Introducción a Base R para Manipulación de Datos II

2023-05-13

Diferencia entre Vectores

Ya que sabemos manipular los nombres de las columnas de los dataframes, es importante conocer las funciones `setdiff()` e `intersect()` de R. Estas nos ayudaran a determinar si las columnas de dos dataframes son tienen nombres similares para poder realizar una concatenacion o nos diran si existe una columna por la cual se pueda hacer un join.

```
##Leemos la tabla de letras
letras_bc <-  read.table("../../../data/letras_bc_consolidado_clean.csv")

##Creamos un nuevo dataframe
nuevas_letras <- data.frame(
  "FechaSubasta"= c("2023-06-01","2023-06-12", "2023-06-14"),
  "FechaLiquidacion"= c("2023-06-01","2023-06-12", "2023-06-14"),
  "MontoSubastado"= c(7400, 8000, 2400),
  "MontoDemandado"= c(2000, 1500, 2222),
  "MontoAdjudicado"= c(1800, 1499, 2200),
  "RendimientoPPA"= c(0.05, 0.0555, 0.087)
  )

##Columnas que no hacen match
setdiff(colnames(letras_bc), colnames(nuevas_letras))
```

```
## [1] "FechadeSubasta"
```

```
##Columnas que si hacen match
intersect(colnames(letras_bc), colnames(nuevas_letras))
```

```
## [1] "FechaLiquidacion" "MontoSubastado"   "MontoDemandado"   "MontoAdjudicado"
## [5] "RendimientoPPA"
```

Concatenación

En R, `rbind()` y `cbind()` son funciones que se utilizan para combinar matrices, dataframes o vectores en una sola estructura.

rbind(): se utiliza para unir dos o más objetos por filas, es decir, apilarlos verticalmente. La función rbind() espera que los objetos tengan el mismo número de columnas, ya que agrega las filas debajo de la última fila de cada objeto.

```r
##No puedo realizar la concatenacion porque el nombre de las columnas no es igual
# rbind(letras_bc, nuevas_letras)

##Dado este analisis  de diferencias de columnas modificoo el nombre
#de la columna para poder realizar la concatenacion
colnames(nuevas_letras)[1] <- "FechadeSubasta"
letras_bc_actualizado <- rbind(letras_bc, nuevas_letras) ##Procedo
#a realizar la concatenacion sin temas

##Verifico que realmente se realizo la concatenacion
cantidad_registros_nuevos <- nrow(letras_bc_actualizado) - nrow(letras_bc)
cantidad_registros_nuevos
```

```
## [1] 3
```

```r
tail(letras_bc_actualizado) ##Veo mi dataset
```

```
##      FechadeSubasta FechaLiquidacion MontoSubastado MontoDemandado
## 534      2023-04-19       2023-04-21           5000        9854.10
## 535      2023-04-26       2023-04-28           5000        9231.47
## 536      2023-05-03       2023-05-05           5000       10624.33
## 1100     2023-06-01       2023-06-01           7400        2000.00
## 2100     2023-06-12       2023-06-12           8000        1500.00
## 3100     2023-06-14       2023-06-14           2400        2222.00
##      MontoAdjudicado RendimientoPPA
## 534          9850.77      0.1235413
## 535          9231.47      0.1235229
## 536          9792.55      0.1187056
## 1100         1800.00      0.0500000
## 2100         1499.00      0.0555000
## 3100         2200.00      0.0870000
```

cbind(): se utiliza para unir dos o más objetos por columnas, es decir, agregarlos horizontalmente. La función cbind() espera que los objetos tengan el mismo número de filas, ya que agrega las columnas a la derecha de la última columna de cada objeto.

```r
# Creamos un nuevo dataframe con informacion nueva de letras,
#en este ejemplo un rankeo
nueva_info_letras <- data.frame(rankeo= seq(1,
                                            nrow(letras_bc), by=1))
#Se utiliza la funcion nrows para obtener el numero de filas
#Lo unimos de manera horizontal al dataframe original
head(cbind(letras_bc, nueva_info_letras))
```

```
##   FechadeSubasta FechaLiquidacion MontoSubastado MontoDemandado MontoAdjudicado
## 1     2007-04-03       2007-04-04            400        2281.80           400.00
## 2     2010-02-17       2010-02-02            300        1088.00           300.00
## 3     2010-02-24       2010-02-02            500         575.70           500.00
## 4     2023-04-12       2023-04-04           5000        9955.15          7951.91
## 5     2019-01-01       2019-01-25           2500        3635.00          1515.00
## 6     2019-02-02       2019-02-15           2500        3823.00          2025.00
##   RendimientoPPA rankeo
```

```
## 1     0.09570000      1
## 2     0.05225600      2
## 3     0.05243900      3
## 4     0.12305392      4
## 5     0.07269412      5
## 6     0.07193004      6
```

Uniones

En R, la función merge() se utiliza para combinar dos o más dataframes en uno solo, basándose en una o más columnas comunes. Es similar a la operación de unión en SQL.

La función merge() en R realiza diferentes tipos de unión (joins) según los valores que se proporcionen en los argumentos all.x y all.y, que se utilizan para especificar qué filas incluir en la unión.

- Si `all.x = TRUE y all.y = FALSE`, se realiza un left join, es decir, se incluyen todas las filas del primer dataframe (x) y solo las filas del segundo dataframe (y) que coinciden con las filas del primer dataframe.
- Si `all.x = FALSE y all.y = TRUE`, se realiza un right join, es decir, se incluyen todas las filas del segundo dataframe (y) y solo las filas del primer dataframe (x) que coinciden con las filas del segundo dataframe.
- Si `all.x = TRUE y all.y = TRUE`, se realiza un full join, es decir, se incluyen todas las filas de ambos dataframes (x e y), aunque no haya coincidencias en las filas.
- Si `all.x = FALSE y all.y = FALSE`, se realiza un inner join, es decir, solo se incluyen las filas que tienen coincidencias en ambas tablas.

Por defecto, si no se especifican los valores de `all.x y all.y`, la función merge() realiza un inner join.

```r
##Creamos una columna de FechaSubasta utilizando una secuencia de
#fechas de la fecha minima de subasta de nuestro dataframe original
#hasta la fecha maxima por dia

##Creamos una segunda columna que aleatoriamente nos dira 1 si
#fue declarado desierto y 0 si no fue declarada desierta

subasta_desierta <- data.frame(
  "FechadeSubasta" = seq(as.Date(min(letras_bc$FechadeSubasta)),
                         as.Date(max(letras_bc$FechadeSubasta)), by="day" ),
  "DeclaradaDesierta"= sample(c(1,0),length(seq(as.Date(min(letras_bc$FechadeSubasta)),
                                                as.Date(max(letras_bc$FechadeSubasta)),
                                                by="day" )), TRUE)
)

##Este nuevo dataframe lo queremos unir con las letras del BC.
#Esta union lo hacemos por la FechadeSubasta
letras_bc$FechadeSubasta <- as.Date(letras_bc$FechadeSubasta)

##En este caso hacemos un left join porque nos interesa toda la
#informacion que esta en el dataframe x (letras_bc), pero no toda
#la informacion que esta en el dataframe y (subasta_desierta)
```

```r
merged_letras_bc <- merge(letras_bc, subasta_desierta,
                          all.x= TRUE, all.y=FALSE)
head(merged_letras_bc)
```

```
##   FechadeSubasta FechaLiquidacion MontoSubastado MontoDemandado MontoAdjudicado
## 1     2007-03-21       2007-03-23            500         3890.0              500
## 2     2007-03-28       2007-03-30            800         3038.1              800
## 3     2007-04-03       2007-04-04            400         2281.8              400
## 4     2007-04-11       2007-04-13            400         1818.0              400
## 5     2007-04-18       2007-04-20            750         3341.2              750
## 6     2007-04-25       2007-04-27            700         2636.7              700
##   RendimientoPPA DeclaradaDesierta
## 1         0.0969                 0
## 2         0.0968                 0
## 3         0.0957                 1
## 4         0.0944                 1
## 5         0.0894                 1
## 6         0.0882                 0
```