



Avaliação de Desempenho de Sistemas de Vigilância: Métricas

IDASH

Acrónimos

ML Machine Learning

IA Inteligencia Artificial

EDA Análisis Exploratorio de Datos

EDA [Análise Exploratória de Dados](#)

IDASH Informática y Ciencia de Datos para la Salud

SIS Sistemas de Información de Salud

Tabela de Conteúdo

1	Conjunto de dados	6
1.1	Leitura de dados	6
2	Análise exploratória de Dados (EDA) e Preparação de dados	8
2.1	Falecido	8
2.2	Hospitalizado	9
2.2.1	Preparação e correção de datas	11
2.3	Positivo	13
2.3.1	Preparação de dados e correção de datas	14
3	Qualidade dos dados	15
3.1	Compleitude	15
3.1.1	Falecido	15
3.1.2	Hospitalizado	15
3.1.3	Positivo	16
3.2	Validez	17
3.2.1	Hospitalizados	17
3.2.2	Positivo	18
4	Conclusões	19

Lista de Figuras

2.1	Categorias de variáveis de tipo fatorial	9
2.2	Categorias de variáveis de tipo fatorial	11
2.3	Categorias de variáveis de tipo fatorial	14

Lista de Tabelas

3.1	Tabla de completitud del conjunto de datos fallecidos.	15
3.2	Tabela de completude do conjunto de dados de hospitalizado.	16
3.3	Tabela de completude do conjunto de dados positivo.	17

1 Conjunto de dados

Os dados são armazenados em 3 arquivos compactados e estão disponíveis no repositório indicado:

- Fallecidos (data/fallecidos.csv.gz)
- Hospitalizados (data/hospitalizados.csv.gz)
- Positivos (data/positivos.csv.gz)

O dicionário de dados está localizado no mesmo repositório, no arquivo `data-dictionary-pt.pdf`

Para o processamento de dados, serão usados os seguintes pacotes: `tidyverse`, `dlookr`, `inspectdf`, `skimr`, `lubridate`, `janitor`, `kableExtra`

1.1 Leitura de dados

```
fallecidos <- read_csv("data/fallecidos.csv.gz",
  col_types = cols(
    fecha_fallecimiento = col_date(format = "%Y-%m-%d"),
    edad_declarada = col_integer(),
    sexo = col_character(),
    clasificacion_def = col_character(),
    departamento = col_factor(),
    provincia = col_factor(),
    distrito = col_factor(),
    uuid = col_character(),
    age_group = col_factor(),
  )
)
```

```
hospitalizados <- read_csv("data/hospitalizados.csv.gz",
  col_types = cols(
    eess_nombre = col_factor(),
    id_persona = col_character(),
    edad = col_integer(),
    sexo = col_factor(),
    fecha_ingreso_hosp = col_character(),
    fecha_ingreso_uci = col_character(),
    fecha_ingreso_ucin = col_character(),
    con_oxigeno = col_logical(),
    con_ventilacion = col_logical(),
    fecha_segumiento_hosp_ultimo = col_character(),
    evolucion_hosp_ultimo = col_factor(),
    flag_vacuna = col_factor(),
    fecha_dosis1 = col_character(),
    fabricante_dosis1 = col_factor(),
    fecha_dosis2 = col_character(),
    fabricante_dosis2 = col_factor(),
    fecha_dosis3 = col_character(),
    fabricante_dosis3 = col_factor(),
    cdc_positividad = col_logical(),
    cdc_fecha_fallecido_covid = col_character(),
    cdc_fallecido_covid = col_logical(),
    dep_domicilio = col_factor(),
    prov_domicilio = col_factor(),
    dist_domicilio = col_factor(),
  )
)
```

```
positivos <- read_csv(
  "data/positivos.csv.gz",
  col_types = cols(
    departamento = col_factor(),
    provincia = col_factor(),
    distrito = col_factor(),
    metododx = col_factor(),
    edad = col_integer(),
    sexo = col_factor(),
    fecha_resultado = col_character(),
    id_persona = col_character()
  ))
```

2 Análise exploratória de Dados (EDA) e Preparação de dados

A Análise Exploratória de Dados ([EDA](#)) é um processo de investigação, visualização e resumo das principais características e padrões de um conjunto de dados, geralmente usando técnicas estatísticas gráficas e descritivas. Seu principal objetivo é entender os dados em profundidade, descobrir anomalias, identificar relações entre variáveis e extrair percepções iniciais que possam orientar análises mais formais subsequentes ou a criação de modelos.

2.1 Falecido

```
diagnose(fallecidos)
```

```
# A tibble: 9 x 6
  variables      types missing_count missing_percent unique_count unique_rate
  <chr>          <chr>         <int>          <dbl>          <int>          <dbl>
1 fecha_fallecimie~ Date           9479          10.0           298          0.00314
2 edad_declarada   inte~           0            0             110          0.00116
3 sexo            char~           0            0              2          0.0000211
4 clasificacion_def char~           0            0              7          0.0000738
5 departamento     fact~           0            0             26          0.000274
6 provincia        fact~           5          0.00527         21          0.000222
7 distrito         fact~           5          0.00527         44          0.000464
8 uuid            char~          1382          1.46          93402         0.985
9 age_group        fact~           0            0              5          0.0000527
```

```
diagnose_numeric(fallecidos)
```

```
# A tibble: 1 x 10
  variables      min    Q1  mean median    Q3   max  zero minus outlier
  <chr>          <int> <dbl> <dbl> <dbl> <dbl> <int> <int> <int> <int>
1 edad_declarada    0    57  66.5   68    77  113   91    0   1526
```


Seis variáveis categóricas foram identificadas no conjunto de dados e a distribuição das categorias de cada variável é mostrada abaixo.

```
var_cat <- inspect_cat(fallecidos[, c(3:7, 9)])
show_plot(var_cat)+
  labs(
    title = "Distribuição das categorias de variáveis",
    subtitle = "Conjunto de dados de falecido",
    x = "Categoria")
```

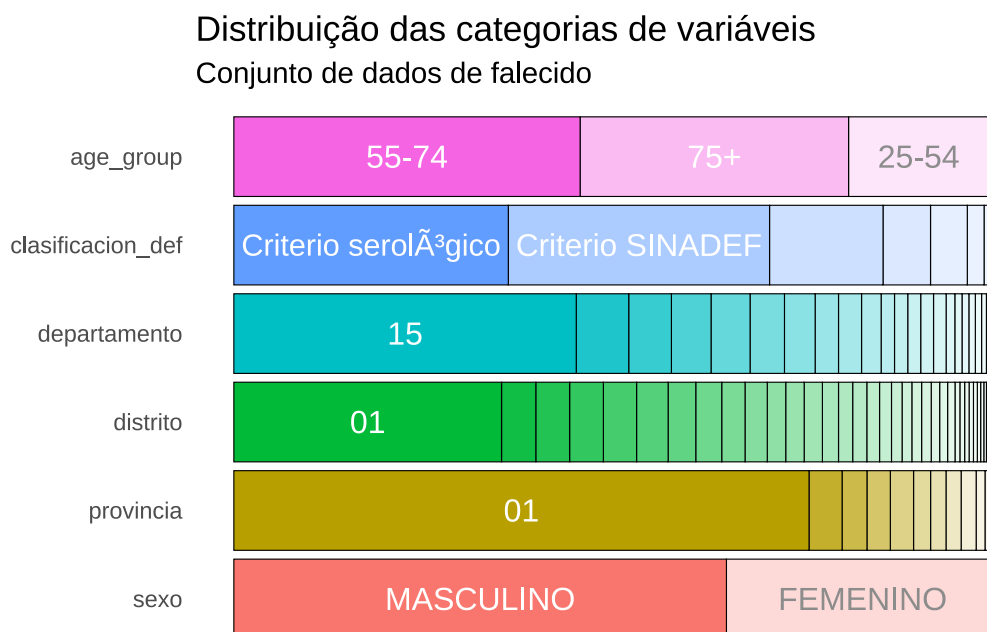


Figura 2.1: Categorias de variáveis de tipo fatorial

2.2 Hospitalizado

```
diagnose(hospitalizados)
```

```
# A tibble: 24 x 6
  variables      types missing_count missing_percent unique_count unique_rate
  <chr>          <chr>         <int>          <dbl>         <int>         <dbl>
1 eess_nombre fact~           6546           12.0           98          0.00180
```

```

2 id_persona      char~           0           0          54242  0.994
3 edad            inte~          3580          6.56           104  0.00191
4 sexo            fact~          2851          5.23            3  0.0000550
5 fecha_ingreso_h~ char~           0           0           295  0.00541
6 fecha_ingreso_u~ char~         50636          92.8           295  0.00541
7 fecha_ingreso_u~ char~         52394          96.0           272  0.00499
8 con_oxigeno      logi~           6          0.0110            3  0.0000550
9 con_ventilacion  logi~          15          0.0275            3  0.0000550
10 fecha_segumient~ char~           1          0.00183          375  0.00687
# i 14 more rows

```

```
tbdhosp<- diagnose(hospitalizados)
```

```
diagnose_numeric(hospitalizados)
```

```

# A tibble: 1 x 10
  variables    min    Q1 mean median    Q3    max zero minus outlier
  <chr>      <int> <dbl> <dbl> <dbl> <dbl> <int> <int> <int> <int>
1 edad            1    35  51.9    54    68   103     0     0     0

```

O número de IDs exclusivos é $c(id_persona = 54242)$ e o número de registros é 54556, portanto, há pacientes que foram hospitalizados mais de uma vez.

As variáveis categóricas do conjunto de dados hospitalizados são mostradas no gráfico abaixo.

```

var_cat_h <- inspect_cat(hospitalizados[, -c(2,5,6,7,10,15,13,17,20)])
show_plot(var_cat_h)+
  labs(
    title = "Distribuição das categorias de variáveis",
    subtitle = "Conjunto de dados de hospitalizado",
    x = "Categoria")

```

Distribuição das categorias de variáveis

Conjunto de dados de hospitalizado

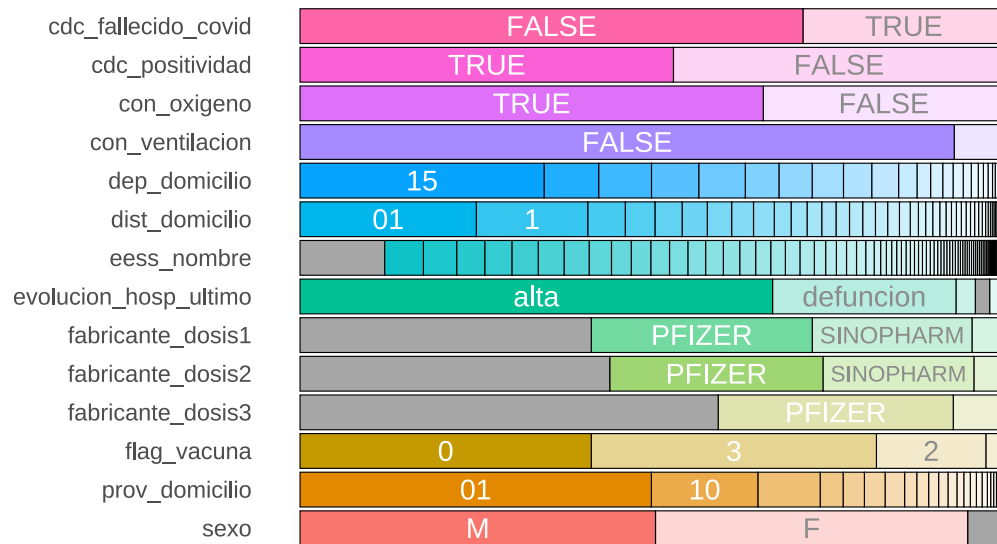


Figura 2.2: Categorias de variáveis de tipo fatorial

2.2.1 Preparação e correção de datas

Agora, analisamos a variabilidade dos registros de variáveis de data que existem no conjunto de dados.

```
fecha_ingreso_hosp_var <- tabyl(hospitalizados$fecha_ingreso_hosp,
                                show_missing_levels = TRUE)

hospitalizados$fecha_ingreso_hosp <- ymd(hospitalizados$fecha_ingreso_hosp)
```

```
fecha_ingreso_uci_var <- tabyl(hospitalizados$fecha_ingreso_uci,
                                show_missing_levels = TRUE)

hospitalizados$fecha_ingreso_uci <- ymd(hospitalizados$fecha_ingreso_uci)
```

```
fecha_ingreso_ucin_var <- tabyl(hospitalizados$fecha_ingreso_ucin,
                                show_missing_levels = TRUE)

hospitalizados$fecha_ingreso_ucin <- ymd(hospitalizados$fecha_ingreso_ucin)
```

```
fecha_segumiento_hosp_ultimo_var <-  
  ↪ tabyl(hospitalizados$fecha_segumiento_hosp_ultimo,  
          show_missing_levels = TRUE)  
  
hospitalizados$fecha_segumiento_hosp_ultimo <-  
  ↪ ymd(hospitalizados$fecha_segumiento_hosp_ultimo)
```

```
fecha_dosis1_var <- tabyl(hospitalizados$fecha_dosis1,  
                          show_missing_levels = TRUE)  
  
#hospitalizados$fecha_dosis1 <- ymd(hospitalizados$fecha_dosis1)
```

Quando atribuímos o formato ano-mês-dia (ymd), confirma-se que as observações não têm esse formato e a coluna está corrompida. Como podemos ver na revisão, as datas estão no formato dia-mês-ano.

```
hospitalizados$fecha_dosis1 <- dmy(hospitalizados$fecha_dosis1)
```

```
fecha_dosis2_var <- tabyl(hospitalizados$fecha_dosis2,  
                          show_missing_levels = TRUE)  
  
hospitalizados$fecha_dosis2 <- ymd(hospitalizados$fecha_dosis2)
```

```
fecha_dosis3_var <- tabyl(hospitalizados$fecha_dosis3,  
                          show_missing_levels = TRUE)  
  
#hospitalizados$fecha_dosis3 <- ymd(hospitalizados$fecha_dosis3)
```

Quando atribuímos o formato ano-mês-dia (ymd), confirma-se que as observações não têm esse formato e a coluna está corrompida. Como podemos ver na revisão, as datas estão no formato dia-mês-ano.

```
hospitalizados$fecha_dosis3 <- dmy(hospitalizados$fecha_dosis3)
```

```
fecha_cdc_fallecido_covid_var <-  
  ↪ tabyl(hospitalizados$cdc_fecha_fallecido_covid,  
          show_missing_levels = TRUE)  
  
hospitalizados$cdc_fecha_fallecido_covid <-  
  ↪ ymd(hospitalizados$cdc_fecha_fallecido_covid)
```

2.3 Positivo

```
diagnose(positivos)
```

```
# A tibble: 8 x 6
```

	variables	types	missing_count	missing_percent	unique_count	unique_rate
	<chr>	<chr>	<int>	<dbl>	<int>	<dbl>
1	departamento	factor	52912	5.17	27	0.0000264
2	provincia	factor	48942	4.79	35	0.0000342
3	distrito	factor	48942	4.79	54	0.0000528
4	metododx	factor	0	0	3	0.00000293
5	edad	integer	55	0.00538	115	0.000112
6	sexo	factor	0	0	2	0.00000196
7	fecha_resultado	charac~	0	0	592	0.000579
8	id_persona	charac~	16980	1.66	989414	0.968

```
diagnose_numeric(positivos)
```

```
# A tibble: 1 x 10
```

	variables	min	Q1	mean	median	Q3	max	zero	minus	outlier
	<chr>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<int>	<int>	<int>
1	edad	0	29	42.0	41	54	120	5920	0	2389

O número de identificações exclusivas é 989414, portanto, há pacientes que foram diagnosticados como Covid positivo mais de uma vez.

As variáveis categóricas são mostradas no gráfico a seguir.

```
var_cat_p <- inspect_cat(positivos[, -c(5,7,8)])
show_plot(var_cat_p)+
  labs(
    title = "Distribuição de categorias de variáveis",
    subtitle = "Conjunto de dados de casos positivo de Covid-19",
    x = "Categoria")
```

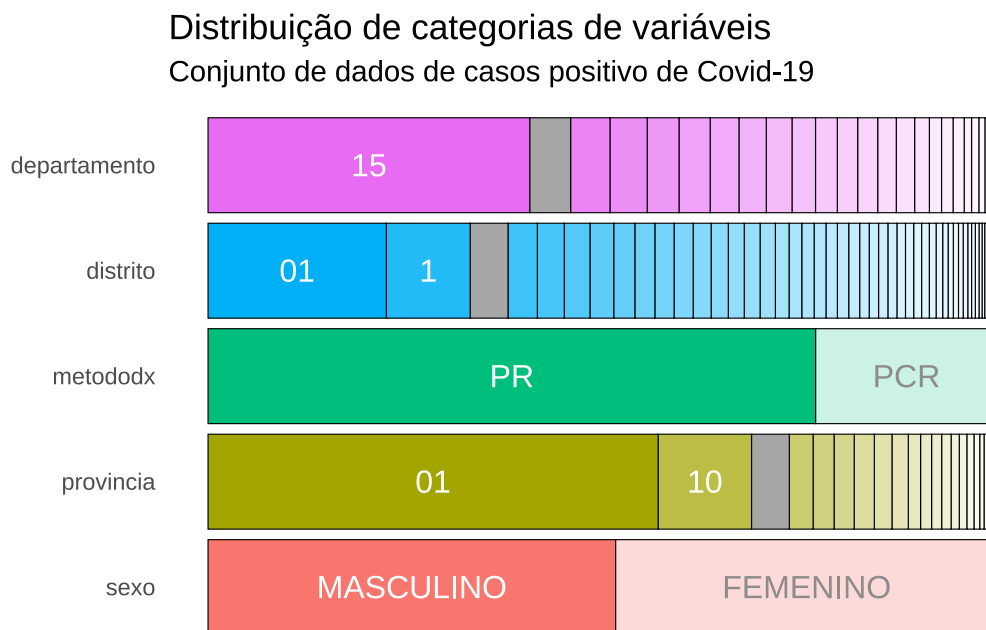


Figura 2.3: Categorias de variáveis de tipo fatorial

2.3.1 Preparação de dados e correção de datas

```

fecha_resultado_var <- tabyl(positivos$fecha_resultado,
                             show_missing_levels = TRUE)

positivos$fecha_resultado_f<- ymd(positivos$fecha_resultado)

num_na_val_fec<-sum(is.na(positivos$fecha_resultado_f))

```

3 Qualidade dos dados

3.1 Completitude

3.1.1 Falecido

A integridade é uma característica de qualidade que se refere ao grau em que um conjunto de dados inclui todos os valores ou atributos esperados.

A tabela a seguir mostra a análise da integridade das variáveis no conjunto de dados.

```
na_fallecidos <- inspect_na(fallecidos)
na_fallecidos <- na_fallecidos |>
  mutate(pcnt = round(pcnt, 2)) |>
  filter(pcnt!=0)

kbl(na_fallecidos, col.names = c("Variável", "Valores ausentes",
  ↪ "Porcentagem(%))" |>
  kable_styling()
```

Tabla 3.1: Tabla de completitud del conjunto de datos fallecidos.

Variável	Valores ausentes	Porcentagem(%)
fecha_fallecimiento	9479	10.00
uuid	1382	1.46
provincia	5	0.01
distrito	5	0.01

As maiores porcentagens de perda de dados são encontradas nas variáveis: `fecha_fallecimiento` a `uuid`.

3.1.2 Hospitalizado

```
tr_na_hosp <- inspect_na(hospitalizados[,c(
  ↪ "eess_nombre", "edad", "sexo", "evolucion_hosp_ultimo", "prov_domicilio"])]

tr_na_hosp <- tr_na_hosp |>
  mutate(pcmt = round(pcmt, 2)) |>
  filter(pcmt!=0)

kbl(tr_na_hosp, col.names = c("Variável", "Valores ausentes",
  ↪ "Porcentagem(%))") |>
  kable_styling()
```

Tabla 3.2: Tabela de completude do conjunto de dados de hospitalizado.

Variável	Valores ausentes	Porcentagem(%)
eess_nombre	6546	12.00
edad	3580	6.56
sexo	2851	5.23
evolucion_hosp_ultimo	1113	2.04
prov_domicilio	45	0.08

3.1.3 Positivo

```
tr_na_pos <- inspect_na(positivos)

tr_na_pos <- tr_na_pos |>
  mutate(pcmt = round(pcmt, 2)) |>
  filter(pcmt!=0)

kbl(tr_na_pos, col.names = c("Variável", "Valores ausentes",
  ↪ "Porcentagem(%))") |>
  kable_styling()
```


Tabla 3.3: Tabela de completude do conjunto de dados positivo.

Variável	Valores ausentes	Porcentagem(%)
fecha_resultado_f	122712	12.00
departamento	52912	5.17
provincia	48942	4.79
distrito	48942	4.79
id_persona	16980	1.66
edad	55	0.01

3.2 Validez

3.2.1 Hospitalizados

```
dosis1_valid <- hospitalizados |>
  group_by(flag_vacuna, fecha_dosis1) |>
  count()
```

```
dosis1_valid
```

```
# A tibble: 1,026 x 3
# Groups:   flag_vacuna, fecha_dosis1 [1,026]
  flag_vacuna fecha_dosis1     n
  <fct>      <date>      <int>
1 3          2021-02-09      21
2 3          2021-02-10      96
3 3          2021-02-11     143
4 3          2021-02-12      69
5 3          2021-02-13      44
6 3          2021-02-14       5
7 3          2021-02-15      37
8 3          2021-02-16      21
9 3          2021-02-17      11
10 3         2021-02-18      28
# i 1,016 more rows
```

Como pode ser visto, não há datas atribuídas à variável `fecha_dosis1` incorretamente, considerando a variável `flag_vacuna` (valores = 0). Portanto, o cálculo da métrica de validade não se aplica.

```
dosis2_valid <- hospitalizados |>
  group_by(flag_vacuna, fecha_dosis2) |>
  count()

dosis2_N0_valid <- dosis2_valid |>
  filter(flag_vacuna == "0" & !is.na(fecha_dosis2))

dosis2_N0_valid$flag_vacuna <- as.character(dosis2_N0_valid$flag_vacuna)
```

Há casos 12 em que a variável `flag_vacuna` registra um valor de 0, portanto, há um problema de validade na variável 0.

A métrica de validade para a variável `flag_vacuna` seria 0.02

```
dosis3_valid <- hospitalizados |>
  group_by(flag_vacuna, fecha_dosis3) |>
  count()

dosis3_valid$flag_vacuna <- as.character(dosis3_valid$flag_vacuna)
```

Todos os registros da variável `fecha_dosis3` são considerados consistentes, considerando as informações da variável `flag_vacuna`. Portanto, o cálculo da métrica de validade não se aplica.

3.2.2 Positivo

```
validade_fecha_resultado <- (num_na_val_fec/nrow(positivos)*100)
```

A métrica de validade da variável de data do resultado no conjunto de dados positivos (casos positivos) corresponde à 12% de datas formatadas incorretamente a serem corrigidas.

4 Conclusões

Nesta seção, desenvolva as conclusões das métricas apresentadas. Se considerar que outras métricas poderiam ser incluídas, comente se os dados são suficientes ou se são necessárias outras fontes.