



Evaluación de Sistemas de Vigilancia: Métricas de Rendimiento

IDASH

Acrónimos

ML Machine Learning

IA Inteligencia Artificial

EDA Análisis Exploratorio de Datos

IDASH Informática y Ciencia de Datos para la Salud

SIS Sistemas de Información de Salud

Tabla de Contenidos

1	Conjunto de datos	6
1.1	Lectura de Datos	6
2	Análisis Exploratorio de Datos (EDA) y preparación de datos	8
2.1	Fallecidos	8
2.2	Hospitalizados	9
2.2.1	Preparación y corrección de fechas	11
2.3	Positivos	13
2.3.1	Preparación de datos corrección de fechas	14
3	Calidad de datos	15
3.1	Compleitud	15
3.1.1	Fallecidos	15
3.1.2	Hospitalizaciones	15
3.1.3	Positivos	16
3.2	Validez	17
3.2.1	Hospitalizados	17
3.2.2	Positivos	18
4	Conclusiones	19

Lista de Figuras

2.1	Categorías de la variables tipo factor	9
2.2	Categorías de la variables tipo factor	11
2.3	Categorías de la variables tipo factor	14

Lista de Tablas

3.1	Tabla de completitud del conjunto de datos fallecidos.	15
3.2	Tabla de completitud del conjunto de datos hospitalizados.	16
3.3	Tabla de completitud del conjunto de datos positivos.	17

1 Conjunto de datos

Los datos se encuentran almacenados en 3 archivos comprimidos y están disponibles en el repositorio indicado:

- Fallecidos
- Hospitalizados
- Positivos

Para el tratamiento de datos se usarán las librerías tidyverse, dlookr, inspectdf,skimr, lubridate,janitor,kableExtra

1.1 Lectura de Datos

```
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 Análisis Exploratorio de Datos (EDA) y preparación de datos

El Análisis Exploratorio de Datos ([EDA](#)) es un proceso que consiste en investigar, visualizar y resumir las principales características y patrones de un conjunto de datos, a menudo utilizando técnicas gráficas y estadísticas descriptivas. Su objetivo principal es entender los datos en profundidad, descubrir anomalías, identificar relaciones entre variables y extraer **insights** iniciales que puedan guiar análisis posteriores más formales o la construcción de modelos.

2.1 Fallecidos

```
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             25      0.000264
6 provincia        fact~           5          0.00527         197      0.00208
7 distrito         fact~           5          0.00527        1390      0.0147
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
```


En el conjunto de datos se identificaron 6 variables categóricas las cuales a continuación se muestra la distribución de las categorías de cada variable.

```
var_cat <- inspect_cat(fallecidos[, c(3:7, 9)])
show_plot(var_cat)+
  labs(
    title = "Distribución de las categorías de las variables",
    subtitle = "Conjunto de datos fallecidos",
    x = "Categoría")
```

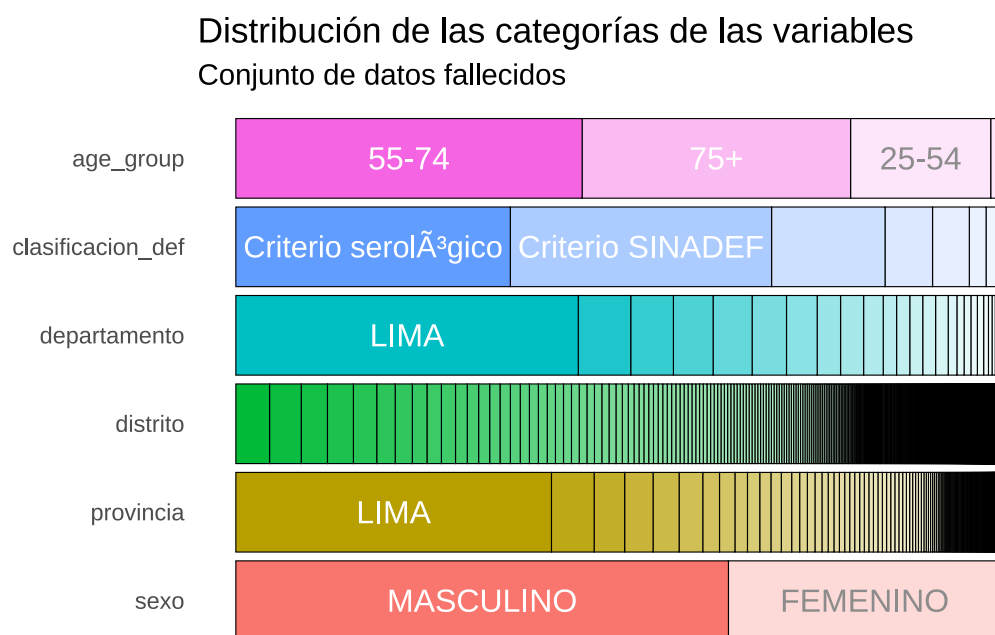


Figura 2.1: Categorías de la variables tipo factor

2.2 Hospitalizados

```
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

```

El número de identificación únicas es `c(id_persona = 54242)` y el número de registros es de 54556, por lo que existen pacientes que se hospitalizaron más de una vez.

Las variables categóricas del conjunto de datos hospitalizados se muestra en el siguiente gráfico.

```

var_cat_h <- inspect_cat(hospitalizados[, -c(2,5,6,7,10,15,13,17,20)])
show_plot(var_cat_h)+
  labs(
    title = "Distribución de las categorías de las variables",
    subtitle = "Conjuntos de datos hospitalizados",
    x = "Categoría")

```

Distribución de las categorías de las variables Conjuntos de datos hospitalizados

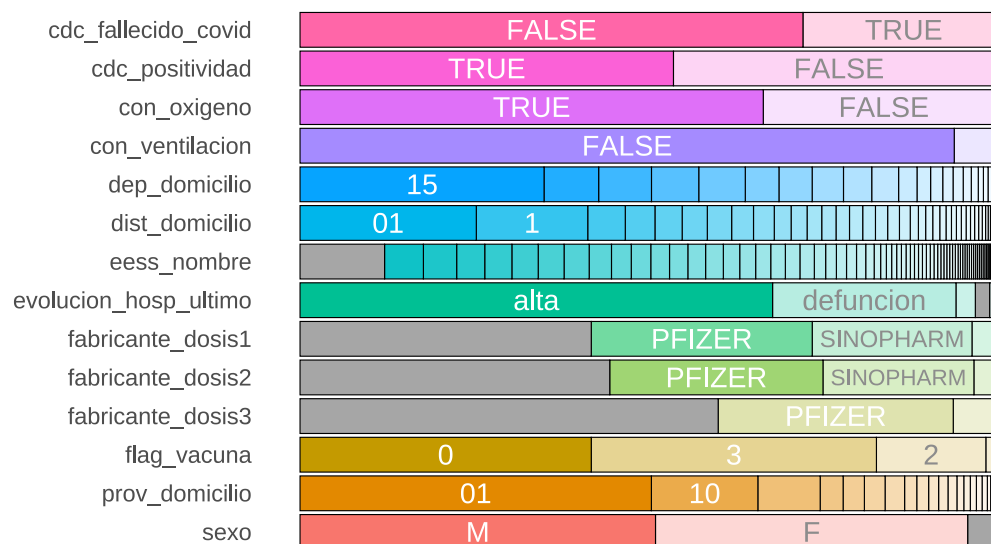


Figura 2.2: Categorías de la variables tipo factor

2.2.1 Preparación y corrección de fechas

Ahora, analizamos la variabilidad de los registros de la variables tipo fechas que existen en el conjunto de datos.

```
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)
```

Cuando asignamos el formato año-mes-día (ymd) se confirma que las observaciones no tienen ese formato y la columna se corrompe. Como apreciamos en la revisión, las fechas están en formato día-mes-año.

```
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)
```

Cuando asignamos el formato año-mes-día (ymd) se confirma que las observaciones no tienen ese formato y la columna se corrompe. Como apreciamos en la revisión, las fechas están en formato día-mes-año.

```
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 Positivos

```
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
```

El número de identificaciones únicas es de 989414, por lo que existen pacientes que fueron diagnosticados como Covid positivo más de una vez.

Las variables categóricas se muestran en el siguiente gráfico

```
var_cat_p <- inspect_cat(positivos[, -c(5,7,8)])
show_plot(var_cat_p)+
  labs(
    title = "Distribución de las categorías de las variables",
    subtitle = "Conjunto de datos de casos positivos de Covid-19",
    x = "Categoría")
```

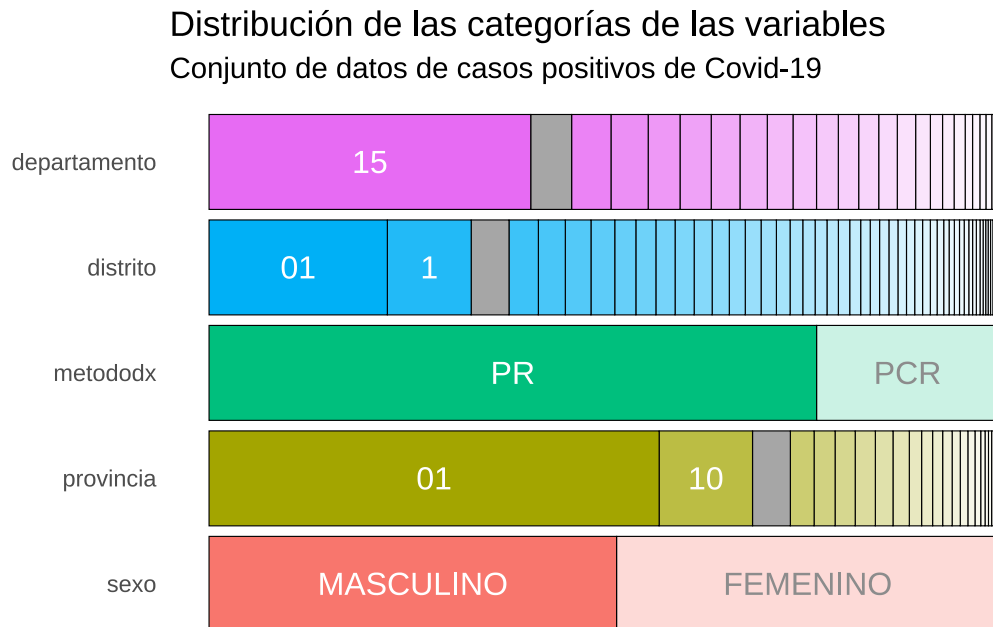


Figura 2.3: Categorías de la variables tipo factor

2.3.1 Preparación de datos corrección de fechas

```
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 Calidad de datos

3.1 Completitud

3.1.1 Fallecidos

La completitud es una característica

En la siguiente tabla

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

kbl(na_fallecidos, col.names = c("Variable", "Valores Perdidos",
  ↪  "Porcentaje(%)" )) |>
  kable_styling()
```

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

Variable	Valores Perdidos	Porcentaje(%)
fecha_fallecimiento	9479	10.00
uuid	1382	1.46
provincia	5	0.01
distrito	5	0.01

Los mayores porcentajes de pérdida de datos se encuentran en las variables: `fecha_fallecimiento` y `uuid`.

3.1.2 Hospitalizaciones

```
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("Variable", "Valores Perdidos",
  ↪ "Porcentaje(%)" )) |>
  kable_styling()
```

Tabla 3.2: Tabla de completitud del conjunto de datos hospitalizados.

Variable	Valores Perdidos	Porcentaje(%)
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 Positivos

```
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("Variable", "Valores Perdidos",
  ↪ "Porcentaje(%)" )) |>
  kable_styling()
```


Tabla 3.3: Tabla de completitud del conjunto de datos positivos.

Variable	Valores Perdidos	Porcentaje(%)
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 se puede apreciar, no existen fechas asignadas a la variable `fecha_dosis1` incorrectamente, considerando la variable `flag_vacuna` (valores = 0). Por tanto, no aplica el cálculo de la métrica de validez.

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

dosis2_NO_valid <- dosis2_valid |>
  filter(flag_vacuna == "0" & !is.na(fecha_dosis2))

dosis2_NO_valid$flag_vacuna <- as.character(dosis2_NO_valid$flag_vacuna)
```

Existen 12 casos en los que la variable `flag_vacuna` registra un valor de 0, por lo que existe un problema de validez en la variable 0.

La métrica de validez de la variable `flag_vacuna` sería 0.02

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

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

Todos los registros de la variable `fecha_dosis3` se encuentran consistentes considerando la información de la variable `flag_vacuna`. Por tanto, no aplica el cálculo de la métrica de validez.

3.2.2 Positivos

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

La métrica de validez para la variable `fecha` resultado en el conjunto de datos positivos (casos positivos) corresponde al 12% de fechas con un formato incorrecto que debe ser coregido.

4 Conclusiones

En este apartado desarrolla las conclusiones de las métricas presentadas, si consideras que se puedan incluir otras métricas comenta si con los datos es suficiente o se necesitarían otras fuentes.