasil é desenvolvida pelo Ministério da Saúde (MS), por meio da Secretaria de Vigilância em Saúde (SVS), desde a pandemia de Influenza A (H1N1) em 2009. Mais informações em <https://coronavirus.saude.gov.br/definicao-de-caso-e-notificacao>.

Os dados do ano de 2020, 2021 e 2022 foram obtidos em 26/Janeiro/2022 no site <https://opendatasus.saude.gov.br>. Os dados de 2020, 2021 e 2022 são carregados e combinados abaixo:

```
##### carregando as bases de dados #####
ckanr::ckanr_setup("https://opendatasus.saude.gov.br")

arqs <- ckanr::package_search("srag 2020")$results %>%
  purrr::map("resources") %>%
  purrr::map(purrr::keep, ~ .x$mimetype == "text/csv") %>%
  purrr::map_chr(purrr::pluck, 1, "url")

arqs2 <- ckanr::package_search("srag 2021")$results %>%
  purrr::map("resources") %>%
  purrr::map(purrr::keep, ~ .x$mimetype == "text/csv") %>%
  purrr::map_chr(purrr::pluck, 2, "url")

dados_a <- fread(arqs[1], sep = ";")

dados_b <- fread(arqs[2], sep = ";")

dados_c <- fread(arqs2[1], sep = ";")

# base de dados semanas epidemiologicas ----
semep_2020 <-
  tabulizer::extract_tables(paste0(getwd(), "/semanas epidemiologicas/2020.pdf"))
semep_2020 <- semep_2020[[1]]
semep_2020 <- as.data.frame(janitor::row_to_names(semep_2020, 1))

semep_2021 <-
  tabulizer::extract_tables(paste0(getwd(), "/semanas epidemiologicas/2021.pdf"))
semep_2021 <- semep_2021[[1]]
semep_2021 <- as.data.frame(janitor::row_to_names(semep_2021, 1))

semep_2022 <-
  tabulizer::extract_tables(paste0(getwd(), "/semanas epidemiologicas/2022.pdf"))
semep_2022 <- semep_2022[[1]]
semep_2022 <- as.data.frame(janitor::row_to_names(semep_2022, 1))

semep <- semep_2020 %>%
  bind_rows(semep_2021) %>%
  bind_rows(semep_2022) %>%
  janitor::clean_names() %>%
  mutate(inicio = lubridate::dmy(inicio),
         termino = lubridate::dmy(termino))

sem <-
  as.numeric(semep$semana[Sys.Date() >= semep$inicio &
    Sys.Date() <= semep$termino]) - 1
```

```
#### Concatenar dados 2020, 2021 e 2022 #####
dados_a <- dados_a %>%
  mutate(FATOR_RISC = case_when(FATOR_RISC == 1 ~ "S",
                                FATOR_RISC == 2 ~ "N"))

dados_b <- dados_b %>%
  mutate(FATOR_RISC = case_when(FATOR_RISC == 1 ~ "S",
                                FATOR_RISC == 2 ~ "N"))

dados_c <- dados_c %>%
  mutate(FATOR_RISC = case_when(FATOR_RISC == 1 ~ "S",
                                FATOR_RISC == 2 ~ "N"))

# COD_IDADE de 2022 para character para padronizar com 2020 e 2021

dados_c$COD_IDADE <- as.character(dados_c$COD_IDADE)

dados1 <- dados_a %>%
  full_join(dados_b) %>%
  full_join(dados_c)

#Criar variavel de ano do caso
dados1 <- dados1 %>%
  dplyr::mutate(
    dt_sint = as.Date(DT_SIN_PRI, format = "%d/%m/%Y"),
    dt_nasc = as.Date(DT_NASC, format = "%d/%m/%Y"),
    ano = lubridate::year(dt_sint),
  )
```

Para acessar os pdf's com informações sobre as semanas epidemiológicas acesse: <http://portalsinan.saude.gov.br/calendario-epidemiologico-2020/43-institucional/171-calendario-epidemiologico-2021>.

Há atualmente 3012568 observações na base de dados e são as variáveis:

```
names(dados1)
```

```
## [1] "DT_NOTIFIC" "SEM_NOT" "DT_SIN_PRI" "SEM_PRI" "SG_UF_NOT"
## [6] "ID_REGIONA" "CO_REGIONA" "ID_MUNICIP" "CO_MUN_NOT" "ID_UNIDADE"
## [11] "CO_UNI_NOT" "CS_SEXO" "DT_NASC" "NU_IDADE_N" "TP_IDADE"
## [16] "COD_IDADE" "CS_GESTANT" "CS_RACA" "CS_ESCOL_N" "ID_PAIS"
## [21] "CO_PAIS" "SG_UF" "ID_RG_RESI" "CO_RG_RESI" "ID_MN_RESI"
## [26] "CO_MUN_RES" "CS_ZONA" "SURTO_SG" "NOSOCOMIAL" "AVE_SUINO"
## [31] "FEBRE" "TOSSE" "GARGANTA" "DISPNEIA" "DESC_RESP"
## [36] "SATURACAO" "DIARREIA" "VOMITO" "OUTRO_SIN" "OUTRO_DES"
## [41] "PUERPERA" "FATOR_RISC" "CARDIOPATI" "HEMATOLOGI" "SIND_DOWN"
## [46] "HEPATICA" "ASMA" "DIABETES" "NEUROLOGIC" "PNEUMOPATI"
## [51] "IMUNODEPRE" "RENAL" "OBESIDADE" "OBES_IMC" "OUT_MORBI"
## [56] "MORB_DESC" "VACINA" "DT_UT_DOSE" "MAE_VAC" "DT_VAC_MAE"
## [61] "M_AMAMENTA" "DT_DOSEUNI" "DT_1_DOSE" "DT_2_DOSE" "ANTIVIRAL"
## [66] "TP_ANTIVIR" "OUT_ANTIV" "DT_ANTIVIR" "HOSPITAL" "DT_INTERNA"
## [71] "SG_UF_INTE" "ID_RG_INTE" "CO_RG_INTE" "ID_MN_INTE" "CO_MU_INTE"
## [76] "UTI" "DT_ENTUTI" "DT_SAIDUTI" "SUPOORT_VEN" "RAIOX_RES"
## [81] "RAIOX_OUT" "DT_RAIOX" "AMOSTRA" "DT_COLETA" "TP_AMOSTRA"
## [86] "OUT_AMOST" "PCR_RESUL" "DT_PCR" "POS_PCRFLU" "TP_FLU_PCR"
## [91] "PCR_FLUASU" "FLUASU_OUT" "PCR_FLUBLI" "FLUBLI_OUT" "POS_PCROUT"
```

```
## [96] "PCR_VSR"      "PCR_PARA1"    "PCR_PARA2"    "PCR_PARA3"    "PCR_PARA4"
## [101] "PCR_ADENO"    "PCR_METAP"    "PCR_BOCA"     "PCR_RINO"     "PCR_OUTRO"
## [106] "DS_PCR_OUT"   "CLASSI_FIN"   "CLASSI_OUT"   "CRITERIO"     "EVOLUCAO"
## [111] "DT_EVOLUCA"   "DT_ENCERRA"   "DT_DIGITA"    "HISTO_VGM"    "PAIS_VGM"
## [116] "CO_PS_VGM"    "LO_PS_VGM"    "DT_VGM"       "DT_RT_VGM"    "PCR_SARS2"
## [121] "PAC_COCBO"    "PAC_DSCBO"    "OUT_ANIM"     "DOR_ABD"      "FADIGA"
## [126] "PERD_OLFT"    "PERD_PALA"    "TOMO_RES"     "TOMO_OUT"     "DT_TOMO"
## [131] "TP_TES_AN"    "DT_RES_AN"    "RES_AN"       "POS_AN_FLU"   "TP_FLU_AN"
## [136] "POS_AN_OUT"   "AN_SARS2"     "AN_VSR"       "AN_PARA1"     "AN_PARA2"
## [141] "AN_PARA3"     "AN_ADENO"     "AN_OUTRO"     "DS_AN_OUT"    "TP_AM_SOR"
## [146] "SOR_OUT"      "DT_CO_SOR"    "TP_SOR"       "OUT_SOR"      "DT_RES"
## [151] "RES_IGG"      "RES_IGM"      "RES_IGA"      "ESTRANG"      "VACINA_COV"
## [156] "DOSE_1_COV"   "DOSE_2_COV"   "LAB_PR_COV"   "LOTE_1_COV"   "LOTE_2_COV"
## [161] "FNT_IN_COV"   "dt_sint"      "dt_nasc"      "ano"
```

#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)
```

Filtragem e tratamento dos dados

A variável que indica a classificação é a `CLASSI_FIN`, com as seguintes categorias: 1-SRAG por influenza, 2-SRAG por outro vírus respiratório, 3-SRAG por outro agente etiológico, 4-SRAG não especificado e 5-SRAG por COVID-19.

#tabela de frequência para a classificação

```
questionr::freq(
  dados1$CLASSI_FIN,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
```

```
kable(caption = "Tabela de frequências para classificação do caso ",
      digits = 2)
```

Table 1: Tabela de frequências para classificação do caso

	n	%
1	16214	0.5
2	24545	0.8
3	8449	0.3
4	816101	27.1
5	1937416	64.3
NA	209843	7.0
Total	3012568	100.0

```
#codificar campo em branco para classificação como 9
dados1$CLASSI_FIN <-
  ifelse(is.na(dados1$CLASSI_FIN) == TRUE, 9, dados1$CLASSI_FIN)
```

Vamos ver a distribuição dos casos por ano e por semana epidemiológica.

```
table(dados1$SEM_PRI, dados1$ano)
```

```
##
##      2019  2020  2021  2022
##  1    144   304 36395 23286
##  2      0   461 35166 27217
##  3      0   470 32331 25663
##  4      0   426 30345 16816
##  5      0   505 36604  5287
##  6      0   507 39881    28
##  7      0   632 50514     0
##  8      0   917 53917     0
##  9      0  1160 75969     0
## 10      0  1990 76883     0
## 11      0  5151 76509     0
## 12      0 12811 61713     0
## 13      0 14972 59007     0
## 14      0 16331 56278     0
## 15      0 19613 48775     0
## 16      0 24900 48869     0
## 17      0 30881 51192     0
## 18      0 34959 51822     0
## 19      0 34671 59411     0
## 20      0 37210 58260     0
## 21      0 33835 55990     0
## 22      0 31306 54816     0
## 23      0 35675 47058     0
## 24      0 34203 40742     0
## 25      0 36792 36102     0
## 26      0 33018 33350     0
## 27      0 37582 31026     0
```

```
## 28 0 37166 26670 0
## 29 0 34575 24872 0
## 30 0 33821 21760 0
## 31 0 32332 23457 0
## 32 0 30163 21012 0
## 33 0 31199 19560 0
## 34 0 28406 16724 0
## 35 0 26450 16204 0
## 36 0 26633 15312 0
## 37 0 24162 14334 0
## 38 0 22334 14052 0
## 39 0 21698 13148 0
## 40 0 22558 11219 0
## 41 0 21116 11480 0
## 42 0 19124 10741 0
## 43 0 19571 9959 0
## 44 0 18835 10165 0
## 45 0 23462 9239 0
## 46 0 25989 9888 0
## 47 0 29507 8905 0
## 48 0 29494 9590 0
## 49 0 33446 10253 0
## 50 0 31105 11545 0
## 51 0 29029 15634 0
## 52 0 31171 15126 4526
## 53 0 22137 13062 0
```

Note que há 13062 casos em 2021 na semana 53. Estes são casos dos dois primeiros dias de 2021, que ainda fazem parte da última semana epidemiológica de 2020 (<http://portalsinan.saude.gov.br/calendario-epidemiologico?layout=edit&id=168>). No entanto, esses casos pertencem à 53ª semana de 2020 e corrigimos no que segue:

```
dados2 <- dados1 %>%
  mutate(ano = ifelse(ano == 2021 & SEM_PRI == 53, 2020, ano))
```

Note também que há 4526 casos em 2022 na semana 52. Estes são casos do primeiro dia de 2022, que ainda fazem parte da última semana epidemiológica de 2021 (<http://portalsinan.saude.gov.br/calendario-epidemiologico?layout=edit&id=168>). No entanto, esses casos pertencem à 52ª semana de 2021 e corrigimos no que segue:

```
dados2 <- dados2 %>%
  mutate(ano = ifelse(ano == 2022 & SEM_PRI == 52, 2021, ano)) %>%
  #só os casos da semana fechada atual
  filter(ano == 2020 | ano == 2021 | (ano == 2022 & SEM_PRI <= sem))
```

```
table(dados2$SEM_PRI, dados2$ano)
```

```
##
##      2020  2021  2022
## 1      304 36395 23286
## 2      461 35166 27217
## 3      470 32331 25663
```

##	4	426	30345	16816
##	5	505	36604	5287
##	6	507	39881	0
##	7	632	50514	0
##	8	917	53917	0
##	9	1160	75969	0
##	10	1990	76883	0
##	11	5151	76509	0
##	12	12811	61713	0
##	13	14972	59007	0
##	14	16331	56278	0
##	15	19613	48775	0
##	16	24900	48869	0
##	17	30881	51192	0
##	18	34959	51822	0
##	19	34671	59411	0
##	20	37210	58260	0
##	21	33835	55990	0
##	22	31306	54816	0
##	23	35675	47058	0
##	24	34203	40742	0
##	25	36792	36102	0
##	26	33018	33350	0
##	27	37582	31026	0
##	28	37166	26670	0
##	29	34575	24872	0
##	30	33821	21760	0
##	31	32332	23457	0
##	32	30163	21012	0
##	33	31199	19560	0
##	34	28406	16724	0
##	35	26450	16204	0
##	36	26633	15312	0
##	37	24162	14334	0
##	38	22334	14052	0
##	39	21698	13148	0
##	40	22558	11219	0
##	41	21116	11480	0
##	42	19124	10741	0
##	43	19571	9959	0
##	44	18835	10165	0
##	45	23462	9239	0
##	46	25989	9888	0
##	47	29507	8905	0
##	48	29494	9590	0
##	49	33446	10253	0
##	50	31105	11545	0
##	51	29029	15634	0
##	52	31171	19652	0
##	53	35199	0	0

O próximo passo é identificar as pessoas gestantes. Para isso, vamos analisar a variável `CS_GESTANT`. Essa variável assume os valores: 1-1º Trimestre; 2-2º Trimestre; 3-3º Trimestre; 4-Idade Gestacional Ignorada; 5-Não; 6-Não se aplica; 9-Ignorado.

```
#tabela de frequência para gestação
questionr::freq(
  dados2$CS_GESTANT,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para variável
    sobre gestação", digits = 2)
```

Table 2: Tabela de frequências para variável sobre gestação

	n	%
0	468	0.0
1	3272	0.1
2	7802	0.3
3	17862	0.6
4	1557	0.1
5	936098	31.1
6	1907111	63.3
9	138226	4.6
Total	3012396	100.0

Há 468 casos com CS_GESTANT=0, em que a categoria 0 não tem código no dicionário.

Vamos ver se há alguma inconsistência ao analisar essa variável conjuntamente com CS_SEX0 (F-feminino, M-masculino e I-ignorado).

```
#tabela de frequência para sexo
questionr::freq(
  dados2$CS_SEX0,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para sexo", digits = 2)
```

Table 3: Tabela de frequências para sexo

	n	%
F	1380291	45.8
I	746	0.0
M	1631359	54.2
Total	3012396	100.0

```
#tabela cruzada para gestação e sexo
table(dados2$CS_GESTANT, dados2$CS_SEX0)
```



```
##
##           F           I           M
##  0       199       171       98
##  1      3272         0         0
##  2      7801         1         0
##  3     17861         1         0
##  4      1556         1         0
##  5    933480        69      2549
##  6    278204       399    1628508
##  9    137918       104       204
```

Veja que há 0 casos de CS_SEXO=M com CS_GESTANT=1,2,3 ou 4, como esperado.

A variável indicadora de puerpério é PUERPERA, com categorias 1-sim, 2-não e 9-Ignorado.

```
#tabela de frequencias para puerpério
questionr::freq(
  dados2$PUERPERA,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para variável indicadora de puérpera",
        digits = 2)
```

Table 4: Tabela de frequências para variável indicadora de puérpera

	n	%
1	10361	0.3
2	1049842	34.9
9	26397	0.9
NA	1925796	63.9
Total	3012396	100.0

```
#tabela cruzada de puerpério e sexo
table(dados2$PUERPERA, dados2$CS_SEX0)
```

```
##
##           F           I           M
##  1    10318         0         43
##  2   507305       191    542346
##  9    12229        16    14152
```

Veja que há 43 casos de CS_SEXO=M com PUERPERA = 1 casos de puérpera e sexo masculino.

A próxima seleção é considerar só as pessoas do sexo feminino:

```
# Filtragem dos casos do sexo feminino
dados3 <- dados2 %>%
  filter(CS_SEX0 == "F")
```

Após essa seleção foram selecionados 1380291 casos.

Agora vamos criar a variável de trimestre gestacional ou puerpério. Veja que para puerpério (**puerp**) são considerados os casos não gestante ou ignorado com **PUERPERA = 1**.

```
#Criação da variável classi_gesta_puerp para o momento gestacional ou puerpério
dados3 <- dados3 %>%
  mutate(
    classi_gesta_puerp = case_when(
      CS_GESTANT == 1 ~ "1tri",
      CS_GESTANT == 2 ~ "2tri",
      CS_GESTANT == 3 ~ "3tri",
      CS_GESTANT == 4 ~ "IG_ig",
      CS_GESTANT == 5 &
        PUERPERA == 1 ~ "puerp",
      CS_GESTANT == 9 & PUERPERA == 1 ~ "puerp",
      TRUE ~ "não"
    )
  )
```

A última filtragem consiste em selecionar os casos de gestantes ou puérperas.

```
#Seleção só dos casos de gestantes ou puérperas
dados4 <- dados3 %>%
  filter(classi_gesta_puerp != "não")
```

A próxima seleção é considerar gestantes e puérperas com idade maior que 10 e menor ou igual a 55 anos.

```
#criando a nossa variável de ano como a diferença entre dt_sint e dt_nasc.
#Nos casos sem dt_nasc, consideramos
#o campo NU_IDADE_N
dados4 <- dados4 %>%
  mutate(
    idade = as.period(interval(start = dt_nasc, end = dt_sint))$year,
    idade_anos = ifelse(is.na(idade), NU_IDADE_N, idade)
  )

# Filtragem dos casos com 55 anos ou menos
dados5 <- dados4 %>%
  filter(idade_anos > 9 & idade_anos <= 55)
```

Ficamos com 37856 casos de gestantes e puérperas, distribuídas nos seguintes grupos de trimestre gestacional e puerpério:

```
#tabela de frequência para grupo gestacional
questionr::freq(
  dados5$classi_gesta_puerp,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para variável
    de trimestre gestacional ou puerpério", digits = 2)
```

Table 5: Tabela de frequências para variável de trimestre gestacional ou puerpério

	n	%
1tri	3239	8.6
2tri	7718	20.4
3tri	17825	47.1
IG_ig	1480	3.9
puerp	7594	20.1
Total	37856	100.0

No que segue tratamos as variáveis consideradas no Observatório Obstétrico Covid-19.

Tipo de diagnóstico:

```
#Caso diagnosticado por PCR
dados5 <- dados5 %>%
  mutate(PCR_SN = case_when(
    (PCR_SARS2 == 1) |
      (
        str_detect(DS_PCR_OUT, "SARS|COVID|COV|CORONA|CIVID")
      ) ~ "sim",
    TRUE ~ "não"
  ))

#Identificar se diagnóstico por sorologia
dados5$res_igg <-
  ifelse(is.na(dados5$RES_IGG) == TRUE, 0, dados5$RES_IGG)

dados5$res_igm <-
  ifelse(is.na(dados5$RES_IGM) == TRUE, 0, dados5$RES_IGM)

dados5$res_iga <-
  ifelse(is.na(dados5$RES_IGA) == TRUE, 0, dados5$RES_IGA)

dados5$sorologia_SN <-
  ifelse(dados5$res_igg == 1 |
    dados5$res_igm == 1 | dados5$res_iga == 1,
    "sim",
    "não")

#Identificar se diagnosticado por antigenio
dados5 <- dados5 %>%
  mutate(antigeno_SN = case_when(
    (AN_SARS2 == 1) | #positivo
      (
        str_detect(DS_AN_OUT, "SARS|COVID|COV|CORONA|CONA")
      ) ~ "sim",
    TRUE ~ "não"
  ))
```

A variável `classi_covid` identifica o tipo de diagnóstico. Essa variável é válida apenas para os casos confirmados de SRAG por COVID-19 (`CLASSI_FIN=5`).

```
#Criação da variável de classificação da covid-19
dados5 <- dados5 %>%
  mutate(
    classi_covid = case_when(
      CLASSI_FIN == 5 & pcr_SN == "sim" ~ "pcr",
      CLASSI_FIN == 5 & pcr_SN == "não" &
        antígeno_SN == "sim" ~ "antigenio",
      CLASSI_FIN == 5 & sorologia_SN == "sim" &
        antígeno_SN == "não" &
        pcr_SN == "não" ~ "sorologia",
      CLASSI_FIN != 5 ~ "não", #não é outro agente etiológico ou não especificado
      TRUE ~ "outro"
    )
  )
```

```
#tabela de frequências para tipo de diagnóstico
questionr::freq(
  dados5$classi_covid,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para o tipo de diagnóstico",
        digits = 2)
```

Table 6: Tabela de frequências para o tipo de diagnóstico

	n	%
antigenio	2987	7.9
não	17263	45.6
outro	4144	10.9
pcr	11865	31.3
sorologia	1597	4.2
Total	37856	100.0

```
ctable(dados5$CLASSI_FIN, dados5$classi_covid, total = FALSE)
```

```
## Cross-Tabulation, Row Proportions
## CLASSI_FIN * classi_covid
## Data Frame: dados5
##
```

```
## -----
##      classi_covid      antigenio      não      outro      pcr      sorol
## CLASSI_FIN
##      1      0 ( 0.0%)      572 (100.0%)      0 ( 0.0%)      0 ( 0.0%)      0 (0
##      2      0 ( 0.0%)      308 (100.0%)      0 ( 0.0%)      0 ( 0.0%)      0 (0
##      3      0 ( 0.0%)      83 (100.0%)      0 ( 0.0%)      0 ( 0.0%)      0 (0
##      4      0 ( 0.0%)      13458 (100.0%)      0 ( 0.0%)      0 ( 0.0%)      0 (0
##      5      2987 (14.5%)      0 ( 0.0%)      4144 (20.1%)      11865 (57.6%)      1597 (7
##      9      0 ( 0.0%)      2842 (100.0%)      0 ( 0.0%)      0 ( 0.0%)      0 (0
## -----
```

Região do Brasil:

```
#Criação da variável de região
regions <- function(state) {
  southeast <- c("SP", "RJ", "ES", "MG")
  south <- c("PR", "SC", "RS")
  central <- c("GO", "MT", "MS", "DF")
  northeast <-
    c("AL", "BA", "CE", "MA", "PB", "PE", "PI", "RN", "SE")
  north <- c("AC", "AP", "AM", "PA", "RO", "RR", "TO")
  out <-
    ifelse(any(state == southeast),
           "southeast",
           ifelse(any(state == south),
                  "south",
                  ifelse(
                    any(state == central),
                    "central",
                    ifelse(any(state == northeast),
                           "northeast", "north")
                  )))
  return(out)
}

dados5$region <- sapply(dados5$SG_UF, regions)
dados5$region <-
  ifelse(is.na(dados5$region) == TRUE, 0, dados5$region)
```

```
#tabela de frequências para região
questionr::freq(
  dados5$region,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para a região do Brasil", digits = 2)
```

Table 7: Tabela de frequências para a região do Brasil

	n	%
central	4050	10.7
north	3816	10.1
northeast	9114	24.1
south	5739	15.2
southeast	15137	40.0
Total	37856	100.0

Veja que há 0 casos sem a informação da região do país (codificado como 0).

No que segue, tratamos as variáveis de caracterização, sintomas, comorbidades, desfechos e variáveis de tempo.

Variáveis de caracterização

```
#Raça
dados5 <- dados5 %>%
  mutate(
    raca = case_when(
      CS_RACA == 1 ~ "branca",
      CS_RACA == 2 ~ "preta",
      CS_RACA == 3 ~ "amarela",
      CS_RACA == 4 ~ "parda",
      CS_RACA == 5 ~ "indigena",
      TRUE ~ NA_character_
    )
  )

#Escolaridade
dados5 <- dados5 %>%
  mutate(
    escol = case_when(
      CS_ESCOL_N == 0 ~ "sem escol",
      CS_ESCOL_N == 1 ~ "fund1",
      CS_ESCOL_N == 2 ~ "fund2",
      CS_ESCOL_N == 3 ~ "medio",
      CS_ESCOL_N == 4 ~ "superior",
      TRUE ~ NA_character_
    )
  )

#Faixa etária
dados5 <- dados5 %>%
  mutate(
    faixa_et = case_when(
      NU_IDADE_N <= 19 ~ "<20",
      NU_IDADE_N >= 20
      & NU_IDADE_N <= 34 ~ "20-34",
      NU_IDADE_N >= 35 ~ ">=35",
      TRUE ~ NA_character_
    )
  )
dados5$faixa_et <-
  factor(dados5$faixa_et, levels = c("<20", "20-34", ">=35"))

#Internação no hospital
dados5 <- dados5 %>%
  mutate(hospital = case_when(HOSPITAL == 1 ~ "sim",
                              HOSPITAL == 2 ~ "não",
                              TRUE ~ NA_character_))

#Histórico de viagem
dados5 <- dados5 %>%
  mutate(hist_viagem = case_when(HISTO_VGM == 1 ~ "sim",
                                  HISTO_VGM == 2 ~ "não",
                                  TRUE ~ NA_character_))
```

```

#Síndrome gripal evoluída para SRAG
dados5 <- dados5 %>%
  mutate(sg_para_srag = case_when(SURTO_SG == 1 ~ "sim",
                                   SURTO_SG == 2 ~ "não",
                                   TRUE ~ NA_character_))

#Infecção adquirida no hospital
dados5 <- dados5 %>%
  mutate(inf_inter = case_when(NOSOCOMIAL == 1 ~ "sim",
                                NOSOCOMIAL == 2 ~ "não",
                                TRUE ~ NA_character_))

#Contato com ave ou suíno
dados5 <- dados5 %>%
  mutate(cont_ave_suino = case_when(AVE_SUINO == 1 ~ "sim",
                                     AVE_SUINO == 2 ~ "não",
                                     TRUE ~ NA_character_))

#Vacina para gripe
dados5 <- dados5 %>%
  mutate(vacina = case_when(VACINA == 1 ~ "sim",
                             VACINA == 2 ~ "não",
                             TRUE ~ NA_character_))

#Antiviral
dados5 <- dados5 %>%
  mutate(
    antiviral = case_when(
      ANTIVIRAL == 1 ~ "Osetamivir",
      ANTIVIRAL == 2 ~ "Zanamivir",
      TRUE ~ NA_character_
    )
  )

#Zona de residência
dados5 <- dados5 %>%
  mutate(zona = case_when(CS_ZONA == 1 ~ "urbana",
                           CS_ZONA == 2 ~ "rural",
                           CS_ZONA == 3 ~ "periurbana",
                           TRUE ~ NA_character_))

#Se mudança de município para atendimento
dados5 <- dados5 %>%
  mutate(mudou_muni = case_when((CO_MUN_RES == CO_MU_INTE) &
                                  !is.na(CO_MU_INTE) &
                                  !is.na(CO_MUN_RES) ~ "não",
                                  (CO_MUN_RES != CO_MU_INTE) &
                                  !is.na(CO_MU_INTE) &
                                  !is.na(CO_MUN_RES) ~ "sim",
                                  TRUE ~ NA_character_
  )
)

```

Sintomas

```
#Febre
dados5 <- dados5 %>%
  mutate(febre = case_when(FEBRE == 1 ~ "sim",
                           FEBRE == 2 ~ "não",
                           TRUE ~ NA_character_))

#Tosse
dados5 <- dados5 %>%
  mutate(tosse = case_when(TOSSE == 1 ~ "sim",
                           TOSSE == 2 ~ "não",
                           TRUE ~ NA_character_))

#Garganta
dados5 <- dados5 %>%
  mutate(garganta = case_when(GARGANTA == 1 ~ "sim",
                              GARGANTA == 2 ~ "não",
                              TRUE ~ NA_character_))

#Dispneia
dados5 <- dados5 %>%
  mutate(dispnéia = case_when(DISPNEIA == 1 ~ "sim",
                              DISPNEIA == 2 ~ "não",
                              TRUE ~ NA_character_))

#Desconforto respiratório
dados5 <- dados5 %>%
  mutate(desc_resp = case_when(DESC_RESP == 1 ~ "sim",
                              DESC_RESP == 2 ~ "não",
                              TRUE ~ NA_character_))

#Saturação
dados5 <- dados5 %>%
  mutate(saturacao = case_when(SATURACAO == 1 ~ "sim",
                              SATURACAO == 2 ~ "não",
                              TRUE ~ NA_character_))

#Diarreia
dados5 <- dados5 %>%
  mutate(diarreia = case_when(DIARREIA == 1 ~ "sim",
                              DIARREIA == 2 ~ "não",
                              TRUE ~ NA_character_))

#Vômito
dados5 <- dados5 %>%
  mutate(vomito = case_when(VOMITO == 1 ~ "sim",
                            VOMITO == 2 ~ "não",
                            TRUE ~ NA_character_))

#Dor abdominal
dados5 <- dados5 %>%
  mutate(dor_abd = case_when(DOR_ABD == 1 ~ "sim",
```



```

DOR_ABD == 2 ~ "não",
TRUE ~ NA_character_))

#Fadiga
dados5 <- dados5 %>%
  mutate(fadiga = case_when(FADIGA == 1 ~ "sim",
                             FADIGA == 2 ~ "não",
                             TRUE ~ NA_character_))

#Perda olfativa
dados5 <- dados5 %>%
  mutate(perd_olft = case_when(PERD_OLFT == 1 ~ "sim",
                                PERD_OLFT == 2 ~ "não",
                                TRUE ~ NA_character_))

#Perda do paladar
dados5 <- dados5 %>%
  mutate(perd_pala = case_when(PERD_PALA == 1 ~ "sim",
                                PERD_PALA == 2 ~ "não",
                                TRUE ~ NA_character_))

```

Comorbidades

```

#Cardiopatia
dados5 <- dados5 %>%
  mutate(cardiopati = case_when(CARDIOPATI == 1 ~ "sim",
                                 CARDIOPATI == 2 ~ "não",
                                 TRUE ~ NA_character_))

#Hematológica
dados5 <- dados5 %>%
  mutate(hematologi = case_when(HEMATOLOGI == 1 ~ "sim",
                                 HEMATOLOGI == 2 ~ "não",
                                 TRUE ~ NA_character_))

#Hepática
dados5 <- dados5 %>%
  mutate(hepatica = case_when(HEPATICA == 1 ~ "sim",
                                HEPATICA == 2 ~ "não",
                                TRUE ~ NA_character_))

#Asma
dados5 <- dados5 %>%
  mutate(asma = case_when(ASMA == 1 ~ "sim",
                           ASMA == 2 ~ "não",
                           TRUE ~ NA_character_))

#Diabetes
dados5 <- dados5 %>%
  mutate(diabetes = case_when(DIABETES == 1 ~ "sim",
                               DIABETES == 2 ~ "não",
                               TRUE ~ NA_character_))

```

```

#Neurológica
dados5 <- dados5 %>%
  mutate(neuro = case_when(NEUROLOGIC == 1 ~ "sim",
                           NEUROLOGIC == 2 ~ "não",
                           TRUE ~ NA_character_))

#Pneumopatia
dados5 <- dados5 %>%
  mutate(pneumopati = case_when(PNEUMOPATI == 1 ~ "sim",
                                PNEUMOPATI == 2 ~ "não",
                                TRUE ~ NA_character_))

#Imunossupressão
dados5 <- dados5 %>%
  mutate(imunodepre = case_when(IMUNODEPRE == 1 ~ "sim",
                                IMUNODEPRE == 2 ~ "não",
                                TRUE ~ NA_character_))

#Renal
dados5 <- dados5 %>%
  mutate(renal = case_when(RENAL == 1 ~ "sim",
                           RENAL == 2 ~ "não",
                           TRUE ~ NA_character_))

#Obesidade
dados5 <- dados5 %>%
  mutate(obesidade = case_when(OBESIDADE == 1 ~ "sim",
                                OBESIDADE == 2 ~ "não",
                                TRUE ~ NA_character_))

```

Desfechos

```

#UTI
dados5 <- dados5 %>%
  mutate(uti = case_when(UTI == 1 ~ "sim",
                         UTI == 2 ~ "não",
                         TRUE ~ NA_character_))

#Suporte ventilatório
dados5 <- dados5 %>%
  mutate(
    suport_ven = case_when(
      SUPORT_VEN == 1 ~ "invasivo",
      SUPORT_VEN == 2 ~ "não invasivo",
      SUPORT_VEN == 3 ~ "não",
      TRUE ~ NA_character_
    )
  )

dados5$suport_ven <- factor(dados5$suport_ven,
                           levels = c("invasivo", "não invasivo", "não"))

```

```

#Evolução
dados5 <-
  dados5 %>% mutate(
    evolucao = case_when(
      EVOLUCAO == 1 ~ "Cura",
      EVOLUCAO == 2 ~ "Obito",
      EVOLUCAO == 3 ~ "Obito",
      TRUE ~ NA_character_
    )
  )

#Classificação final
dados5 <-
  dados5 %>% mutate(
    classi_fin1 = case_when(
      CLASSI_FIN == 5 ~ "COVID-19",
      CLASSI_FIN == 1 ~ "Influenza",
      CLASSI_FIN == 2 ~ "Outro vírus",
      CLASSI_FIN == 3 ~ "Outro agente",
      TRUE ~ NA_character_
    )
  )

```

```

#tabela de frequência para UTI
questionr::freq(
  dados5$uti,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para UTI",
        digits = 2)

```

Table 8: Tabela de frequências para UTI

	n	%
não	25063	66.2
sim	7882	20.8
NA	4911	13.0
Total	37856	100.0

```

#tabela de frequência para suporte ventilatório
questionr::freq(
  dados5$suport_ven,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para

```

```
suporte ventilatório",
digits = 2)
```

Table 9: Tabela de frequências para suporte ventilatório

	n	%
invasivo	3393	9.0
não invasivo	10152	26.8
não	18974	50.1
NA	5337	14.1
Total	37856	100.0

```
#tabela de frequência para evolução
questionr::freq(
  dados5$evolucao,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para evolução",
        digits = 2)
```

Table 10: Tabela de frequências para evolução

	n	%
Cura	29054	76.7
Obito	2423	6.4
NA	6379	16.9
Total	37856	100.0

```
#tabela de frequência para classificação final
questionr::freq(
  dados5$classi_fin1,
  cum = FALSE,
  total = TRUE,
  na.last = FALSE,
  valid = FALSE
) %>%
  kable(caption = "Tabela de frequências para classificação final",
        digits = 2)
```

Table 11: Tabela de frequências para classificação final

	n	%
COVID-19	20593	54.4
Influenza	572	1.5
Outro agente	83	0.2

	n	%
Outro vírus	308	0.8
NA	16300	43.1
Total	37856	100.0

Variáveis de tempo

Vamos criar as variáveis de tempo: tempo entre primeiros sintomas e internação, tempo entre primeiros sintomas e notificação, tempo de permanência na UTI e tempo entre primeiros sintomas e evolução. Para isso, vamos primeiro tratar as variáveis de data no que segue.

```
##### datas
#dt_notific - Data do preenchimento da ficha de notificação
#dt_sint - Data de 1ºs sintomas (deve ser menor que dt_notific)
#dt_interna - Data da internação por SRAG
#dt_pcr - Data do Resultado RT-PCR/outro método por Biologia Molecular
#dt_entuti - Data da entrada na UTI
#dt_saiduti - Data da saída da UTI
#dt_evoluca - Data da alta ou óbito
#dt_digita - Preenchido automaticamente pelo sistema com a data da digitação da ficha.
##Não é a data de preenchimento da ficha manualmente e sim a data em que é digitado no sistema.
dados5 <- dados5 %>%
  mutate(dt_notific = as.Date(DT_NOTIFIC, format = "%d/%m/%Y"),
         dt_sint = as.Date(DT_SIN_PRI, format = "%d/%m/%Y"),
         dt_interna = as.Date(DT_INTERNA, format = "%d/%m/%Y"),
         dt_pcr = as.Date(DT_PCR, format = "%d/%m/%Y"),
         dt_entuti = as.Date(DT_ENTUTI, format = "%d/%m/%Y"),
         dt_saiduti = as.Date(DT_SAIDUTI, format = "%d/%m/%Y"),
         dt_evoluca = as.Date(DT_EVOLUCA, format = "%d/%m/%Y"),
         dt_encerra = as.Date(DT_ENCERRA, format = "%d/%m/%Y"),
         dt_digita = as.Date(DT_DIGITA, format = "%d/%m/%Y"))

hoje <- Sys.Date() #data de hoje

#Arrumando as datas de internação inconsistentes.
#Quando for maior que a data de hoje e menor que a data dos primeiros sintomas é NA.
dados5 <- dados5 %>%
  mutate(dt_interna = case_when((dt_interna <= hoje &
                                dt_interna >= dt_sint) ~ dt_interna,
                                TRUE ~ NA_Date_))
```

Para a variável de internação, vamos avaliar os casos que tem informação sobre a data de internação. Há casos que foram para hospital mas não tem informação sobre a data.

```
#Criando a variável que indica se o caso tem data de internação
dados5 <- dados5 %>%
  mutate(info_dt_interna = ifelse(!is.na(dt_interna), "sim", "não"))

#Tabela cruzada de internação e informação sobre data de internação
ctable(dados5$hospital, dados5$info_dt_interna)
```

```
## Cross-Tabulation, Row Proportions
## hospital * info_dt_interna
## Data Frame: dados5
##
## -----
##      info_dt_interna      não      sim      Total
## hospital
##      não      1232 (100.0%)      0 ( 0.0%)      1232 (100.0%)
##      sim      749 ( 2.1%)      35015 (97.9%)      35764 (100.0%)
##      <NA>      860 (100.0%)      0 ( 0.0%)      860 (100.0%)
##      Total      2841 ( 7.5%)      35015 (92.5%)      37856 (100.0%)
## -----
```

```
#Criando as variáveis de tempo a partir da diferença entre as datas.
# tempo_sintomas_hosp: tempo entre primeiros sintomas e internação.
# tempo_sintomas_notific: tempo entre primeiros sintomas e notificação.
# tempo_uti: tempo de permanência na UTI.
# tempo_sint_evolucao: tempo entre primeiros sintomas e evolução
dados5 <- dados5 %>%
  mutate(
    tempo_sintomas_hosp = as.numeric(dt_interna - dt_sint),
    tempo_sintomas_notific = as.numeric(dt_notific - dt_sint),
    tempo_uti = as.numeric(dt_saiduti - dt_entuti),
    tempo_sint_evolucao = as.numeric(dt_evolucao - dt_sint)
  )
```

Pela análise abaixo, podemos observar que há casos que foram para UTI, mas não tem informação de tempo de permanência.

```
#Número de dados faltantes do tempo de UTI por grupo de UTI
dados5 %>%
  group_by(uti) %>%
  count(!is.na(tempo_uti))
```

```
## # A tibble: 4 x 3
## # Groups:   uti [3]
##   uti   'is.na(tempo_uti)'      n
##   <chr> <lgl>              <int>
## 1 não   FALSE                25063
## 2 sim   FALSE                 3342
## 3 sim   TRUE                  4540
## 4 <NA>  FALSE                 4911
```

Pela análise abaixo, podemos observar que há casos finalizados (óbito ou cura), mas não tem informação sobre o tempo entre primeiros sintomas e evolução.

```
#Número de dados faltantes do tempo entre sintomas e evolução por grupo de evolução
dados5 %>%
  group_by(evolucao) %>%
  count(!is.na(tempo_sint_evolucao))
```

```
## # A tibble: 5 x 3
```

```
## # Groups:   evolucao [3]
##   evolucao '!is.na(tempo_sint_evolucao)'      n
##   <chr>    <lgl>                          <int>
## 1 Cura     FALSE                          1321
## 2 Cura     TRUE                           27733
## 3 Obito    FALSE                           7
## 4 Obito    TRUE                          2416
## 5 <NA>     FALSE                         6379
```

A seguir apresentamos as medidas descritivas das quatro variáveis de tempo consideradas:

```
datasummary((
  tempo_sintomas_hosp + tempo_sintomas_notific + tempo_uti + tempo_sint_evolucao
) ~ 1 * (n + faltantes + media + DP + mediana + minimo + maximo),
data = dados5,
output = 'markdown'
)
```

	n	faltantes	media	DP	mediana	minimo	maximo
tempo_sintomas_hosp	35015.00	2841.00	5.62	12.61	4.00	0.00	397.00
tempo_sintomas_notific	37856.00	0.00	7.90	17.08	5.00	0.00	459.00
tempo_uti	4540.00	33316.00	10.17	12.29	6.00	0.00	183.00
tempo_sint_evolucao	30149.00	7707.00	13.95	18.46	10.00	0.00	512.00

Exportando as bases de dados

```
##### concatenando as informações base que usamos para o painel OOB Covid-19
dados6 <- dados5 %>%
  select(
    SEM_PRI,
    idade_anos,
    SG_UF,
    ID_MN_RESI,
    CO_MUN_RES,
    CO_MU_INTE,
    CLASSI_FIN,
    classi_fin1,
    DT_SIN_PRI,
    DT_EVOLUCA,
    dt_evoluca,
    dt_sint,
    ano,
    classi_gesta_puerp,
    classi_covid,
    region,
    raca,
    escol,
    mudou_muni,
    zona,
```

```

    faixa_et,
    hospital,
    hist_viagem,
    sg_para_srag,
    inf_inter,
    cont_ave_suino,
    vacina,
    antiviral,
    febre,
    tosse,
    garganta,
    dispneia,
    desc_resp,
    saturacao,
    diarreia,
    vomito,
    dor_abd,
    fadiga,
    perd_olft,
    perd_pala,
    cardiopati,
    hematologi,
    hepatica,
    asma,
    diabetes,
    neuro,
    pneumopati,
    imunodepre,
    renal,
    obesidade,
    uti,
    suport_ven,
    evolucao,
    dt_evoluca,
    tempo_sintomas_hosp,
    tempo_sintomas_notific,
    tempo_uti,
    tempo_sint_evolucao
)

#Exportando
write_xlsx(dados6, "dados5.xlsx")

```

```

#Dados gerais
internacoes_covid <-
data.frame(
  covid = c("confirmado", "não espec"),
  int_covid = c(nrow(dados2[dados2$CLASSI_FIN ==
    5, ]),
    nrow(dados2[(dados2$CLASSI_FIN ==
    4 |
    dados2$CLASSI_FIN ==
    9), ])),

```



```
fem_covid = c(nrow(dados3[dados3$CLASSI_FIN ==  
                    5, ]),  
              nrow(dados3[(dados3$CLASSI_FIN ==  
                    4 |  
                    dados3$CLASSI_FIN ==  
                    9), ]))  
  
)  
  
#Exportando  
write_xlsx(internacoes_covid, "dados2_nrow.xlsx")
```