

14.- Feature Data Transform_04_19_vacunacion_completo_v_01

June 8, 2023

#

CU04_Optimización de vacunas

Citizenlab Data Science Methodology > III - Feature Engineering Domain *** > # 14.- Feature Data Transform

Feature Data Transform is the process that allows change (if is required) the type and/or distribution of data features (e.g. scaling, normalizing o standardizing data features).

0.1 Tasks

Perform Basic Data Transforms

Perform Categorical Variable Transformation

- Encode Transformation
- One-hot encoding
- Ordinal encoding
- Dummy encoding
- Evaluate a Logistic Regression model
- Consider Embedding if text mining context

Perform Numeric Variable Transformation

- Scale Transformation
- Normalization
- Standardization
- IQR Robust Scaler Transform
- Evaluate a KNN model
- Distribution Transformation
- Discretization
- Uniform
- Clustered(k-Means)
- Quantile
- Normal Quantile
- Uniform Quantile
- Evaluate a KNN model
- Evaluate a KNN model
- Power transforms (Make Distributions More Gaussian)
- Box-Cox Transform
- Yeo-Johnson Transform
- Evaluate a KNN model

0.2 Consideraciones casos CitizenLab programados en R

- Algunas de las tareas de este proceso se han realizado en los notebooks del proceso 05 Data Collection porque eran necesarias para las tareas ETL. En esos casos, en este notebook se referencia al notebook del proceso 05 correspondiente
- Otras tareas típicas de este proceso se realizan en los notebooks del dominio IV al ser más eficiente realizarlas en el propio pipeline de modelización.
- Por tanto en los notebooks de este proceso de manera general se incluyen las comprobaciones necesarias, y comentarios si procede
- Las tareas del proceso se van a aplicar solo a los archivos que forman parte del despliegue, ya que hay muchos archivos intermedios que no procede pasar por este proceso
- El nombre de archivo del notebook hace referencia al nombre de archivo del proceso 05 al que se aplica este proceso, por eso pueden no ser correlativa la numeración
- Las comprobaciones se van a realizar teniendo en cuenta que el lenguaje utilizado en el despliegue de este caso es R

0.3 File

- Input File: CU_04_08_20_vacunacion_gripe_train_and_test.csv
- Output File: No aplica

0.3.1 Encoding

Con la siguiente expresión se evitan problemas con el encoding al ejecutar el notebook. Es posible que deba ser eliminada o adaptada a la máquina en la que se ejecute el código.

```
[7]: Sys.setlocale(category = "LC_ALL", locale = "es_ES.UTF-8")
```

```
Warning message in Sys.setlocale(category = "LC_ALL", locale = "es_ES.UTF-8"):  
"OS reports request to set locale to "es_ES.UTF-8" cannot be honored"  
"
```

0.4 Settings

0.4.1 Libraries to use

```
[8]: library(readr)  
library(dplyr)  
library(tidyr)  
library(forcats)  
library(lubridate)
```

0.4.2 Paths

```
[9]: iPath <- "Data/Input/"  
oPath <- "Data/Output/"
```

0.5 Data Load

OPCION A: Seleccionar fichero en ventana para mayor comodidad

Data load using the {tcltk} package. Uncomment the line if using this option

```
[10]: # file_data <- tcltk::tk_choose.files(multi = FALSE)
```

OPCION B: Especificar el nombre de archivo

```
[11]: iFile <- "CU_04_08_20_vacunacion_gripe_train_and_test.csv"
file_data <- paste0(iPath, iFile)

if(file.exists(file_data)){
  cat("Se leerán datos del archivo: ", file_data)
} else{
  warning("Cuidado: el archivo no existe.")
}
```

Se leerán datos del archivo:

Data/Input/CU_04_08_20_vacunacion_gripe_train_and_test.csv

Data file to dataframe Usar la función adecuada según el formato de entrada (xlsx, csv, json, ...)

```
[12]: data <- read_csv(file_data)
```

Rows: 21736 Columns: 49

Column specification

Delimiter: ","

chr (3): GEOCODIGO, DESBDT, nombre_zona

dbl (45): ano, semana, n_vacunas, n_citas, tmed, prec, velmedia,
presMax, be...

lgl (1): is_train

Use `spec()` to retrieve the full column specification for this data.

Specify the column types or set `show_col_types = FALSE` to quiet this message.

Estructura de los datos:

```
[13]: data |> glimpse()
```

Rows: 21,736

Columns: 49

\$ GEOCODIGO <chr> "259", "260", "041", "025", "046",

```

"159", "065", "09...
$ DESBDT      <chr> "V Centenario", "Valdeacederas",
"Canillejas", "Bara...
$ ano         <dbl> 2022, 2022, 2022, 2022, 2022, 2022,
2022, 2021, 2023...
$ semana      <dbl> 34, 8, 9, 49, 24, 3, 8, 47, 1, 2,
52, 39, 16, 50, 34...
$ n_vacunas   <dbl> 0, 0, 0, 292, 0, 524, 0, 248, 204,
205, NA, 0, 0, 51...
$ n_citas     <dbl> 0, 0, 0, 280, 0, 498, 0, 228, 198,
187, NA, 0, 0, 51...
$ tmed        <dbl> 27.278748, 9.577289, 8.536554,
9.065363, 29.905728, ...
$ prec        <dbl> 0.169955881, 1.264910043,
3.122881160, 7.313886680, ...
$ velmedia    <dbl> 2.297067, 1.890425, 2.418071,
1.562328, 2.564749, 1...
$ presMax     <dbl> 940.0420, 944.1770, 949.7179,
941.8342, 940.5669, 95...
$ benzene     <dbl> 0.1764413, 0.4591543, 0.4099159,
0.4224172, 0.195865...
$ co          <dbl> 0.4987735, 0.3960647, 0.3951587,
NA, 0.2891224, 0.50...
$ no          <dbl> NA, 6.611337, 9.331224, 14.007722,
4.063517, 24.4756...
$ no2         <dbl> 14.21113, 34.67671, 30.29999,
32.54832, 26.06913, 44...
$ nox         <dbl> 18.00109, 48.94660, 45.22346,
56.75574, 30.35311, 74...
$ o3          <dbl> 80.90659, 42.06663, 48.88088,
26.68276, 64.55205, 31...
$ pm10        <dbl> 20.117087, 15.042152, 14.002432,
18.032354, 55.79346...
$ pm2.5       <dbl> 10.628064, 5.539590, 7.124192,
6.793868, 19.520373, ...
$ so2         <dbl> 2.794934, 3.507164, 2.692125,
2.351139, 3.397640, 2...
$ campana     <dbl> NA, NA, NA, 2022, NA, 2021, NA,
2021, 2022, 2021, 20...
$ scampana    <dbl> NA, NA, NA, 14, NA, 20, NA, 12, 18,
19, 17, 4, NA, 1...
$ capacidad_zona <dbl> 7957, 6537, 7167, 5633, 3864,
12583, 8544, 5077, 494...
$ prop_riesgo <dbl> 0.11393237, 0.15763986, 0.25500690,
0.14452370, 0.26...
$ tasa_riesgo <dbl> 0.013477754, 0.015731142,
0.009177382, 0.013099129, ...
$ tasa_mayores <dbl> 0.023033610, 0.032817374,

```

```

0.028147027, 0.020829657, ...
$ poblacion_mayores <dbl> 0.10330662, 0.14362062, 0.23161874,
0.13058449, 0.24...
$ nombre_zona <chr> "V Centenario", "Valdeacederas",
"Canillejas", "Bara...
$ nsec <dbl> 17, 18, 22, 13, 14, 42, 32, 13, 17,
11, NA, 15, 15, ...
$ t3_1 <dbl> 36.73039, 41.41412, 45.44882,
39.78001, 46.13171, 46...
$ t1_1 <dbl> 31778, 26202, 28658, 22492, 15450,
50478, 34148, 202...
$ t2_1 <dbl> 0.5084658, 0.5329728, 0.5316594,
0.5189021, 0.551191...
$ t2_2 <dbl> 0.4915342, 0.4670272, 0.4683406,
0.4810979, 0.448809...
$ t4_1 <dbl> 0.22551283, 0.12790298, 0.12603707,
0.18104432, 0.11...
$ t4_2 <dbl> 0.6711962, 0.7284970, 0.6423306,
0.6883785, 0.641173...
$ t4_3 <dbl> 0.10330662, 0.14362062, 0.23161874,
0.13058449, 0.24...
$ t5_1 <dbl> 0.1063332, 0.2295250, 0.1655070,
0.1266086, 0.165893...
$ t6_1 <dbl> 0.1706875, 0.3477631, 0.2511757,
0.1998911, 0.261480...
$ t7_1 <dbl> 0.05131106, 0.04606911, 0.04379644,
0.05585777, 0.06...
$ t8_1 <dbl> 0.03892836, 0.03586418, 0.03207779,
0.04434976, 0.05...
$ t9_1 <dbl> 0.5151383, 0.3863876, 0.3129631,
0.4611972, 0.701812...
$ t10_1 <dbl> 0.09258503, 0.13151901, 0.13926119,
0.10460043, 0.06...
$ t11_1 <dbl> 0.6406787, 0.5451465, 0.4600730,
0.5920292, 0.471769...
$ t12_1 <dbl> 0.7028586, 0.6277335, 0.5346482,
0.6590530, 0.502531...
$ area <dbl> 2100118.9, 1164622.0, 1597474.5,
3816572.0, 870986.8...
$ densidad_hab_km <dbl> 15131.52443, 22498.28643,
17939.56640, 5893.24662, 1...
$ tuits_gripe <dbl> 60, 56, 72, 196, 46, 382, 56, 280,
24, 508, NA, 126,...
$ interes_gripe <dbl> 24, 15, 24, 77, 21, 42, 15, 64, 64,
69, NA, 42, 40, ...
$ Target <dbl> 24, 15, 24, 77, 21, 42, 15, 64, 64,
69, NA, 42, 40, ...
$ is_train <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE,

```

TRUE, TRUE, TRUE...

Muestra de los primeros datos:

```
[14]: data |> slice_head(n = 5)
```

| | GEOCODIGO <chr> | DESBDT <chr> | ano <dbl> | semana <dbl> | n_vacunas <dbl> | n_citas <dbl> | tmed <dbl> | |
|-----------------------|--------------------|-----------------|--------------|-----------------|--------------------|------------------|---------------|---|
| | 259 | V Centenario | 2022 | 34 | 0 | 0 | 27.278748 | 0 |
| A spec_tbl_df: 5 × 49 | 260 | Valdeacederas | 2022 | 8 | 0 | 0 | 9.577289 | 1 |
| | 041 | Canillejas | 2022 | 9 | 0 | 0 | 8.536554 | 3 |
| | 025 | Barajas | 2022 | 49 | 292 | 280 | 9.065363 | 7 |
| | 046 | Castelló | 2022 | 24 | 0 | 0 | 29.905728 | 0 |

0.6 Basic Data Transforms

0.6.1 Data Selecting

```
[68]: data |> select(1)
```

GEOCODIGO

<chr>

259

260

041

025

046

159

065

092

221

038

041

174

126

139

188

040

084

261

132

172

266

022

233

195

286

209

128

137

124

A tibble: 21736 × 1

210

121

123

125

141

143

145

154

155

163

166

175

178

180

183

186

190

196

197

211

215

0.6.2 Data Filtering

```
[69]: data |> filter(ano = 2021)
```

```
Error in `filter()`:  
! We detected a named input.  
  This usually means that you've used `=` instead of `==`.  
  Did you mean `ano == 2021`?  
Traceback:  
  
1. filter(data, ano = 2021)  
2. filter.data.frame(data, ano = 2021)  
3. check_filter(dots)  
4. abort(bullets, call = error_call)  
5. signal_abort(cnd, .file)
```

0.6.3 Insert New Column

```
[70]: data |>  
      mutate(x = TRUE)
```


| | GEOCODIGO <chr> | DESBDT <chr> | ano <dbl> | semana <dbl> | n_vacunas <dbl> | n_citas <dbl> | tmed <dbl> |
|----------------------|--------------------|------------------------|--------------|-----------------|--------------------|------------------|---------------|
| | 259 | V Centenario | 2022 | 34 | 0 | 0 | 27.278 |
| | 260 | Valdeacederas | 2022 | 8 | 0 | 0 | 9.5772 |
| | 041 | Canillejas | 2022 | 9 | 0 | 0 | 8.5365 |
| | 025 | Barajas | 2022 | 49 | 292 | 280 | 9.0653 |
| | 046 | Castelló | 2022 | 24 | 0 | 0 | 29.905 |
| | 159 | Mar Báltico | 2022 | 3 | 524 | 498 | 4.7172 |
| | 065 | Daroca | 2022 | 8 | 0 | 0 | 10.233 |
| | 092 | Felipe II | 2021 | 47 | 248 | 228 | 6.0786 |
| | 221 | Ramón y Cajal | 2023 | 1 | 204 | 198 | 8.1459 |
| | 038 | Campo Real | 2022 | 2 | 205 | 187 | 4.2450 |
| | 041 | NA | 2023 | 52 | NA | NA | 10.700 |
| | 174 | Monforte de Lemos | 2022 | 39 | 0 | 0 | 16.474 |
| | 126 | La Plata | 2022 | 16 | 0 | 0 | 10.784 |
| | 139 | Las Rozas | 2022 | 50 | 515 | 510 | 8.8428 |
| | 188 | Opañel | 2022 | 34 | 0 | 0 | 28.005 |
| | 040 | Canal de Panamá | 2021 | 37 | 0 | 0 | 19.878 |
| | 084 | Embajadores | 2022 | 11 | 0 | 0 | 10.896 |
| | 261 | Valdebernardo | 2021 | 42 | 185 | 183 | 15.813 |
| | 132 | Las Américas | 2022 | 5 | 409 | 397 | 9.6411 |
| | 172 | Miraflores | 2022 | 50 | 291 | 265 | 10.074 |
| | 266 | Valdezarza | 2022 | 33 | 0 | 0 | 24.314 |
| | 022 | Arganda - Felicidad | 2022 | 43 | 418 | 393 | 18.313 |
| | 233 | San Blas | 2022 | 50 | 333 | 326 | 10.574 |
| | 195 | Paracuellos del Jarama | 2022 | 32 | 0 | 0 | 28.061 |
| | 286 | Zofío | 2022 | 42 | 368 | 357 | 18.393 |
| | 209 | Portazgo | 2021 | 45 | 215 | 213 | 10.457 |
| | 128 | La Ribota | 2022 | 18 | 0 | 0 | 16.293 |
| | 137 | Las Matas | 2021 | 36 | 0 | 0 | 21.968 |
| | 124 | La Marazuela | 2022 | 52 | 151 | 149 | 8.4116 |
| A tibble: 21736 × 50 | 210 | Potosí | 2022 | 12 | 0 | 0 | 10.009 |
| | 121 | NA | 2023 | 52 | NA | NA | 7.3067 |
| | 123 | NA | 2023 | 52 | NA | NA | 10.211 |
| | 125 | NA | 2023 | 52 | NA | NA | 10.544 |
| | 141 | NA | 2023 | 52 | NA | NA | 11.357 |
| | 143 | NA | 2023 | 52 | NA | NA | 11.203 |
| | 145 | NA | 2023 | 52 | NA | NA | 10.166 |
| | 154 | NA | 2023 | 52 | NA | NA | 10.223 |
| | 155 | NA | 2023 | 52 | NA | NA | 11.044 |
| | 163 | NA | 2023 | 52 | NA | NA | 11.014 |
| | 166 | NA | 2023 | 52 | NA | NA | 10.142 |
| | 175 | NA | 2023 | 52 | NA | NA | 11.012 |
| | 178 | NA | 2023 | 52 | NA | NA | 11.259 |
| | 180 | NA | 2023 | 52 | NA | NA | 10.654 |
| | 183 | NA | 2023 | 52 | NA | NA | 11.527 |
| | 186 | NA | 2023 | 52 | NA | NA | 11.146 |
| | 190 | NA | 2023 | 52 | NA | NA | 11.068 |
| | 196 | NA | 2023 | 52 | NA | NA | 11.219 |
| | 197 | NA | 2023 | 52 | NA | NA | 10.361 |
| | 211 | NA | 2023 | 52 | NA | NA | 10.965 |
| | 215 | NA | 2023 | 52 | NA | NA | 11.205 |

0.6.4 Delete Column

```
[71]: data |> select(-x)
```

```
Error in `select()`:  
! Can't subset columns that don't exist.  
Column `x` doesn't exist.  
Traceback:  
  
1. select(data, -x)  
2. select.data.frame(data, -x)  
3. tidyselect::eval_select(expr(c(...)), data = .data, error_call = error_call)  
4. eval_select_impl(data, names(data), as_quosure(expr, env), include = include  
  .     exclude = exclude, strict = strict, name_spec = name_spec,  
  .     allow_rename = allow_rename, allow_empty = allow_empty, allow_predicates  
  ↪= allow_predicates,  
  .     error_call = error_call, )  
5. with_subscript_errors(out <- vars_select_eval(vars, expr, strict = strict,  
  .     data = x, name_spec = name_spec, uniquely_named = uniquely_named,  
  .     allow_rename = allow_rename, allow_empty = allow_empty, allow_predicates  
  ↪= allow_predicates,  
  .     type = type, error_call = error_call), type = type)  
6. try_fetch(expr, vctrs_error_subscript = function(cnd) {  
  .     cnd$subscript_action <- subscript_action(type)  
  .     cnd$subscript_elt <- "column"  
  .     cnd_signal(cnd)  
  . })  
7. withCallingHandlers(expr, condition = function(cnd) {  
  .     {  
  .         __handler_frame__ <- TRUE  
  .         __setup_frame__ <- frame  
  .         if (inherits(cnd, "message")) {  
  .             except <- c("warning", "error")  
  .         }  
  .         else if (inherits(cnd, "warning")) {  
  .             except <- "error"  
  .         }  
  .         else {  
  .             except <- ""  
  .         }  
  .     }  
  .     while (!is_null(cnd)) {  
  .         if (inherits(cnd, "vctrs_error_subscript")) {  
  .             out <- handlers[[1L]](cnd)  
  .             if (!inherits(out, "rlang_zap"))  
  .                 throw(out)  
  .         }  
  .         inherit <- .subset2(.subset2(cnd, "rlang"), "inherit")
```

```

.      if (is_false(inherit)) {
.          return()
.      }
.      cnd <- .subset2(cnd, "parent")
.  }
. })
8. vars_select_eval(vars, expr, strict = strict, data = x, name_spec = name_spec,
.    uniquely_named = uniquely_named, allow_rename = allow_rename,
.    allow_empty = allow_empty, allow_predicates = allow_predicates,
.    type = type, error_call = error_call)
9. walk_data_tree(expr, data_mask, context_mask)
10. eval_c(expr, data_mask, context_mask)
11. reduce_sels(node, data_mask, context_mask, init = init)
12. walk_data_tree(new, data_mask, context_mask)
13. as_indices_sel_impl(out, vars = vars, strict = strict, data = data,
.    allow_predicates = allow_predicates, call = error_call, arg =
↳as_label(expr))
14. as_indices_impl(x, vars, call = call, arg = arg, strict = strict)
15. chr_as_locations(x, vars, call = call, arg = arg)
16. vctrs::vec_as_location(x, n = length(vars), names = vars, call = call,
.    arg = arg)
17. (function ()
.  stop_subscript_oob(i = i, subscript_type = subscript_type, names = names,
.    subscript_action = subscript_action, subscript_arg = subscript_arg,
.    call = call))()
18. stop_subscript_oob(i = i, subscript_type = subscript_type, names = names,
.    subscript_action = subscript_action, subscript_arg = subscript_arg,
.    call = call)
19. stop_subscript(class = "vctrs_error_subscript_oob", i = i, subscript_type =
↳subscript_type,
.    ..., call = call)
20. abort(class = c(class, "vctrs_error_subscript"), i = i, ...,
.    call = call)
21. signal_abort(cnd, .file)
22. signalCondition(cnd)
23. (function (cnd)
.  {
.    {
.      __handler_frame__ <- TRUE
.      __setup_frame__ <- frame
.      if (inherits(cnd, "message")) {
.        except <- c("warning", "error")
.      }
.      else if (inherits(cnd, "warning")) {
.        except <- "error"
.      }
.      else {
.        except <- ""

```

```

.     }
.   }
.   while (!is_null(cnd)) {
.     if (inherits(cnd, "vctrs_error_subscript")) {
.       out <- handlers[[1L]](cnd)
.       if (!inherits(out, "rlang_zap"))
.         throw(out)
.     }
.     inherit <- .subset2(.subset2(cnd, "rlang"), "inherit")
.     if (is_false(inherit)) {
.       return()
.     }
.     cnd <- .subset2(cnd, "parent")
.   }
. }) (structure(list(message = "", trace = structure(list(call = list(
.   IRkernel::main(), kernel$run(), handle_shell(), executor$execute(msg),
.   tryCatch(evaluate(request$content$code, envir = .GlobalEnv,
.     output_handler = oh, stop_on_error = 1L), interrupt = function(cond
↪{
.     log_debug("Interrupt during execution")
.     interrupted <-< TRUE
.   }, error = .self$handle_error), tryCatchList(expr, classes,
.     parentenv, handlers), tryCatchOne(tryCatchList(expr,
.     names[-nh], parentenv, handlers[-nh]), names[nh], parentenv,
.     handlers[[nh]]), doTryCatch(return(expr), name, parentenv,
.     handler), tryCatchList(expr, names[-nh], parentenv, handlers[-nh]),
.     tryCatchOne(expr, names, parentenv, handlers[[1L]]),
↪doTryCatch(return(expr),
.     name, parentenv, handler), evaluate(request$content$code,
.     envir = .GlobalEnv, output_handler = oh, stop_on_error = 1L),
.     evaluate_call(expr, parsed$src[[i]], envir = envir, enclos = enclos,
.     debug = debug, last = i == length(out), use_try = stop_on_error !=
.     2L, keep_warning = keep_warning, keep_message = keep_message,
.     log_echo = log_echo, log_warning = log_warning, output_handler =
↪output_handler,
.     include_timing = include_timing), timing_fn(handle(ev <-
↪withCallingHandlers(withVisible(eval_with_user_handlers(expr,
.     envir, enclos, user_handlers)), warning = wHandler, error = eHandle,
.     message = mHandler))), handle(ev <-
↪withCallingHandlers(withVisible(eval_with_user_handlers(expr,
.     envir, enclos, user_handlers)), warning = wHandler, error = eHandle,
.     message = mHandler)), try(f, silent = TRUE), tryCatch(expr,
.     error = function(e) {
.       call <- conditionCall(e)
.       if (!is.null(call)) {
.         if (identical(call[[1L]], quote(doTryCatch)))
.           call <- sys.call(-4L)
.         dcall <- deparse(call, nlines = 1L)

```

```

.         prefix <- paste("Error in", dcall, ": ")
.         LONG <- 75L
.         sm <- strsplit(conditionMessage(e), "\n")[[1L]]
.         w <- 14L + nchar(dcall, type = "w") + nchar(sm[1L],
.             type = "w")
.         if (is.na(w))
.             w <- 14L + nchar(dcall, type = "b") + nchar(sm[1L],
.                 type = "b")
.         if (w > LONG)
.             prefix <- paste0(prefix, "\n ")
.     }
.     else prefix <- "Error : "
.     msg <- paste0(prefix, conditionMessage(e), "\n")
.     .Internal(seterrmessage(msg[1L]))
.     if (!silent && isTRUE(getOption("show.error.messages"))) {
.         cat(msg, file = outFile)
.         .Internal(printDeferredWarnings())
.     }
.     invisible(structure(msg, class = "try-error", condition = e))
.     }), tryCatchList(expr, classes, parentenv, handlers),
.     tryCatchOne(expr, names, parentenv, handlers[[1L]]),
↳doTryCatch(return(expr),
.     name, parentenv, handler),
↳withCallingHandlers(withVisible(eval_with_user_handlers(expr,
.     envir, enclos, user_handlers)), warning = wHandler, error = eHandle,
.     message = mHandler), withVisible(eval_with_user_handlers(expr,
.     envir, enclos, user_handlers)), eval_with_user_handlers(expr,
.     envir, enclos, user_handlers), eval(expr, envir, enclos),
.     eval(expr, envir, enclos), select(data, -x), select.data.frame(data,
.     -x), tidyselect::eval_select(expr(c(...)), data = .data,
.     error_call = error_call), eval_select_impl(data, names(data),
.     as_quosure(expr, env), include = include, exclude = exclude,
.     strict = strict, name_spec = name_spec, allow_rename = allow_rename,
.     allow_empty = allow_empty, allow_predicates = allow_predicates,
.     error_call = error_call, ), with_subscript_errors(out <-
↳vars_select_eval(vars,
.     expr, strict = strict, data = x, name_spec = name_spec,
.     uniquely_named = uniquely_named, allow_rename = allow_rename,
.     allow_empty = allow_empty, allow_predicates = allow_predicates,
.     type = type, error_call = error_call), type = type),
.     try_fetch(expr, vctrs_error_subscript = function(cnd) {
.         cnd$subscript_action <- subscript_action(type)
.         cnd$subscript_elt <- "column"
.         cnd_signal(cnd)
.     })), withCallingHandlers(expr, condition = function(cnd) {
.     {
.         .__handler_frame__. <- TRUE
.         .__setup_frame__. <- frame

```

```

.         if (inherits(cnd, "message")) {
.             except <- c("warning", "error")
.         }
.         else if (inherits(cnd, "warning")) {
.             except <- "error"
.         }
.         else {
.             except <- ""
.         }
.     }
.     while (!is_null(cnd)) {
.         if (inherits(cnd, "vctrs_error_subscript")) {
.             out <- handlers[[1L]](cnd)
.             if (!inherits(out, "rlang_zap"))
.                 throw(out)
.         }
.         inherit <- .subset2(.subset2(cnd, "rlang"), "inherit")
.         if (is_false(inherit)) {
.             return()
.         }
.         cnd <- .subset2(cnd, "parent")
.     }
. }, vars_select_eval(vars, expr, strict = strict, data = x,
.     name_spec = name_spec, uniquely_named = uniquely_named,
.     allow_rename = allow_rename, allow_empty = allow_empty,
.     allow_predicates = allow_predicates, type = type, error_call =
↳error_call),
.     walk_data_tree(expr, data_mask, context_mask), eval_c(expr,
.     data_mask, context_mask), reduce_sels(node, data_mask,
.     context_mask, init = init), walk_data_tree(new, data_mask,
.     context_mask), as_indices_sel_impl(out, vars = vars,
.     strict = strict, data = data, allow_predicates = allow_predicates,
.     call = error_call, arg = as_label(expr)), as_indices_impl(x,
.     vars, call = call, arg = arg, strict = strict), chr_as_locations(x,
.     vars, call = call, arg = arg), vctrs::vec_as_location(x,
.     n = length(vars), names = vars, call = call, arg = arg),
.     `<fn>`() , stop_subscript_oob(i = i, subscript_type = subscript_type,
.     names = names, subscript_action = subscript_action, subscript_arg =
↳subscript_arg,
.     call = call), stop_subscript(class = "vctrs_error_subscript_oob",
.     i = i, subscript_type = subscript_type, ..., call = call),
.     abort(class = c(class, "vctrs_error_subscript"), i = i, ...,
.     call = call)), parent = c(0L, 1L, 2L, 3L, 4L, 5L, 6L,
.     7L, 6L, 9L, 10L, 4L, 12L, 13L, 13L, 15L, 16L, 17L, 18L, 19L,
.     13L, 13L, 13L, 23L, 24L, 0L, 0L, 27L, 28L, 29L, 30L, 31L, 29L,
.     33L, 34L, 35L, 36L, 37L, 38L, 39L, 40L, 0L, 42L, 43L, 44L), visible = c(TRU ,
.     TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE,
.     TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE,

```

```

. TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
. FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
. FALSE, FALSE, FALSE), namespace = c("IRkernel", NA, "IRkernel",
. NA, "base", "base", "base", "base", "base", "base", "base", "base", "evaluate",
. "evaluate", "evaluate", "evaluate", "base", "base", "base", "base",
. "base", "base", "base", "evaluate", "base", "base", "dplyr",
. "dplyr", "tidyselect", "tidyselect", "tidyselect", "rlang", "base",
. "tidyselect", "tidyselect", "tidyselect", "tidyselect", "tidyselect",
. "tidyselect", "tidyselect", "tidyselect", "vctrs", "vctrs", "vctrs",
. "vctrs", "rlang"), scope = c("::", NA, "local", NA, "::", "local",
. "local", "local", "local", "local", "local", "::", ":::", "local",
. "local", "::", "::", "local", "local", "local", "::", "::", ":::",
. "::", "::", "::", ":::", "::", ":::", ":::", "::", "::", ":::",
. ":::", ":::", ":::", ":::", ":::", ":::", ":::", "::", "local",
. ":::", ":::", "::"), error_frame = c(FALSE, FALSE, FALSE, FALSE,
. FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
. FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
. FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, FALSE, FALSE, FALSE,
. FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
. FALSE, FALSE, FALSE, FALSE, FALSE)), row.names = c(NA, -45L), version = 2L,
↪class = c("rlang_trace",
. "rlib_trace", "tbl", "data.frame")), parent = NULL, i = "x",
. subscript_type = "character", names = c("GEOCODIGO", "DESBDT",
. "ano", "semana", "n_vacunas", "n_citas", "tmed", "prec",
. "velmedia", "presMax", "benzene", "co", "no", "no2", "nox",
. "o3", "pm10", "pm2.5", "so2", "campana", "scampana", "capacidad_zona",
. "prop_riesgo", "tasa_riesgo", "tasa_mayores", "poblacion_mayores",
. "nombre_zona", "nsec", "t3_1", "t1_1", "t2_1", "t2_2", "t4_1",
. "t4_2", "t4_3", "t5_1", "t6_1", "t7_1", "t8_1", "t9_1", "t10_1",
. "t11_1", "t12_1", "area", "densidad_hab_km", "tuits_gripe",
. "interes_gripe", "Target", "is_train"), subscript_action = NULL,
. subscript_arg = "x", rlang = list(inherit = TRUE), call = select(data,
. -x)), class = c("vctrs_error_subscript_oob", "vctrs_error_subscript",
. "rlang_error", "error", "condition"))
24. handlers[[1L]](cnd)
25. cnd_signal(cnd)
26. signal_abort(cnd)

```

0.6.5 Rank Data

Operation

```
[76]: data |> mutate(rank = order(Target))
```

| | GEOCODIGO <chr> | DESBDT <chr> | ano <dbl> | semana <dbl> | n_vacunas <dbl> | n_citas <dbl> | tmed <dbl> |
|----------------------|--------------------|------------------------|--------------|-----------------|--------------------|------------------|---------------|
| | 259 | V Centenario | 2022 | 34 | 0 | 0 | 27.278 |
| | 260 | Valdeacederas | 2022 | 8 | 0 | 0 | 9.5772 |
| | 041 | Canillejas | 2022 | 9 | 0 | 0 | 8.5365 |
| | 025 | Barajas | 2022 | 49 | 292 | 280 | 9.0653 |
| | 046 | Castelló | 2022 | 24 | 0 | 0 | 29.903 |
| | 159 | Mar Báltico | 2022 | 3 | 524 | 498 | 4.7172 |
| | 065 | Daroca | 2022 | 8 | 0 | 0 | 10.233 |
| | 092 | Felipe II | 2021 | 47 | 248 | 228 | 6.0786 |
| | 221 | Ramón y Cajal | 2023 | 1 | 204 | 198 | 8.1459 |
| | 038 | Campo Real | 2022 | 2 | 205 | 187 | 4.2450 |
| | 041 | NA | 2023 | 52 | NA | NA | 10.700 |
| | 174 | Monforte de Lemos | 2022 | 39 | 0 | 0 | 16.474 |
| | 126 | La Plata | 2022 | 16 | 0 | 0 | 10.784 |
| | 139 | Las Rozas | 2022 | 50 | 515 | 510 | 8.8428 |
| | 188 | Opañel | 2022 | 34 | 0 | 0 | 28.005 |
| | 040 | Canal de Panamá | 2021 | 37 | 0 | 0 | 19.878 |
| | 084 | Embajadores | 2022 | 11 | 0 | 0 | 10.896 |
| | 261 | Valdebernardo | 2021 | 42 | 185 | 183 | 15.813 |
| | 132 | Las Américas | 2022 | 5 | 409 | 397 | 9.6411 |
| | 172 | Miraflores | 2022 | 50 | 291 | 265 | 10.074 |
| | 266 | Valdezarza | 2022 | 33 | 0 | 0 | 24.314 |
| | 022 | Arganda - Felicidad | 2022 | 43 | 418 | 393 | 18.313 |
| | 233 | San Blas | 2022 | 50 | 333 | 326 | 10.574 |
| | 195 | Paracuellos del Jarama | 2022 | 32 | 0 | 0 | 28.061 |
| | 286 | Zofío | 2022 | 42 | 368 | 357 | 18.393 |
| | 209 | Portazgo | 2021 | 45 | 215 | 213 | 10.457 |
| | 128 | La Ribota | 2022 | 18 | 0 | 0 | 16.293 |
| | 137 | Las Matas | 2021 | 36 | 0 | 0 | 21.968 |
| | 124 | La Marazuela | 2022 | 52 | 151 | 149 | 8.4116 |
| A tibble: 21736 × 50 | 210 | Potosí | 2022 | 12 | 0 | 0 | 10.009 |
| | 121 | NA | 2023 | 52 | NA | NA | 7.3067 |
| | 123 | NA | 2023 | 52 | NA | NA | 10.211 |
| | 125 | NA | 2023 | 52 | NA | NA | 10.544 |
| | 141 | NA | 2023 | 52 | NA | NA | 11.357 |
| | 143 | NA | 2023 | 52 | NA | NA | 11.203 |
| | 145 | NA | 2023 | 52 | NA | NA | 10.166 |
| | 154 | NA | 2023 | 52 | NA | NA | 10.223 |
| | 155 | NA | 2023 | 52 | NA | NA | 11.044 |
| | 163 | NA | 2023 | 52 | NA | NA | 11.014 |
| | 166 | NA | 2023 | 52 | NA | NA | 10.142 |
| | 175 | NA | 2023 | 52 | NA | NA | 11.012 |
| | 178 | NA | 2023 | 52 | NA | NA | 11.259 |
| | 180 | NA | 2023 | 52 | NA | NA | 10.654 |
| | 183 | NA | 2023 | 52 | NA | NA | 11.527 |
| | 186 | NA | 2023 | 52 | NA | NA | 11.146 |
| | 190 | NA | 2023 | 52 | NA | NA | 11.068 |
| | 196 | NA | 2023 | 52 | NA | NA | 11.219 |
| | 197 | NA | 2023 | 52 | NA | NA | 10.361 |
| | 211 | NA | 2023 | 52 | NA | NA | 10.965 |
| | 215 | NA | 2023 | 52 | NA | NA | 11.205 |

0.7 Numeric Variable Transformation: Scale

0.7.1 Normalization Transform

Select columns

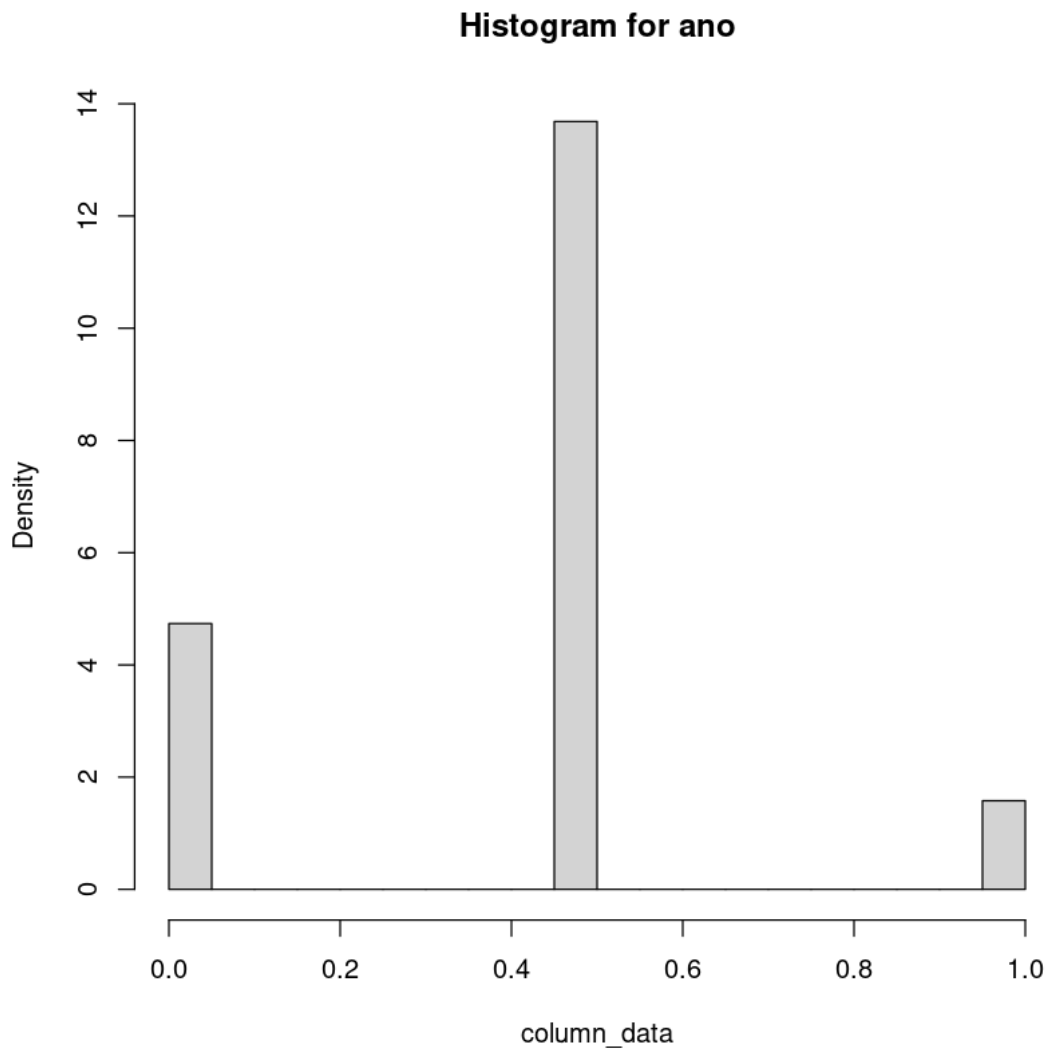
```
[21]: cols <- sapply(data, is.numeric)
```

Operation

```
[25]: data_normalized <- data
for (col_name in names(numeric_cols)[numeric_cols]) {
  data_normalized[[col_name]] <- (data[[col_name]] - min(data[[col_name]])) /
                                (max(data[[col_name]]) - min(data[[col_name]]))
}

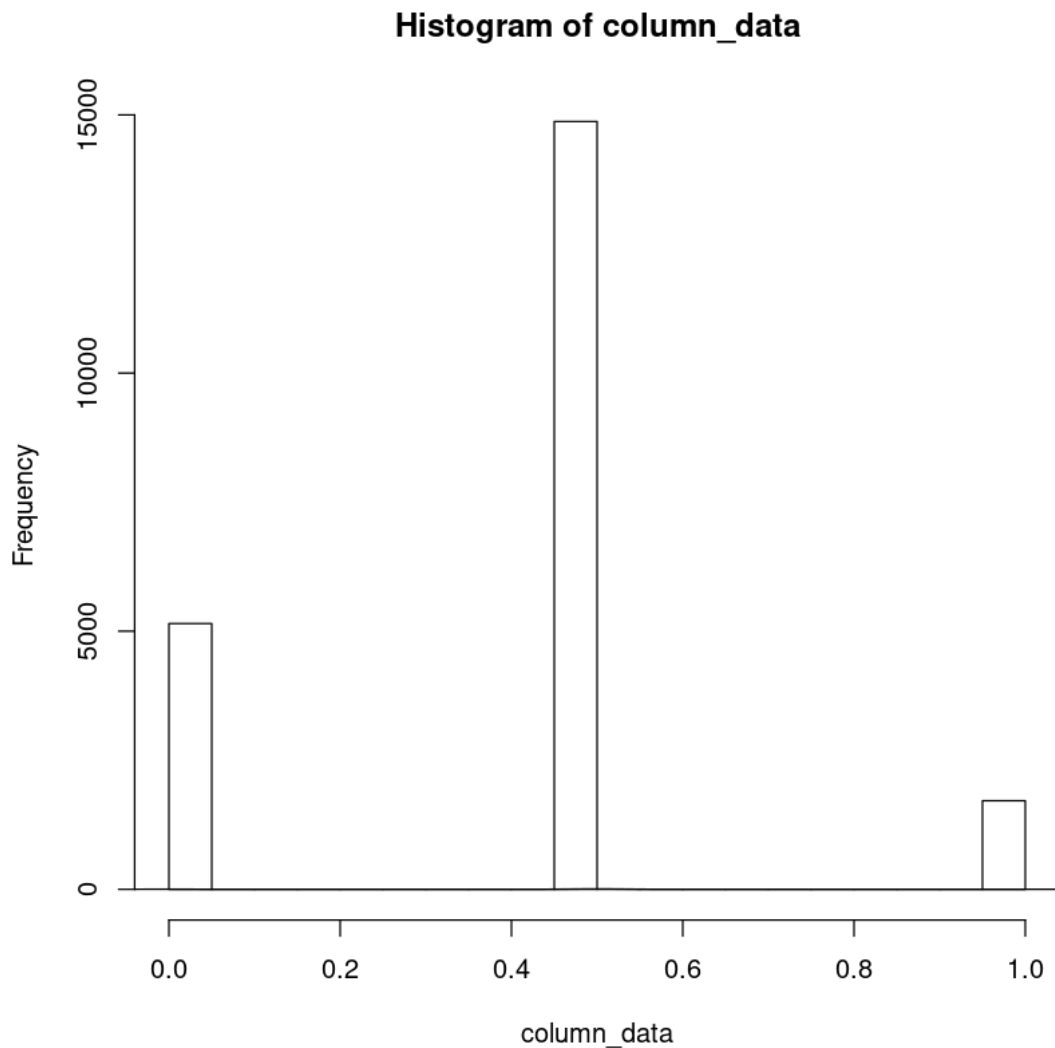
for (col_name in names(cols)[cols]) {
  cat("Processing column:", col_name, "\n")
  column_data <- data_normalized[[col_name]]
  hist_plot <- hist(column_data, freq = FALSE, main = paste("Histogram for", col_name))
  print(plot(hist_plot))
  lines(density(column_data))
}
```

Processing column: ano



NULL

Processing column: semana



NULL

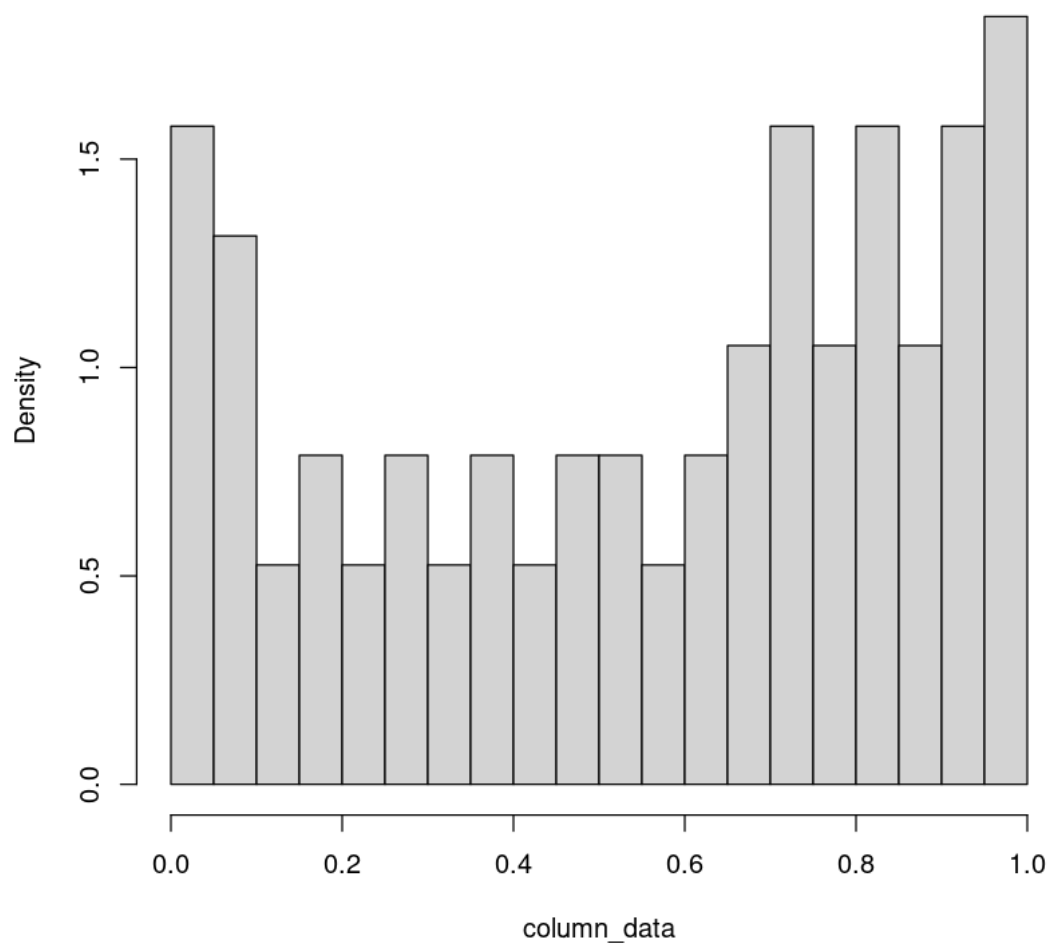
Processing column: n_vacunas

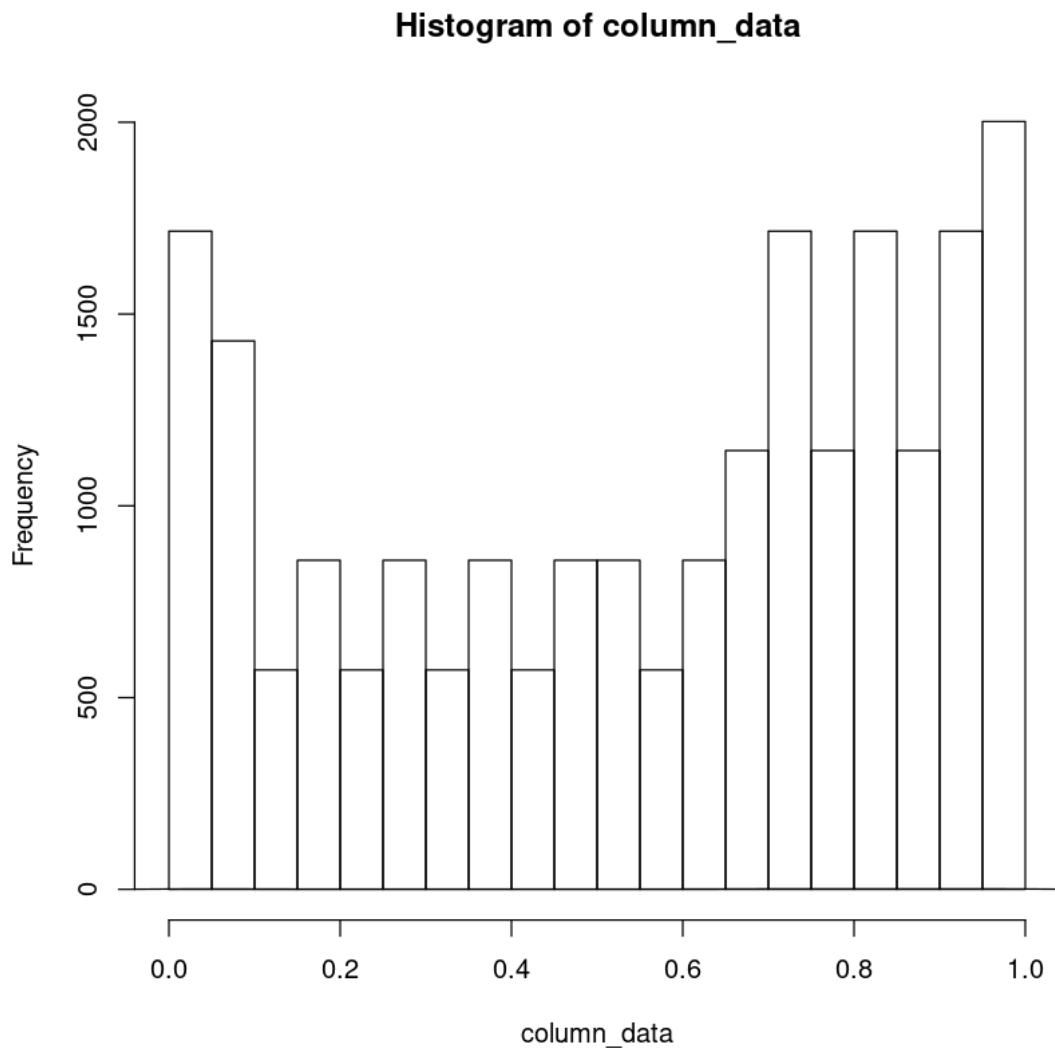
```
Error in hist.default(column_data, freq = FALSE, main = paste("Histogram for",  
  ↪invalid number of 'breaks')
```

Traceback:

```
1. hist(column_data, freq = FALSE, main = paste("Histogram for",  
  . col_name))  
2. hist.default(column_data, freq = FALSE, main = paste("Histogram for",  
  . col_name))  
3. stop("invalid number of 'breaks'")
```

Histogram for semana





[]:

0.7.2 Standarization Transform

Select columns

```
[27]: cols <- sapply(data, is.numeric)
```

Operation

```
[26]: data_standardized <- data

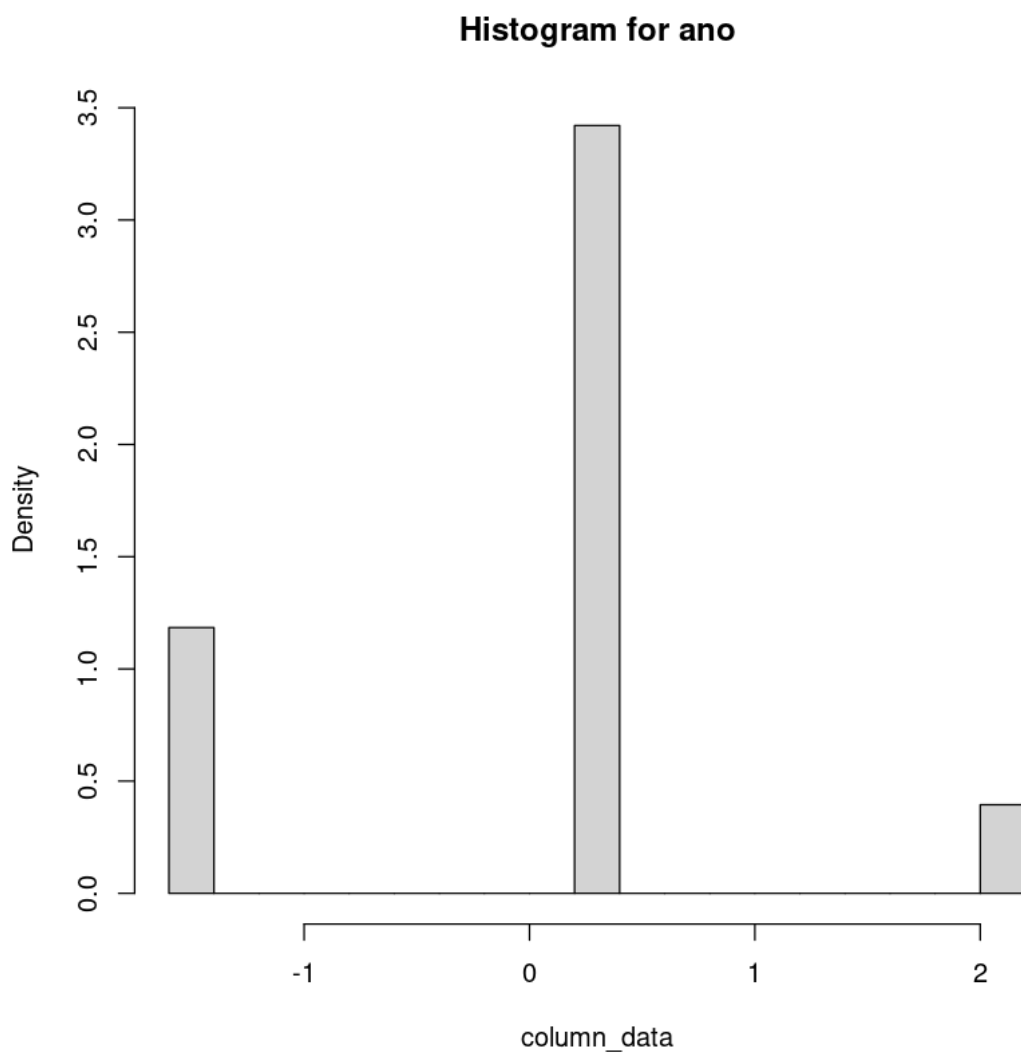
for (col_name in names(numeric_cols)[numeric_cols]) {
  data_standardized[[col_name]] <- scale(data[[col_name]])
}
```

```

}
for (col_name in names(cols)[cols]) {
  cat("Processing column:", col_name, "\n")
  column_data <- data_standardized[[col_name]]
  hist_plot <- hist(column_data, freq = FALSE, main = paste("Histogram for", col_name))
  print(plot(hist_plot))
  lines(density(column_data))
}

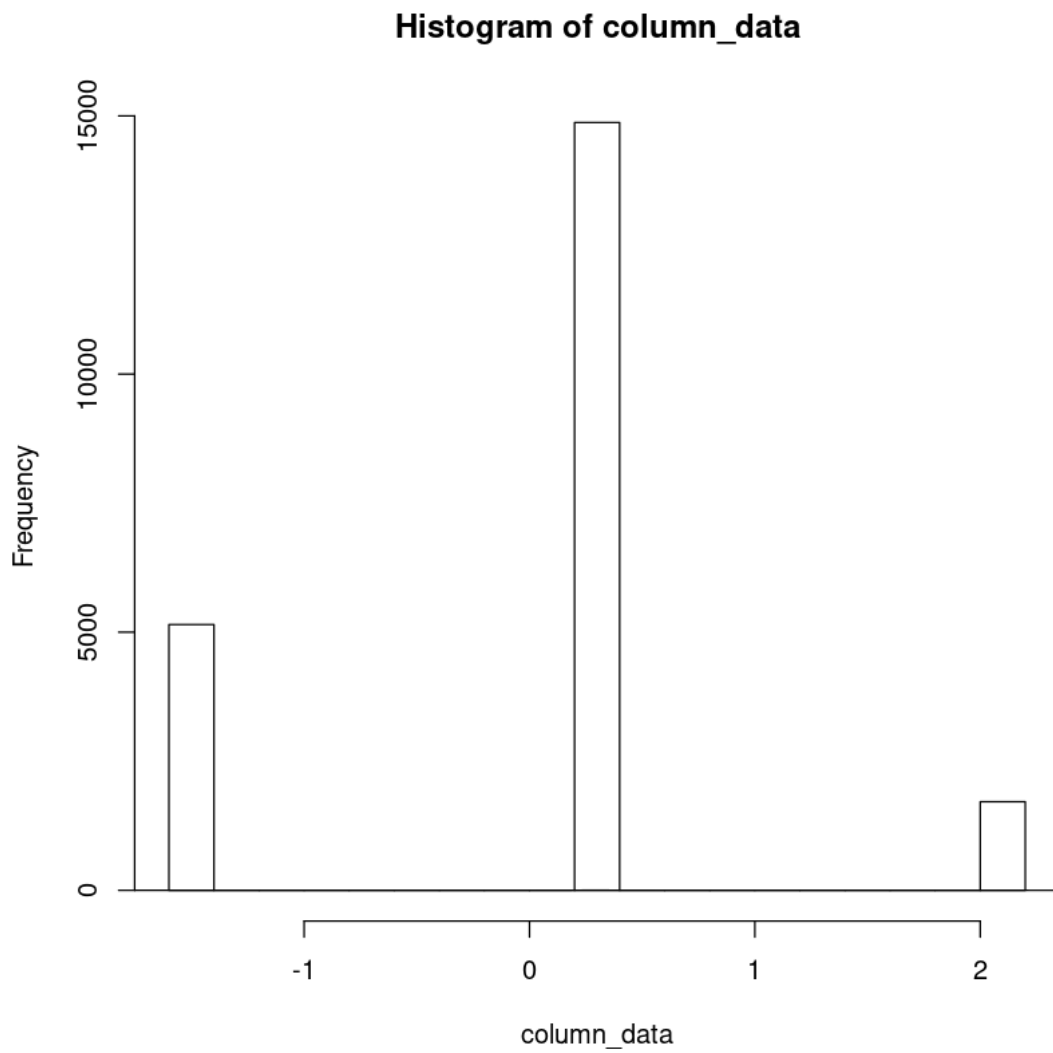
```

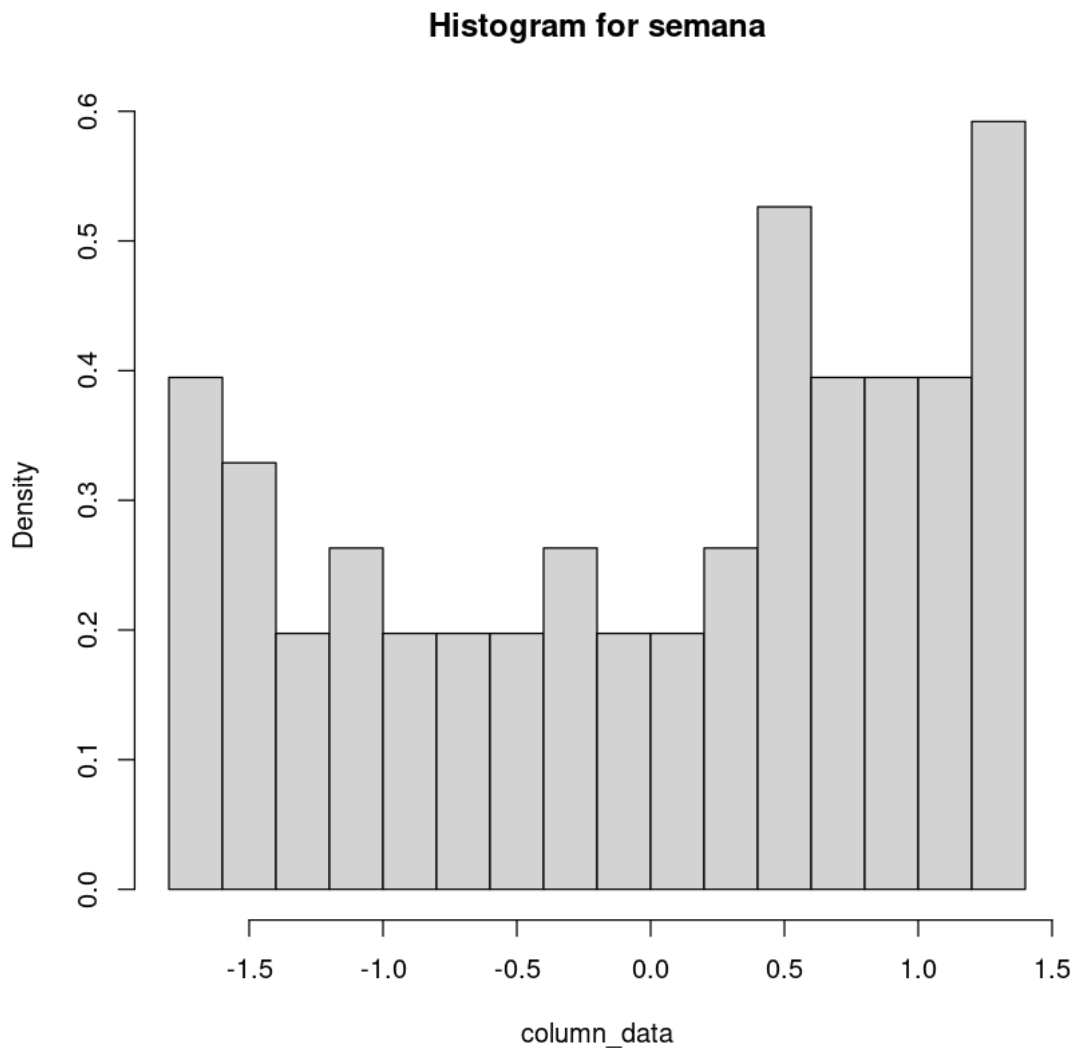
Processing column: ano



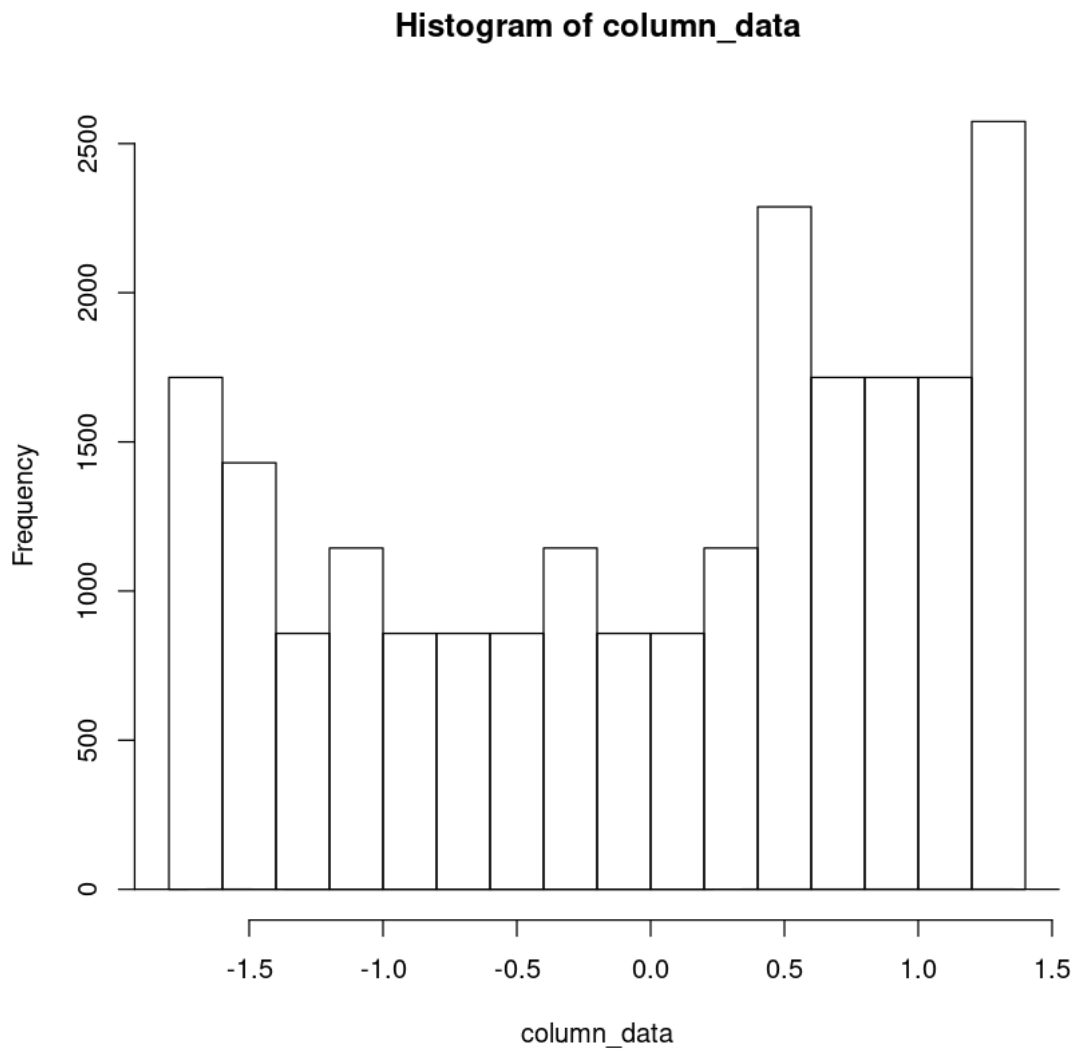
NULL

Processing column: semana





NULL
Processing column: n_vacunas

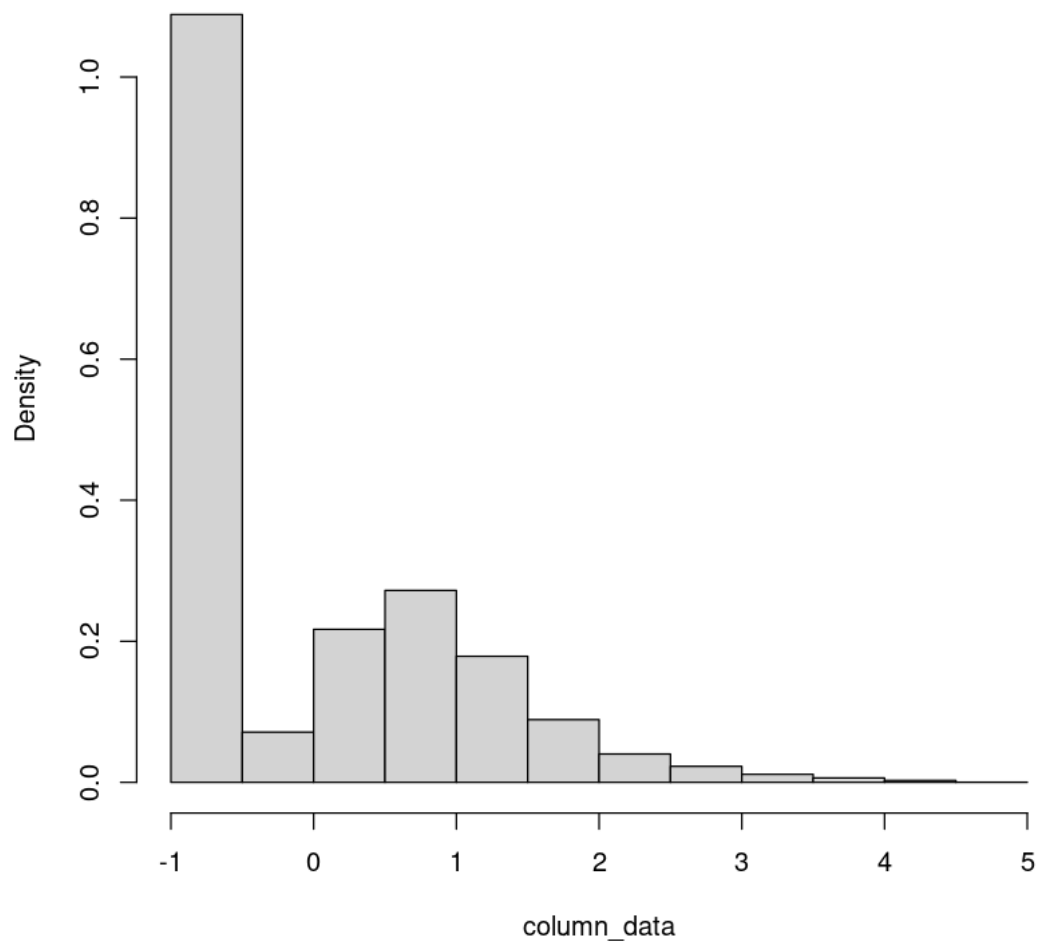


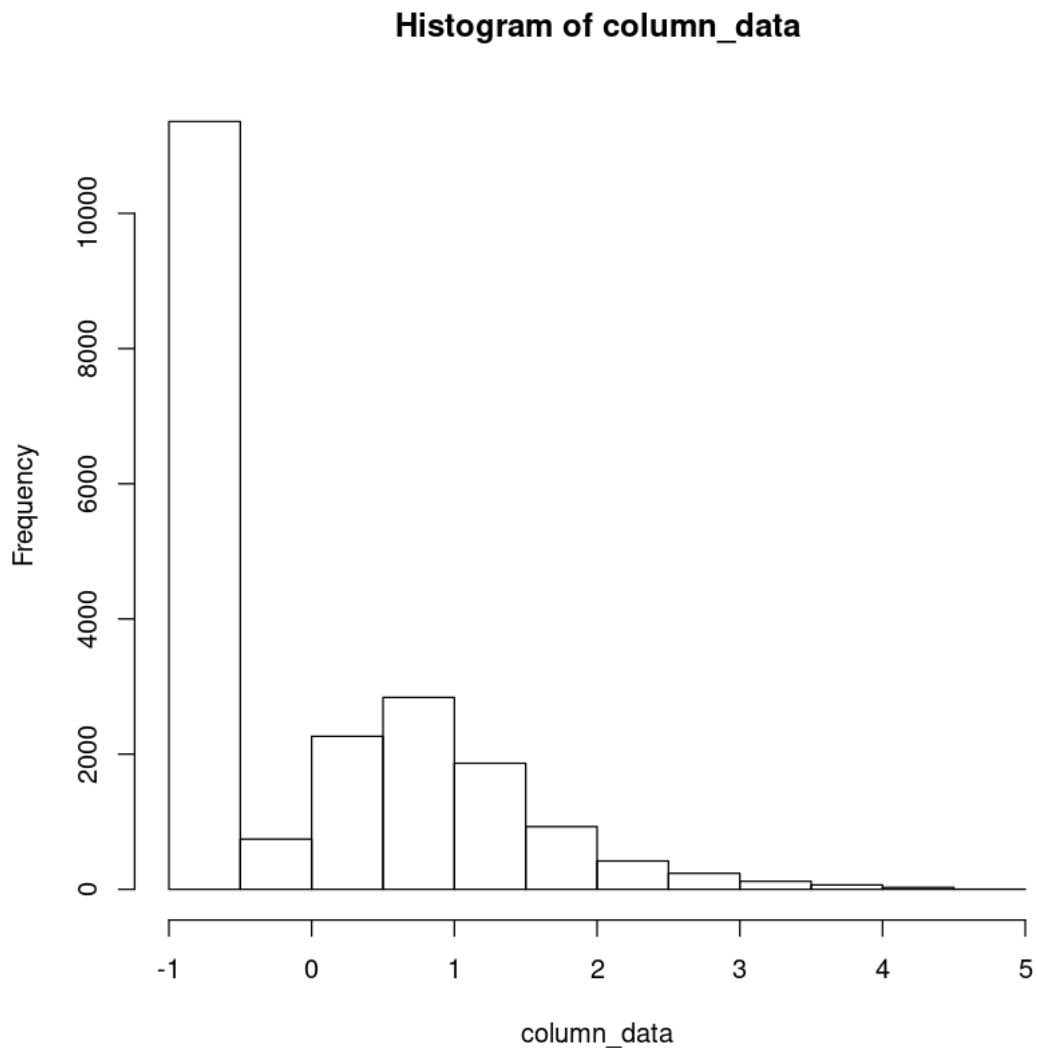
NULL

```
Error in density.default(column_data): 'x' contains missing values
Traceback:
```

```
1. lines(density(column_data))
2. density(column_data)
3. density.default(column_data)
4. stop("'x' contains missing values")
```

Histogram for n_vacunas





0.8 Numeric Variable Transformation: Distribution

0.8.1 Discretization Transform

Evaluating Discretization Transformations

Uniform Discretization Transform Select columns

```
[28]: cols <- sapply(data, is.numeric)
```

```
[ ]:
```

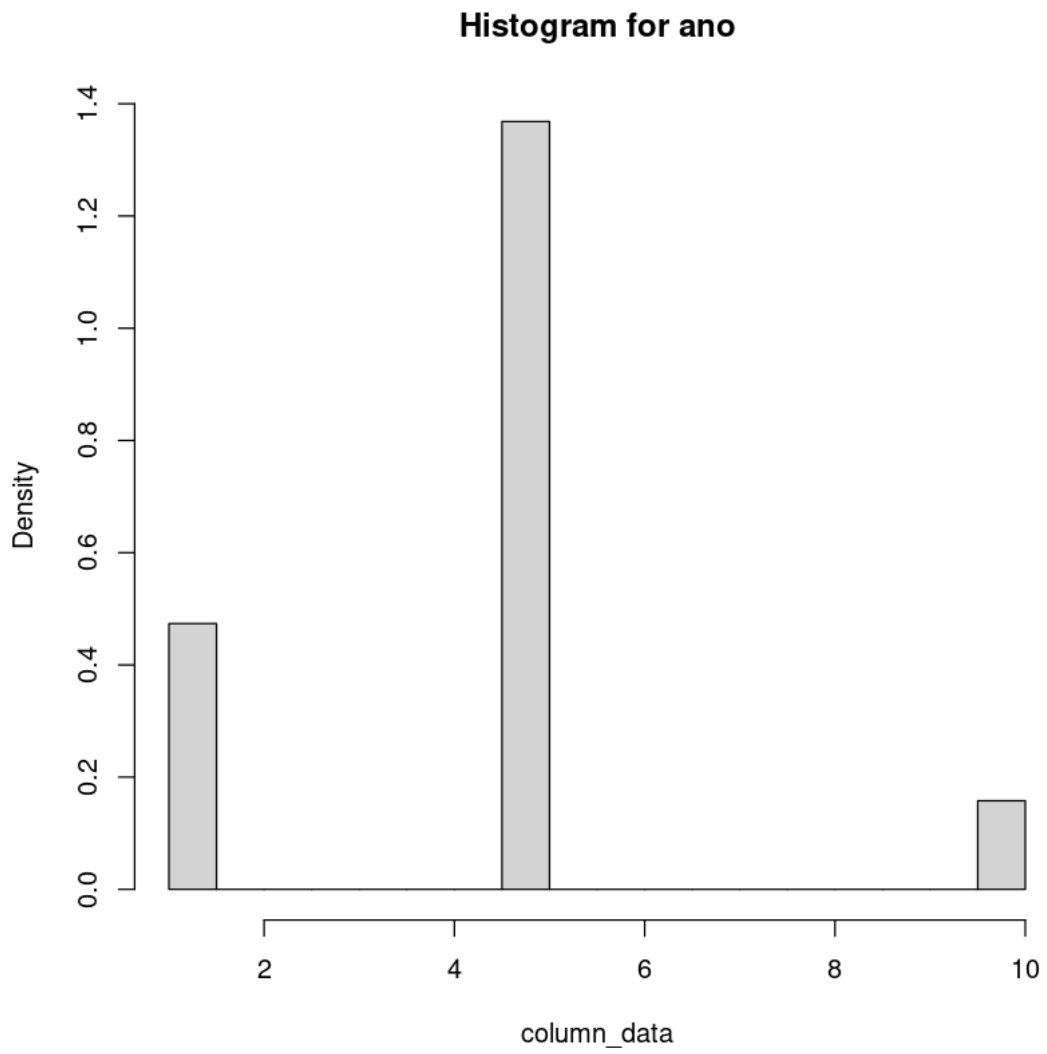
Operation

```
[32]: # create a copy of the original data frame
data_discretized <- data

# number of intervals
k <- 10 # change this value according to your needs

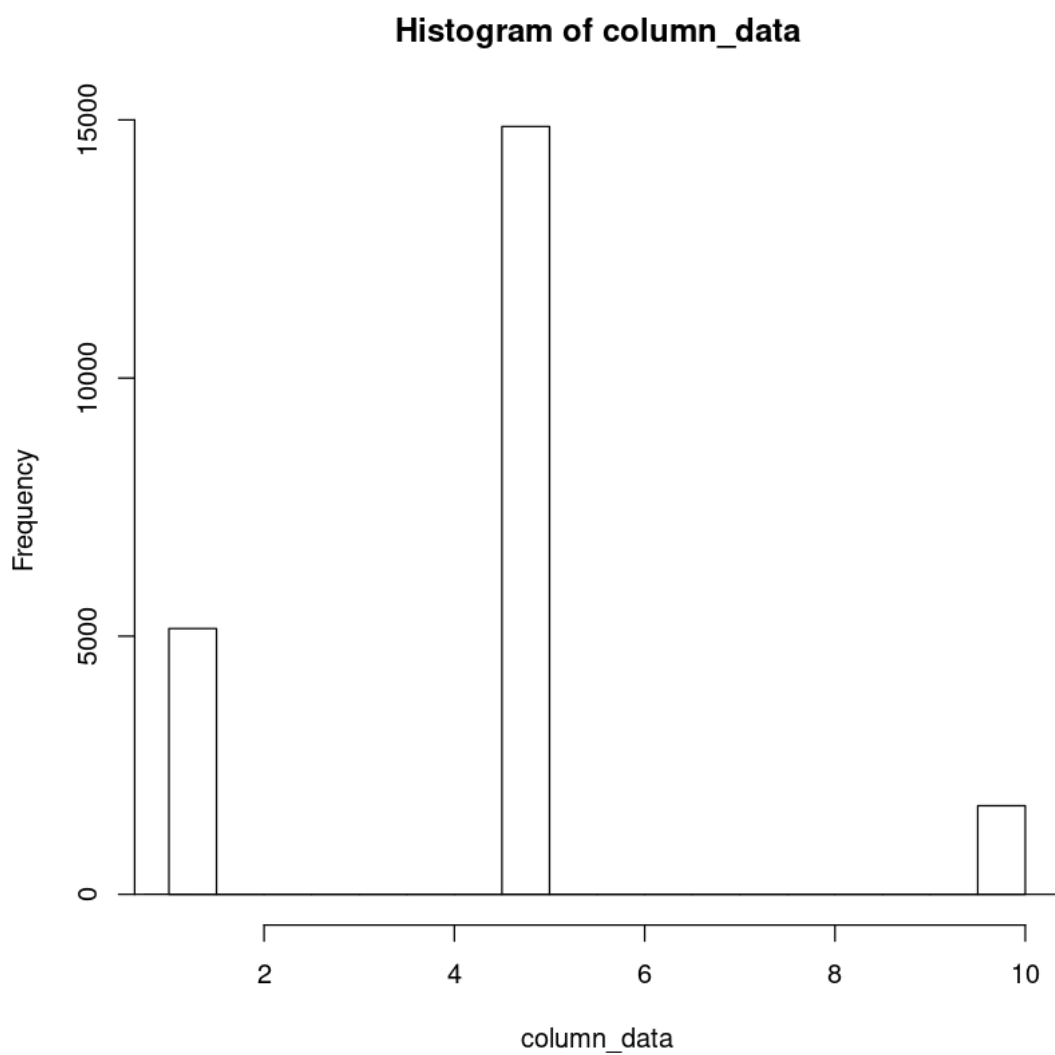
# discretize each numeric column
for (col_name in names(numeric_cols)[numeric_cols]) {
  data_discretized[[col_name]] <- cut(data[[col_name]], breaks = k, labels = FALSE)
}
for (col_name in names(cols)[cols]) {
  cat("Processing column:", col_name, "\n")
  column_data <- data_discretized[[col_name]]
  hist_plot <- hist(column_data, freq = FALSE, main = paste("Histogram for", col_name))
  print(plot(hist_plot))
  lines(density(column_data))
}
```

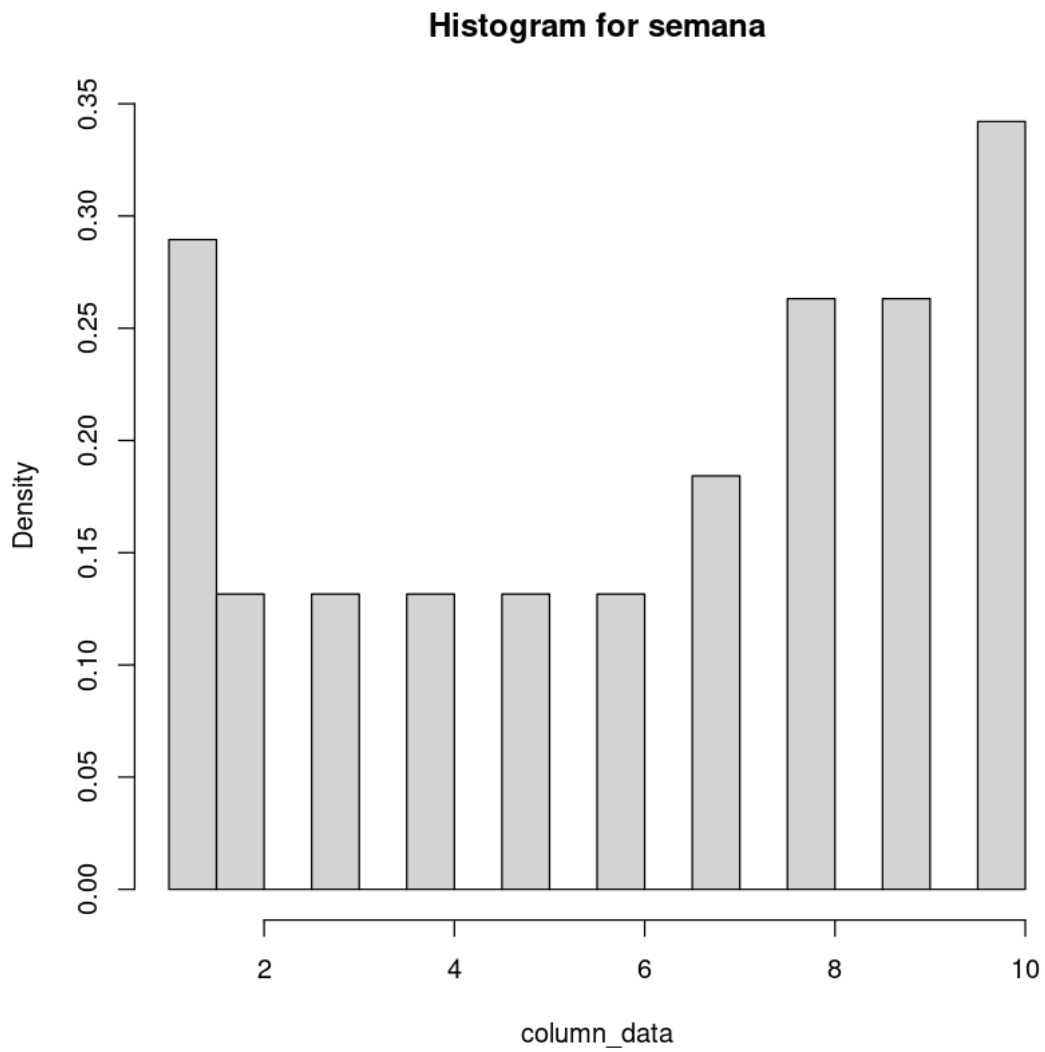
Processing column: ano



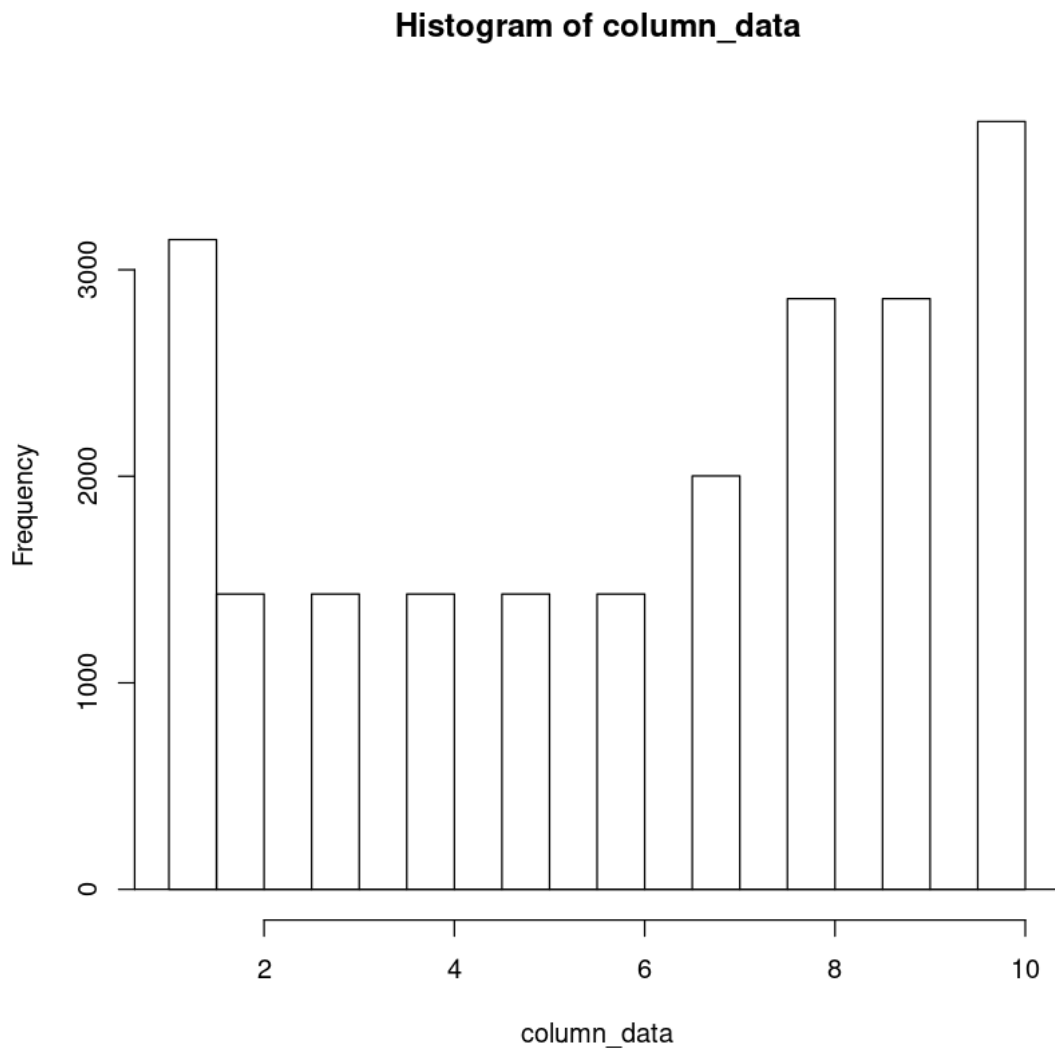
NULL

Processing column: semana





NULL
Processing column: n_vacunas

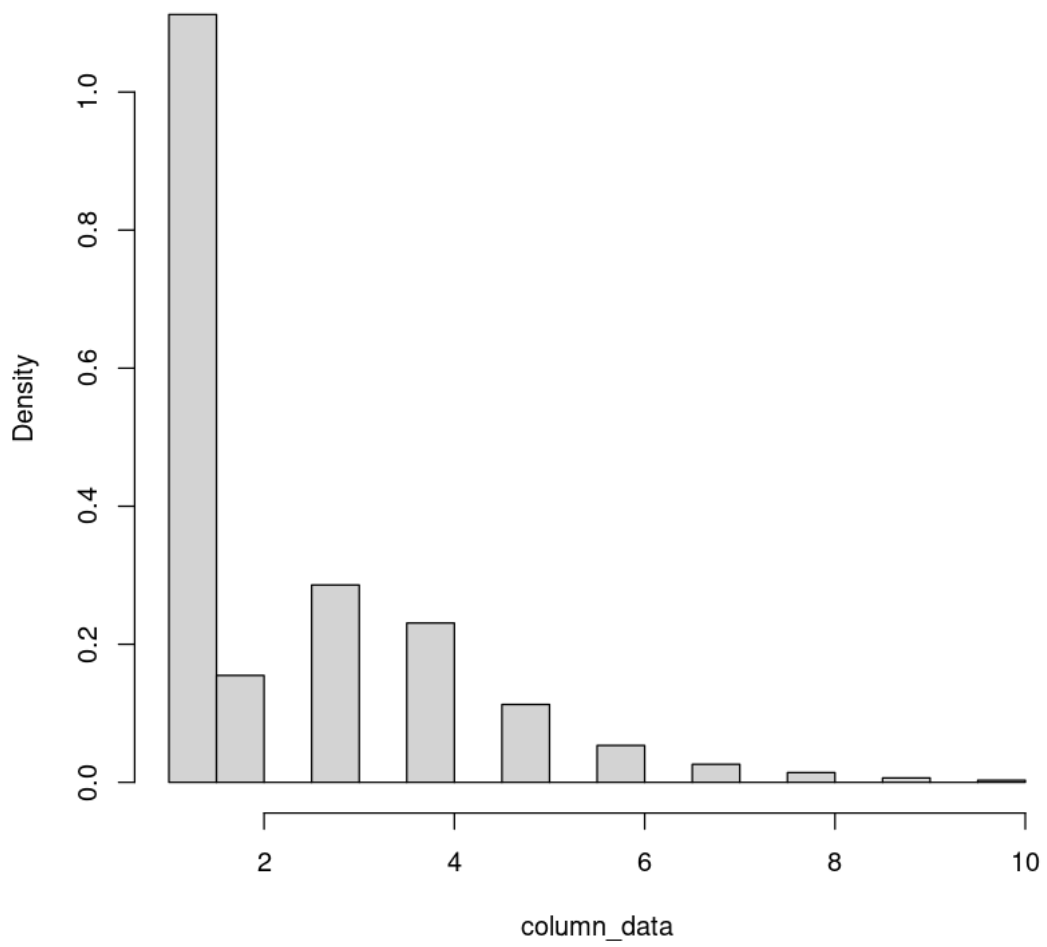


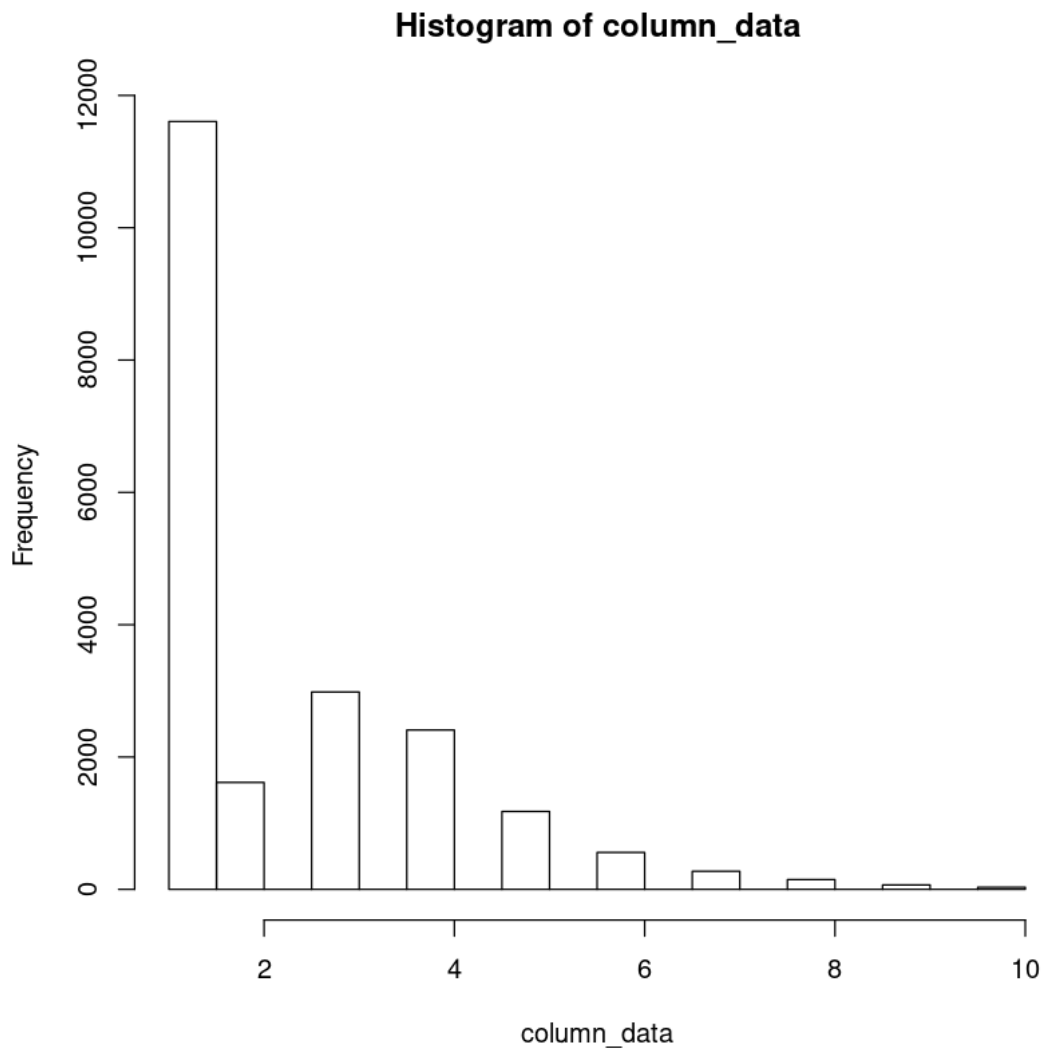
NULL

```
Error in density.default(column_data): 'x' contains missing values
Traceback:
```

```
1. lines(density(column_data))
2. density(column_data)
3. density.default(column_data)
4. stop("'x' contains missing values")
```


Histogram for n_vacunas





0.8.2 Power Transform

Data to Transform

Evaluating Yeo-Johnson tranform

Yeo-Johnson Transform Select columns

```
[37]: cols <- sapply(data, is.numeric)
```

Operation

```
[39]: if (!require(bestNormalize)) {
      install.packages('bestNormalize')
    }

    # load the bestNormalize package
    library(bestNormalize)

    # assuming 'data' is your data frame

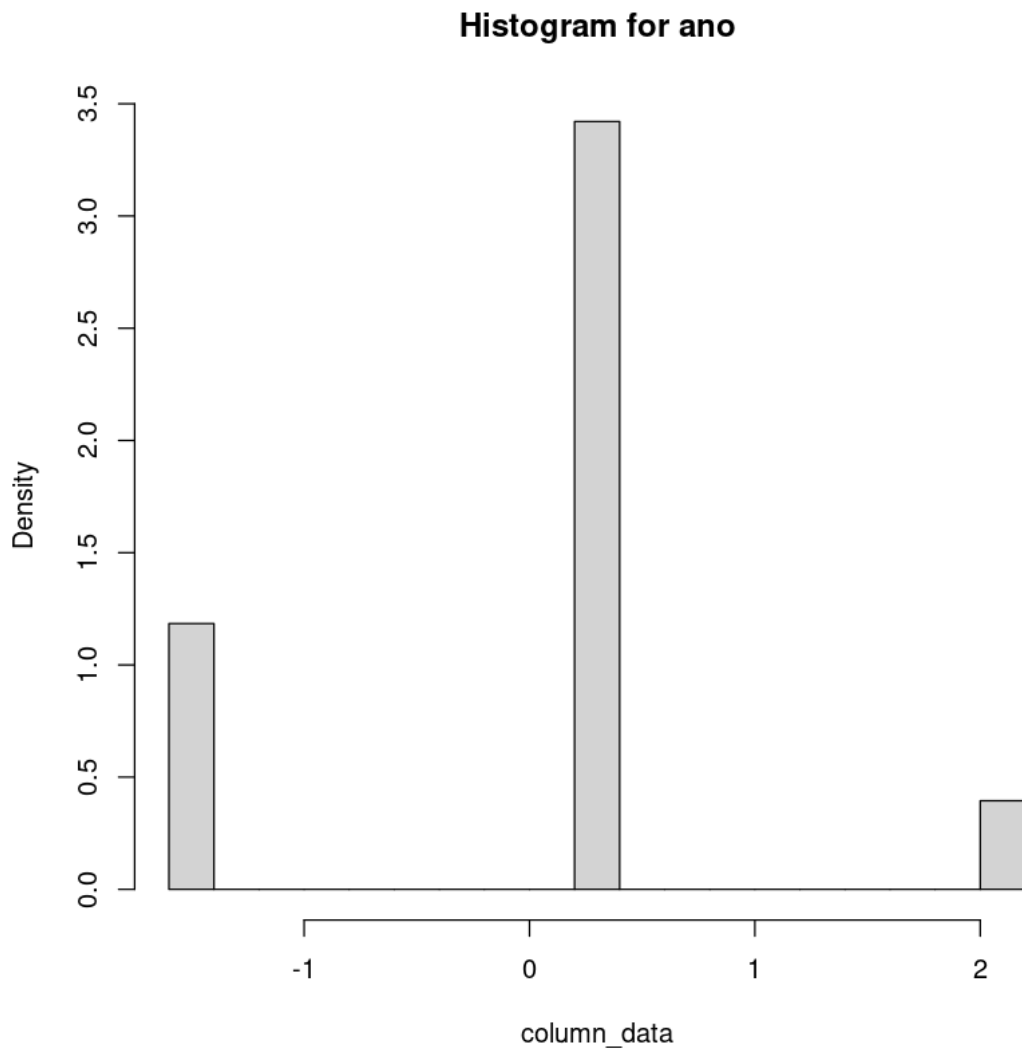
    # identify the numeric columns
    numeric_cols <- sapply(data, is.numeric)

    # create a copy of the original data frame
    data_yeojohnson <- data

    # apply the Yeo-Johnson transformation to each numeric column
    for (col_name in names(numeric_cols)[numeric_cols]) {
      yj <- yeojohnson(data[[col_name]])
      data_yeojohnson[[col_name]] <- yj$x.t
    }
    for (col_name in names(cols)[cols]) {
      cat("Processing column:", col_name, "\n")
      column_data <- data_yeojohnson[[col_name]]
      hist_plot <- hist(column_data, freq = FALSE, main = paste("Histogram for", col_name))
      print(plot(hist_plot))
      lines(density(column_data))
    }
  }
```

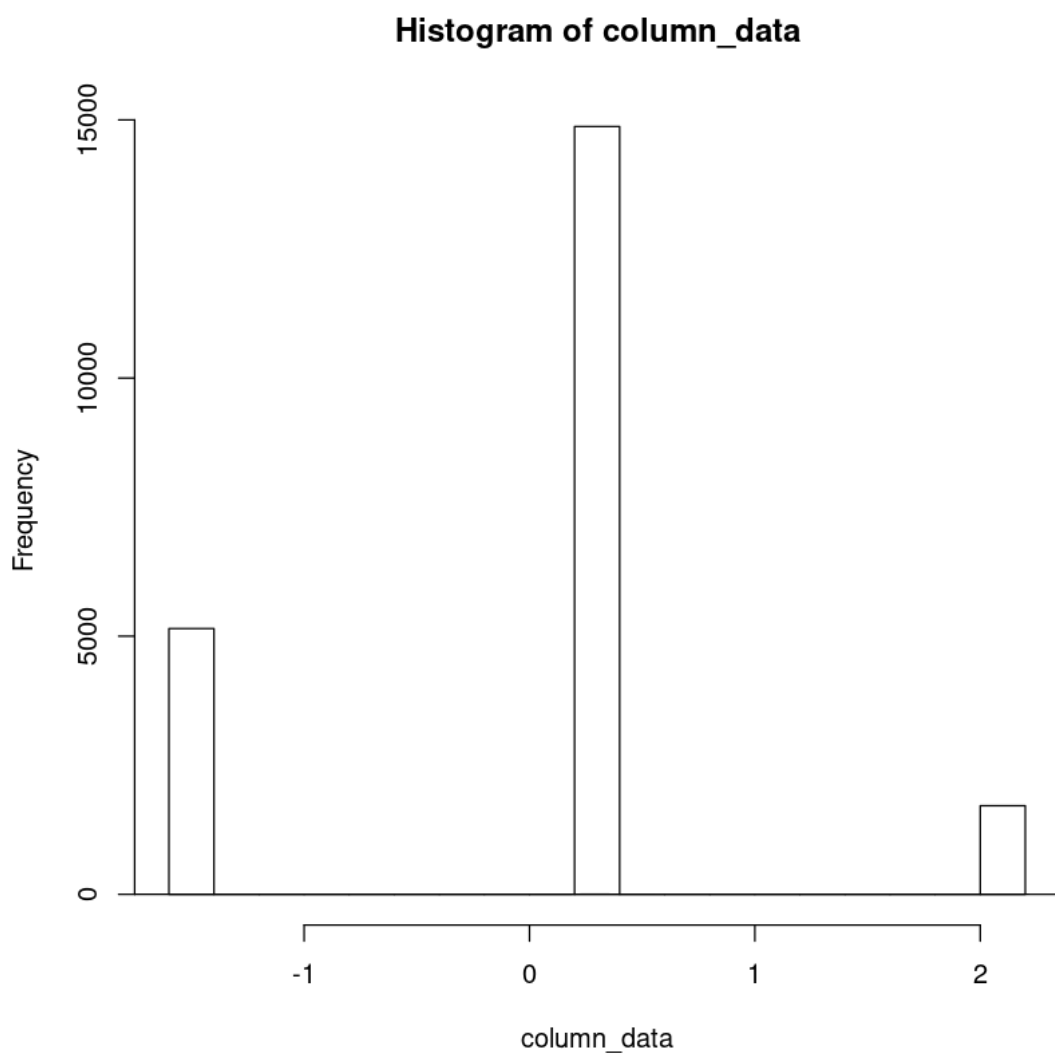
```
Warning message in optimize(yj_loglik, lower = lower, upper = upper, maximum =
TRUE, :
"NA/Inf replaced by maximum positive value"
Warning message in optimize(yj_loglik, lower = lower, upper = upper, maximum =
TRUE, :
"NA/Inf replaced by maximum positive value"
Warning message in optimize(yj_loglik, lower = lower, upper = upper, maximum =
TRUE, :
"NA/Inf replaced by maximum positive value"
Warning message in optimize(yj_loglik, lower = lower, upper = upper, maximum =
TRUE, :
"NA/Inf replaced by maximum positive value"
Warning message in optimize(yj_loglik, lower = lower, upper = upper, maximum =
TRUE, :
"NA/Inf replaced by maximum positive value"
Warning message in optimize(yj_loglik, lower = lower, upper = upper, maximum =
TRUE, :
"NA/Inf replaced by maximum positive value"
```

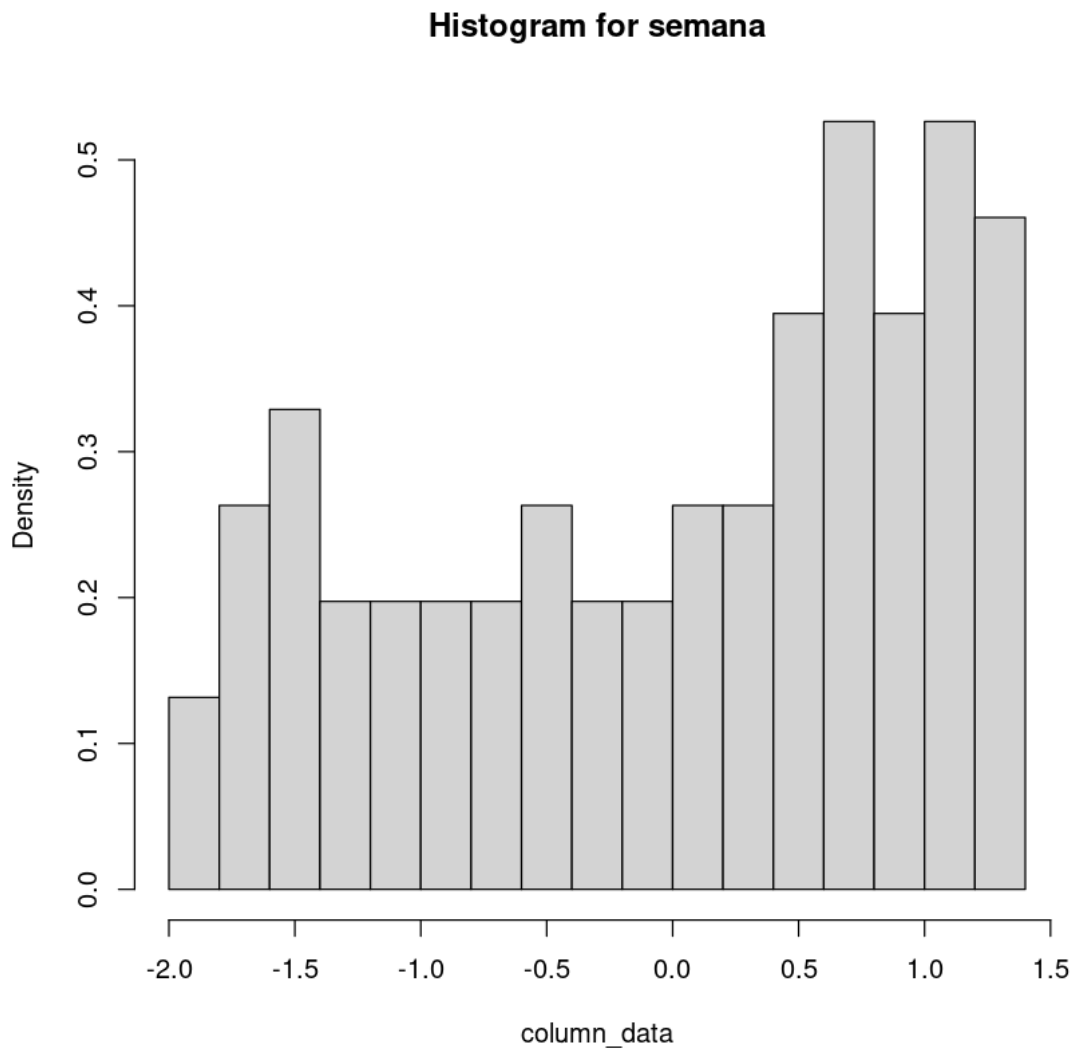
[illegible]



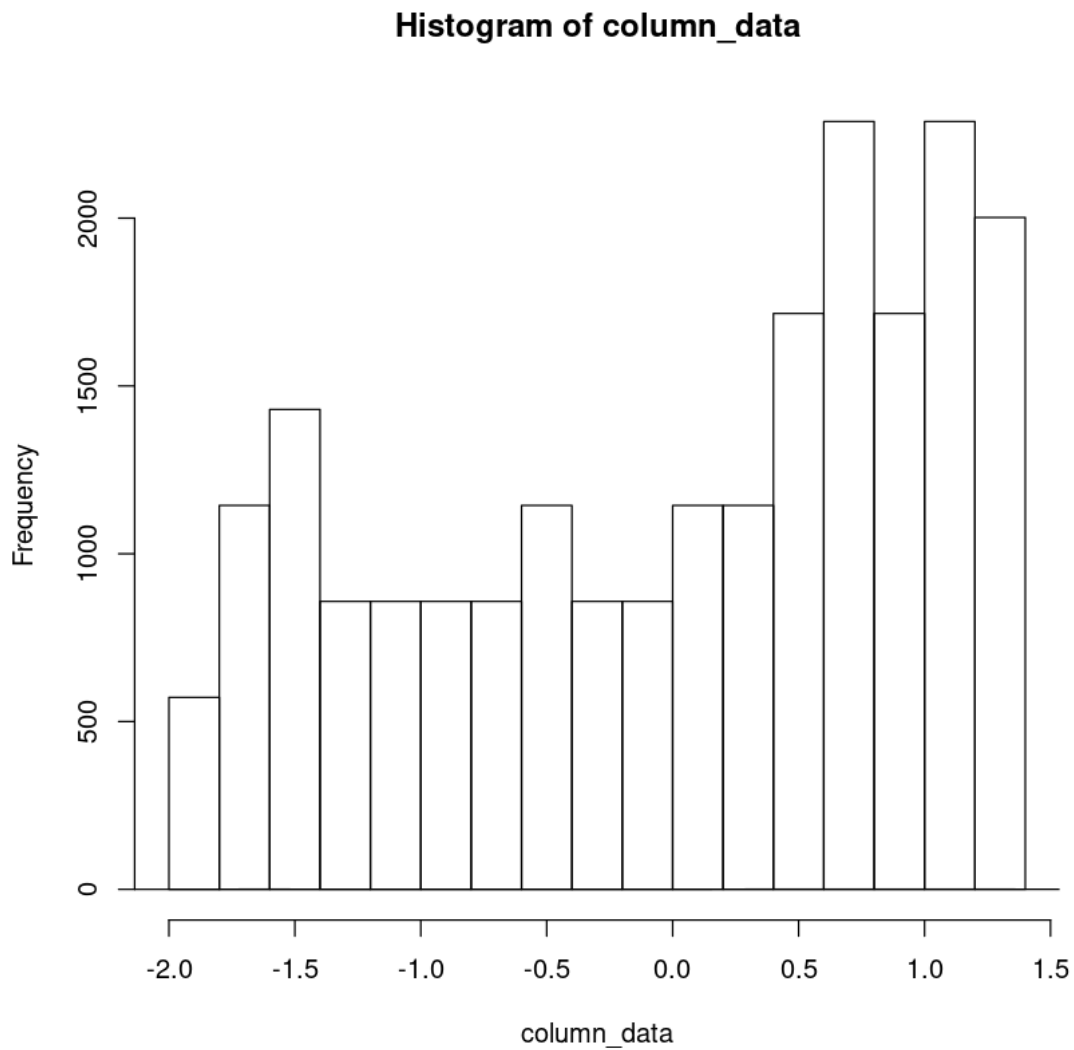
NULL

Processing column: semana





NULL
Processing column: n_vacunas

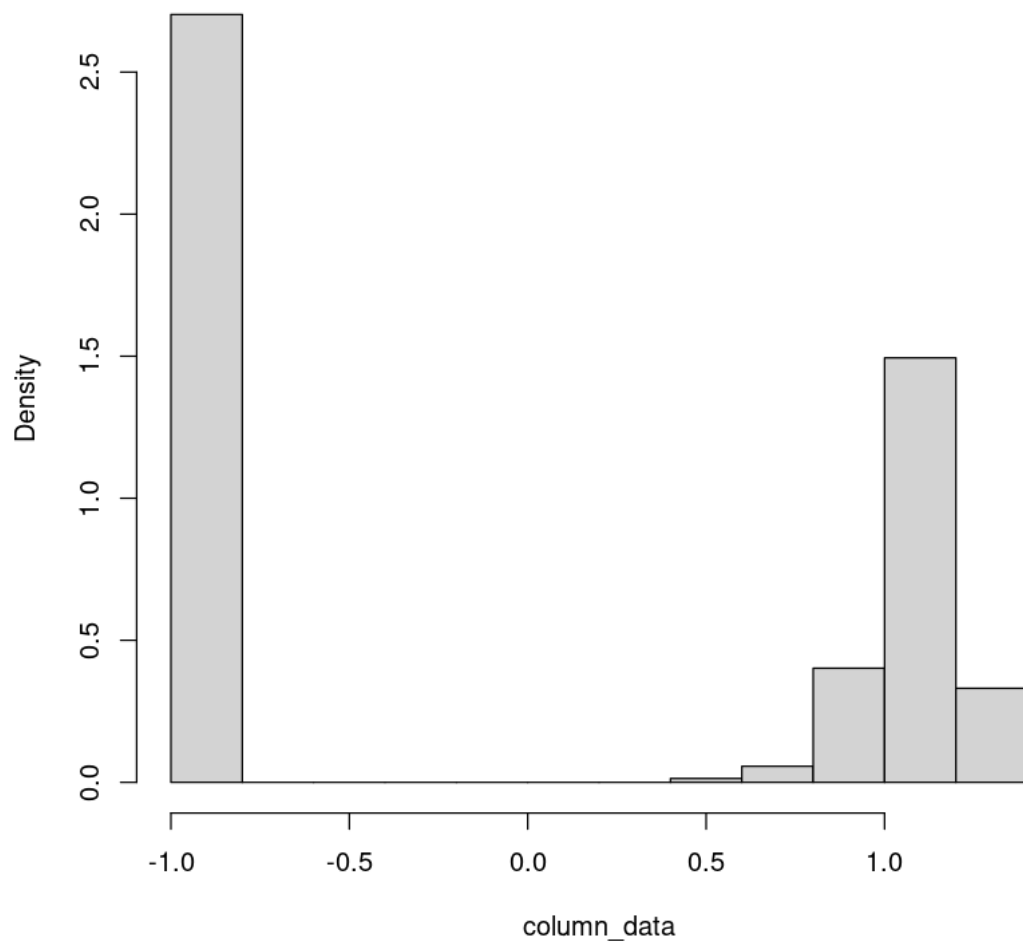


NULL

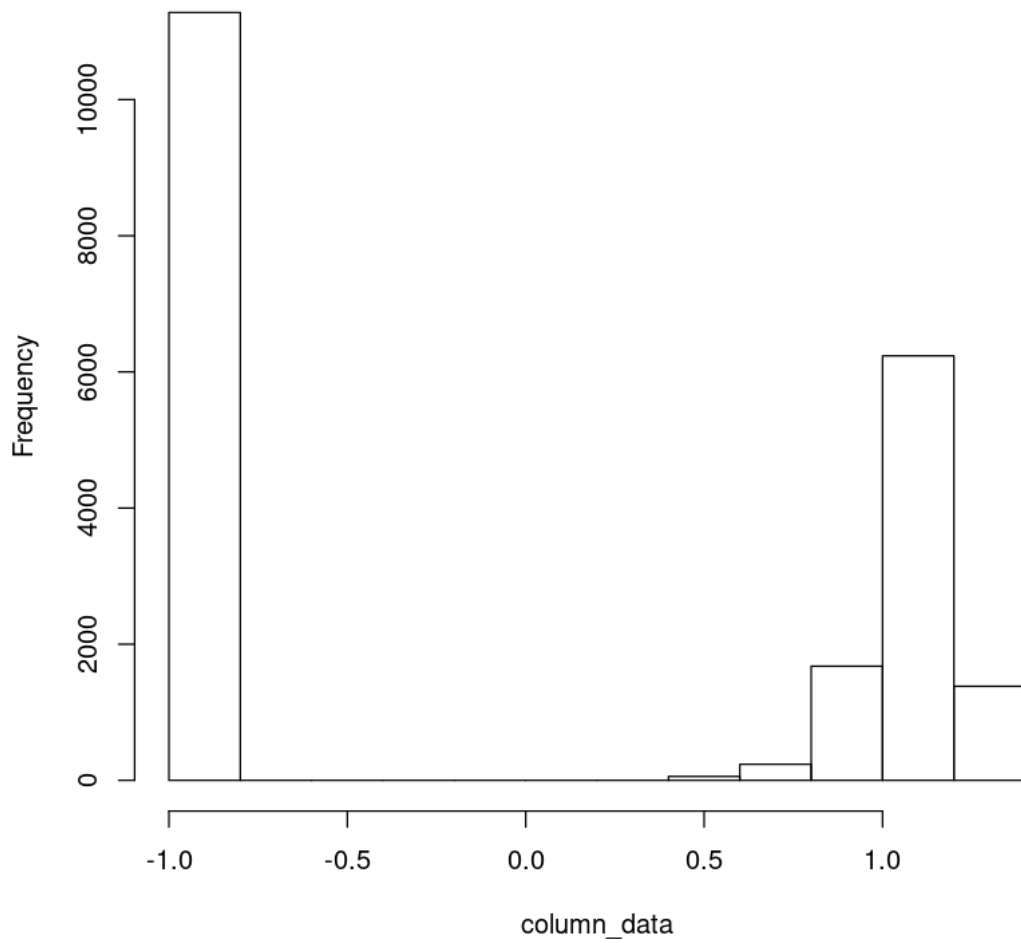
```
Error in density.default(column_data): 'x' contains missing values
Traceback:
```

```
1. lines(density(column_data))
2. density(column_data)
3. density.default(column_data)
4. stop("'x' contains missing values")
```


Histogram for n_vacunas



Histogram of column_data



[]: