

# Visualización de datos en una pirámide poblacional

Miguel Herreros

2024-12-16

## Índice

Introducción	1
Contraste	2
Reordenando los grupos de edad.	3
Retocando Ejes	6
Accesibilidad	7
Daltonismo . . . . .	7
Dislexia . . . . .	7
Corrección de clase	9
Escala del eje x. . . . .	9
Añadir pie de gráfico. . . . .	10

## Introducción

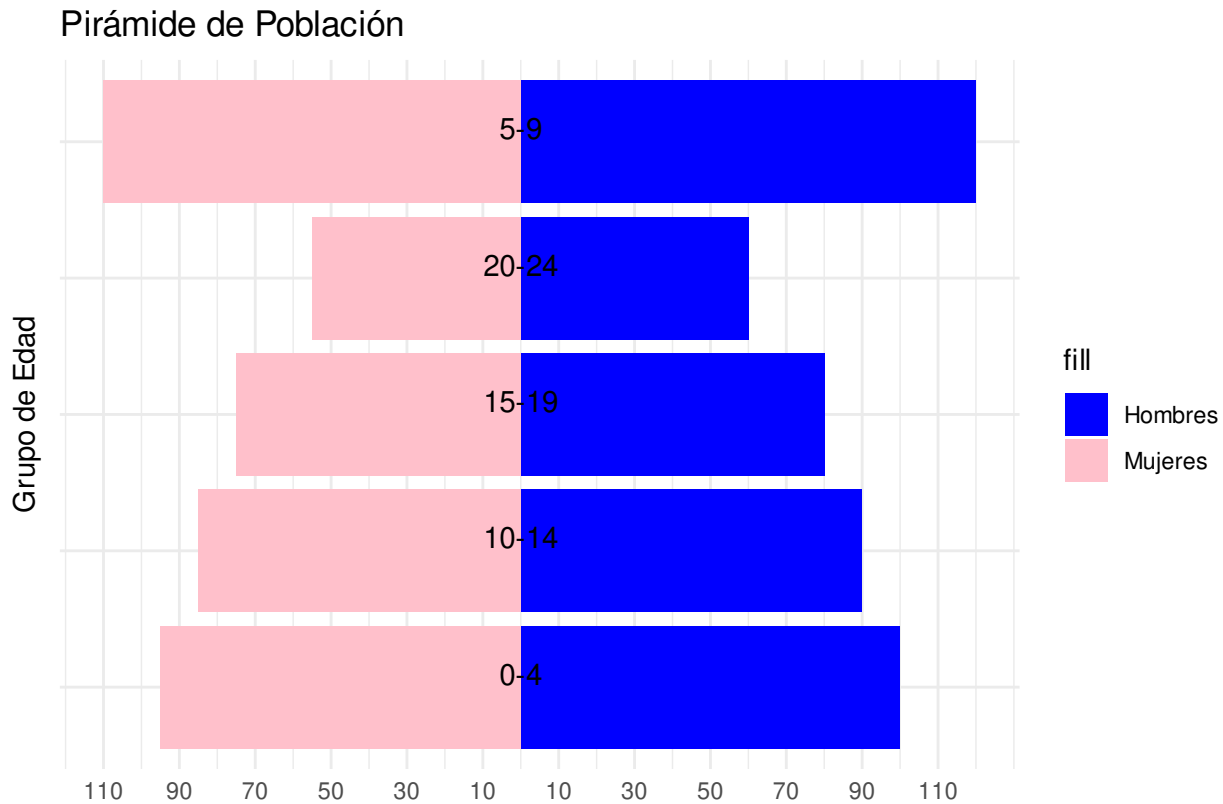
En esta práctica vamos a tratar la visualización de datos siguiendo como ejemplo la siguiente pirámide de población.

```
# Creación de un conjunto de datos de ejemplo
data <- data.table(
  Edad = c("0-4", "5-9", "10-14", "15-19", "20-24"),
  Hombres = c(100, 120, 90, 80, 60),
  Mujeres = c(95, 110, 85, 75, 55)
)

# Creación de la pirámide de población
p <- ggplot(data, aes(y = Edad)) +
  geom_bar(aes(x = Hombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -Mujeres, fill = "Mujeres"), stat = "identity",
    position = "identity") +
  scale_fill_manual(values = c("Hombres" = "blue", "Mujeres" = "pink")) +
  labs(title = "Pirámide de Población", x = "", y = "Grupo de Edad") +
  theme_minimal() +
  scale_x_continuous(breaks = seq(-max(data$Mujeres), max(data$Hombres), by = 20),
    labels = abs(seq(-max(data$Mujeres), max(data$Hombres), by = 20))) +
  geom_text(aes(y = Edad, x = 0, label = Edad), hjust = 0.5, vjust = 0) +
```

```
theme(axis.text.y = element_blank(), axis.ticks.y = element_blank())
```

p



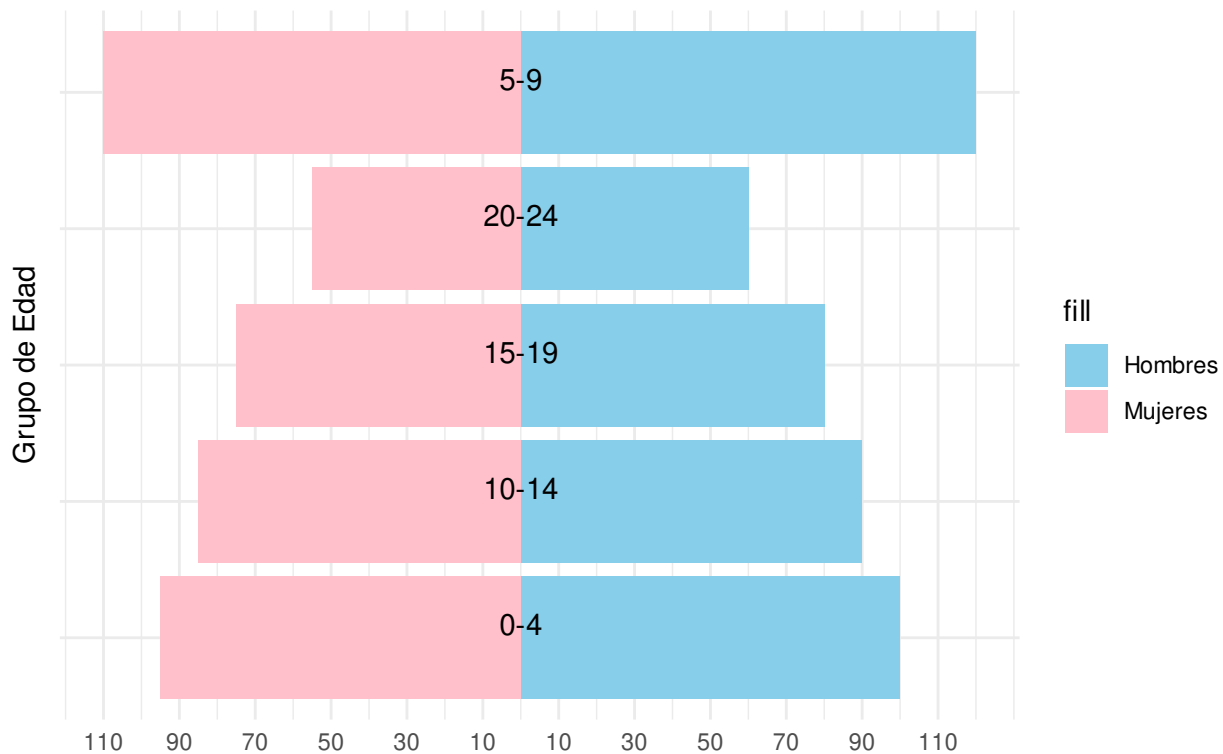
## Contraste

A primera vista lo que más me llama la atención es el poco contraste que hay entre el azul de las barras de la derecha y el negro de las etiquetas, dificultando la lectura de las mismas.

```
p <- ggplot(data, aes(y = Edad)) +
  geom_bar(aes(x = Hombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -Mujeres, fill = "Mujeres"), stat = "identity",
    position = "identity") +
  scale_fill_manual(values = c("Hombres" = "skyblue", "Mujeres" = "pink")) +
  labs(title = "Pirámide de Población", x = "", y = "Grupo de Edad") +
  theme_minimal() +
  scale_x_continuous(breaks = seq(-max(data$Mujeres), max(data$Hombres), by = 20),
    labels = abs(seq(-max(data$Mujeres), max(data$Hombres),
      by = 20))) +
  geom_text(aes(y = Edad, x = 0, label = Edad), hjust = 0.5, vjust = 0) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank())
```

p

## Pirámide de Población



## Reordenando los grupos de edad.

Tal y como está en el `data.table`, los grupos de edad son cadenas y, por tanto, las ordena lexicográficamente. Sin embargo eso no tiene mucho sentido pues el grupo de 5 – 9 años no debería ser el mayor. El paquete `stringr` nos permite crear una columna con el orden de las cadenas pero ordenadas numéricamente en lugar de lexicográficamente.

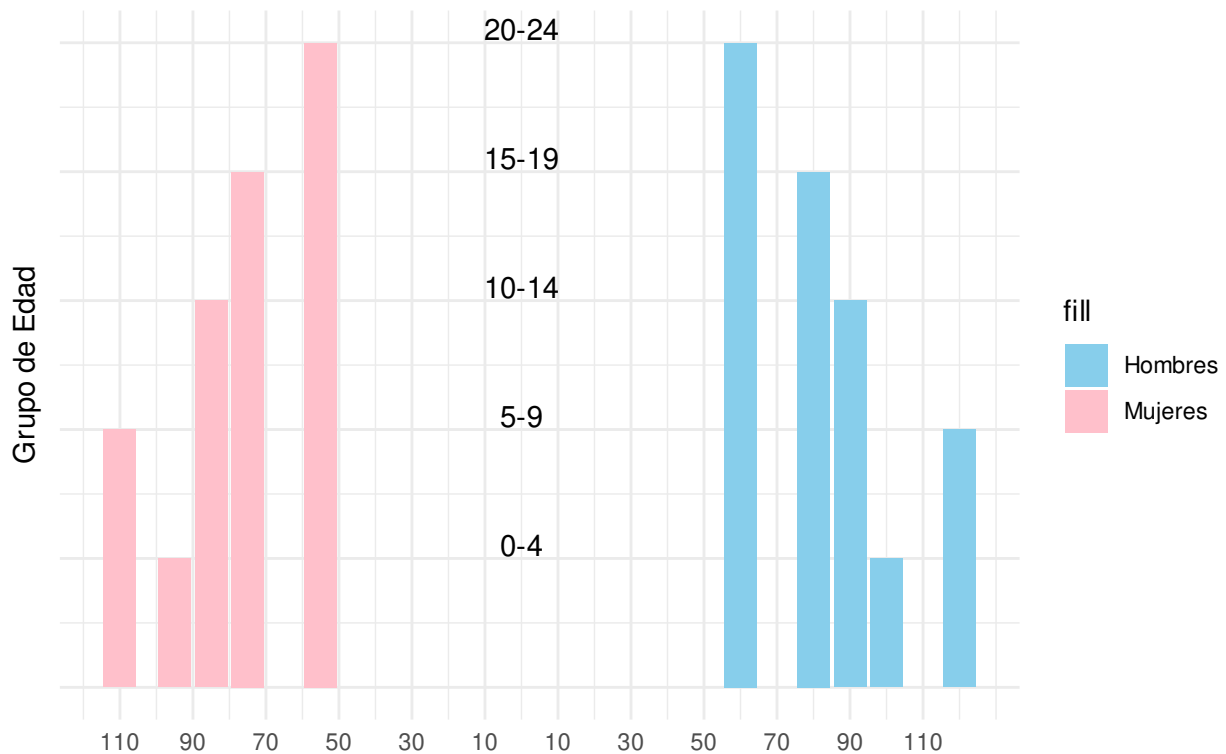
```
data[, order:= str_order(x = Edad, numeric = TRUE)]
```

Con esta nueva columna podemos ordenar correctamente los grupos de edad.

```
p <- ggplot(data, aes(y = order)) +
  geom_bar(aes(x = Hombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -Mujeres, fill = "Mujeres"), stat = "identity",
    position = "identity") +
  scale_fill_manual(values = c("Hombres" = "skyblue", "Mujeres" = "pink")) +
  labs(title = "Pirámide de Población", x = "", y = "Grupo de Edad") +
  theme_minimal() +
  scale_x_continuous(breaks = seq(-max(data$Mujeres), max(data$Hombres), by = 20),
    labels = abs(seq(-max(data$Mujeres), max(data$Hombres),
      by = 20))) +
  geom_text(aes(y = order, x = 0, label = Edad), hjust = 0.5, vjust = 0) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank())
```

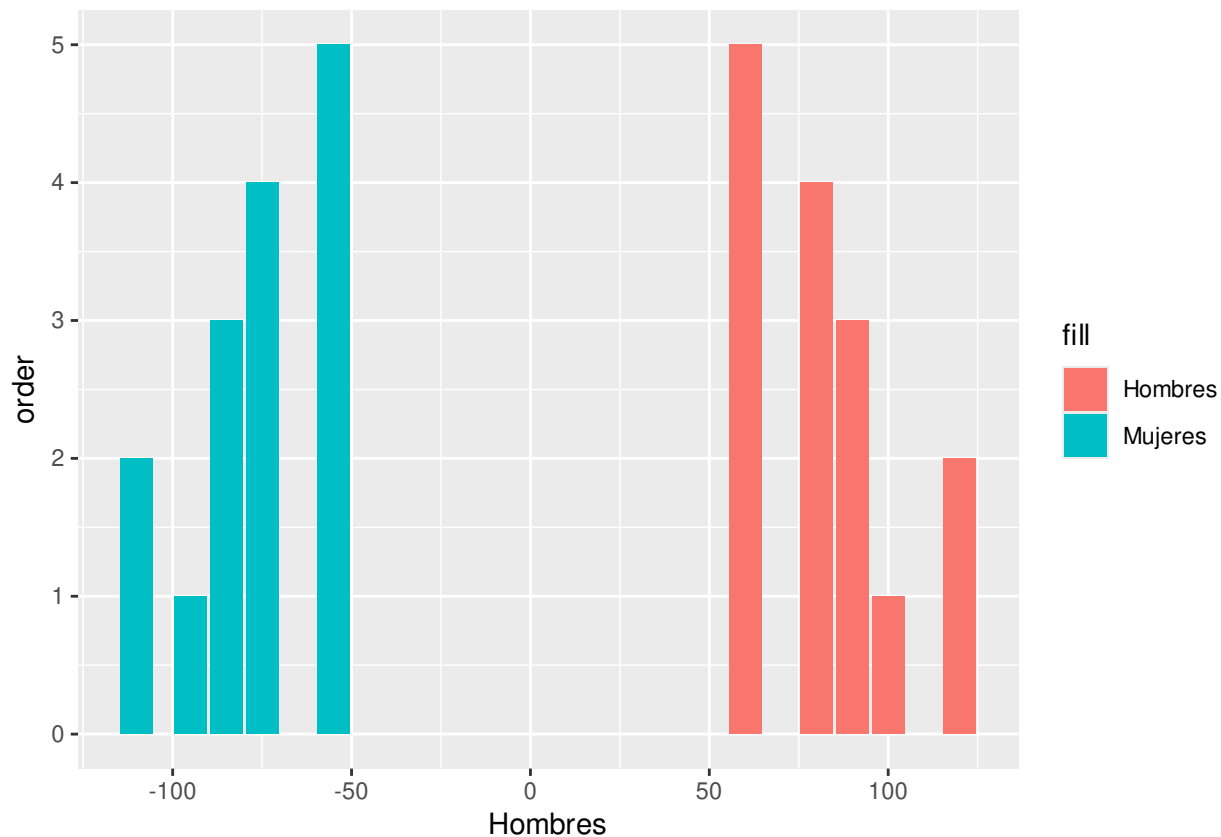
p

## Pirámide de Población



Como podemos ver, ggplot no está terminando de interpretar bien lo que queremos mostrar. Si quitamos las modificaciones que hemos hecho al gráfico y dejamos los valores por defecto se puede ver bien qué está sucediendo.

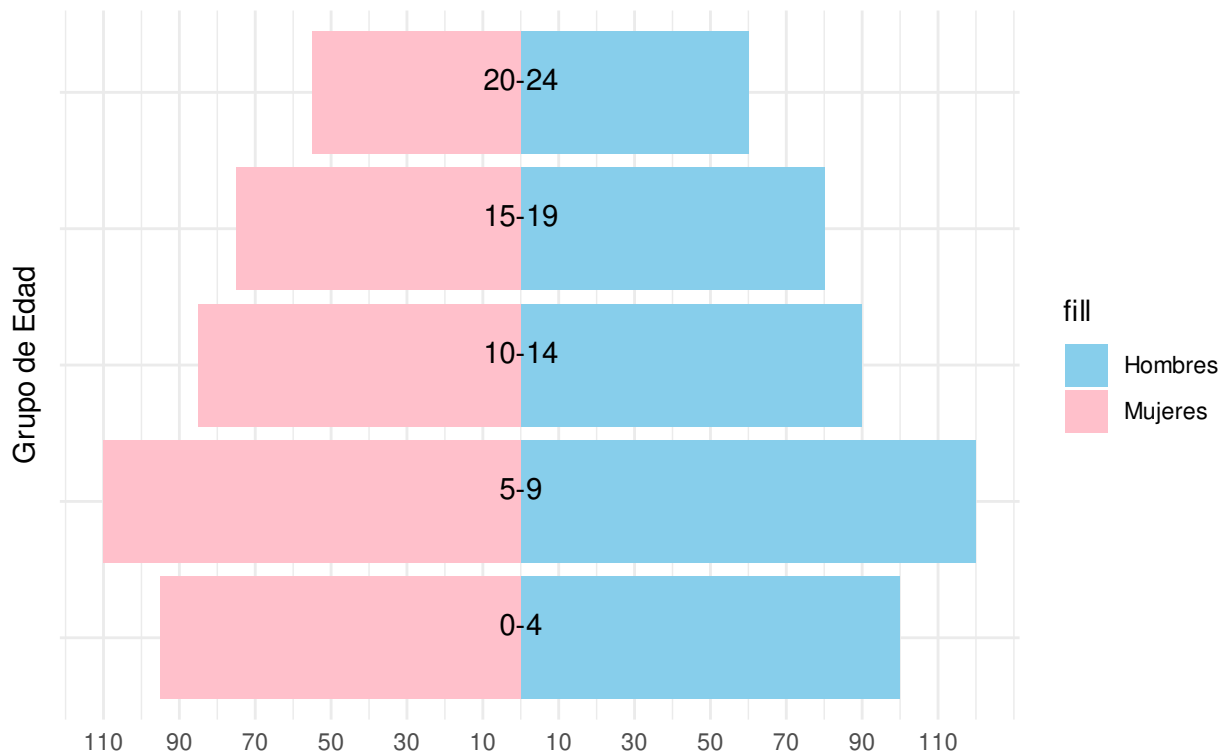
```
p <- ggplot(data, aes(y = order)) +
  geom_bar(aes(x = Hombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -Mujeres, fill = "Mujeres"), stat = "identity",
    position = "identity")
p
```



Como podemos ver, al ser `order` una variable numérica nos está haciendo un histograma al uso, con las barras subiendo desde el eje x hasta el valor del “orden” en el eje y. Esto se soluciona fácilmente si simplemente especificamos a R que la variable `order` es categórica.

```
data[, order:= as.factor(order)]
p <- ggplot(data, aes(y = order)) +
  geom_bar(aes(x = Hombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -Mujeres, fill = "Mujeres"), stat = "identity",
    position = "identity") +
  scale_fill_manual(values = c("Hombres" = "skyblue", "Mujeres" = "pink")) +
  labs(title = "Pirámide de Población", x = "", y = "Grupo de Edad") +
  theme_minimal() +
  scale_x_continuous(breaks = seq(-max(data$Mujeres), max(data$Hombres), by = 20),
    labels = abs(seq(-max(data$Mujeres), max(data$Hombres),
      by = 20))) +
  geom_text(aes(y = order, x = 0, label = Edad), hjust = 0.5, vjust = 0) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank())
p
```

## Pirámide de Población



## Retocando Ejes

El rango de las edades no hace falta tenerlo en el gráfico, es muy fácil de ver si las etiquetas están en el eje y, dejando así espacio en el gráfico para etiquetar cada grupo con otro tipo de información que pueda ser útil. Por ejemplo, el porcentaje de personas de cada género en cada grupo.

Además, también se ha añadido un título descriptivo a la leyenda y al eje x y se ha modificado el título del gráfico.

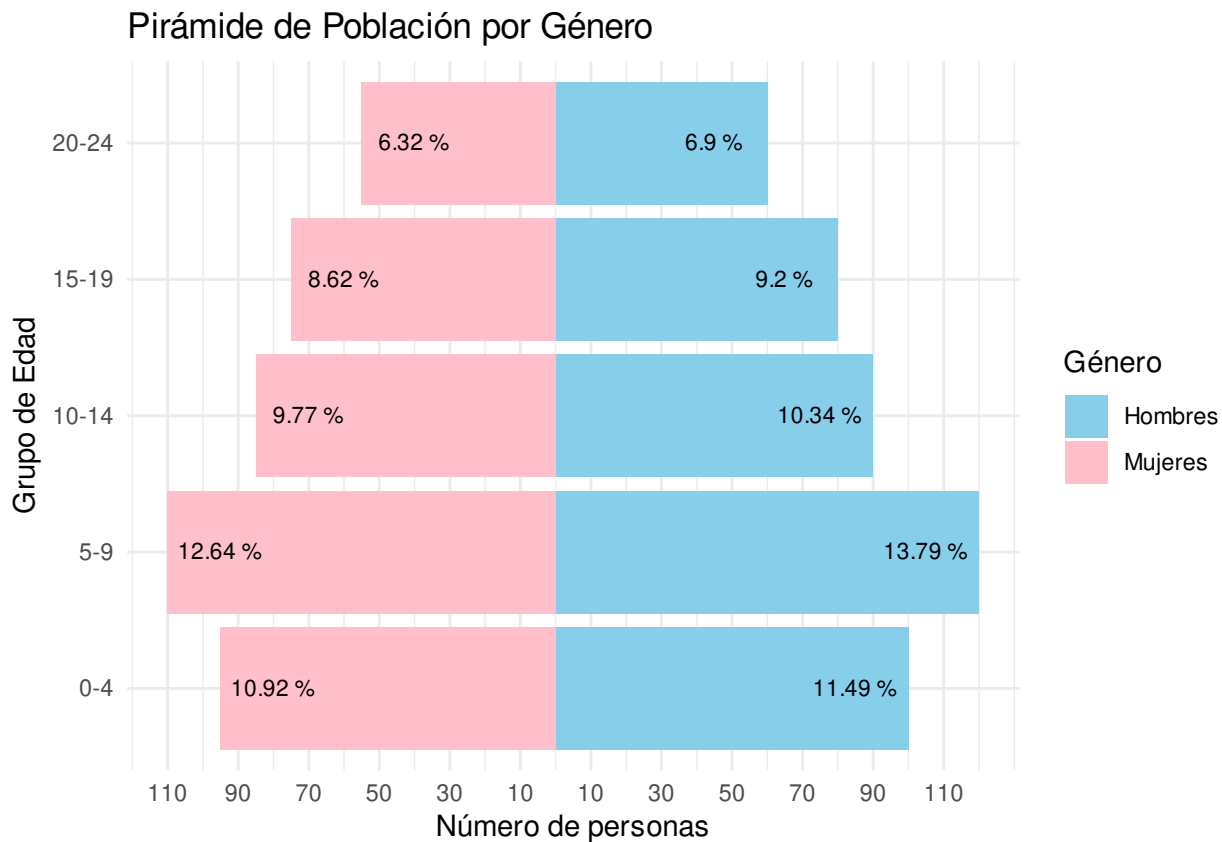
```
total_population = sum(data[, Hombres]) + sum(data[, Mujeres])
data[, ratiosHombres := Hombres / total_population]
data[, ratiosMujeres := Mujeres / total_population]
data[, percentHombres :=
  paste(as.character(round(ratiosHombres*100, digits = 2)), "%")]
data[, percentMujeres :=
  paste(as.character(round(ratiosMujeres*100, digits = 2)), "%")]
p <- ggplot(data, aes(y = order)) +
  geom_bar(aes(x = Hombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -Mujeres, fill = "Mujeres"), stat = "identity",
    position = "identity") +
  scale_fill_manual(values = c("Hombres" = "skyblue", "Mujeres" = "pink"),
    name="Género") +
  labs(title = "Pirámide de Población por Género", x = "Número de personas",
    y = "Grupo de Edad") +
  theme_minimal() +
  scale_x_continuous(breaks = seq(-max(data$Mujeres), max(data$Hombres), by = 20),
    labels = abs(seq(-max(data$Mujeres), max(data$Hombres),
```

```

    by = 20))) +
scale_y_discrete(breaks = data$order, labels = data$Edad) +
geom_text(aes(y = order, x = Hombres -15, label = percentHombres),
    hjust = 0.5, vjust = 0.5, size = 3) +
geom_text(aes(y = order, x = -Mujeres + 15, label = percentMujeres),
    hjust = 0.5, vjust = 0.5, size = 3)

```

p



## Accesibilidad

Por último se va a estudiar el modificar los colores del gráfico y el tipo de letra, con el objetivo de hacer más accesible el gráfico.

### Daltonismo

Resulta que, tras comprobar diversas paletas de colores en Coblis, la paleta de colores escogida ya es muy robusta ante los ocho tipos de daltonismo que te permite comprobar la web. Luego en este aspecto no hace falta cambiar nada.

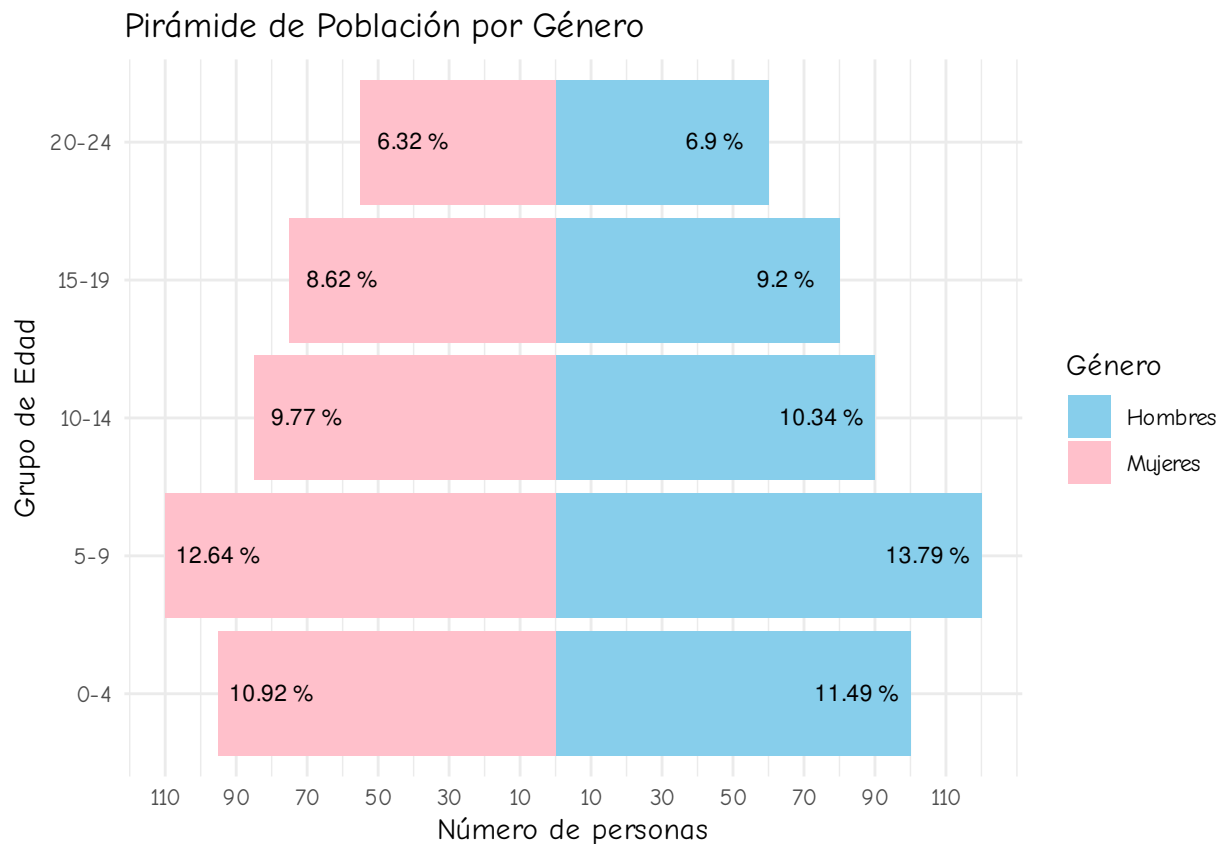
### Dislexia

Para que una persona con dislexia tenga las menores dificultades posibles para leer correctamente un texto se suelen exigir tres condiciones. Que no tengan serifas, que el texto sea monoespaciado y que no sea un tipo de letra cursiva u oblicua. Pese a la mala fama que suele tener, Comic Sans cumple las tres condiciones anteriores y es de las más legibles para personas con dislexia. Sin embargo, al formar parte de los tipos de

letra propietarios de Microsoft, usaremos Comic Neue, una fuente inspirada en Comic Sans y con las mismas virtudes pero que busca ser menos “infantil” que la primera. Además de estar disponible en Linux.

```
p <- ggplot(data, aes(y = order)) +
  geom_bar(aes(x = Hombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -Mujeres, fill = "Mujeres"), stat = "identity",
    position = "identity") +
  scale_fill_manual(values = c("Hombres" = "skyblue", "Mujeres" = "pink"),
    name="Género") +
  labs(title = "Pirámide de Población por Género", x = "Número de personas",
    y = "Grupo de Edad") +
  theme_minimal() +
  theme(text = element_text(family = "Comic Neue")) +
  scale_x_continuous(breaks = seq(-max(data$Mujeres), max(data$Hombres), by = 20),
    labels = abs(seq(-max(data$Mujeres), max(data$Hombres),
      by = 20))) +
  scale_y_discrete(breaks = data$order, labels = data$Edad) +
  geom_text(aes(y = order, x = Hombres -15, label = percentHombres),
    hjust = 0.5, vjust = 0.5, size = 3) +
  geom_text(aes(y = order, x = -Mujeres + 15, label = percentMujeres),
    hjust = 0.5, vjust = 0.5, size = 3)
```

p





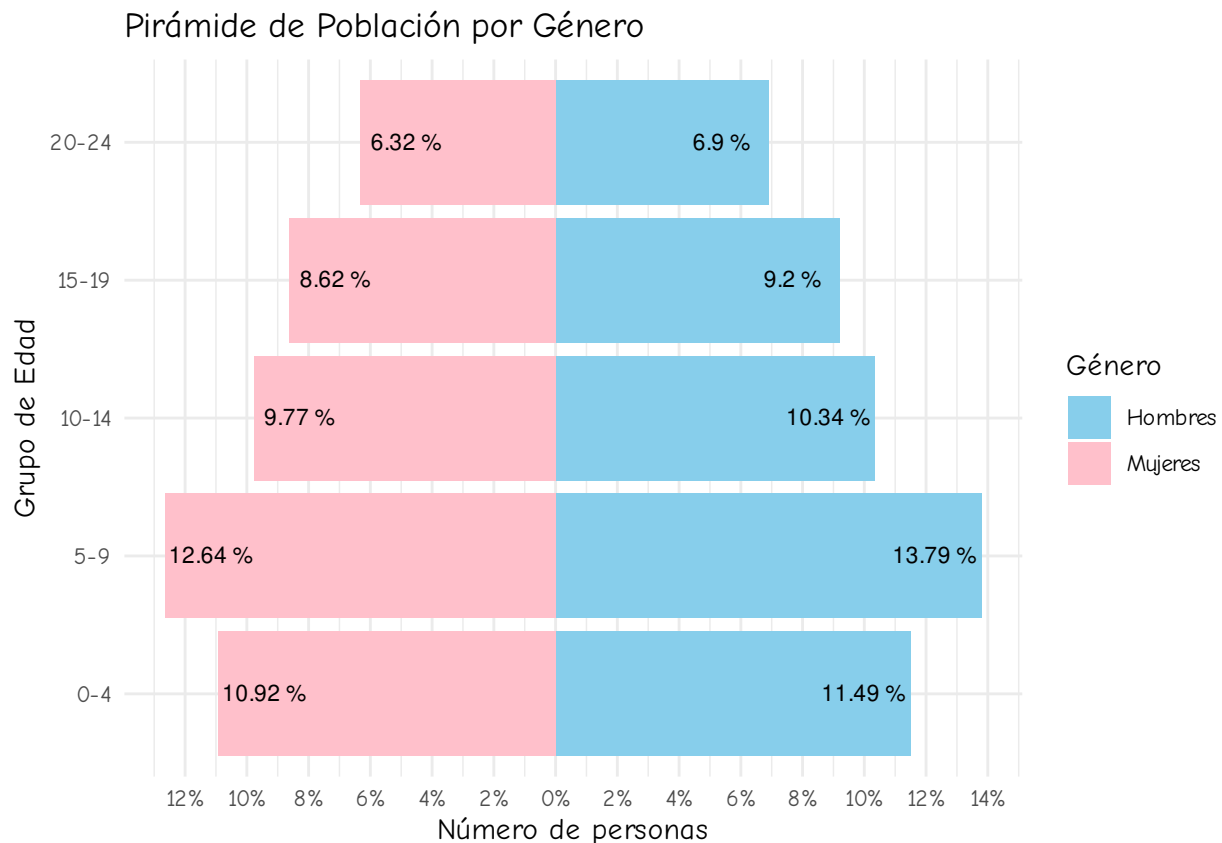
## Corrección de clase

### Escalas del eje x.

Tal y como se ha comentado en clase, la idea de representar a la vez el número absoluto de personas y su proporción puede llegar a ser confuso. De hecho, es una estrategia habitualmente usada para distorsionar la realidad. Para evitar confusiones, cambiamos el eje x.

```
rango = seq(-14,14, by = 2)
labelsRango = paste(abs(rango), "%", sep = "")
p <- ggplot(data, aes(y = order)) +
  geom_bar(aes(x = 100*ratiosHombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -100*ratiosMujeres, fill = "Mujeres"), stat = "identity",
    position = "identity") +
  scale_fill_manual(values = c("Hombres" = "skyblue", "Mujeres" = "pink"),
    name="Género") +
  labs(title = "Pirámide de Población por Género", x = "Número de personas",
    y = "Grupo de Edad") +
  theme_minimal() +
  theme(text = element_text(family = "Comic Neue")) +
  scale_x_continuous(breaks = rango,
    labels = labelsRango) +
  scale_y_discrete(breaks = data$order, labels = data$Edad) +
  geom_text(aes(y = order, x = 100*ratiosHombres-1.5, label = percentHombres),
    hjust = 0.5, vjust = 0.5, size = 3) +
  geom_text(aes(y = order, x = -100*ratiosMujeres+1.5, label = percentMujeres),
    hjust = 0.5, vjust = 0.5, size = 3)
```

p



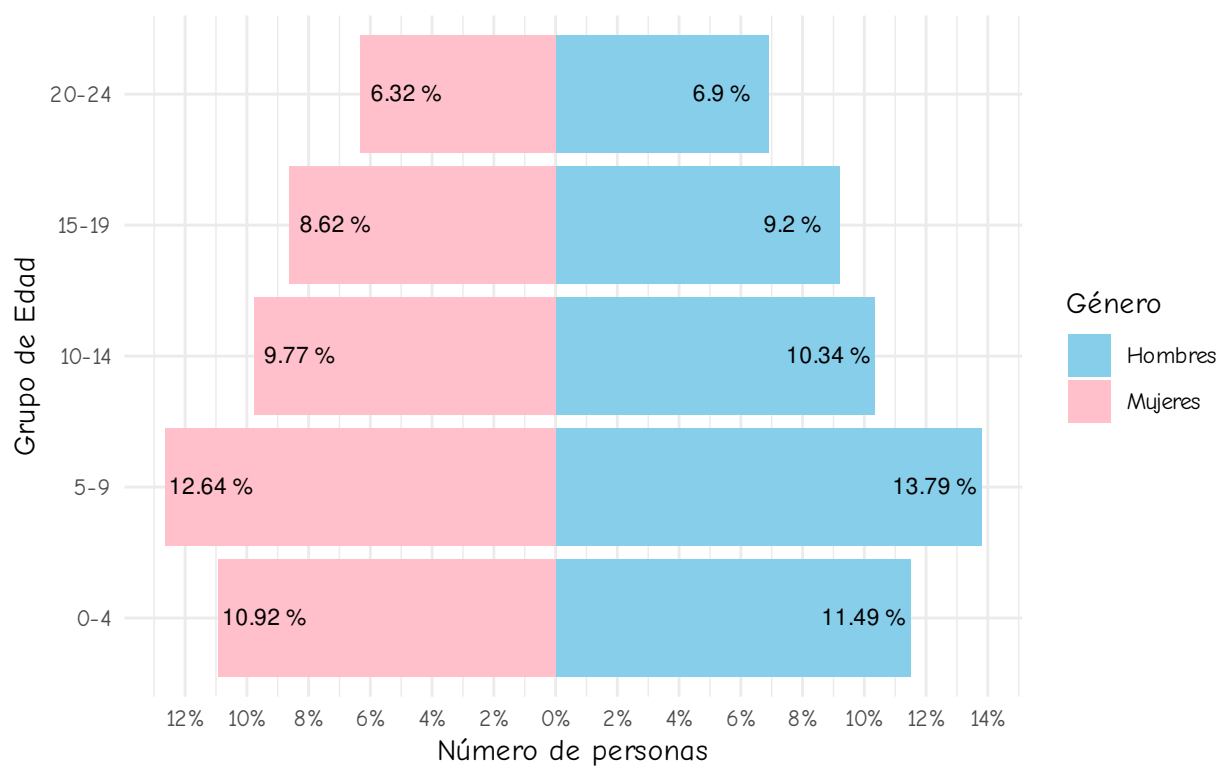
### Añadir pie de gráfico.

Además también se ha comentado la necesidad de, en un caso real, mostrar la fuente de los datos.

```
rango = seq(-14,14, by = 2)
labelsRango = paste(abs(rango), "%", sep = "")
p <- ggplot(data, aes(y = order)) +
  geom_bar(aes(x = 100*ratiosHombres, fill = "Hombres"), stat = "identity",
    position = "identity") +
  geom_bar(aes(x = -100*ratiosMujeres, fill = "Mujeres"), stat = "identity",
    position = "identity") +
  scale_fill_manual(values = c("Hombres" = "skyblue", "Mujeres" = "pink"),
    name="Género") +
  labs(title = "Pirámide de Población por Género", x = "Número de personas",
    y = "Grupo de Edad", caption = "Fuente: Datos de ejemplo para el trabajo.") +
  theme_minimal() +
  theme(text = element_text(family = "Comic Neue")) +
  scale_x_continuous(breaks = rango,
    labels = labelsRango) +
  scale_y_discrete(breaks = data$order, labels = data$Edad) +
  geom_text(aes(y = order, x = 100*ratiosHombres-1.5, label = percentHombres),
    hjust = 0.5, vjust = 0.5, size = 3) +
  geom_text(aes(y = order, x = -100*ratiosMujeres+1.5, label = percentMujeres),
    hjust = 0.5, vjust = 0.5, size = 3)
```

p

## Pirámide de Población por Género



Fuente: Datos de ejemplo para el trabajo.