SEMINARI 3.1 Gràfiques Exploratòries (Respostes)

1. OBJECTIUS

Aquest seminari introdueix l'ús de gràfiques exploratòries.

2. PART 1. Diagrames de barres o diagrames de sectors circulars

Com sempre, si obriu R de nou, primer de tot recordeu que heu de tornar a carregar la llibreria tidyverse.

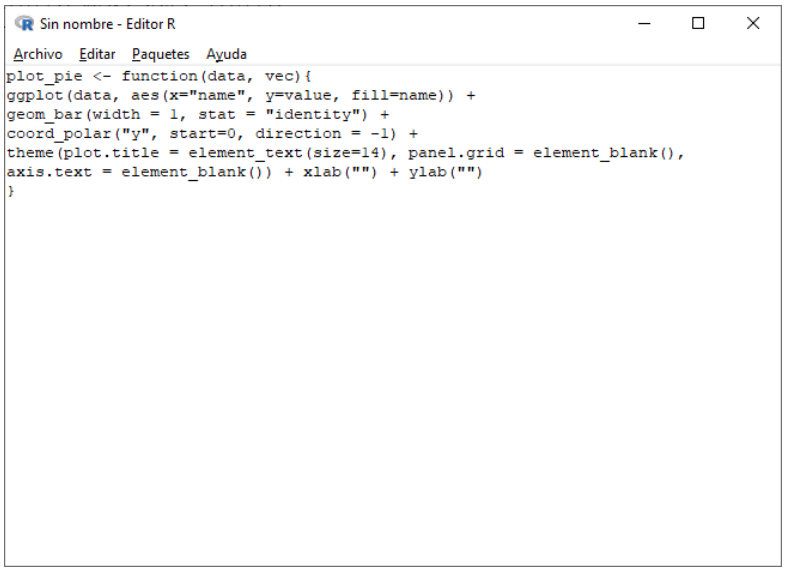
En aquesta primera part del seminari compararem l'ús de `pie()` i ggplot per crear un diagrama de sectors. Després compararem diagrames de sectors circulars amb diagrames de barres

1.- Diagrames de sectors circulars amb ggplot

a) Feu un script en R que contingui la funció `plot_pie` creada amb ggplot següent, i guardeu-la amb el nom plot_pie.R:

```
plot_pie <- function(data, vec){
      ggplot(data, aes(x="name", y=value, fill=name)) +
      geom_bar(width = 1, stat = "identity") +
      coord_polar("y", start=0, direction = -1) +
      theme(plot.title = element_text(size=14), panel.grid =
      element_blank(),
      axis.text = element_blank()) + xlab("") + ylab("")
      }
```

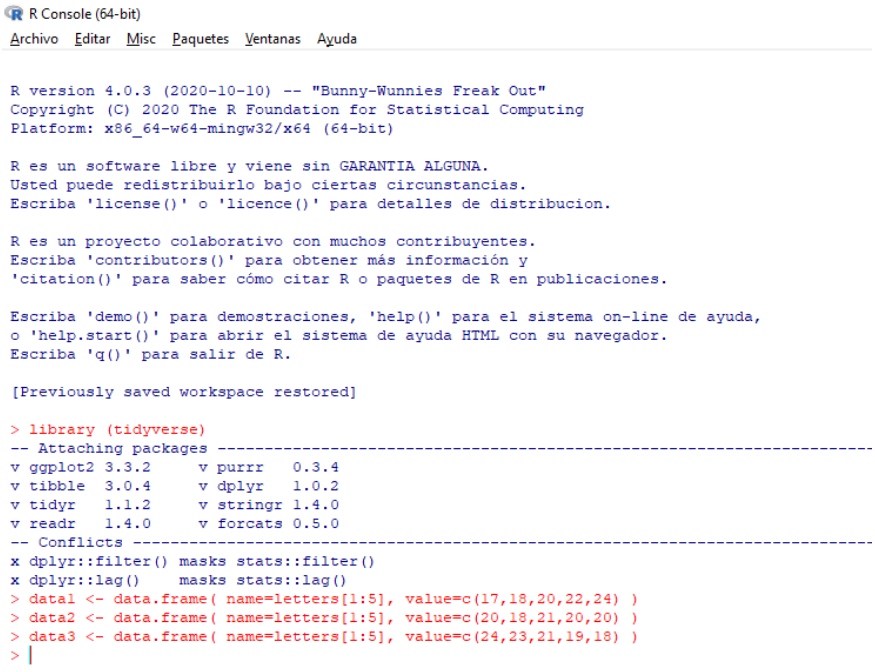
Fem Archivo-> Nuevo script i se'ns obre una pantalla on copiem el text



Guardem l'arxiu posant com a nom (per exemple) plot_pie.R (Recordeu triar en quin directori el guardeu, ja que després haureu de saber el directori per tal de fer-ne us)

b) Creeu tres datasets

```
data1 <- data.frame( name=letters[1:5], value=c(17,18,20,22,24) )
data2 <- data.frame( name=letters[1:5], value=c(20,18,21,20,20) )
data3 <- data.frame( name=letters[1:5], value=c(24,23,21,19,18) )
```



c) Utilitzant el script que heu creat, dibuixeu els següents gràfics de sectors i guardeu-los en fitxers .png:

- Un pie chart on *data* sigui el primer dataset creat (`data1`) i *vec* sigui `c(10,35,55,75,93)`

- Un pie chart on *data* sigui el segon dataset creat (`data2`) i *vec* sigui `c(10,35,53,75,93)`

- Un pie chart on *data* sigui el tercer dataset creat (`data3`) i `vec` sigui `c(10,29,50,75,93)`

Com heu vist a les diapositives primer hem de cridar el script

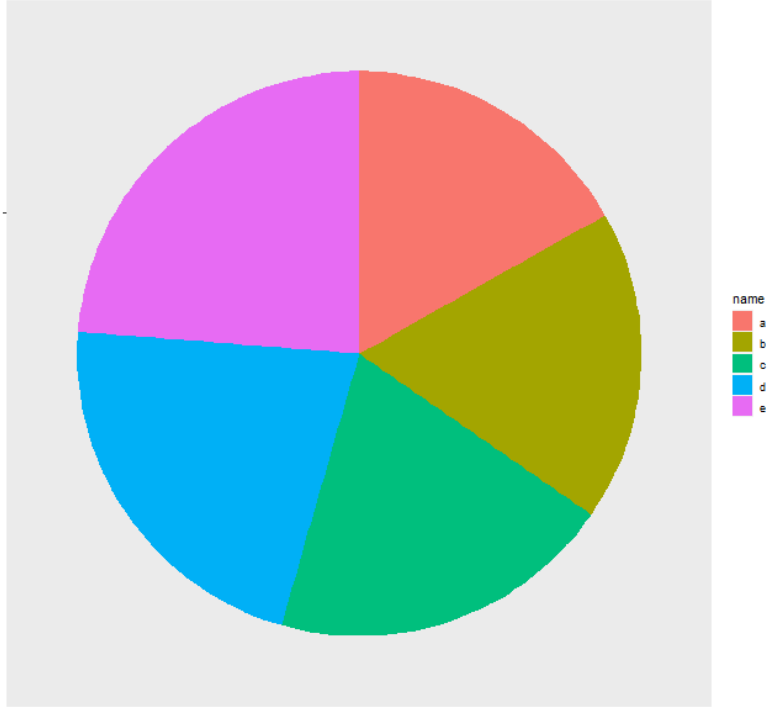
```
> source("<name_path>/plot_pie.R") # Recordeu especificar en <name_path> l'adreça/directori on heu guardat el script
```

Ara només us cal cridar la funció `plot_pie()` com ens indica cada subapartat. Però a més volem guardar els gràfics amb extensió .png Per tant, com heu vist a les diapositives, per cada subapartat primer crideu a la funció `png()`, després crideu `plot_pie()` i després tanqueu i guardeu amb `dev.off()`. Exemple:

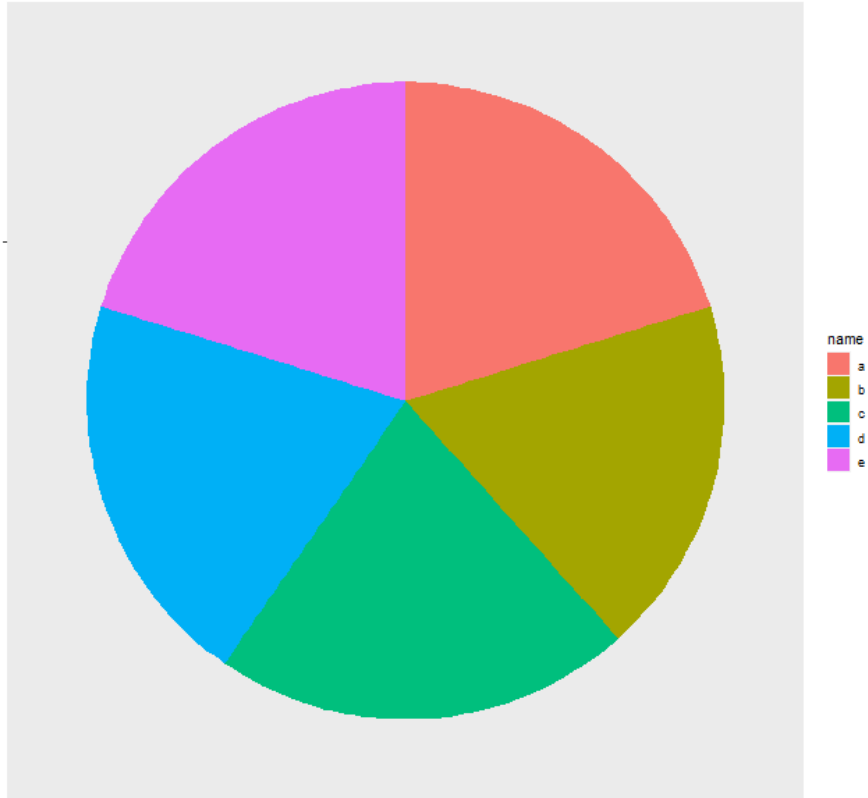
```
> png("<name_path>/pie_data1.png",width=800,height=800) # Actualitzar path i nom
> plot_pie(data1, c(10,35,55,75,93))
> dev.off()
```

Tindreu tres fitxers .png:

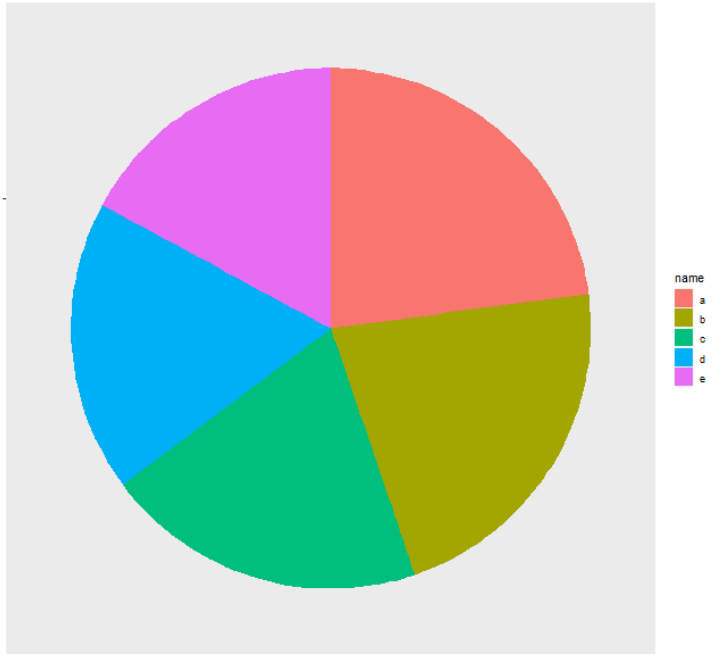
pie_data1.png



pie_data2.png



pie_data3.png



2.- Mostreu la mateixa informació que se us demanava en l'exercici 1.c, però ara utilitzeu un diagrama de barres. Compareu un a un amb el respectiu diagrama de sector circular que heu guardat en l'exercici 1. Si tinguéssiu que triar entre utilitzar diagrames de barres o diagrames de sectors circulars, què utilitzaríeu, per què?

Per cada dataset fem:

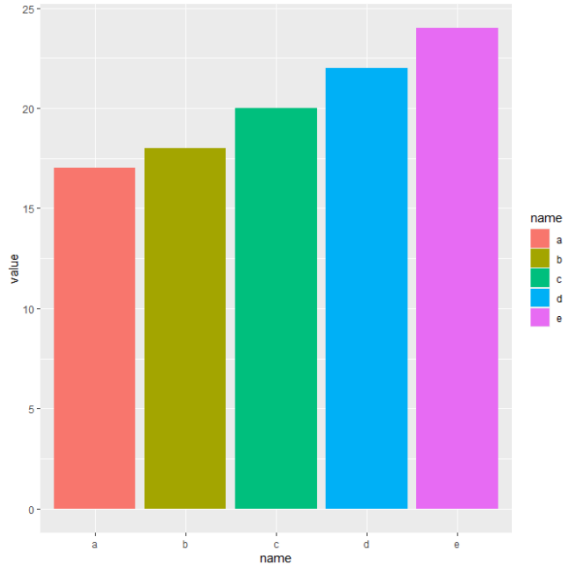
```
> ggplot(data1, aes(x=name, y=value, fill=name))+ geom_bar( stat = "identity")
```

o el que és el mateix en aquest cas on `stat=` "identity":

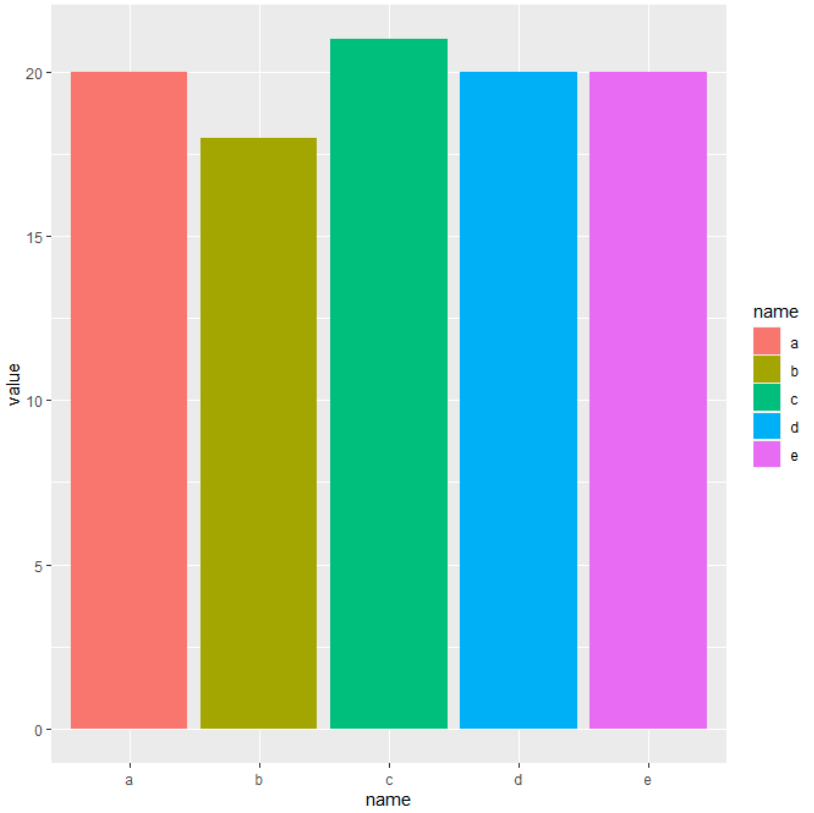
```
> ggplot(data1, aes(x=name, y=value, fill=name))+ geom_col()
```

Obtindrem les gràfiques següents:

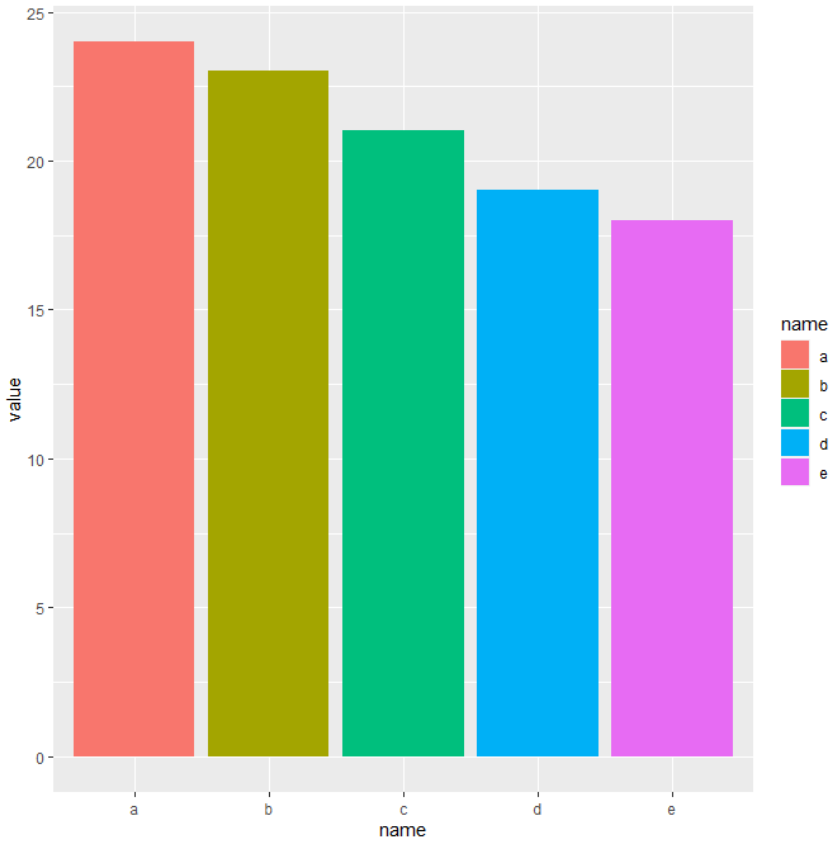
- Per data1



- Pel data2



- Pel data3



En el diagrama de barres veiem diferencies significatives que no es veien en els diagrames de sectors circulars.

! L'ull humà és més sensible a diferenciar alçades de barres que no pas a diferenciar entre angles. Només utilitzem diagrames de sectors circulars quan les diferencies proporcionals siguin molt grans. En un exemple com el d'aquest exercici, clarament els diagrames de barres ens ajuden més a mostrar la informació que tenim en les nostres dades.

3. PART 2. Distribucions & comparacions

En aquesta segona part del seminari anem a explorar un nou d*ataframe* i fer algunes gràfiques tot practicant amb comandes que ja hem vist. També farem us de la nova llibreria forcats que hem vist avui

Starwars és un *dataframe* on cada fila és una observació i cada columna és una variable

```
> starwars
# A tibble: 87 x 14
   name               height  mass hair_color    skin_color  eye_color birth_year sex    gender    homeworld species films     vehicles  starships
   <chr>               <int> <dbl> <chr>         <chr>       <chr>          <dbl> <chr>  <chr>     <chr>     <chr>   <list>    <list>    <list>
 1 Luke Skywalker        172    77 blond         fair        blue            19   male   masculine Tatooine  Human   <chr [5]> <chr [2]> <chr [2]>
 2 C-3PO                 167    75 <NA>          gold        yellow         112   none   masculine Tatooine  Droid   <chr [6]> <chr [0]> <chr [0]>
 3 R2-D2                  96    32 <NA>          white, blue red             33   none   masculine Naboo     Droid   <chr [7]> <chr [0]> <chr [0]>
 4 Darth Vader           202   136 none          white       yellow          41.9 male   masculine Tatooine  Human   <chr [4]> <chr [0]> <chr [1]>
 5 Leia Organa           150    49 brown         light       brown           19   female feminine  Alderaan  Human   <chr [5]> <chr [1]> <chr [0]>
 6 Owen Lars             178   120 brown, grey   light       blue            52   male   masculine Tatooine  Human   <chr [3]> <chr [0]> <chr [0]>
 7 Beru Whitesun lars    165    75 brown         light       blue            47   female feminine  Tatooine  Human   <chr [3]> <chr [0]> <chr [0]>
 8 R5-D4                  97    32 <NA>          white, red  red             NA   none   masculine Tatooine  Droid   <chr [1]> <chr [0]> <chr [0]>
 9 Biggs Darklighter     183    84 black         light       brown           24   male   masculine Tatooine  Human   <chr [1]> <chr [0]> <chr [1]>
10 Obi-Wan Kenobi        182    77 auburn, white fair        blue-gray       57   male   masculine Stewjon   Human   <chr [6]> <chr [1]> <chr [5]>
# ... with 77 more rows
>
```

Si volem saber quantes files i columnes tenim, fem:

```
>glimpse(starwars)
```

NOTA: En molts dels exercicis R ens tornarà un *warning*:

```
Warning message:
Removed 6 rows containing non-finite values (stat_density).
```

Veurem més endavant, quan veiem data massatge com treure aquestes files de forma que NO ens surtin *warnings*.

1.- Mostra la distribució de les alçades dels personatges i indica quina és la moda.

a) Quin tipus de gràfica és més evident de les que hem vist?

Height és una variable numèrica contínua. Vam veure que l'histograma era una bona manera de mostrar distribucions d'una variable contínua.

Si fem:

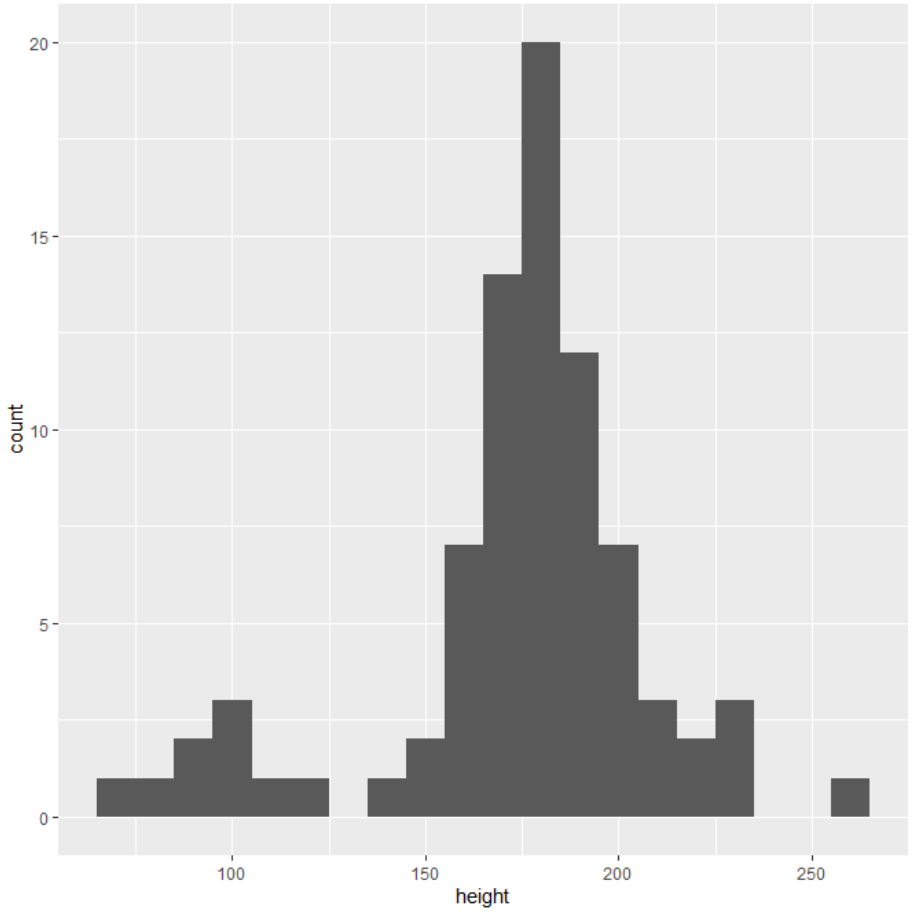
```
> ggplot(starwars,aes(x=height)) +geom_histogram()
```

Ens diu efectivament:

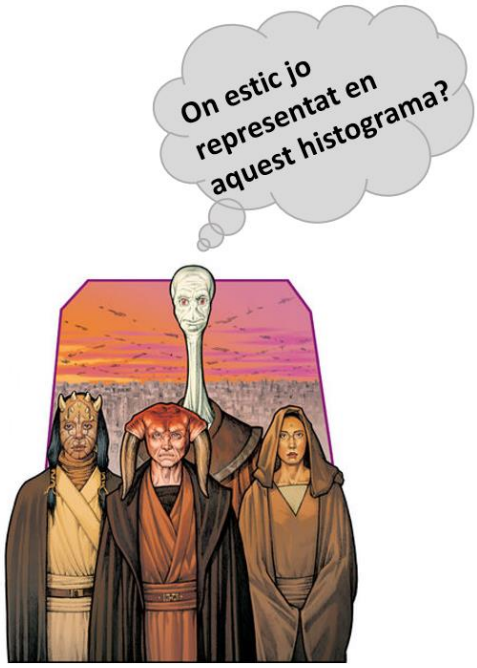
```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Provem amb `binwidth=10` per exemple (o altre que creguem adients):

```
> ggplot(starwars,aes(x=height))+geom_histogram(binwidth=10)
```

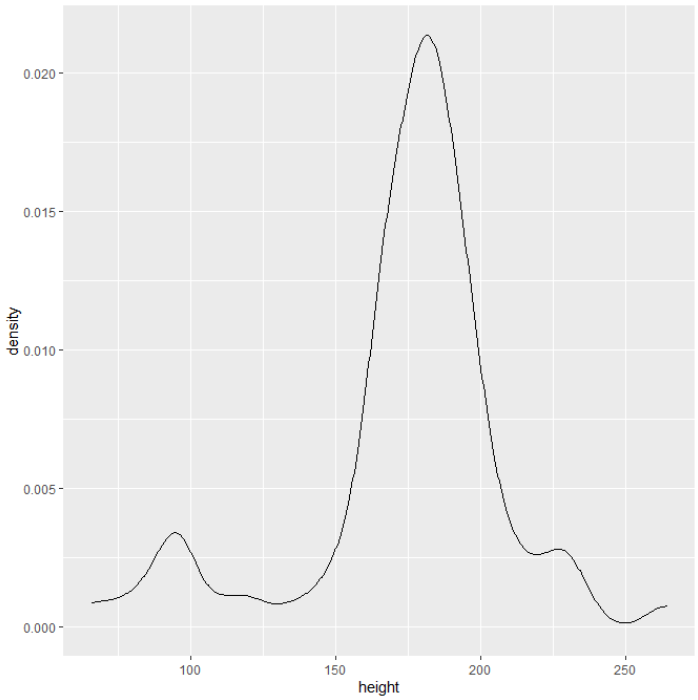


En quant a la moda, veiem que la majoria dels personatges tenen alçades entre 160-190 cm, podria ser degut a que en les primeres pel·lícules de Star Wars no hi havia tants efectes especials, i alguns personatges eren humans disfressats. Segur que podeu intuir en quina part de l'histograma estava algun personatge com en Yoda i/o algun altre personatge.



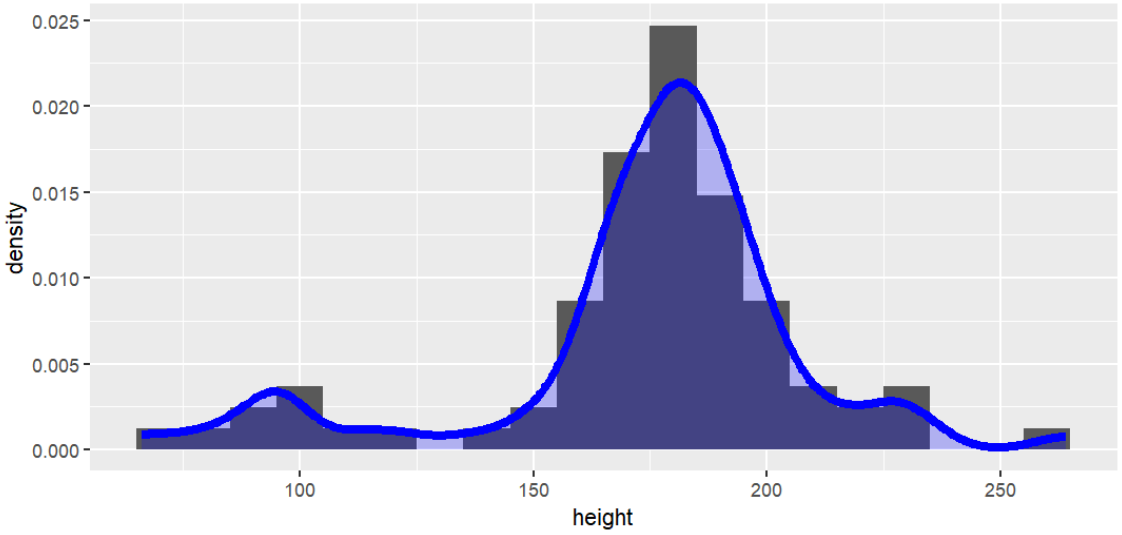
b) Ara proba la `geom_density()` i explica quina informació et dona

```
> ggplot(starwars,aes(x=height)) +geom_density()
Warning message:
Removed 6 rows containing non-finite values (stat_density).
```



Aquí semblaria que hi ha personatges amb alçades entre 230-240 cm mentre en l'histograma veiem que no n'hi ha.

c) Una altra opció és adjuntar ambdues gràfiques en una fent servir una transparència. Per això podeu posar **`aes(y=..density..)`** en el `geom_histogram`.



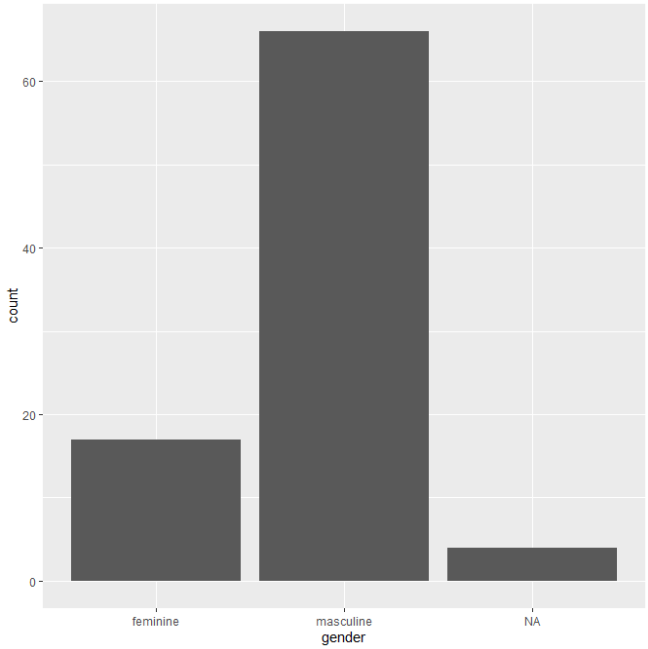

```
>ggplot(starwars,aes(x=height))+geom_histogram(binwidth=10, aes(y=..density..))+geom_density(lwd = 2, colour = 'blue', fill = 'blue', alpha = 0.25)
```

Nota: `lwd` només marca el gruix de la línia de `geom_density`

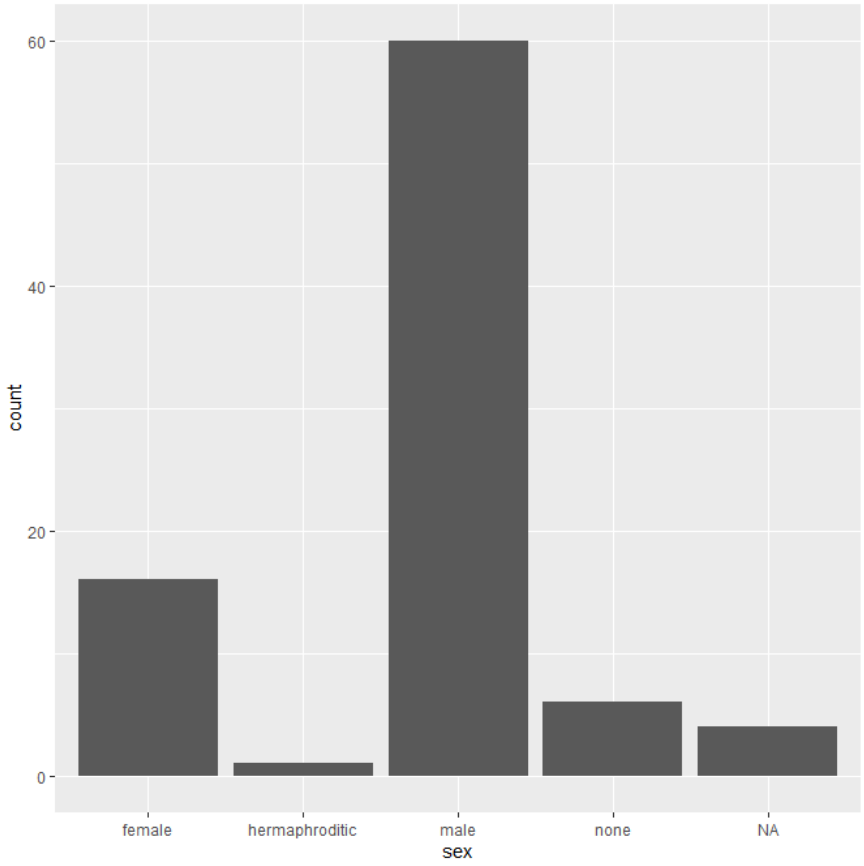
2.- Mostra la proporció dels diferents gèneres dels personatges de starwars

Gender és una variable discreta qualitativa. Volem comparar quants personatges hi ha de cada gènere, per tant sembla adient fer un diagrama de barres

```
>ggplot(starwars,aes(x=gender))+geom_bar()
```




```
>ggplot(starwars,aes(x=sex))+geom_bar()
```



Hi ha menys actors amb sexe masculí que amb gènere masculí. El gènere si feu `?starwars` veureu que es refereix al rol que adapten els personatges. Personatges com R2-D2 no tenen sexe però adapten un gènere masculí.

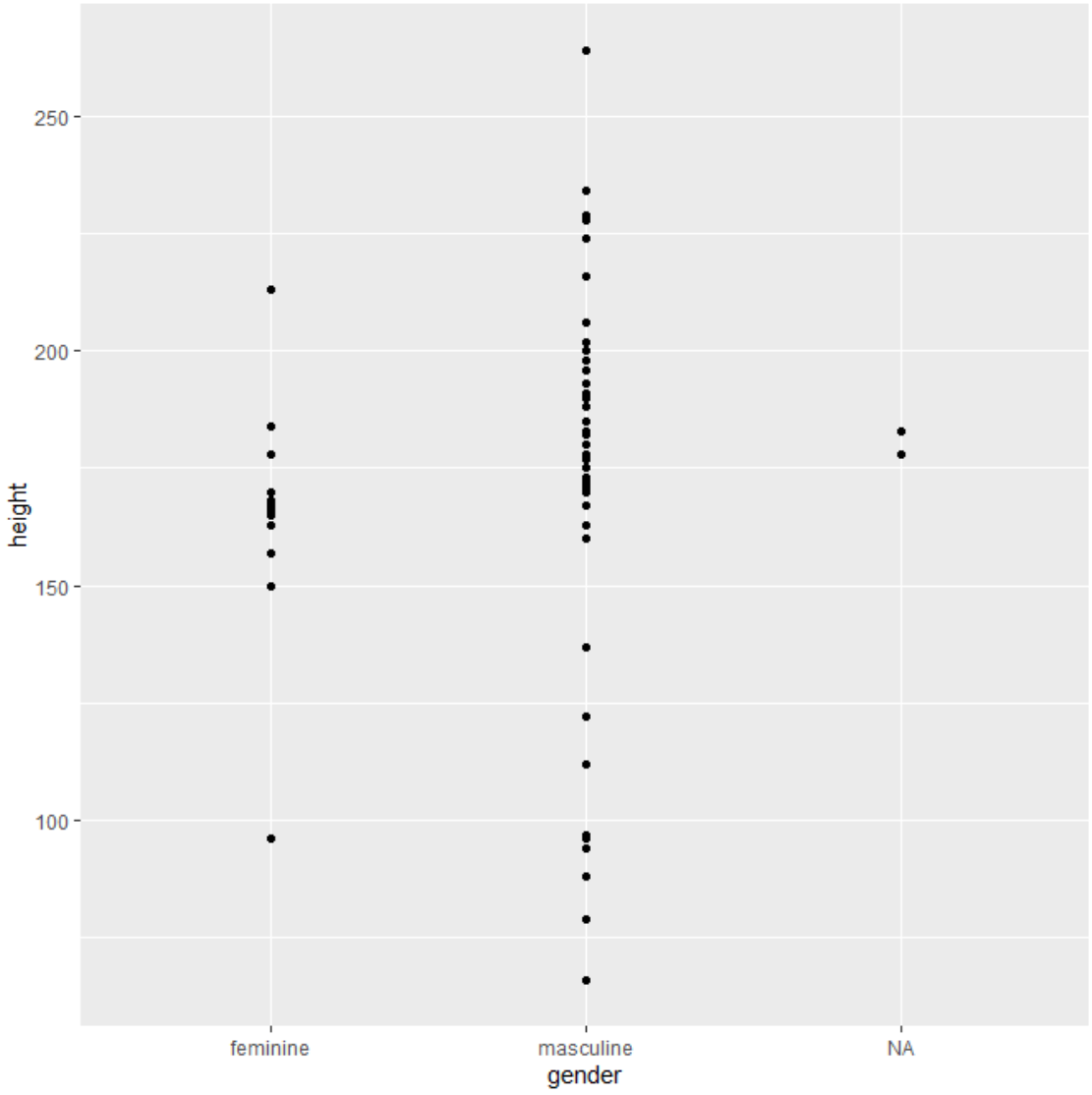
3.- Mostra la distribució de les alçades versus el gènere

a) Quin gènere té un alçada mitjana més alta?

b) De quin gènere és el personatge més alt?

`height` és una variable numèrica contínua. `gender` és una variable discreta-categòrica. Si fem un scatter plot, on l'eix x representi el gènere i l'eix y l'alçada, la visualització no ens dona gaire informació:

```
> ggplot(starwars,aes(x=gender,y=height))+geom_point()
Warning message:
Removed 6 rows containing missing values (geom_point).
```

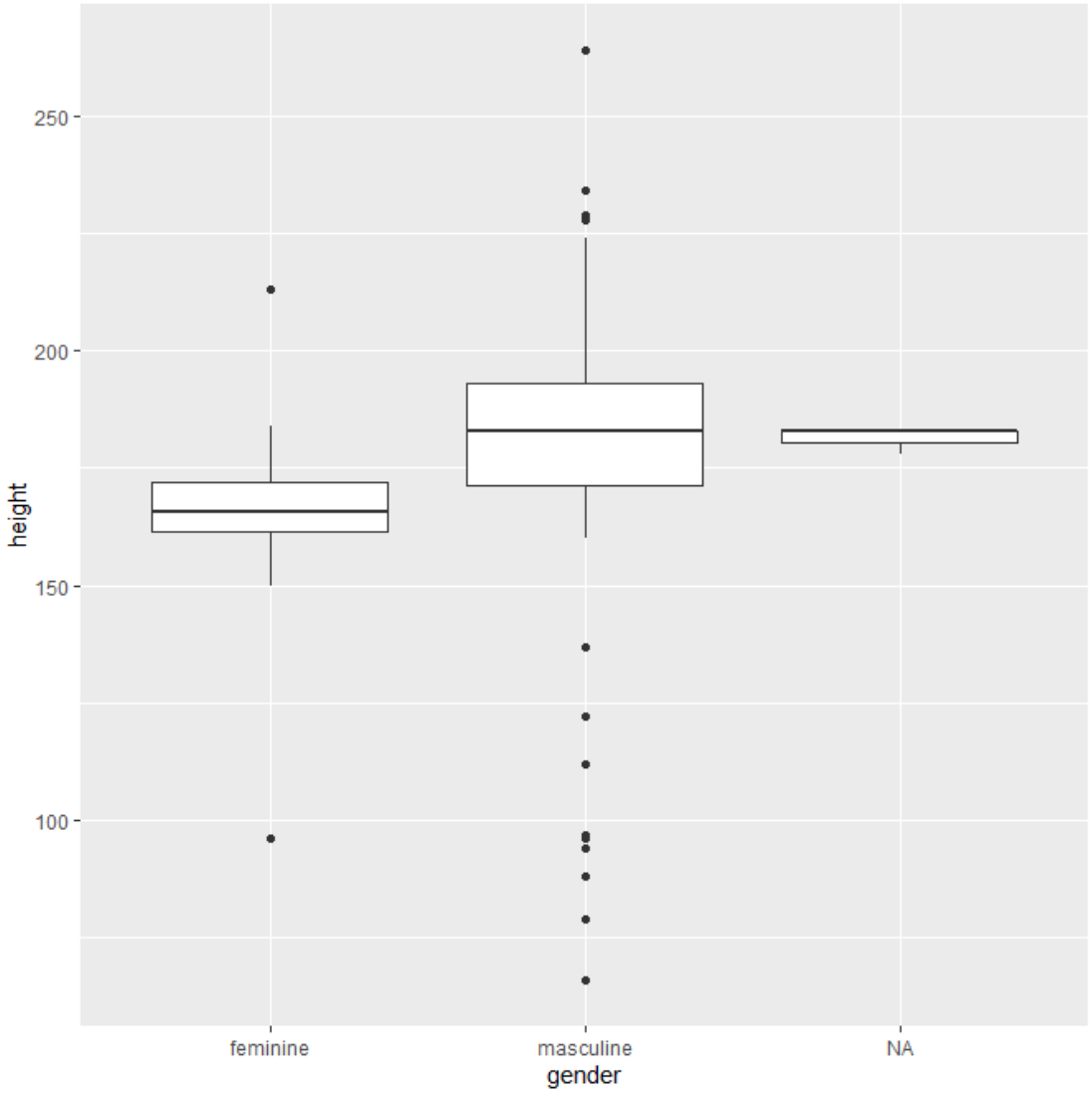


Si fem un scatter plot, utilitzant `geom_point()`, contestar la pregunta *b* és fàcil (també ho era pels seguidors de Star Wars o els que ja heu vist la imatge que us he posat en la resposta de l'exercici 1). El personatge més alt és de gènere masculí. Però contestant

l'apartat *a* ens equivocaríem segurament si no féssim un diagrama de caixes (boxplot). A més el boxplot ens dóna molta més informació sobre la distribució de les alçades versus el gènere.

Per tant:

```
> ggplot(starwars,aes(x=gender,y=height))+geom_boxplot()
Warning message:
Removed 6 rows containing non-finite values (stat_boxplot).
```



I curiosament la mitjana del gènere masculí i la de NA és la mateixa (cosa que no es veia en el *scatterplot*)

4.- Mostra la proporció dels diferents colors d'ulls dels personatges de starwars

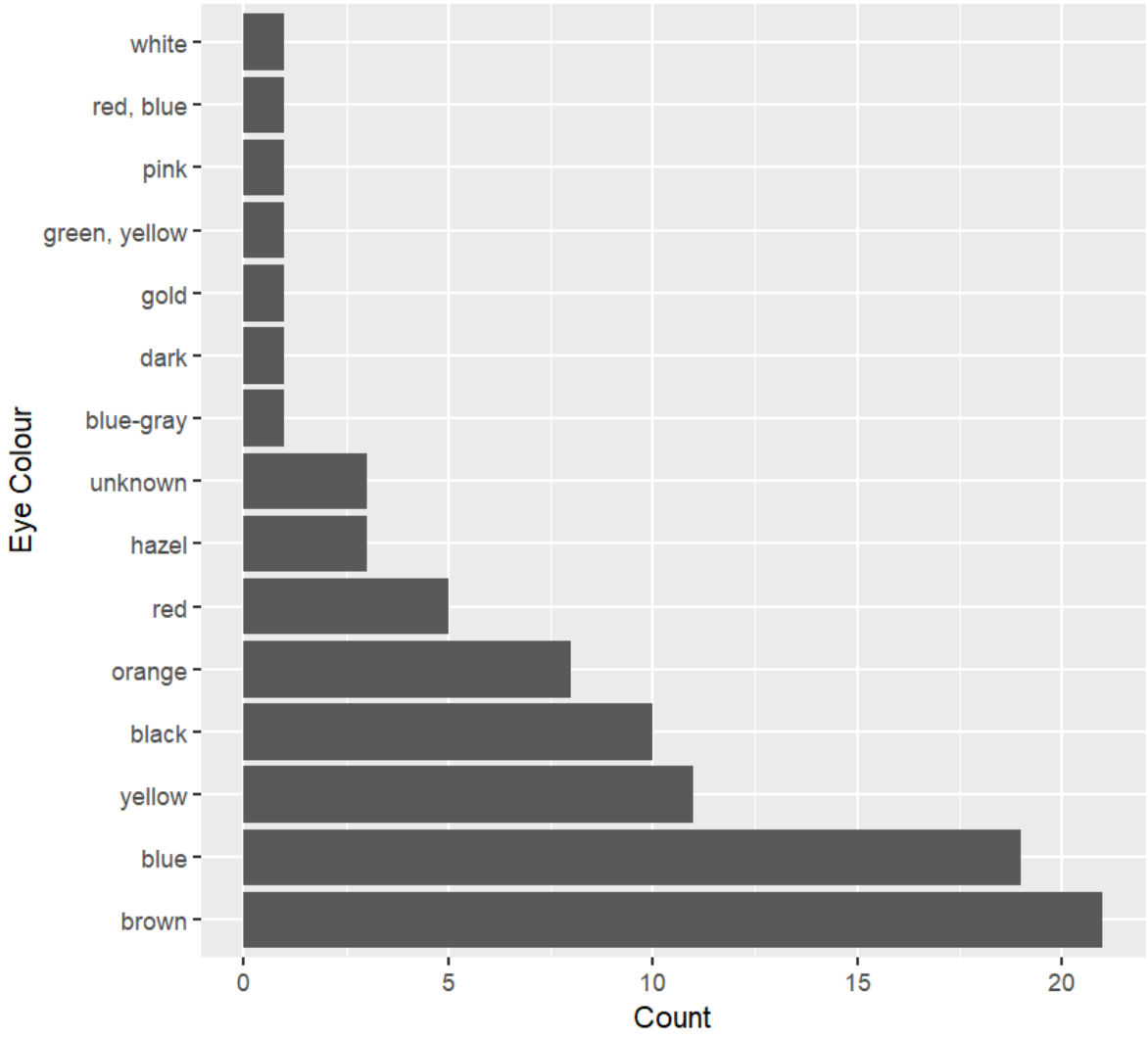
`eye_color` és una variable discreta. Volem comparar quants personatges hi ha de cada color d'ulls, per tant sembla adient fer un diagrama de barres. Ara bé si ho fem:

```
> ggplot(starwars, mapping = aes(x = eye_color)) + geom_bar()
```

Veure'm que se'ns amunteguen els noms de les variables de color d'ulls a l'eix x.

Sembla adient fer un `coord_flip()`, com heu vist en les diapositives, i reordenar el diagrama de barres per freqüències amb la llibreria forcats que heu vist avui. **NOTA:** És com l'exemple que heu vist a les diapositives del color de cabell (que també podeu refer).

```
> library(forcats) #si heu carregat tidyverse ja hi és, si sol heu carregat ggplot2 l'haurieu de carregar.
> ggplot(starwars, aes(x = fct_infreq(eye_color))) + geom_bar()+labs(x='Eye Colour', y='Count')+coord_flip()
```



Els dos colors majoritaris són el marró seguit del blau. Un color minoritari podria ser el rosa.

NOTA: Com sempre, aquest solucionari presenta una solució, no vol dir que per alguns casos, no hi hagi més solucions adequades.

Judit Chamorro Servent
Bellaterra, Març 2025