

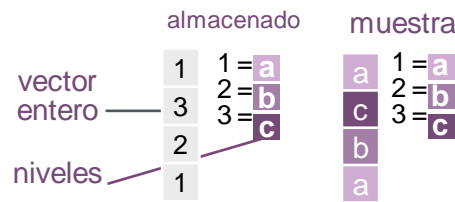
Factores con forcats : : GUÍA RÁPIDA



El paquete **forcats** proporciona herramientas para trabajar con factores, que son la estructura de datos de R para datos categóricos.

Factores

R representa datos categóricos con factores. Un **factor** es un vector entero con **niveles** que almacenan un conjunto de asignaciones entre enteros y valores categóricos. Al ver un factor, R no muestra los números enteros, sino los niveles asociados a ellos.



Crear un factor con factor()

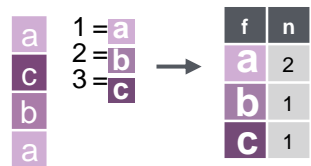
factor(x = character(), levels, labels = levels, exclude = NA, ordered = is.ordered(x), nmax = NA) Convertir un vector a un factor. Además **as_factor()**.

f <- factor(c("a", "c", "b", "a"), levels = c("a", "b", "c"))

Devuelve sus niveles con levels()
levels(x) Devolver/establecer los niveles de un factor.
levels(f); levels(f) <- c("x", "y", "z")

Use unclass() to see its structure

Inspeccionar Factores



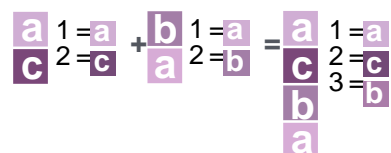
fct_count(f, sort = FALSE, prop = FALSE) Cuento el número de valores en cada nivel. **fct_count(f)**

fct_match(f, lvs) Comprobar si hay niveles en f.
fct_match(f, "a")

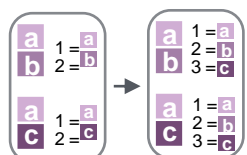


fct_unique(f) Devolver los valores únicos, eliminando los duplicados. **fct_unique(f)**

Combinar Factores

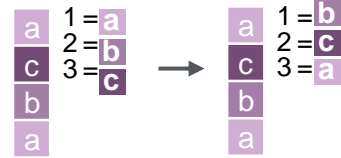


fct_c(...) Combinar factores con diferentes niveles. Además **fct_cross()**.
f1 <- factor(c("a", "c"))
f2 <- factor(c("b", "a"))
fct_c(f1, f2)



fct_unify(fs, levels = lvs_union(fs)) Estandarizar los niveles a través de una lista de factores.
fct_unify(list(f2, f1))

Cambiar el Orden de los Niveles



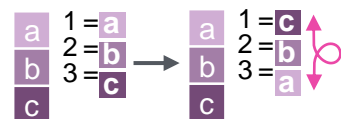
fct_relevel(f, ..., after = 0L) Reordenar manualmente los niveles de factor.
fct_relevel(f, c("b", "c", "a"))



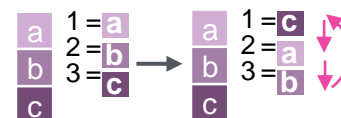
fct_infreq(f, ordered = NA) Reordene los niveles por la frecuencia con la que aparecen en los datos (primero la frecuencia más alta). Además, **fct_inseq()**.
f3 <- factor(c("c", "c", "a"))
fct_infreq(f3)



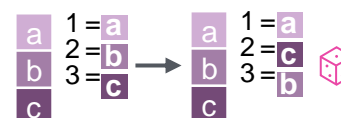
fct_inorder(f, ordered = NA) Reordenar los niveles por el orden en el que aparecen en los datos. **fct_inorder(f2)**



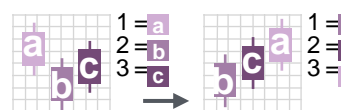
fct_rev(f) Orden de nivel inverso.
f4 <- factor(c("a", "b", "c")) **fct_rev(f4)**



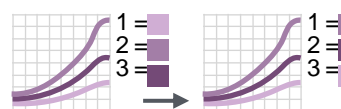
fct_shift(f) Cambia los niveles a la izquierda o a la derecha, envolviéndolos alrededor del extremo. **fct_shift(f4)**



fct_shuffle(f, n = 1L) Permutar aleatoriamente el orden de los niveles de factor. **fct_shuffle(f4)**

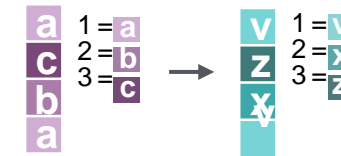


fct_reorder(.f, .x, .fun = median, ..., .desc = FALSE) Reordenar los niveles por su relación con otra variable. **boxplot(PlantGrowth, weight ~ fct_reorder(group, weight))**

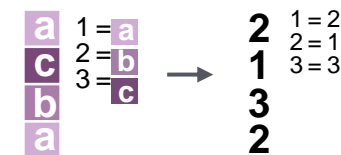


fct_reorder2(.f, .x, .y, .fun = last2, ..., .desc = TRUE) Reordene los niveles por sus valores finales cuando se trazan con otras dos variables.
ggplot(diamonds, aes(carat, price, color = fct_reorder2(color, carat, price))) + geom_smooth()

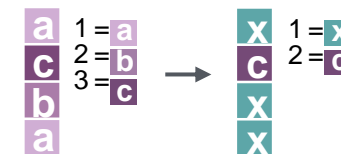
Cambiar el Valor de los Niveles



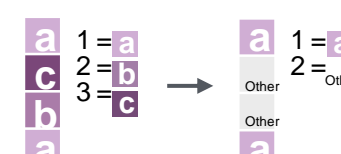
fct_recode(f, ...) Cambie manualmente los niveles. También **fct_relabel()** que obedece a la sintaxis purrr::map para aplicar una función o expresión a cada nivel.
fct_recode(f, v = "a", x = "b", z = "c")
fct_relabel(f, ~ paste0("x", .x))



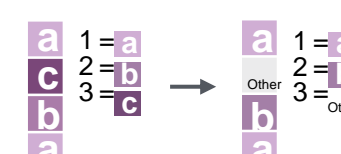
fct_anon(f, prefix = "") Anonimizar niveles con enteros aleatorios. **fct_anon(f)**



fct_collapse(f, ..., other_level = NULL) Contraer niveles en grupos definidos manualmente. **fct_collapse(f, x = c("a", "b"))**

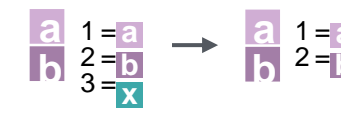


fct_lump_min(f, min, w = NULL, other_level = "Other") Agrupa los factores que aparecen menos de un mínimo de veces. Además, **fct_lump_n()**, **fct_lump_prop()**, y **fct_lump_lowfreq()**.
fct_lump_min(f, min = 2)

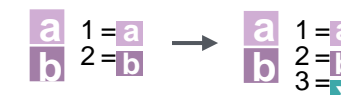


fct_other(f, keep, drop, other_level = "Other") Reemplace los niveles con "Other".
fct_other(f, keep = c("a", "b"))

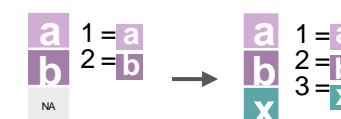
Agregar o Quitar Niveles



fct_drop(f, only) Eliminar los niveles no utilizados.
f5 <- factor(c("a", "b"), c("a", "b", "x"))
f6 <- fct_drop(f5)



fct_expand(f, ...) Agregar niveles a un factor. **fct_expand(f6, "x")**



fct_na_value_to_level(f, level = "(Missing)") Asigna un nivel a los NA para asegurarse de que aparezcan en las gráficas, etc.
f7 <- factor(c("a", "b", NA))
fct_na_value_to_level(f7, level = "(Missing)")