

2025
10-13 JULY

MAD
COOL
FESTIVAL



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

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 dataset:

```
> 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 del nombre de rutes de cada companyia, siguin de sortida o arribada a qualsevol ciutat australiana (gràfica 1) i fixa't amb les tres companyies aèries amb més rutes l'any 2017. Diques 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 Massaging.

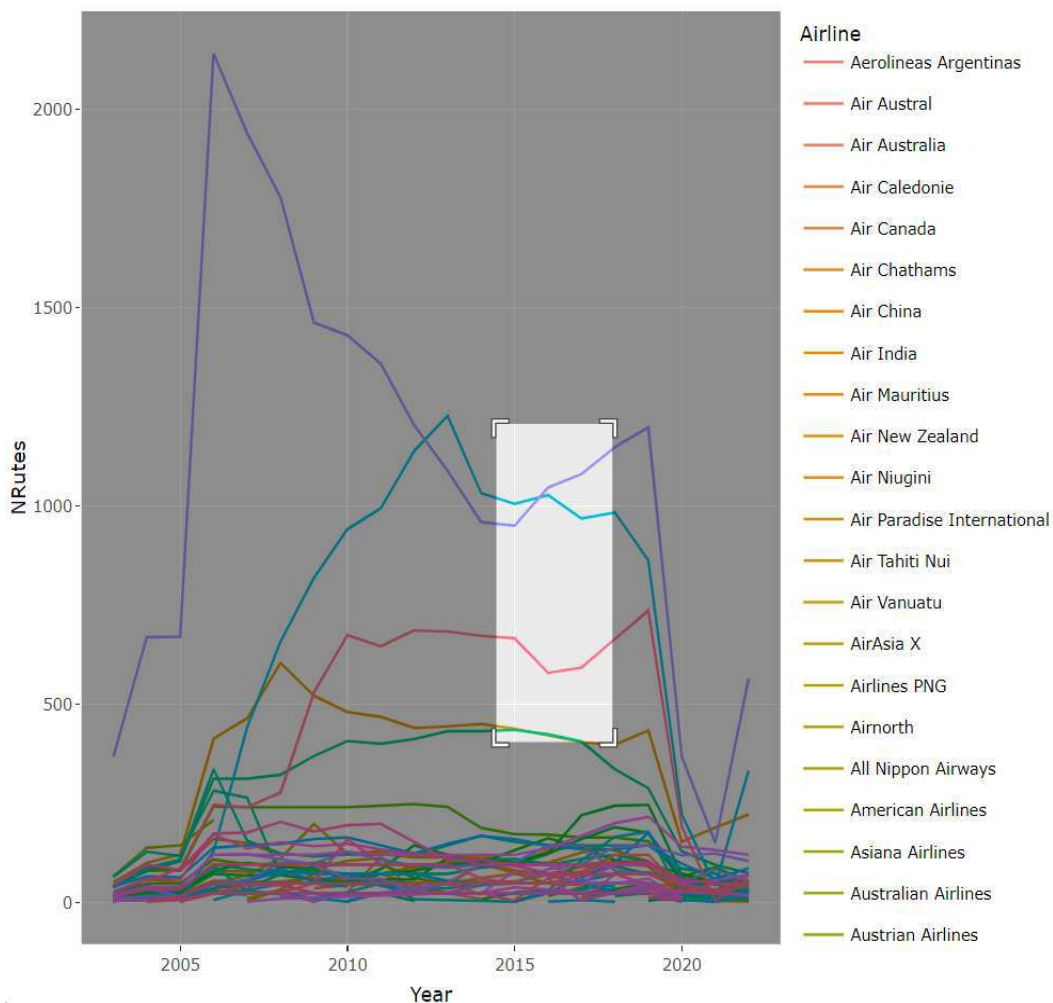
RESPOSTA: *En primer lloc, hem de fer data massaging sobre el data set original.*

DATA MASSAGING: *Agrupar per (Any, Companyia Aèria, Sexe) calculant el nombre de rutes:*

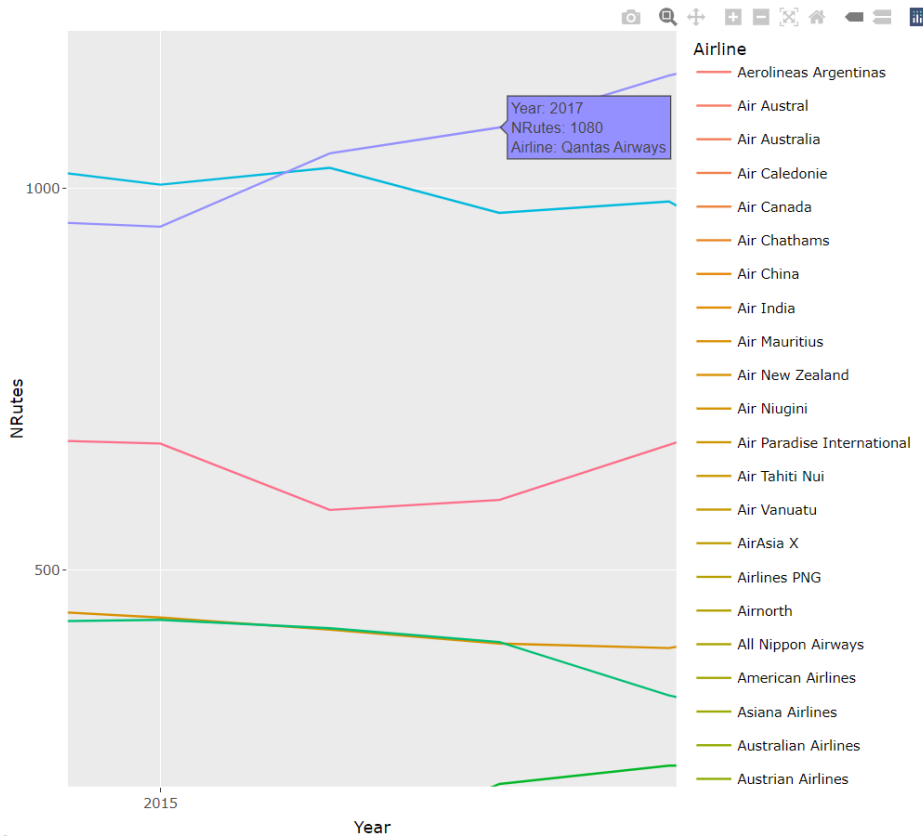
```
> Australian_FD_Rutes <- Australian_FD %>% group_by(Year,Airline) %>%
summarize(NRutes = n())
```

GRÀFICA 1 de línies amb ggplotly:

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



I fer un zoom en la zona propera a l'any 2017:



*Els noms de les companyies són **Qantas Airways (1080 rutes)**, **Jetstar (968)** i **Virgin Australia (592)***

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

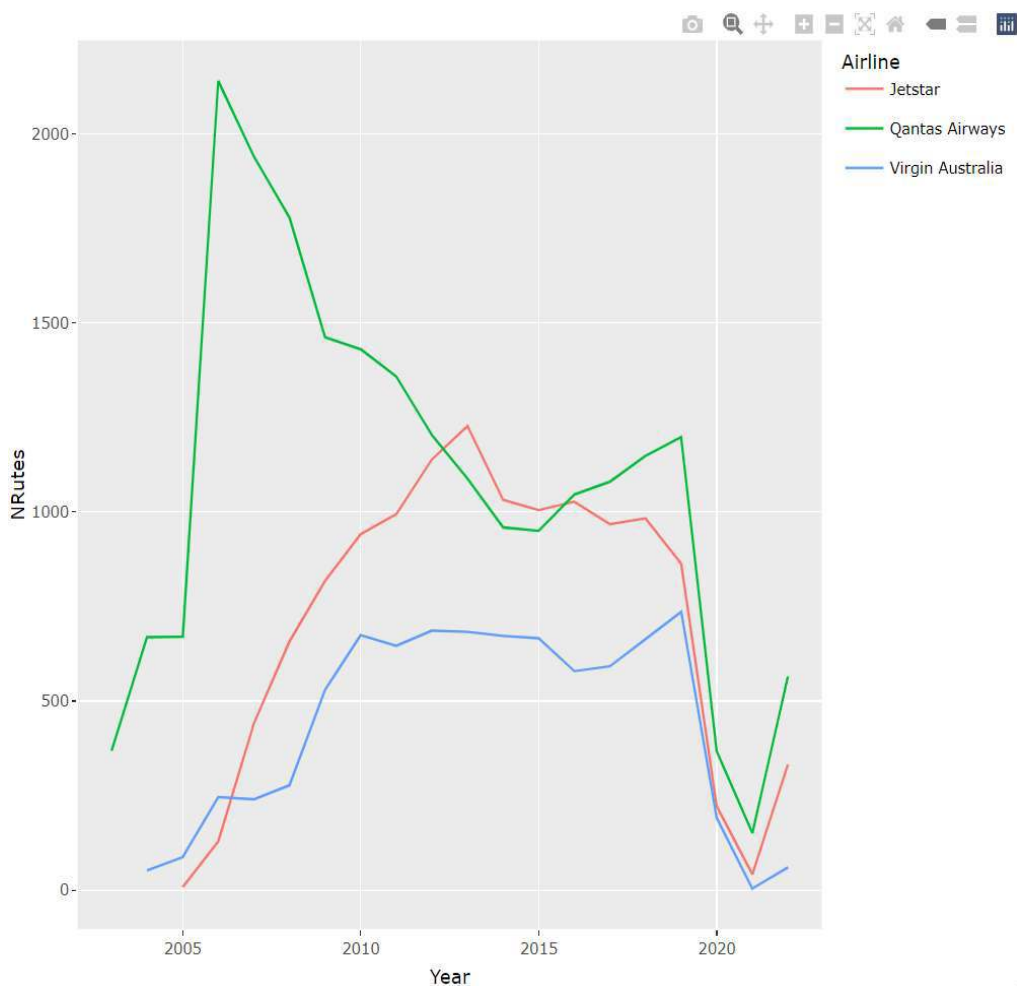
AMB ggplotly (abans filtrem el data set per les tres companyies aèries):

```
> Australian_FD_Rutes_QJV <- Australian_FD_Rutes %>% filter(Airline == "Qantas Airways" | Airline == "Jetstar" | Airline == "Virgin Australia")
> ggplotQJV <- ggplot(Australian_FD_Rutes_QJV, aes(x=Year, y=NRutes, color=Airline)) + geom_line()
> ggplotly(ggplotQJV)
>
```

2025
10-13 JULY



MAD
COOL
FESTIVAL



AMB shiny + plotly:

```
ui <- fluidPage(
  selectizeInput( inputId = "Airline",
    label = "Selecciona una COMPANIAIA:",
    choices = unique(Australian_FD_Rutes_QJV$Airline),
    selected=c("Qantas Airways","Jetstar",
      "Virgin Australia"),
    multiple = TRUE
  ),
  plotlyOutput(outputId = "p")
)

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

Selecciona una COMPANYIA:

Qantas Airways Virgin Australia Jetstar



1.2 (0,5 pts) 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
Qantas Airways	2021 - 151	2006 - 2140
Jetstar	2005 - 8	2013 - 1227
Virgin Australia	2021 - 4	2019 - 736

b) **(0,1 pts)** Comenta la taula anterior. Trobes alguna singularitat?. Sabries el per què?.

RESPOSTA: El mínim nombre de rutes de les companyies coincideix el 2021, amb 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
Virgin Australia-Jetstar	2006	2007
Jetstar-Qantas Airways	2012	2013
Qantas Airways-Jetstar	2015	2016

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 rutes per anys, del 2003 al 2020.

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
```

PAS 1: DATA MASSAGING:

- Filtrar el dataset pels anys 2003 al 2020:

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

PAS 2: ANIMATED BAR RACE RANKING:

```
> anim <- ggplot(Australian_FD_Rutes_formatted, aes(rank, group = Airline,
  fill = as.factor(Airline), color = as.factor(Airline))) +
  geom_tile(aes(y = NRutes/2,
    height = NRutes,
    #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=NRutes,label = NRutes, 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" ),
    plot.title=element_text(size=25, hjust=0.5, face="bold", colour="grey",
vjust=-1),
```


2025
10-13 JULY

MAD
COOL
FESTIVAL



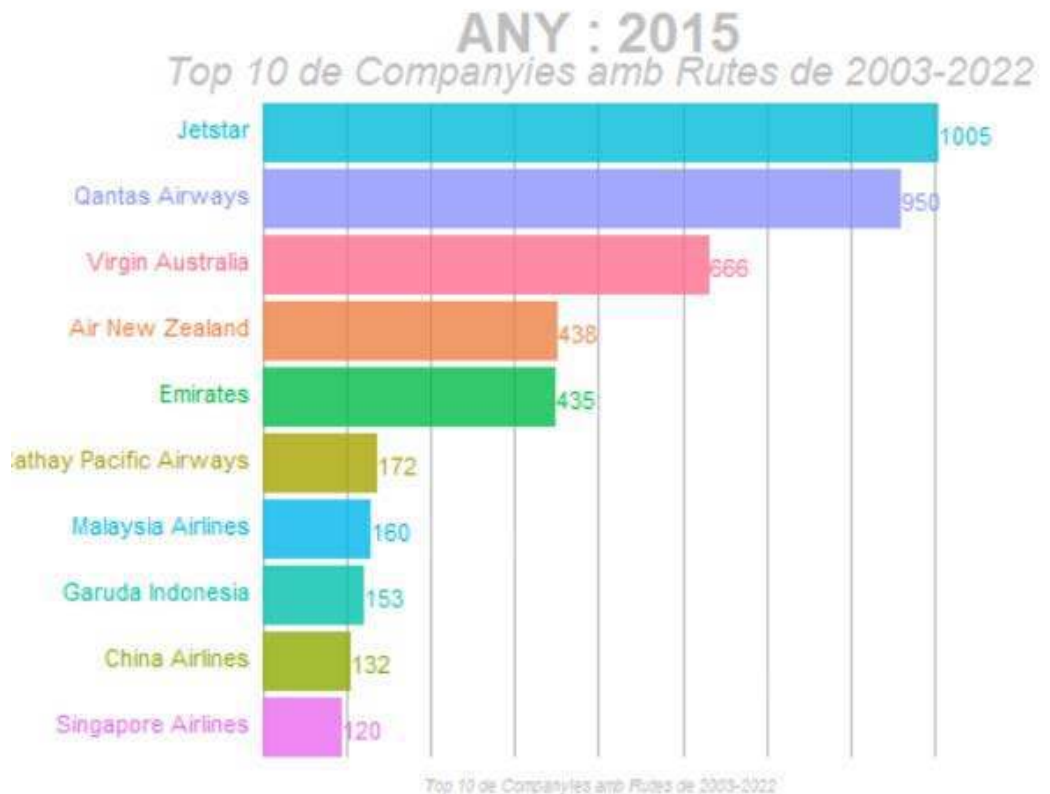
```

        plot.subtitle=element_text(size=18,      hjust=0.5,      face="italic",
color="grey"),
        plot.caption      =element_text(size=8,      hjust=0.5,      face="italic",
color="grey"),
        plot.background=element_blank(),
        plot.margin = margin(2,2, 2, 4, "cm")) +
        transition_states(Year, transition_length = 4, state_length = 1, wrap = FALSE)
+
        view_follow(fixed_x = TRUE) +
        labs(title = 'ANY : {closest_state}',
        subtitle = "Top 10 de Companyies amb Rutes de 2003-2022",
        caption = "Top 10 de Companyies amb Rutes de 2003-2022")
> anim

# PAS 2. Exportar frames a fitxer GIF
> animate(anim, 200, fps = 20, duration= 30, width = 1200, height = 1000,
        renderer = gifski_renderer("USNamesF.gif"), end_pause = 15, start_pause
= 15)
>

# PAS 3. Exportar frames a fitxer AVI
> animate(anim, 200, fps = 20, duration= 30, width = 1200, height = 1000,
        renderer = av_renderer("USNamesF.avi"), end_pause = 15, start_pause =
15)

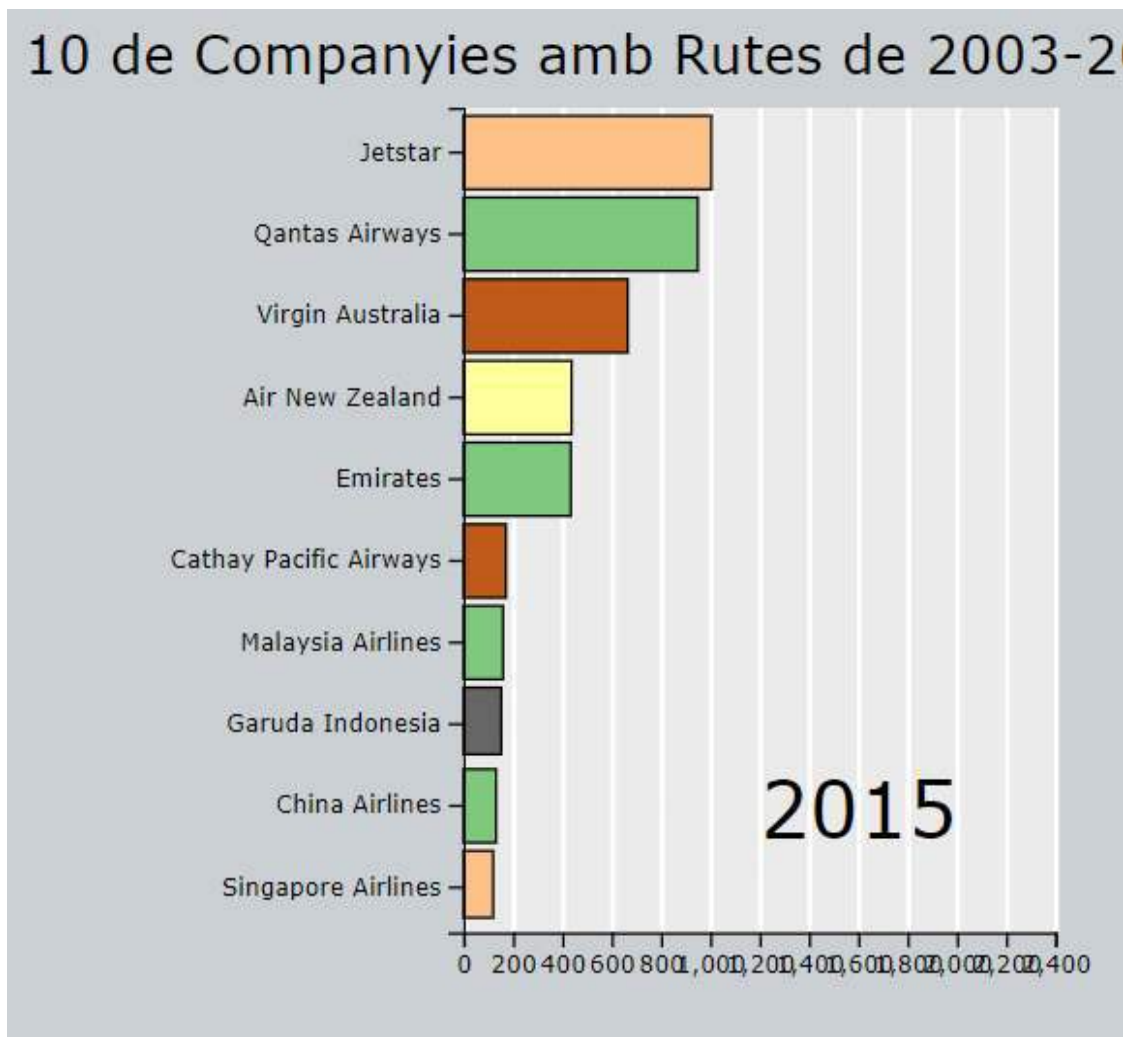
```



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_Rutes_formatted %>%
  barChartRace(
    x = "NRutes",
    y = "Airline",
    time = "Year",
    title = "Top 10 de Companyies amb Rutes 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. *Complicated 1 2 3 4 5 6 7 Simple*
2. *Non-inclusive 1 2 3 4 5 6 7 Inclusive*
3. *The design looks attractive*
4. *It was easy to find the information I needed*
5. *I thought the System was wasy to use*
6. *Conservative 1 2 3 4 5 6 7 Innovative*
7. *The product is creatively designed*
8. *This System has all the functions and capabilities I expect it to have*
9. *I found the various functions in this System very well integrated*
10. *The product is stylish: Strongly Disagree 2 3 4 5 6 Strongly Agree*

RESPOSTA:

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

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

- Event
- *plotly_hover*
- *Data linking*
- *Slider*
- *Learnability*
- *Toggle*

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 2022. Un cop tingueu el dataframe, 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_Rutes2022<- Australian_FD%>%filter(Australian_City %in%
c("Adelaide","Brisbane", "Cairns", "Darwin", "Melbourne", "Perth",
"Sydney")) %>% filter (Year==2022) %>%filter(In_Out=="O")
%>% group_by(Australian_City) %>% summarize(NRutes = n())
```

(No es penalitza si no es fa el `filter(In_Out=="O")`)

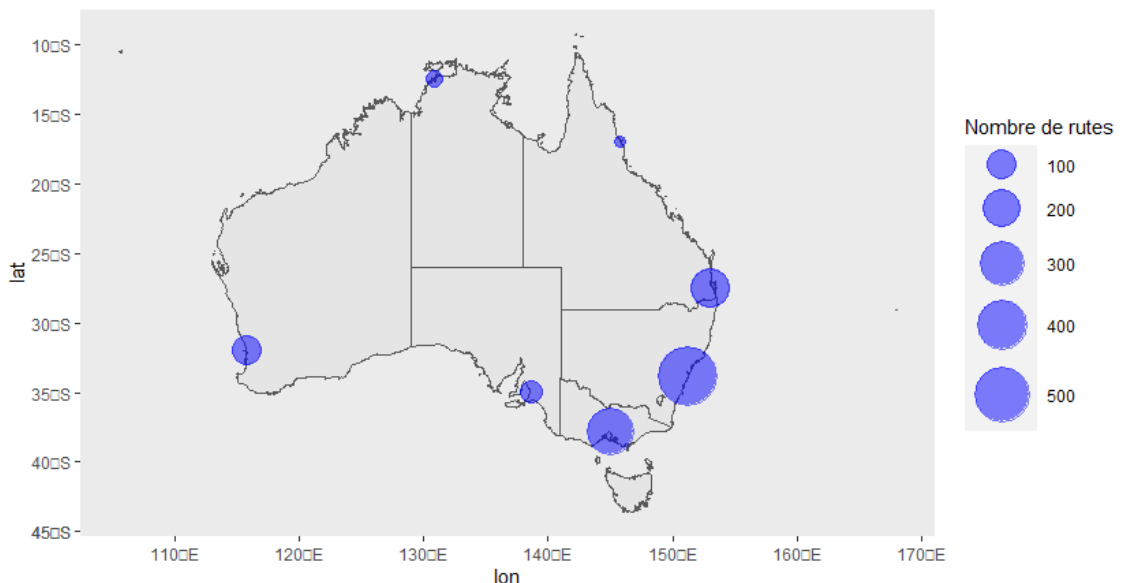
```
> dades<-
inner_join(Australian_FD_Rutes2022,oz_cities,by="Australian_City")
```

(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

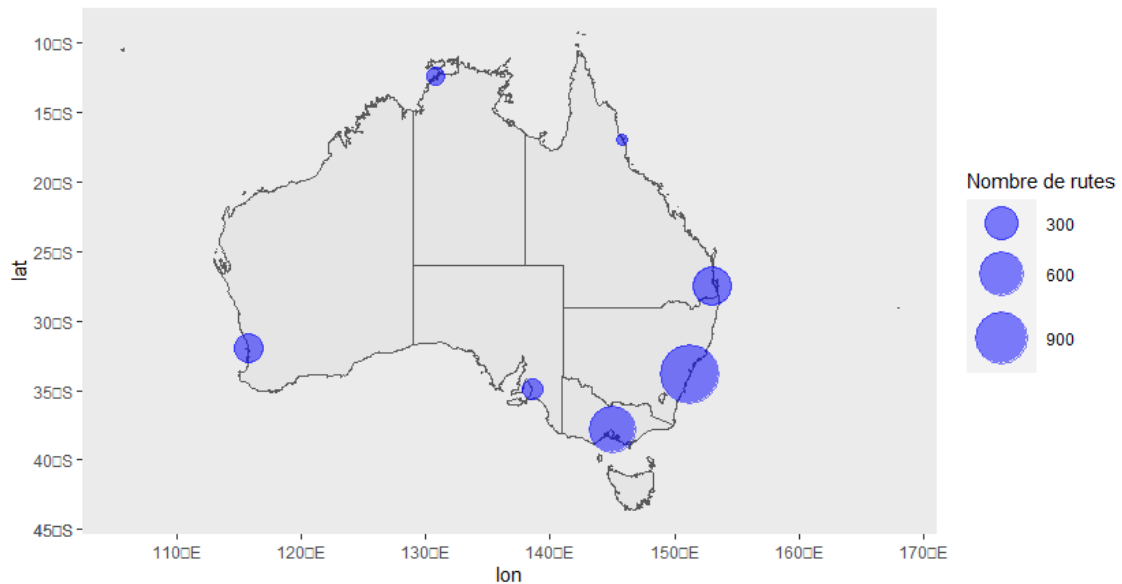
```
> library (ozmaps)
> library (sf)

> 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, Melbourne o Sidney, van volar més vols l'any 2022 a cada regió de servei (Service_Region).

- Fes el datamassatge necessari (0.5 pt).
- 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).
- 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 2022 no van volar vols des de Melbourne? (0.5 pt).

RESPOSTA:

a)

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

```
> Aust_FD_regiodesiti2022<- Australian_FD%>%filter(Australian_City %in%  
c("Melbourne", "Sydney")) %>% filter (Year==2022)%>%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.

Per tant, una possibilitat amb aquest dataset seria:



- Per definir el color i actuar doncs com un 'grup pare' utilitzem la variable *Australian_City* (Agafem aquesta com grup de color, ja que sol té 2 nivells ("Melbourne", "Sydney") i serà més fàcil entendre la llegenda, que si posem *Service_Region* que en té més).
- Com a 'subgrup' utilitzaríem la regió de servei (*Service_Region*, que té més nivells que *Australian_City*).
- Com a variable que descriu l'àrea de les caselles triarem *All_Flights*. Aquesta tria ve donada perquè ens estan preguntant per regions amb més/menys vols des d'una certa ciutat. I de totes les variables que ens pregunten, és l'única numèrica.

Si ho fem amb R, el primer que hem de fer és carregar les llibreries necessàries. La llibreria específica aquí és *treemapify* (abans necessitarem tenir instal·lat el paquet com vam veure al seminari corresponent). També haurem de fer ús de *geom_treemap*.

Per fer un treemap basic doncs:

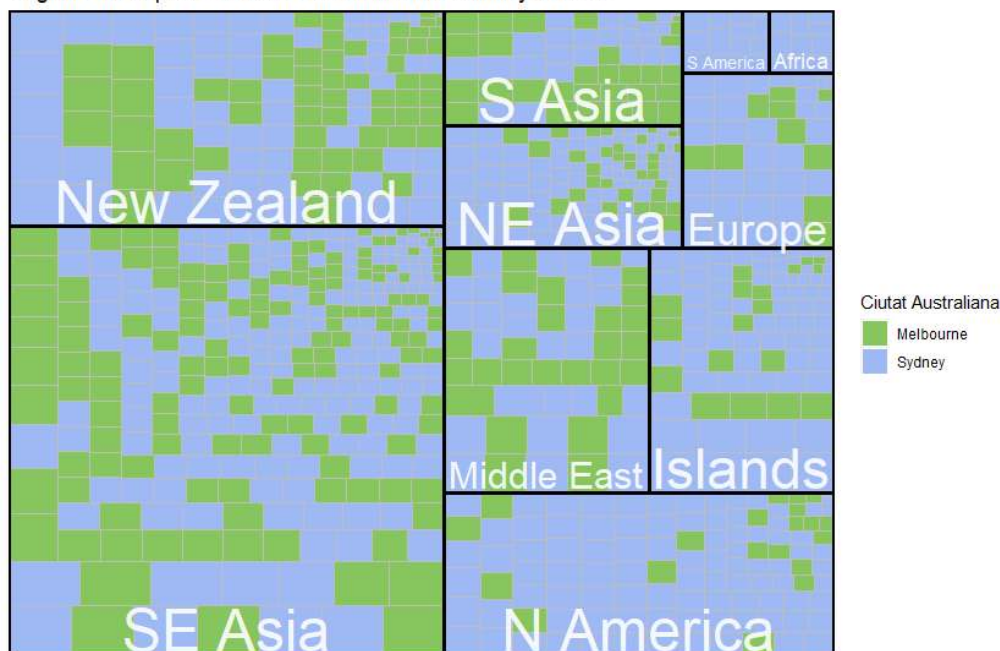
```
> library("treemapify")

> ggplot(Aust_FD_regiodesti2022, aes(area = All_Flights, fill =
  Australian_City, subgroup=Service_Region))+geom_treemap()+
  geom_treemap_subgroup_border(colour = "black",size=3)
+geom_treemap_subgroup_text(alpha=0.9,colour = "white")+labs(title =
  "Regions on es podia volar des de Melbourne i Sidney al 2022", fill="Ciutat
  Australiana") +scale_fill_manual(values=c('#87C55F','#9EB9F3'))
```

Fixeu-vos que hem posat el color amb *fill*:

```
scale_fill_manual(values=c('#87C55F','#9EB9F3'))
```

Regions on es podia volar des de Melbourne i Sidney al 2022

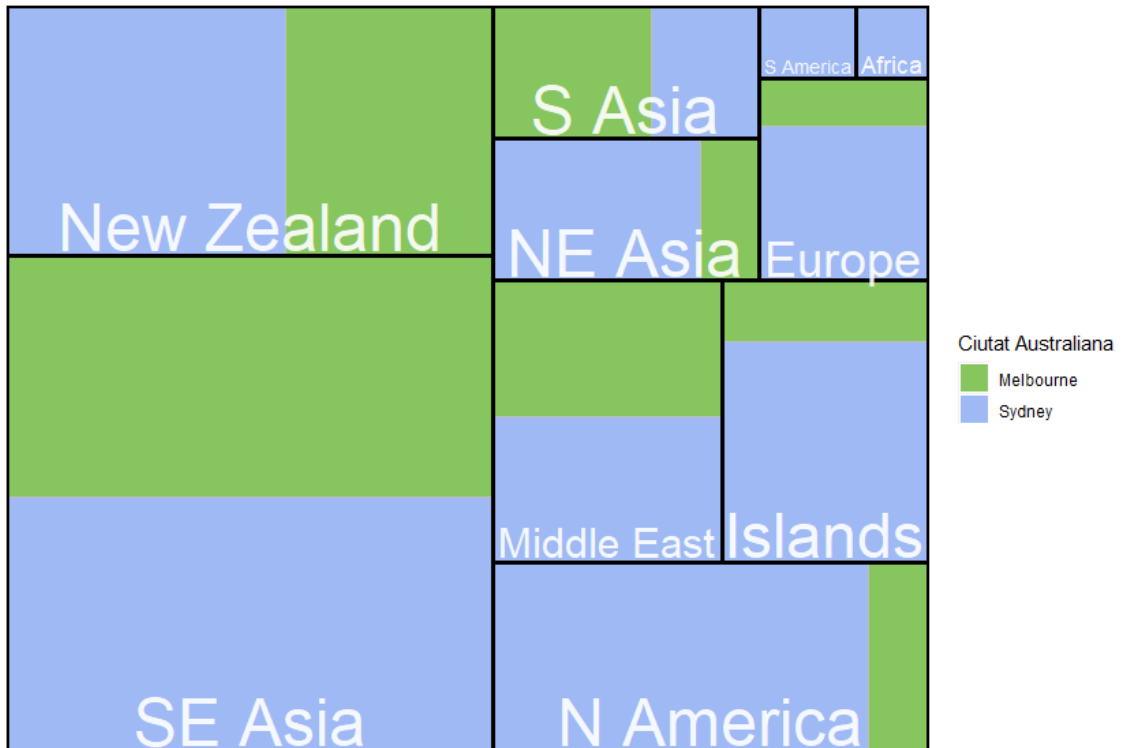


Podeu a més ordenar la visualització:

```
> Aust_FD_regiodesiti2022<-  
Aust_FD_regiodesiti2022%>%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 Melbourne i Sidney al 2022



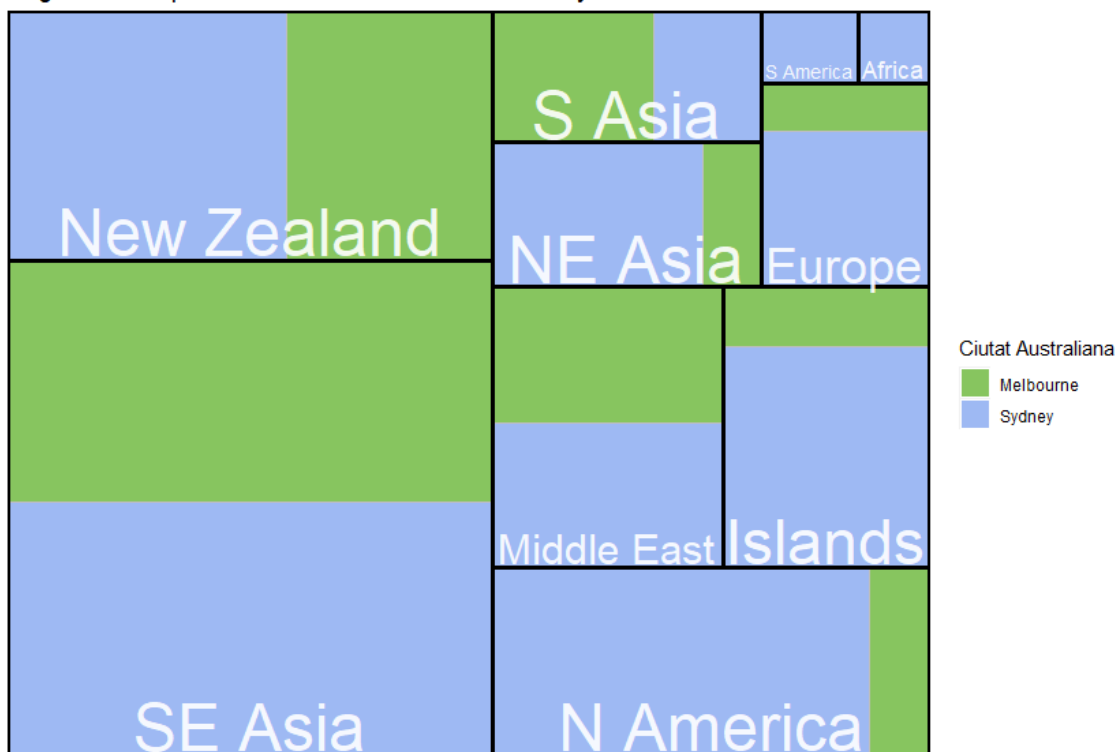
I si no es fa el filter(In_Out=="O"):

Regions on es podia volar des de Melbourne i Sidney al 2022



O ordenant com abans:

Regions on es podia volar des de Melbourne i Sidney al 2022



En qualsevol cas, la resposta és: A Àfrica i Sud Amèrica no hi van volar vols des de Melbourne al 2022.