TIBBLE

El tibble es formato de datos por defecto de Tidyverse.
Es como un dataframe pero con algunas mejoras, como el imprimir sólo una muestra de los datos que quepa en pantalla, darnos info sobre el tipo de variables o no convertir cadenas a factores automáticamente

Principales funciones:

- as_tibble(df): para convertir a tibble
- as.data.frame(tibble): para convertir a data frame
- print(tibble, n = num registros a imprimir, width = num columnas a imprimir)
- options(tibble.print_min = Inf, tibble.width = Inf): para definir las opciones a nivel general
- rownames_to_column(df): para poner los rownames como una variable
- glimpse(df): para un primer vistazo a los datos
- %>% para encadenar pasos (pipes)

READR

DS4B

Es el paquete de Tidyverse para importar datos que están en formato de fichero

Leer archivos:

- read_csv() para ficheros separados por comas
- read_csv2() para ficheros separados por punto y coma
- read_tsv() para ficheros separados por tabulador
- read_table() para ficheros separados por espacios
- read_table2() para ficheros separados por espacios múltiples
- read_delim() para ficheros separados por otro caracter
- read_fwf() para ficheros separados por un ancho fijo

cread_lines() para leer archivos que no conocemos todavía

- read_excel() para leer excels
- read_rds() para leer este formato interno de R

Definir tipo de las columnas:

- col_guess() para que R decida
- col_skip() para no importar esa variable
- col_character() para que sea tipo caracter
- col_double para decimales con separador de punto
- col_euro_double para decimales con separador de coma
- col_datetime para fecha y hora
- col_date() para fecha
- col_time() para hora
- col_factor() para que sea tipo factor
- col_integer() para que sea número entero
- col_logical() para que sea V o F

READR

DS4B

Es el paquete de Tidyverse para importar datos que están en formato de fichero

Parsear variables:

- parse_guess()
- parse_character()
- parse_datetime()
- parse_date()
- parse_time()
- parse_integer()
- parse_double()
- parse_number() ignora lo que está antes y después del primer número que encuentra
- parse_factor()
- parse_logical()

Formatos para importar fechas y horas:

- Año 4 digitos: %Y
- Año 2 digitos: %y
- Mes 2 digitos: %m
- Día 2 digitos: %d
- Hora (0-23): %H
- Hora (0-12): %I
- Hora con am-pm: %p
- Minutos: %M
- Segundos: %S

Escribir archivos:

- write_csv()
- write_delim()
- write_excel_csv()
- write_rds()

DPLYR

DS4B

Es la navaja suiza de Tidyverse. Lo usaremos para todo tipo de gestión y transformación de datos

Funciones Base de Dplyr:

- filter() para filtrar registros
- select() para seleccionar columnas
- select_if() para seleccionar variables si cumplen una condición
- rename() para renombrar variables
- mutate() para modificar datos o crear nuevas variables
- mutate_if() para modificar variables si cumplen una condición
- mutate_at() para modificar las variables especificadas
- mutate_all() para modificar todas las variables
- transmute() para obtener sólo la variable modificada
- summarize() para crear cálculos y agregados
- arrange() para ordenar, en descendente con desc()
- group_by() para usar cualquiera de las anteriores por grupos en lugar de sobre todo el dataset
- ungroup() para desagrupar
- count() para hacer conteos de los valores de la variable

Funciones ayudantes:

- starts_with('texto'): selecciona variables cuyos nombres empiecen por texto
- ends_with('texto'): selecciona variables cuyos nombres terminen por texto
- contains('texto'): selecciona variables que contengan el texto en su nombre
- matches('texto'): selecciona variables mediante expresiones regulares
- num_range('texto', rangonumeros): selecciona variables con la combinación del texto con todos los
números del rango. Ej num_range('Enero_',2017:2019) seleccionaría varaibles que tuvieran Enero_2018
o Enero_2019
- everything(): selecciona todas las variables que no estén ya incluidas
- one_of(varios_nombres): selecciona las variables que estén entre varios_nombres

DPLYR

DS4B

Es la navaja suiza de Tidyverse. Lo usaremos para todo tipo de gestión y transformación de datos

Funciones a usar con summarize:

- sum() para calcular totales
- mean() para calcular la media
- median() para calcular la mediana
- min() y max()
- first() para extraer el primer registro
- last() para extraer el último registro
- nth() para extraer cualquier otra posición

Principales operadores en Dplyr:

- Igual a: ==
- Diferente a: !=
- Mayor que y mayor o igual que: >, >=
- Menor que y menor o igual que: <, <=
- Operador Y: &
- Operador O: |
- Operador No: !
- Está en: %in%
- Es nulo: is.na()
- No es nulo: !is.na()
- Entre: between()

Otras funciones útiles de Dplyr:

- glimpse(): vistazo rápido de una dataset
- n_distint(): calcular el número de registros diferentes
- distint(): eliminar registros duplicados
- sample_n(): muestra aleatoria de n registros
- sample_frac(): muestra aleatoria de un % del total de registros
- slice(): seleccionar unos registros determinados
- top_n(): seleccionar y ordenar los primeros n registros según el orden de una variable
- top_frac(): seleccionar y ordenar los primeros frac porcentaje de registros según el orden de una variable
- pull(): extrae una variable como un vector (vs select que la extrae como dataframe)
- near(): compara si dos vectores son iguales

DPLYR (UNIONES)

DS4B

Es la navaja suiza de Tidyverse. Lo usaremos para todo tipo de gestión y transformación de datos

Uniones sin cruzar variables:

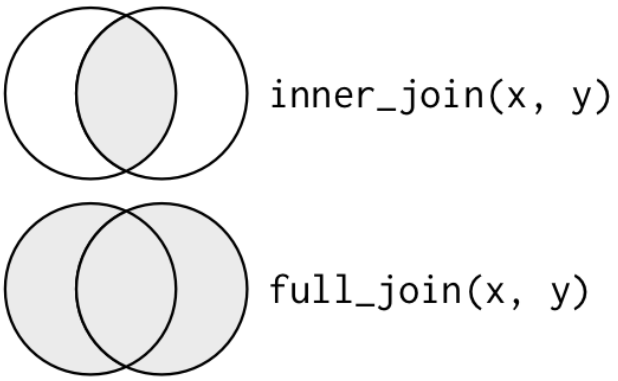
- bind_rows(): para apilar filas con filas
- bind_cols(): para apilar columnas con columnas

Uniones cruzando variables:

- left_join(): para que mande la tabla de la izquierda
- right_join(): para que mande la tabla de la derecha
- inner_join(): para coger sólo los registros comunes
- full_join(): para coger todos los registros
- anti_join(): para coger sólo los registros que estén en la izquierda pero no estén en la derecha

Lógica de conjuntos:

- union(): une todos los elementos únicos de ambos conjuntos
- intesect(): solo los elementos comunes en ambos conjuntos
- setdiff(a,b): solo los elementos que están en a pero no en b



TIDYR

DS4B

Tidyr nos permite sobre todo cambiar la estructura de los datos de ancho a largo o viceversa

Principales funciones:

- pivot_longer(): transforma de ancho a largo
- pivot_wider(): transforma de largo a ancho
- separate(): separa una columna de texto en varias mediante un separador
- unite(): integra varias columnas en una mediante un separador
- drop_na(): elimina los registros que tienen algún nulo
- fill(): imputa un nulo con el valor del registro anterior o del siguiente

FORCATS

DS4B

Forcats nos da funciones para trabajar de forma mucho más cómoda y eficiente con los factores (variable categóricas

Principales funciones:

- fct_reorder(): reordena los niveles según otra variable, muy útil para reordenar gráficos al vuelo
- fct_infreq(): reordena los niveles según los valores más frecuentes
- fct_relevel(): reordena los niveles según nuestras indicaciones manuales
- fct_rev(): invierte el orden actual de los niveles
- fct_recode(): recodifica los valores uno a uno
- fct_collapse(): recodifica los valores varios a la vez
- fct_lump(): agrupa automáticamente los valores más o menos frecuentes, útil para crear categoría
"otros"
- fct_relabel(): cambia los nombres de los factores de forma masiva
- fct_cross(): crea nuevos factores a partir de la combinación de otros

LUBRIDATE

DS4B

Lubridate nos da funciones para trabajar de forma mucho más cómoda y eficiente con las variables de tipo fecha

Principales funciones:

- today(): fecha del sistema
- now(): fecha - hora del sistema
- as_date(): Convierte a formato fecha
- make_date: Crea una fecha a partir de sus componentes
- dmy(): Y sus variantes, parsean fechas a partir de texto
- dmy_hms(): Y sus variantes, parsean fechas - horas a partir de texto
- date(): Extrae la fecha de una fecha - hora
- year(): Extrae el año de una fecha
- semester(): Extrae el semestre de una fecha
- month(): Extrae el mes de una fecha
- week(): Extrae la semana de una fecha
- day(): Extrae el día de una fecha
- wday(): Extrae el día de la semana de una fecha
- hour(): Extrae la hora de una fecha - hora
- minute(): Extrae el minuto de una fecha - hora
- second(): Extrae el segundo de una fecha - hora
- ddays(), dyears(), dweeks(), . . . : durations, transforman el valor dado a segundos
- as.duration(): transforma una diferencia de fechas a segundos
- time_length(): calcula la diferencia entre dos fechas a las unidades que le pidamos
- years(), months(), days(): periods. Son como los durations pero en lugar de trabajar en segundos

trabajan en unidades superiores y más “humanas”

- days_in_month(): Calcula el número de días en el mes de la fecha
- leap_year(): Indica si el año de la fecha es bisiesto

STRINGR

DS4B

Stringr tiene varias funciones para abordar las principales tareas que nos encontremos cuando estemos trabajando con variables en formato texto

Detectar palabras o patrones:

- str_detect(): Devuelve un 1 si detecta el patrón en el texto del registro y un 0 si no lo detecta
- str_which(): Devuelve el índice de los registros en los que encuentra el patrón
- str_count(): Cuenta cuantas veces aparece el patrón en el texto dentro del mismo registro
- str_locate(): Devuelve una matriz con la posición en la que empieza el patrón y la posición en la que termina
- str_locate_all(): Igual pero devuelve la posición de inicio y fin de todas las ocurrencias detectadas y en formato de lista

Extraer palabras o patrones:

- str_subset(): Extrae sólo los registros en los que ha encontrado el patrón (o no lo ha encontrado con negate = T)
- str_sub(): Extrae la parte del texto entre las posiciones de inicio y fin que le tenemos que pasar
- word(): Extrae las palabras entre las posiciones de inicio y fin que definamos
- str_extract(): Extrae la parte del texto que matchea con un patrón, especialmente una expresión regular, como un vector de caracteres
- str_match(): Igual pero devuelve una matriz

STRINGR

DS4B

Stringr tiene varias funciones para abordar las principales tareas que nos encontremos cuando estemos trabajando con variables en formato texto

Unir o separar textos:

- str_c(): Con sep, une varias variables de texto en una sola
- str_c(): Con collapse, une todos los valores de una variable en un único vector de caracteres
- str_glue(): Une textos pero además nos permite incluir variables usando llaves
- str_split(): Separa el texto cuando encuentra el patrón y devuelve una lista. Si ponemos simplify = T
devuelve una matriz

Cambiar el contenido o formato del texto:

- str_replace(): Reemplaza en cada registro la primera ocurrencia del patrón definido por un nuevo texto
- str_replace_all(): Igual pero reemplaza en cada registro todas las ocurrencias
- str_to_lower(): Cambia todo a minúsculas
- str_to_upper (): Cambia todo a mayúsculas
- str_to_title(): Cambia todo a títulos (la primera letra de cada palabra en mayúsculas)

Ordenar textos:

- str_sort(): Devuelve directamente el texto en ascendente o descendente
- str_order(): Devuelve el índice de los registos según el orden ascendente o descendente

STRINGR

DS4B

Stringr tiene varias funciones para abordar las principales tareas que nos encontremos cuando estemos trabajando con variables en formato texto

Gestionar la longitud del texto:

- str_length(): Calcular la longitud del texto
- str_pad(): Rellenar con caracteres adicionales para que todos los registros tengan la misma longitud en el texto
- str_trunc(): Recortar para que todos los registros tengan la misma longitud en el texto
- str_trim(): Eliminar espacios en blanco
- str_wrap(): Formatear el texto en párrafos de una longitud definida

Funciones útiles de Stringi:

- stri_compare(): Detectar si dos vectores de caracteres son exactamente iguales
- stri_startswith_fixed(): Ver si el texto empieza con el patrón
- stri_endswith_fixed(): Ver si el texto termina con el patrón
- stri_remove_empty(): Eliminar los elementos vacíos en un vector de caracteres
- stri_unique(): Extrae los elementos únicos del vector (eliminando los duplicados)