

Lubridate Lab

Michelle

10/6/2024

Laboratorio de la libreria Lubridate

```
##cargar librerias
library(nycflights13)
library(lubridate)
library(dplyr)
```

Resuelva las siguientes preguntas:

```
### Ejercicio 1: Convertir columnas de hora en fecha-hora

#Problema: Convierte las columnas dep_time (hora de salida) y arr_time (hora de llegada) en objetos de
#Ayuda: Investiga la funcion matematica de modulo de r.

# Crear una nueva columna 'dep_datetime' y 'arr_datetime' con la conversión
flights <- nycflights13::flights

flights <- flights %>%
  mutate(
    dep_hour = dep_time %/% 100,
    dep_minute = dep_time %% 100,
    arr_hour = arr_time %/% 100,
    arr_minute = arr_time %% 100,
    dep_datetime = make_datetime(year = year, month = month, day = day, hour = dep_hour, min = dep_minu
    arr_datetime = make_datetime(year = year, month = month, day = day, hour = arr_hour, min = arr_minu
  )

# Ver los resultados
head(flights[c("dep_time", "dep_datetime", "arr_time", "arr_datetime")])
```

```
## # A tibble: 6 x 4
##   dep_time dep_datetime      arr_time arr_datetime
##   <int> <dtm>              <int> <dtm>
## 1     517 2013-01-01 05:17:00      830 2013-01-01 08:30:00
## 2     533 2013-01-01 05:33:00      850 2013-01-01 08:50:00
## 3     542 2013-01-01 05:42:00      923 2013-01-01 09:23:00
## 4     544 2013-01-01 05:44:00     1004 2013-01-01 10:04:00
```

```
## 5      554 2013-01-01 05:54:00      812 2013-01-01 08:12:00
## 6      554 2013-01-01 05:54:00      740 2013-01-01 07:40:00
```

Ejercicio 2: Duracion del vuelo

#Calcula el tiempo de vuelo total en minutos entre las columnas dep_time y arr_time que calculaste en e

Calcula la duración del vuelo en minutos

```
flights <- flights %>%
  mutate(
    flight_duration = as.numeric(difftime(arr_datetime, dep_datetime, units = "mins"))
  )
```

Ver los primeros resultados

```
head(flights[c("dep_datetime", "arr_datetime", "flight_duration")])
```

```
## # A tibble: 6 x 3
##   dep_datetime      arr_datetime      flight_duration
##   <dtm>            <dtm>            <dbl>
## 1 2013-01-01 05:17:00 2013-01-01 08:30:00      193
## 2 2013-01-01 05:33:00 2013-01-01 08:50:00      197
## 3 2013-01-01 05:42:00 2013-01-01 09:23:00      221
## 4 2013-01-01 05:44:00 2013-01-01 10:04:00      260
## 5 2013-01-01 05:54:00 2013-01-01 08:12:00      138
## 6 2013-01-01 05:54:00 2013-01-01 07:40:00      106
```

Ejercicio 3: Extraer componentes de fechas

```
flights <- flights %>%
  mutate(
    dep_day_of_week = wday(dep_datetime, label = TRUE), # Día de la semana (con etiqueta)
    dep_hour = hour(dep_datetime) # Hora de salida
  )
```

Ver los primeros resultados

```
head(flights[c("dep_datetime", "dep_day_of_week", "dep_hour")])
```

```
## # A tibble: 6 x 3
##   dep_datetime      dep_day_of_week dep_hour
##   <dtm>            <ord>            <int>
## 1 2013-01-01 05:17:00 mar              5
## 2 2013-01-01 05:33:00 mar              5
## 3 2013-01-01 05:42:00 mar              5
## 4 2013-01-01 05:44:00 mar              5
## 5 2013-01-01 05:54:00 mar              5
## 6 2013-01-01 05:54:00 mar              5
```

Ejercicio 4: Crear nuevas columnas con el día de la semana y la semana del año

#Problema: Usando la columna `time_hour`, crea una nueva columna que indique el día de la semana y otra

#Ayuda: Invesitga la funcion wday de lubridate.

```
# Crear nuevas columnas para el día de la semana y la semana del año
flights <- flights %>%
  mutate(
    flight_day_of_week = wday(time_hour, label = TRUE), # Día de la semana
    flight_week_of_year = week(time_hour) # Semana del año
  )

# Ver los primeros resultados
head(flights[c("time_hour", "flight_day_of_week", "flight_week_of_year")])
```

```
## # A tibble: 6 x 3
##   time_hour      flight_day_of_week flight_week_of_year
##   <dtm>          <ord>                <dbl>
## 1 2013-01-01 05:00:00 mar                1
## 2 2013-01-01 05:00:00 mar                1
## 3 2013-01-01 05:00:00 mar                1
## 4 2013-01-01 05:00:00 mar                1
## 5 2013-01-01 06:00:00 mar                1
## 6 2013-01-01 05:00:00 mar                1
```

Ejercicio 5: Encontrar los vuelos que salieron los fines de semana

#Problema: Filtra los vuelos que despegaron un sábado o domingo y devuelve el total de vuelos en fines

```
Sys.setlocale("LC_TIME", "es_ES.UTF-8")
```

```
## [1] "es_ES.UTF-8"
```

```
# Filtrar vuelos que salieron en sábado o domingo
weekend_flights <- flights %>%
  filter(flight_day_of_week %in% c("sáb", "dom"))
```

```
# Contar el total de vuelos en fines de semana
total_weekend_flights <- nrow(weekend_flights)
```

```
# Mostrar el total de vuelos
total_weekend_flights
```

```
## [1] 85077
```

```
# Mostrar el total de vuelos
head(weekend_flights)
```

```
## # A tibble: 6 x 29
##   year month  day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>   <int>         <int>      <dbl>    <int>         <int>
## 1  2013     1     5      14          2359        15      503          445
## 2  2013     1     5      37          2230       127      341          131
```

```
## 3 2013      1      5      458          500          -2      640          650
## 4 2013      1      5      516          515           1      821          816
## 5 2013      1      5      534          530           4      829          829
## 6 2013      1      5      537          540          -3      831          850
## # i 21 more variables: arr_delay <dbl>, carrier <chr>, flight <int>,
## #   tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>,
## #   hour <dbl>, minute <dbl>, time_hour <dtm>, dep_hour <int>,
## #   dep_minute <dbl>, arr_hour <dbl>, arr_minute <dbl>, dep_datetime <dtm>,
## #   arr_datetime <dtm>, flight_duration <dbl>, dep_day_of_week <ord>,
## #   flight_day_of_week <ord>, flight_week_of_year <dbl>
```

```
head(total_weekend_flights)
```

```
## [1] 85077
```