

2025
10-13 JULY



MAD
COOL
FESTIVAL



Visualització de dades (Enginyeria de Dades - EE - UAB)
Examen Segon Parcial - 19 Juny 2023
RESPOSTES MODEL B

Nom i Cognom: _____

NIU: _____

Grup de Matrícula: _____

Només es permet l'ús d'internet per l'accés al campus virtual en el moment de descarregar el full d'enunciats y d'entregar l'examen.

Sólo se permite el uso de internet para el acceso al campus virtual en el momento de descargar la hoja de enunciados y de entregar el examen.

PART 1 (6 pt)

Dataset: Australian_Flight_dataset_2003-2022.csv

Agafarem aquest dataset que conté totes les rutes (cada ruta conté un o més vols diaris, setmanals o mensuals) d'arribada i/o sortida als aeroports d'Austràlia durant els anys 2003 al 2022. Utilitzeu les llibreries (plotly, gganimate, shiny, etc.) que creieu convenientes i dibuixeu les gràfiques que us facin falta.

Cada registre conté informació d'una ruta d'una companyia aèria. Conté les variables:

1. Month ---> Date in month and day
2. In_Out ---> Status of flight [Incoming/Outgoing]
3. Australian_City ---> Australian city name
4. International_City ---> International city name
5. Airline ---> Airline owning the flight
6. Route ---> Route taken by the flight
7. Port_Country ---> Port country
8. Port_Region ---> Port region
9. Service_Country ---> Service country
10. Service_Region ---> Service region
11. Stops ---> Number of stops taken by the flight
12. All_Flights ---> Total number of flights
13. Max_Seats ---> Total capacity of seats in flight
14. Year ---> Date in year
15. Month_num ---> Date in month number

Com a principals variables que utilitzarem en aquesta 1a part, cal ressaltar Airline, Max_Seats i Year.

Càrrega de les llibreries i data set:

```
> library(tidyverse)
> library(dplyr)
> library(plotly)
> library(shiny)
> getwd()
> setwd("C:/Users/enric/Documents/R")
```

```
> Australian_FD <- read.csv('./Australian_Flight_Dataset_2003-2022.csv')
```

1.1 (2 pts) Mostra el codi (en ggplotly) i la gràfica de línies de l'evolució temporal en anys de la capacitat en seients de cada companyia, siguin de rutes de sortida o arribada a qualsevol ciutat australiana (gràfica 1) i fixa't amb les tres companyies aèries amb més seients l'any 2021. Digue's quines companyies són per ordre decreixent i mostra el codi (en ggplotly o shiny+plotly) i la gràfica de línies de l'evolució temporal d'aquestes tres companyies (gràfica 2). Especifica si fas Data Messaging.

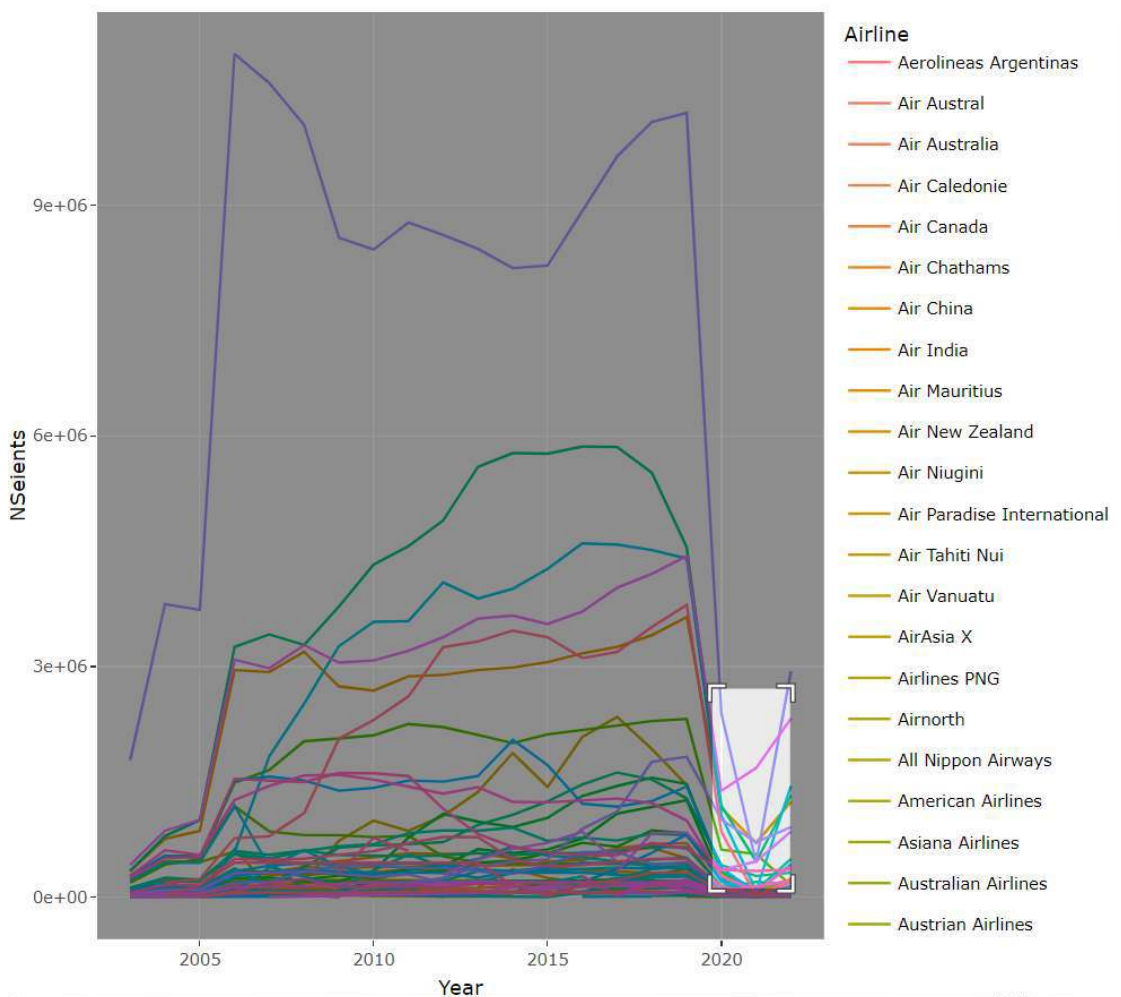
RESPOSTA: En primer lloc, hem de fer data messaging sobre el data set original.

DATA MASSAGING: Agrupar per (Any, Companyia Aèrea, Sexe) calculant el nombre de seients (Max_Seats):

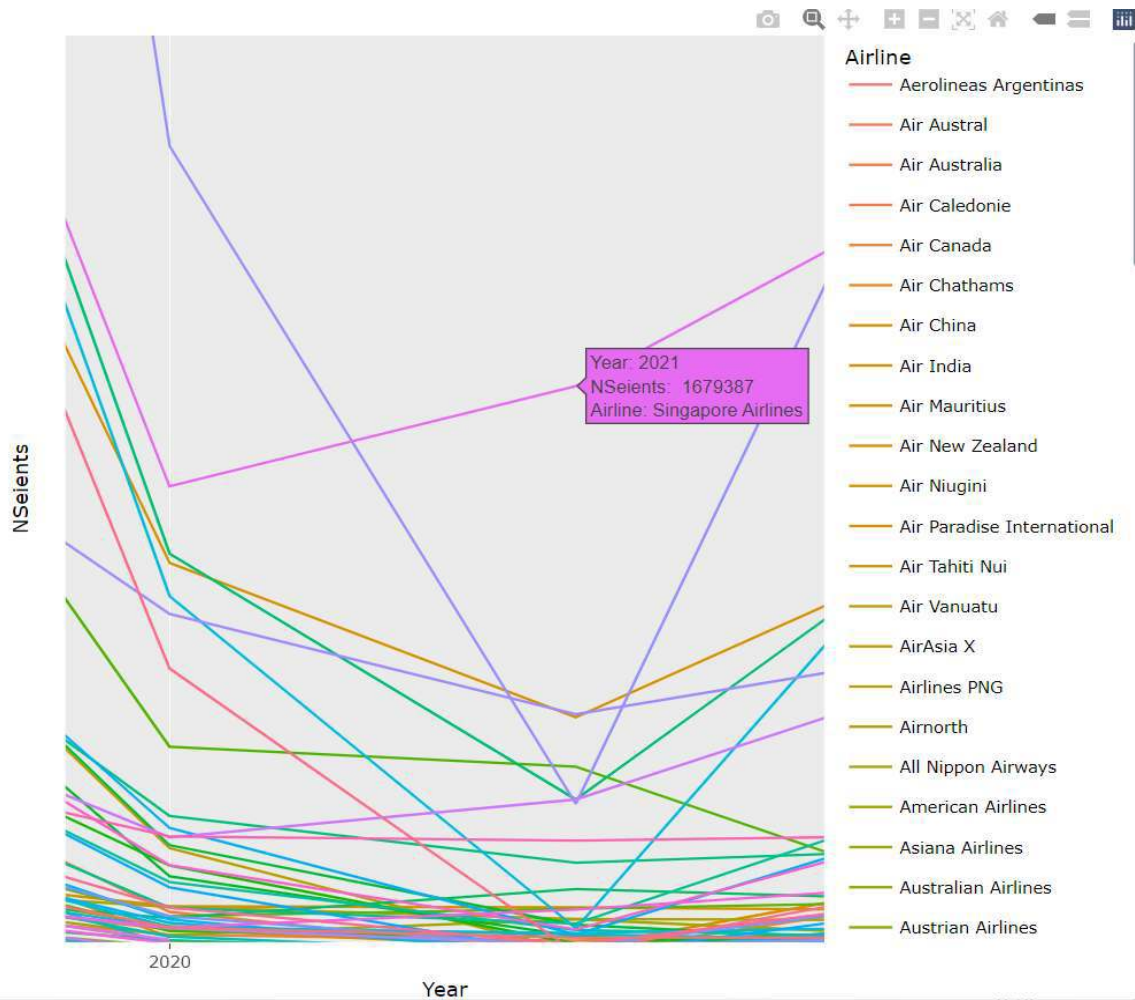
```
> Australian_FD_Rutes <- Australian_FD %>% group_by(Year,Airline) %>%  
summarize(NSeients = sum(Max_Seats))
```

GRÀFICA 1 de línies amb ggplotly:

```
> ggplotR <- ggplot(Australian_FD_Rutes, aes(x=Year, y=NSeients,  
color=Airline)) + geom_line()  
> ggplotly(ggplotR)
```



I fer un zoom en la zona propera a l'any 2021:



*Els noms de les companyies són **Singapore Airlines (1679387 seients)**, **Qatar Airways (712357)** i **Air New Zealand (703436)***

GRÀFICA 2: Després podem llistar la gràfica de dues formes:

AMB ggplotly (abans filtrem el data set per les tres companyies aèries):

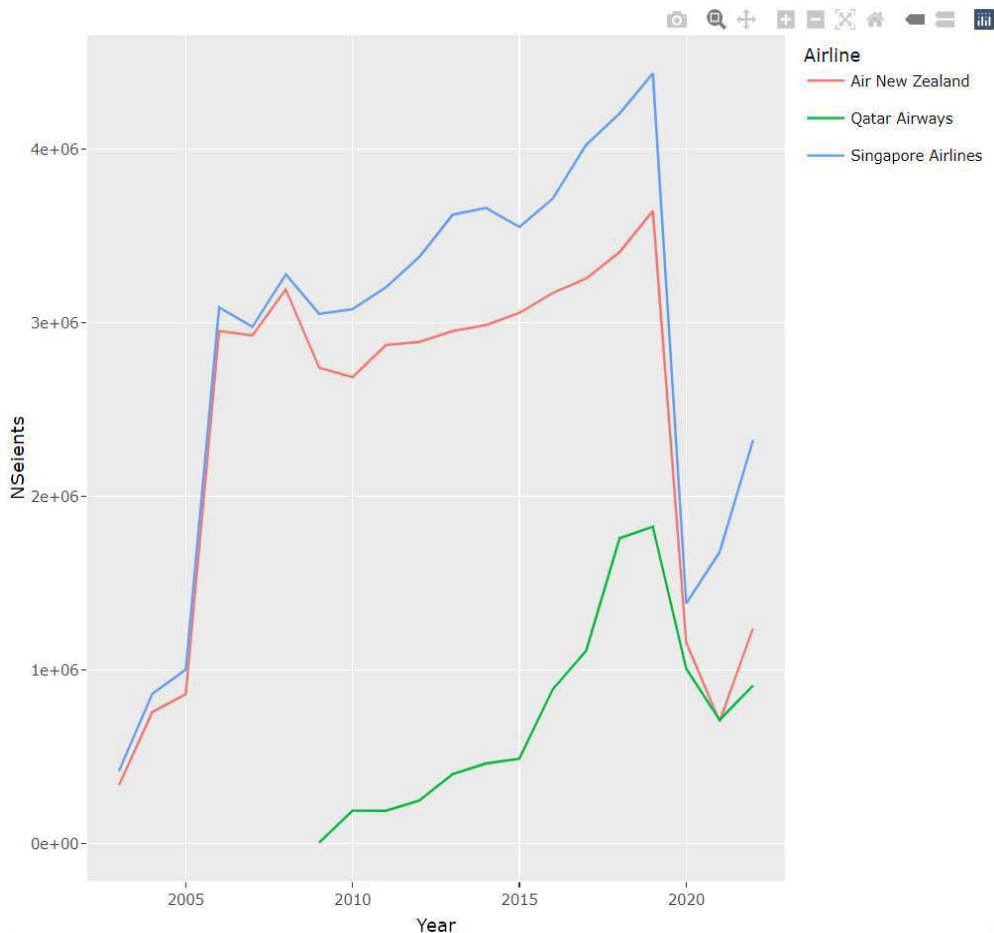
```
> Australian_FD_Seients_SQA <- Australian_FD_Seients %>% filter(Airline ==
"Singapore Airlines" | Airline == "Qatar Airways" | Airline == "Air New Zealand")

> ggplotSQA <- ggplot(Australian_FD_Seients_SQA, aes(x=Year, y=NSeients,
color=Airline)) + geom_line()
> ggplotly(ggplotSQA)
>
```

2025
10-13 JULY



MAD
COOL
FESTIVAL



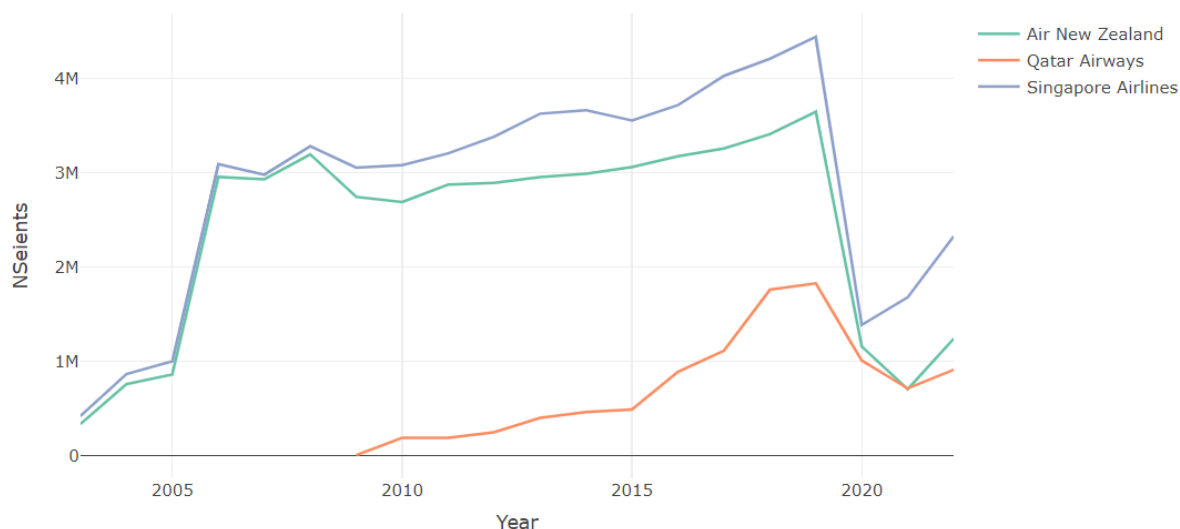
AMB shiny + plotly:

```
ui <- fluidPage(
  selectizeInput( inputId = "Airline",
    label = "Selecciona una COMPANIA:",
    choices = unique(Australian_FD_Seients_SQA$Airline),
    selected = c("Singapore Airlines", "Qatar Airways",
      "Air New Zealand"),
    multiple = TRUE
  ),
  plotlyOutput(outputId = "p")
)

server <- function(input, output, ...)
{
  output$p <- renderPlotly (
    {plot_ly(Australian_FD_Seients_SQA, x = ~Year, y = ~NSeients,
      color=~Airline) %>%
      filter(Airline %in% input$Airline) %>%
      group_by(Airline) %>%
      add_lines()
    })
  }
shinyApp(ui, server)
```


Selecciona una COMPANYIA:

Air New Zealand Singapore Airlines
Qatar Airways



1.2 (0,5 pt) Compara les tres companyies en la gràfica 2 segons els següents criteris:

a) (0,2 pts) Anys en que assoleixen el màxim i el mínim.

RESPOSTA:

NOM	MÍNIM: Any - Nombre	MÀXIM: Any - Nombre
Singapore Airlines	2003 – 418696	2019 – 4436641
Qatar Airways	2009 – 5698	2019 – 1824887
Air New Zealand	2003 – 335642	2019 – 3644470

b) (0,1 pts) Comenta la taula anterior. Trobes alguna singularitat?.

RESPOSTA: El màxim nombre de rutes de les companyies coincideix el 2019, poc abans la pandèmia del Covid-19.

c) (0,2 pts) Les tres gràfiques tenen creuaments. De cada creuament, digues quin parell de companyies creuen i l'interval dels dos anys en que es produeix.

RESPOSTA:

NOMS	Any Anterior	Any Posterior
Air New Zealand-Qatar Airways	2020	2021
Qatar Airways-Air New Zealand	2021	2022

1.3 (1,5 pts) Mostra el codi i adjunta el fitxer AVI o GIF del Ranking de Barres Animades (*Animated Bar Race Ranking*) sobre les 10 companyies amb més seients per anys, del 2003 al 2020.

RESPOSTA:

Càrrega de les llibreries

```
> library(tidyverse)
> library(gganimate)      # Generació de frames i la seva compilació per a
generar fitxer animació
> library(gifski)         # Generació fitxer animació GIF
```

PAS1: DATA MASSAGING:

- Filtrar el dataset pels anys 2003 al 2020:

```
> Australian_FD_Seients <- Australian_FD_Seients %>% filter(Year < 2021)
```

- Definir nova variable rank amb la posició decreixent en nombre de rutes i filtrar les 10 primeres companyies segons rank:

```
> Australian_FD_Seients_formatted <- Australian_FD_Seients %>%
  group_by(Year) %>%
  # The * 1 makes it possible to have non-integer ranks while sliding
  mutate(rank = rank(-NSeients)) %>%
  group_by(Airline) %>%
  filter(rank <=10) %>%
  ungroup()
```

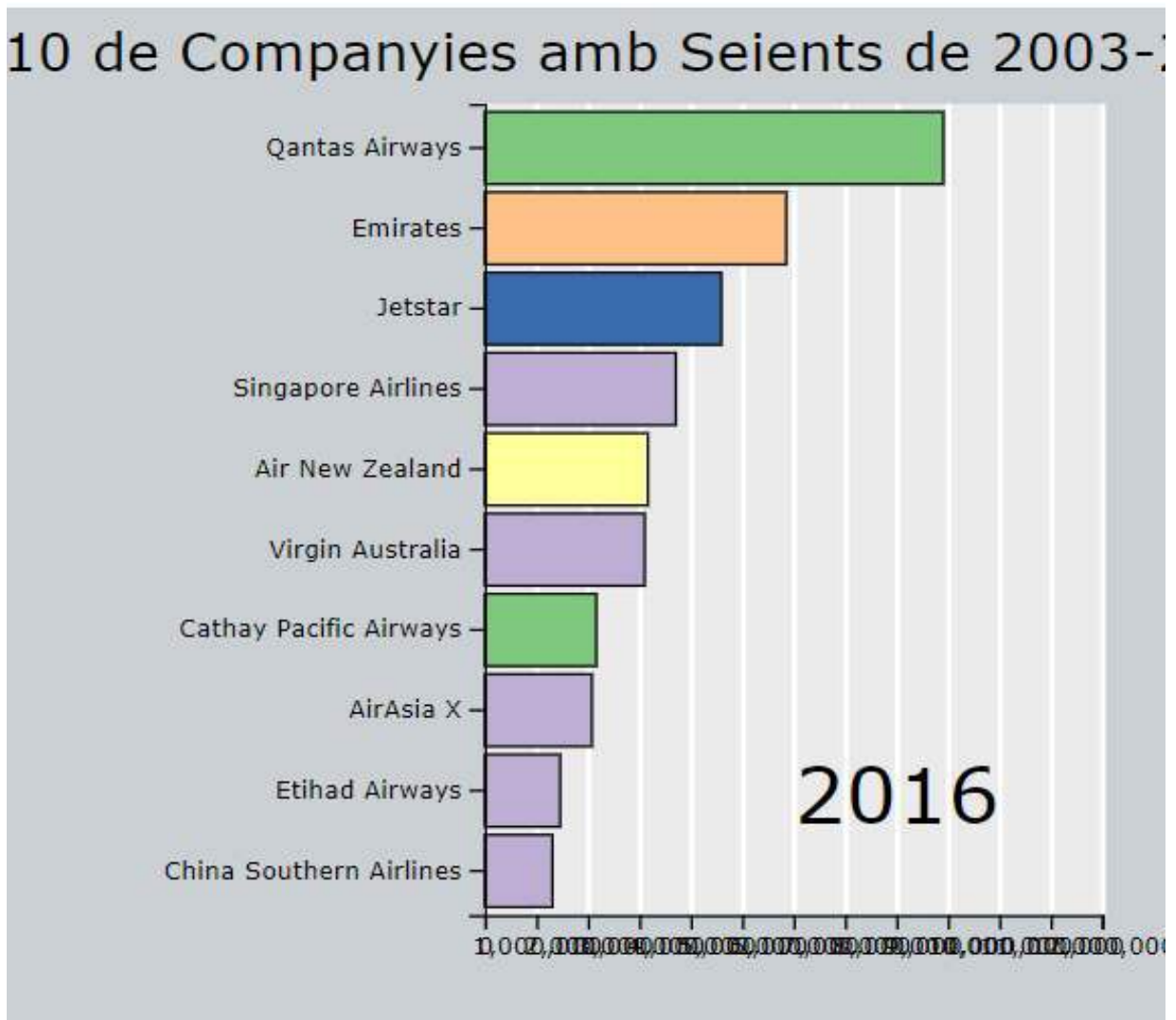
PAS 2: ANIMATED BAR RACE RANKING:

```
> anim <- ggplot(Australian_FD_Seients_formatted, aes(rank, group = Airline,
  fill = as.factor(Airline), color = as.factor(Airline))) +
  geom_tile(aes(y = NSeients/2,
    height = NSeients,
    #width = 0.9), alpha = 0.8, color = "black") +
    width = 0.9), alpha = 0.8, color = NA) +
  geom_text(aes(y = 0, label = paste(Airline, " ")), vjust = 0.2, hjust = 1) +
  geom_text(aes(y=NSeients,label = NSeients, hjust=0)) +
  coord_flip(clip = "off", expand = FALSE) +
  scale_y_continuous(labels = scales::comma) +
  scale_x_reverse() +
  guides(color = "none", fill = "none") +
  theme(axis.line=element_blank(),
    axis.text.x=element_blank(),
    axis.text.y=element_blank(),
    axis.ticks=element_blank(),
    axis.title.x=element_blank(),
    axis.title.y=element_blank(),
    legend.position="none",
    panel.background=element_blank(),
    panel.border=element_blank(),
    panel.grid.major=element_blank(),
    panel.grid.minor=element_blank(),
    panel.grid.major.x = element_line( size=.1, color="grey" ),
    panel.grid.minor.x = element_line( size=.1, color="grey" ),
```

ANIMATED BAR RACE RANKING. ALTERNATIVA (no avaluable, doncs no s'ha explicat, però es posa per a que ho coneixeu): Llibreria ddplot que defineix una gràfica de forma més senzilla i clara

```
> install.packages("ddplot") # Càrrega paquet ddplot (primera vegada)
> library(ddplot)           # Generació de Racing Bar Charts

> Australian_FD_Seients_formatted %>%
  barChartRace(
    x = "NSeients",
    y = "Airline",
    time = "Year",
    title = "Top 10 de Companyies amb Seients de 2003-2022"
  )
```



No hem trobat encara la forma de generar fitxer de vídeo amb aquesta funció.

1.4 (1 pt) Classifica les següents preguntes o mesures que s'utilitzen en tests d'Usabilitat o d'Experiència d'Usuari (UX) en les següents dues categories:

- a) Instrumentals / Usabilitat / Pragmàtiques
- b) No instrumentals / Hedònics / Emocionals

PREGUNTES:

1. *Conservative 1 2 3 4 5 6 7 Innovative*
2. *Non-inclusive 1 2 3 4 5 6 7 Inclusive*
3. *This System has all the functions and capabilities I expect it to have*
4. *Complicated 1 2 3 4 5 6 7 Simple*
5. *I thought the System was wasy to use*
6. *The product is stylish: Strongly Disagree 2 3 4 5 6 Strongly Agree*
7. *The design looks attractive*
8. *I found the various functions in this System very well integrated*
9. *It was easy to find the information I needed*
10. *The product is creatively designed*

RESPOSTA:

PREGUNTA	CATEGORIA	TIPUS DE TEST
1	b)	Attrakdiff, Hedonic
2	b)	Attrakdiff, Hedonic
3	a)	PSSUQ
4	a)	Attrakdiff, Pragmatic
5	a)	SUS
6	b)	meCUE, A3, Aesthetic
7	b)	meCUE, A2, Aesthetic
8	a)	SUS
9	a)	PSSUQ
10	b)	meCUE, A1, Aesthetic

1.5 (1 pt) Defineix 4 dels següents conceptes en animació, interactivitat, usabilitat i Experiència d'Usuari en Visualització de Dades:

- Usabilitat
- *plotly_click*
- *Utility*
- *Toggle*
- *Efficiency*

2025
10-13 JULY



MAD
COOL
FESTIVAL



- Control

RESPOSTA:

PART 2 (4 pt)

Dataset: *Australian_Flight_dataset_2003-2022.csv*

Continuarem utilitzant el mateix dataset que en la part 1 de l'examen.

2.1 (2 pt) Fes un bubble map per representar el nombre de rutes que van sortir l'any 2018 de cadascuna de les següents ciutats d'Austràlia: "Adelaide", "Brisbane", "Cairns", "Darwin", "Melbourne", "Perth", "Sydney".

Primer necessites crear una *tibble* amb les latituds i longituds d'aquestes ciutats, les quals et donem:

```
~ Australian_City, ~lat, ~lon,  
"Sydney", -33.8688, 151.2093,  
"Melbourne", -37.8136, 144.9631,  
"Brisbane", -27.4698, 153.0251,  
"Adelaide", -34.9285, 138.6007,  
"Perth", -31.9505, 115.8605,  
"Cairns", -16.9230, 145.7662,  
"Darwin", -12.4612, 130.8418,
```

Aquesta *tibble* l'hauràs d'ajuntar amb un dataframe construït contenint el nombre de rutes per ciutat de l'any 2018. Per ajuntar ambdós dataframes pots usar la comanda de R:

```
> inner_join (nom_tibble, nom_dataframe, by="nom_variable_compartida")
```

- Indica el datamassage necessari (0,25 pts).
- Feu un bubble map d'Austràlia, que doni informació sobre el nombre de rutes per cada ciutat. Podeu usar el mapa *oz_states* que vam usar al seminari. Al dibuixar-lo treu la graella del fons (1,5 pts).

RESPOSTA:

(a) Feu la *tibble*

```
> library(tidyverse)  
> library(dplyr)  
> oz_cities <- tibble::tribble(  
~ Australian_City, ~lat, ~lon,  
"Sydney", -33.8688, 151.2093,  
"Melbourne", -37.8136, 144.9631,  
"Brisbane", -27.4698, 153.0251,  
"Adelaide", -34.9285, 138.6007,  
"Perth", -31.9505, 115.8605,  
"Cairns", -16.9230, 145.7662,  
"Darwin", -12.4612, 130.8418,)
```

Guardant només les ciutats de les quals us hem donat la longitud i latitud, construïu un dataframe, que contingui el nombre de rutes que surten (In_Out=O) de cada ciutat de l'any 2018. Un cop el tingueu, ajunteu-lo amb el dataframe creat amb dites latituds i longituds. Per això últim, si ho feu en R, podeu utilitzar la comanda `inner_join` donada:

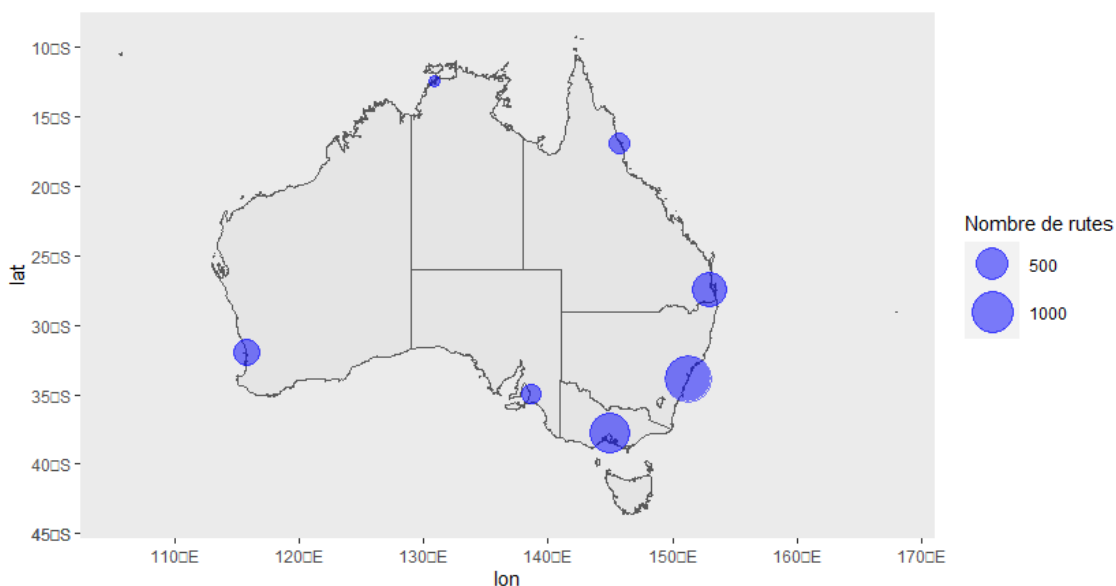
```
> Australian_FD_Rutes2018<- Australian_FD%>%filter(Australian_City %in%
c("Adelaide","Brisbane", "Cairns", "Darwin", "Melbourne", "Perth", "Sydney")) %>%
filter (Year==2018) %>%filter(In_Out=="O")%>% group_by(Australian_City) %>%
summarize(NRutes = n())

> dades<-inner_join(Australian_FD_Rutes2018,oz_cities,by="Australian_City")
```

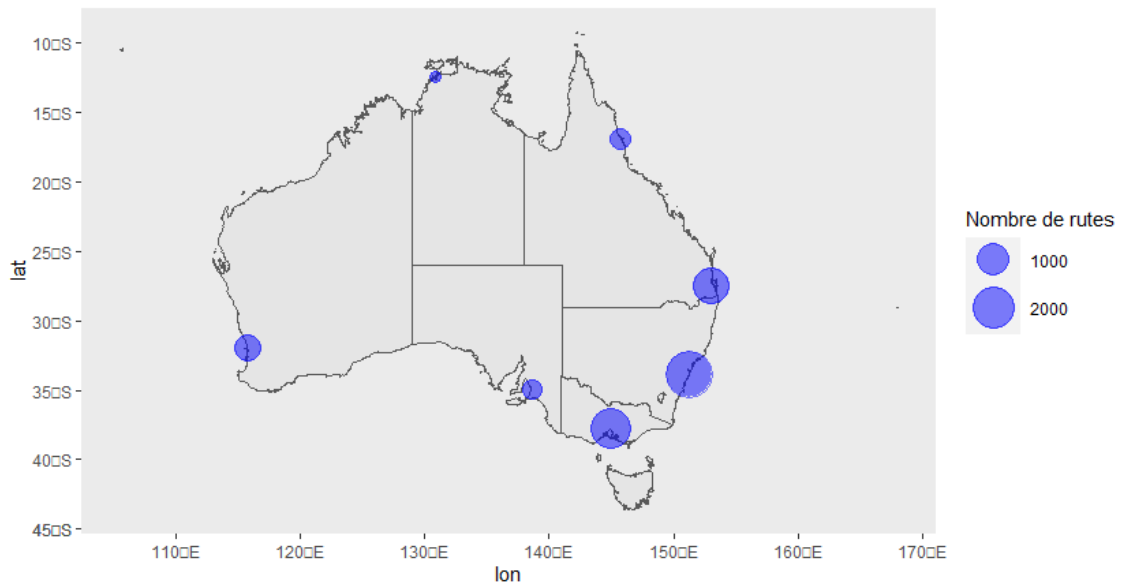
b) Sobre el mapa de `oz_states` Dibuixeu tot traient la graella del fons. Feu un bubble map on la talla de les bombolles creix amb el nombre de rutes per ciutat

```
> library(ozmaps)
> library(sf)
> oz_states <- ozmaps::ozmap_states

> ggplot() + geom_sf(data = oz_states)+geom_point(data = dades, mapping =
aes(x = lon, y = lat,size=NRutes), alpha=0.5,
color="blue")+coord_sf()+theme(panel.grid.major = element_blank(),
panel.grid.minor = element_blank()) +scale_size_continuous(range=c(3,15),
"Nombre de rutes")
```



També es compta bé si no s'ha especificat el filtre per agafar només les rutes de sortida. En aquest cas el mapa seria:



2.2 (2 pt) Mostra un mapa d'arbre (treemap) que et permeti saber des de quina de les dues ciutats, Perth o Sydney, van volar més vols l'any 2018 a cada regió de servei (Service_Region).

a) Fes el datamassage necessari (0.5 pt).

b) Argumenta com és un *treemap* en general i els passos que has de fer per construir la visualització d'aquest exercici (és a dir, quina variable utilitzes per l'àrea de les graelles, variables d'agrupació, etc). Inclou el codi si el fas en R (1 pt).

c) Mostra el mapa d'arbre, posa llegenda i títol i posa una paleta de color (en R podeu usar una paleta manual, per exemple amb els colors `c('#87C55F', '#9EB9F3')`, o una altra al vostre gust). Quina és/són la/les regions de destí on l'any 2018 no van volar vols des de Perth? (0.5 pt).

RESPOSTA:

a)

```
> library("tidyverse")
```

```
> Aust_FD_regiodesiti2018<- Australian_FD%>%filter(Australian_City %in%
c("Perth", "Sydney")) %>% filter (Year==2018) %>%filter(In_Out=="0")
```

(Com abans, no es penalitza si no es fa el `filter(In_Out=="0")`)

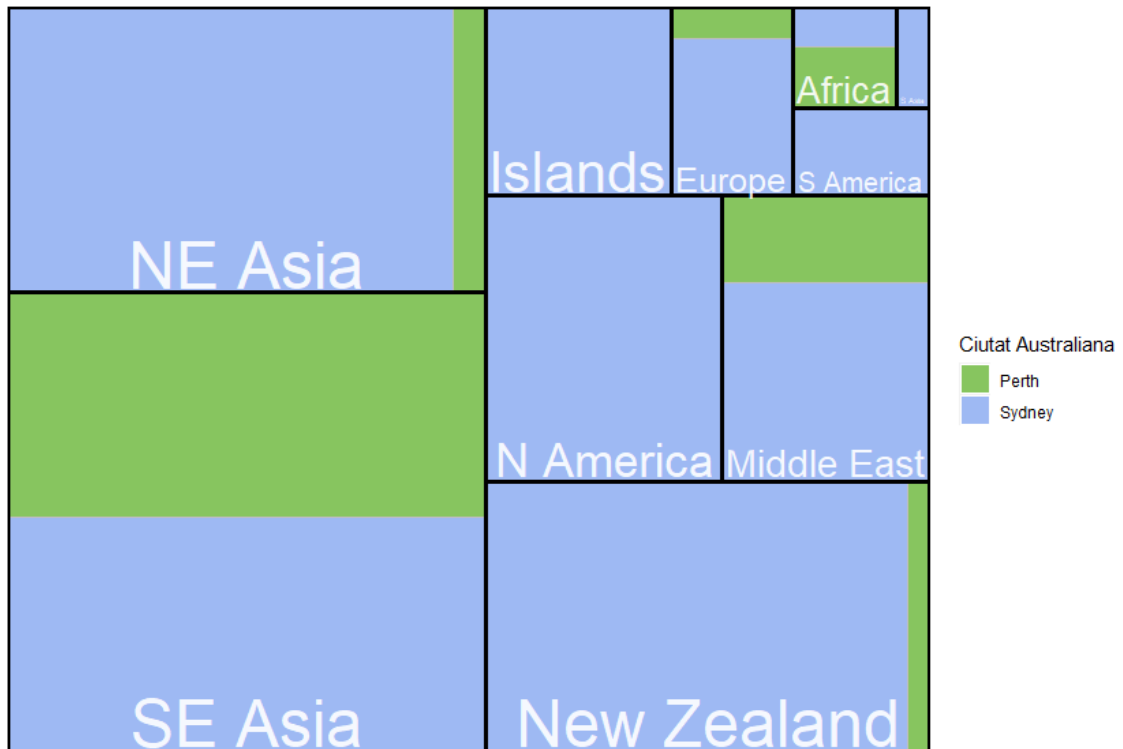
b) Com vam veure a classe, un mapa d'arbre és un dibuix rectangular dividit en caselles, i cada casella representa una sola observació. Vam veure que era una bona manera de mostrar dades jeràrquiques mitjançant rectangles imbricats. I l'àrea relativa de cada casella expressava una variable contínua. També vam veure que era òptim quan hi ha com a màxim dues variables d'agrupació, per tant no en definirem més.

Podeu a més ordenar la visualització:

```
>Aust_FD_regiodesiti2018<-Aust_FD_regiodesiti2018%>%group_by(Australian_City,  
Service_Region)%>%summarise(ALL_Flights=sum(ALL_Flights))
```

Us sortirà un warning que no té major importància, però que podeu treure (veure: <https://statisticsglobe.com/dplyr-message-summarise-has-grouped-output-r>)

Regions on es podia volar des de Perth i Sidney al 2018



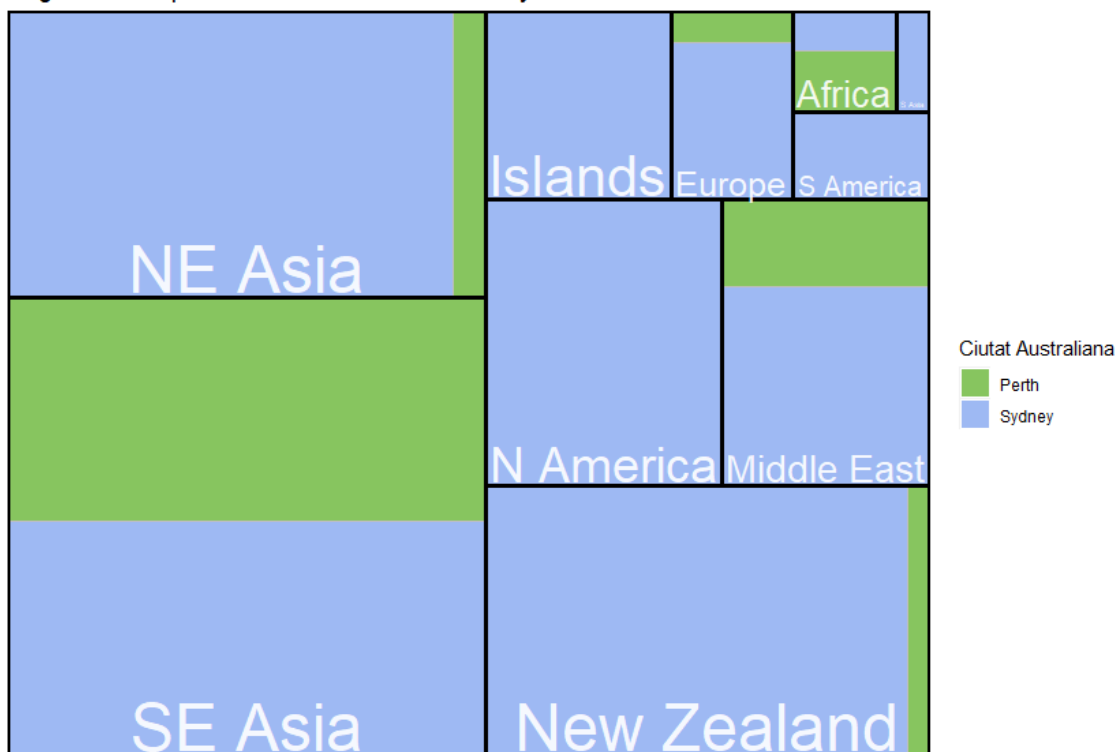
I si no es fa el `filter(In_Out=="O")`:

Regions on es podia volar des de Perth i Sidney al 2018



O ordenant com abans:

Regions on es podia volar des de Perth i Sidney al 2018



En qualsevol cas, al 2018 no hi va haver vols des de Perth a les regions: d'Amèrica del Nord (N America), Amèrica del Sud (S America), Islands i Àsia del Sud (S Asia)

2025
10-13 JULY



MAD
COOL
FESTIVAL



OLIVIA RODRIGO



GRACIE ABRAMS



JUSTICE



GLASS ANIMALS



ROYEL OTIS



ARDE BOGOTÁ



KINGS OF LEON



NINE INCH NAILS



ALANIS MORISSETTE



NSQK



BENSON BOONE