

Graficando 2

David H. Duncan

January 20, 2016

Hola, bienvenido a esta lección. En esta vamos a desglosar nuestro tramo de datos del número de hijos y hijas con diagramas de barras y otros tipos aliados.

Dejamos la última lección después de haber resumido nuestra variable del número de hijos y hijas que tuvo una mujer ecuatoriana de promedio en 2010 por dos tipos de gráfica, o dos y media si cuenta con el stripchart. Estas resúmenes dejaron las provincias anónimas, y de esta manera pudimos enfocar en la variación en los promedios del número de hijos.

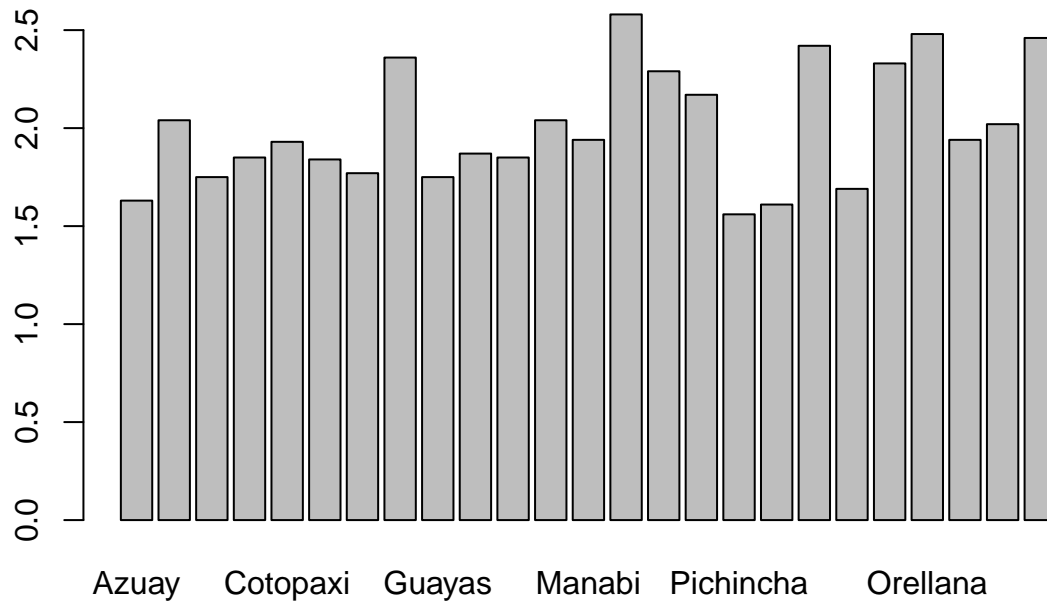
Para preparar los datos necesarios, copie y pegue este código en la consola:

```
.pathToFile <- function(course_, lesson_, file_){
  if(as.character(packageVersion("swirl")) > "2.2.21"){
    file.path(get_swirl_option("courses_dir"), course_, lesson_, file_)
  } else {
    file.path(find.package("swirl"), "Courses", course_, lesson_, file_)
  }
}

# Make path to xlsx available to user
path2csv1 <- file.path(path.package('swirl'), 'Courses',
                        'ConoceR',
                        'Graficando_1',
                        'hijosResumen.csv')
```

Para empezar, podemos realizar una gráfica tipo diagrama de barras fácilmente por ingresar nuestro objeto resumenhijos como el argumento 'height' a la función `barplot()`. Después de ingresar hijos como 'height', se puede especificar el argumento 'region'. Acuérdesse que siempre hay que separar argumentos de funciones con una coma. Dese la oportunidad de crear el barplot que le comenté arriba.

```
barplot(height = resumen$hijos, names.arg = resumen$region)
```



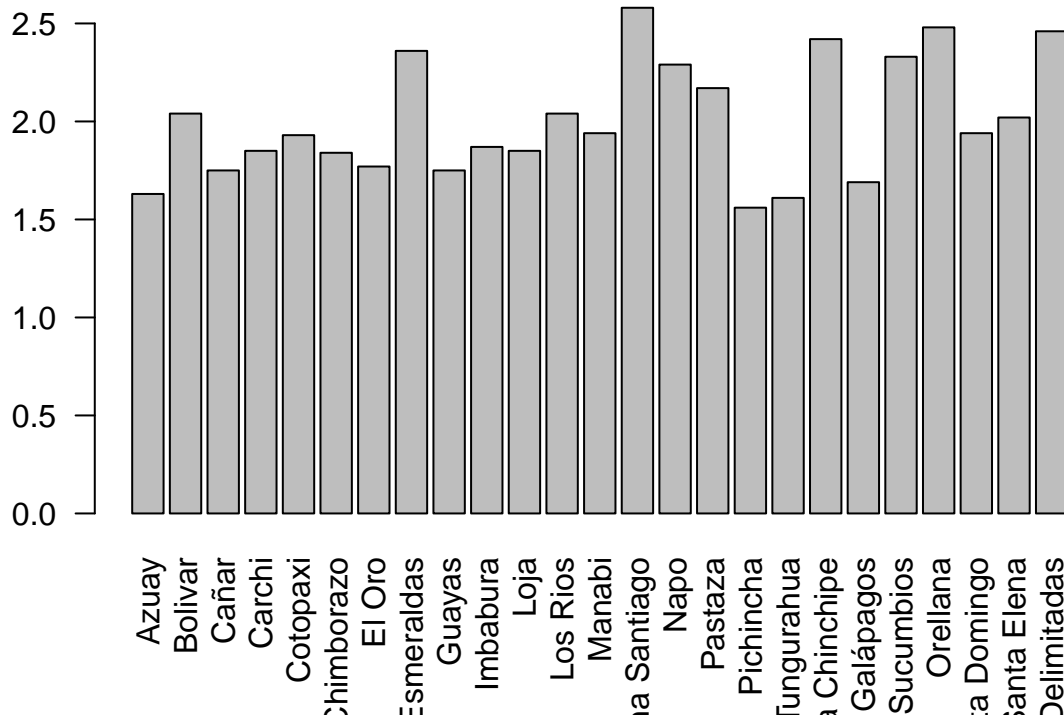
Hmmm, esta gráfica no se agrada mucho, no se puede leer muchos de los nombres de las provincias. ¿Se acuerda Usted como se llama el argumento para girar las etiquetas o valores de los ejes? Es algo irónico que con este tipo de gráfica los números del eje Y salieron bueno por defecto mientras que los nombres en el eje X no. Bueno, ¿cómo se llama el argumento que necesitamos entonces? Escriba las letras no más, sin o el símbolo = o un valor. ¡Intente contestar!

las

Le comento como funciona este argumento 'las'. Tome nota en el álbum de recortes para su propia referencia. Acepta cuatro valores posibles {0, 1, 2, 3}. Por defecto el ajuste es en 0, lo que hace que las etiquetas de los ejes corren en paralelo con los ejes. El argumento las = 1 hace que las etiquetas salen en el plano horizontal para los dos ejes, como usted vió en lecciones anteriores. En cambio las = 2 arregla que las etiquetas sean perpendiculares a los ejes y por último las = 3 arreglaría las etiquetas fueran todas en el plano vertical.

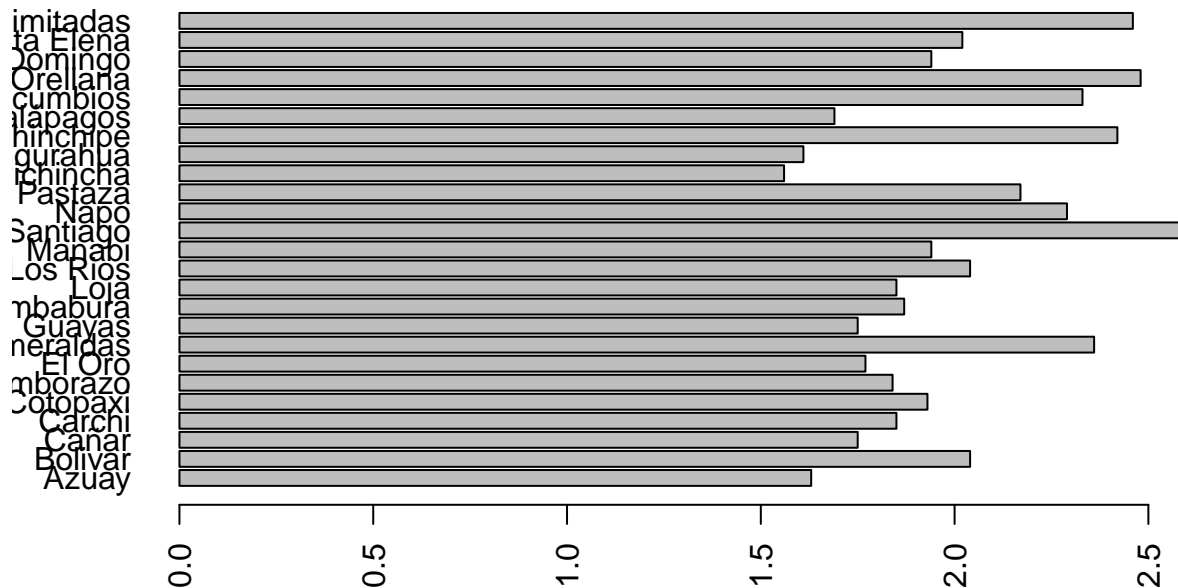
¿Entendido? A ver, recupere usted el comando para el barplot y añada un argumento con las para girar las etiquetas. Queremos que las etiquetas del eje Y no cambien, pero las de el eje X si giraran. Ponga un argumento con las que cree vaya a realizar este cambio.

```
barplot(resumen$hijos, names.arg = resumen$region, las=2)
```



Pues, mejor, pero con etiquetas largas como estos nombres de las provincias suele ser mejor girar la gráfica misma. Recupere la última llamada y añadir el argumento `horiz = TRUE`, para que las barras salgan en horizontal.

```
barplot(resumen$ hijos, names.arg = resumen$region, las=2, horiz = TRUE)
```



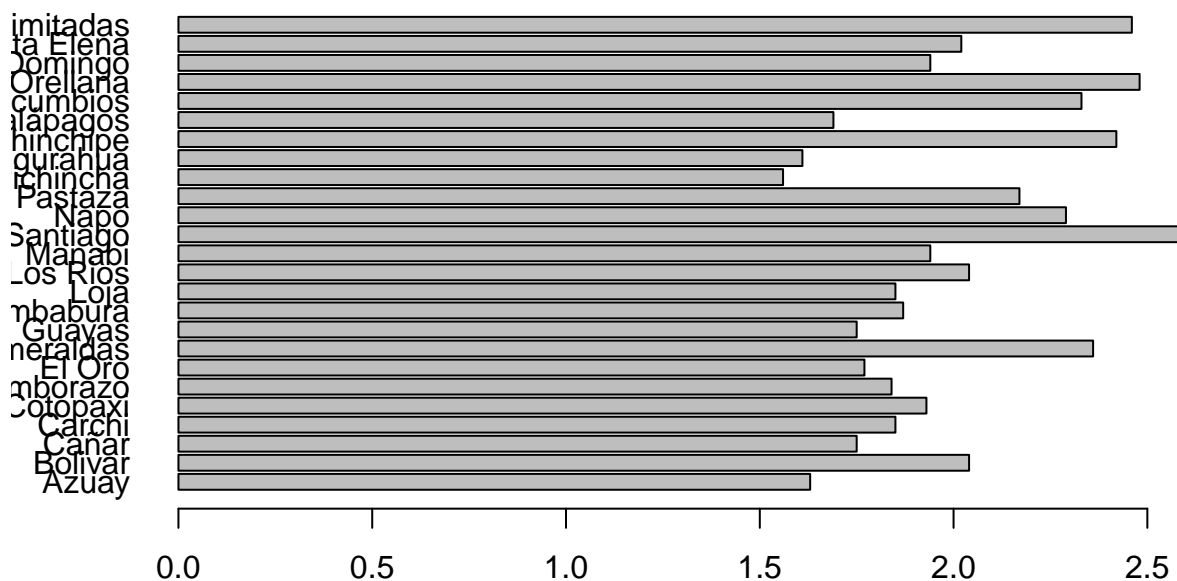
Eso tampoco funcionó, ¿verdad? Con etiquetas tan largas hay que hacer algo más drástico. Vamos a ajustar a las márgenes del espacio de la gráfica por teclear el siguiente comando `par(mar=c(4, 10, 4, 2))` en el indicador, aparte de la llamada a la gráfica.

```
par(mar=c(4, 10, 4, 2))
```

Le explico como funciona la cosa. En el sistema base de graficar, la en que nos encontramos de momento, la función ‘par’ (para parÁMETROS) controla algunas ajustes importantes. El argumento ‘mar’ (para márgenes) tal vez es lo más útil entre ellos. Con este comando usted cambiara las margenes desde sus valores por defect de (5, 4, 4, 2), el órden de los valores sigue el modelo que hubiera visto en la lección Anatomía de una gráfica (parte inferior, izquierda, parte superior, derecha). Con el último comando usted hubiera quitado una unidad de la parte inferior, añadido 6 unidades a la izquierda para acomodar las etiquetas, y dejado los demás en paz.

Ahora usted puede recuperar la última llamada a `barplot()` y repetirla sin cambiar nada más que el argumento `'las'` a `las = 1`. Ahora si, queremos que las etiquetas de ambos ejes sean horizontales.

```
barplot(resumen$hiijos, names.arg = resumen$region, las=1, horiz = TRUE)
```



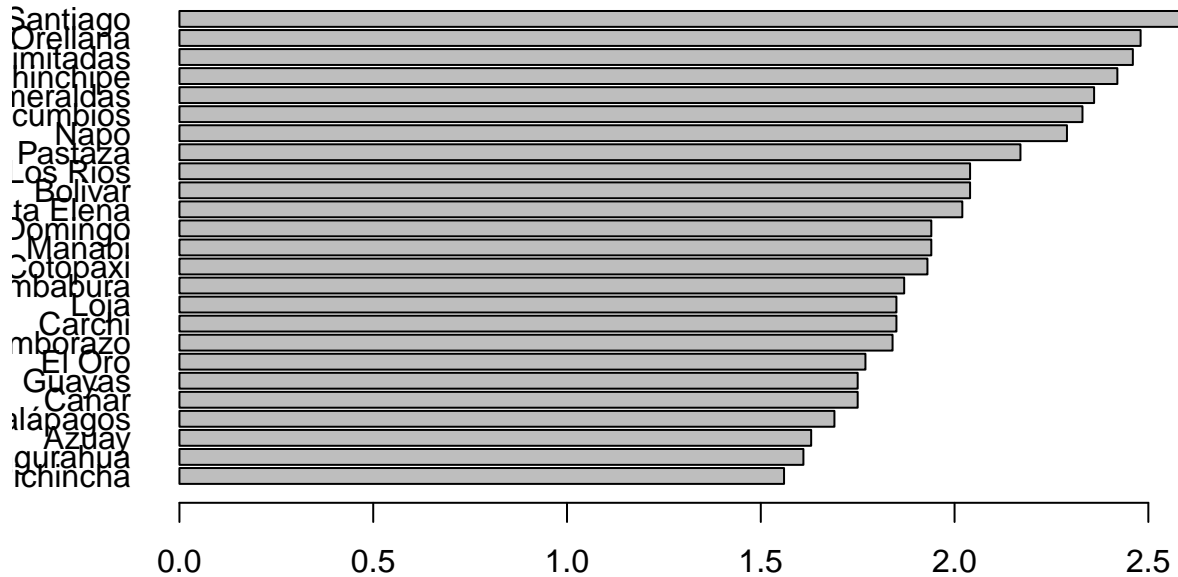
Depende el tamaño de la pantalla que usted tiene, puede que todavía se la vea apretada. Haga click en ZOOM para convencerse que no es nada grave. Se puede arrastrar la ventanilla PLOTS para que ella tenga mas espacio, pero no lo hagas tanto que apretara a la consola.

La grafica ya esta mucho mejor, pero nos queda un problema muy común que afecta el poder de la gráfica contar su historia. Acuérdense a las gráficas de la última lección, ¿estuvo muy facil leer el rango de valores de ellas, verdad? Antes de seguir, entonces, ¿le parece más difícil ahora? ¿Cómo son ordenadas las provincias?

- alfabeticamente
- aleatoriamente
- geograficamente

Supongo que este orden podría ser útil para ubicar rápidamente una cierta provincia, pero compárelo con el efecto de ordenar las provincias por el valor de la variable de respuesta, el número de hijos y hijas. Vamos a cambiar la llamada a la función `barplot` para controlar el orden, como sigue ‘`with(resumen[order(resumen[,2]),], barplot(hijos, names.arg = region, horiz = TRUE, las=1))`’. Hágalo, en seguida se lo explico a usted.

```
with(resumen[order(resumen[,2]), ], barplot(hijos, names.arg = region, horiz = TRUE, las=1))
```



No sé como se siente usted, pero a mi me tranquilizó los ojos este cambio, aunque todavía el producto podría ser mejorado. Ya se puede leer mejor las etiquetas y también el rango de valores. Creo yo que podemos dejar esta gráfica acá, aunque le toca a usted añadir etiquetas y títulos para completarla.

Vamos a comparar una opción más antes de partir. Pero antes de eso, quiero que vea el efecto del último comando que usted ha hecho en la gráfica esta. Teclee este parte del comando solo `resumen[order(resumen[,2]),]` y presione INTRO.

```
resumen[order(resumen[,2]), ]
```

```
##           region hijos
## 17      Pichincha  1.56
## 18    Tungurahua  1.61
##  1         Azuay  1.63
## 20      Galápagos  1.69
##  3         Cañar  1.75
##  9         Guayas  1.75
##  7        El Oro  1.77
##  6    Chimborazo  1.84
##  4         Carchi  1.85
## 11         Loja  1.85
## 10      Imbabura  1.87
##  5      Cotopaxi  1.93
## 13        Manabi  1.94
## 23    Santa Domingo  1.94
## 24      Santa Elena  2.02
##  2         Bolivar  2.04
## 12        Los Rios  2.04
## 16        Pastaza  2.17
## 15         Napo  2.29
## 21      Sucumbios  2.33
##  8      Esmeraldas  2.36
## 19    Zamora Chinchipe  2.42
```

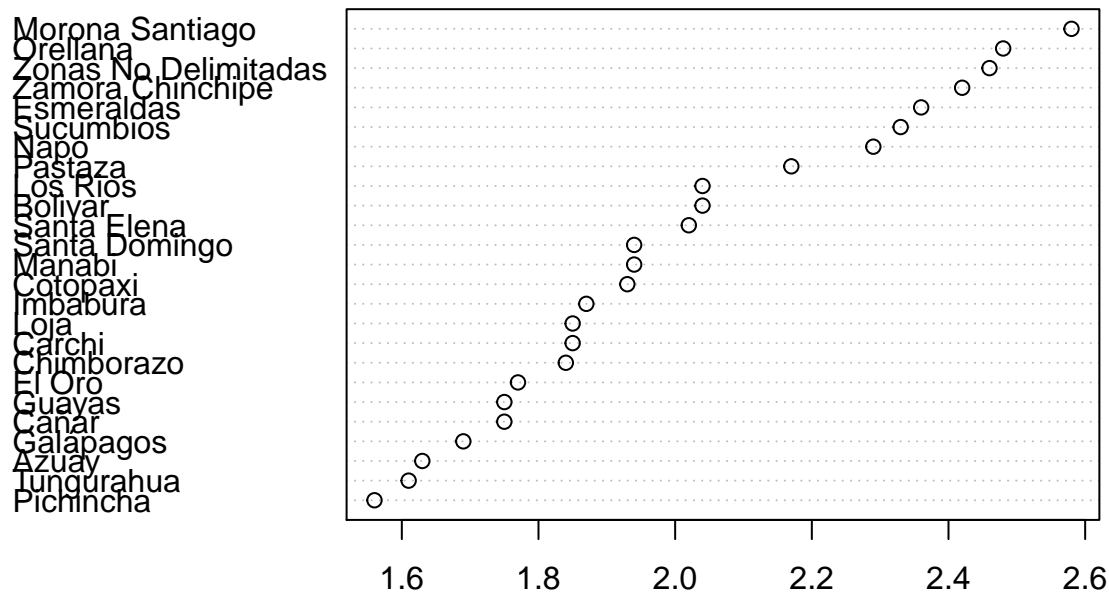
```
## 25 Zonas No Delimitadas 2.46
## 22 Orellana 2.48
## 14 Morona Santiago 2.58
```

Este comando no es nada transparente ¿verdad?. Dice a R algo como... con todo el objeto 'resumen', ordénalo por la segunda columna en sí. Un comando equivalente, tal vez más de fácil leer sería 'resumen[with(resumen, order(hijos)),]'. En sí, ninguno de ellos afectará un cambio permanente a nuestro tramo de datos, sino una reconfiguración temporal de conveniencia. Para hacer la próxima y final gráfica más sencilla realizar, vamos a guardar una versión del tramo con este orden por número de hijos y hijas. Solo hay que asignar el último comando su propio nombre, así resumenOrd <- resumen[with(resumen, order(hijos)),]. Pruébalo, el resultado aparecerá en la ventanilla ENVIRONMENT.

```
resumenOrd <- resumen[with(resumen, order(hijos)), ]
```

He guardado lo mejor para el fin. En el vídeo de principios en que le destaqué algunos heroes del campo de visualización de datos, mencioné William Cleveland. Según los principios de la percepción humana, él desarrolló una alternativa a la digrama de barras la que lleva su nombre 'Cleveland Plot'. En R, con nuestro conjunto reordenado podemos realizarlo así dotchart(x = resumenOrd\$hijos, labels = resumenOrd\$region). Pruébalo, puede que le quitara las ganas de hacer diagramas de barras para toda la vida.

```
dotchart(x = resumenOrd$hijos, labels = resumenOrd$region)
```



Para mi, esta gráfica es lo mejor de las opciones para comunicar este resumen de información. Los valores sobresaltan mejor de ella que de una tabla de valores, y también comunica su contenido mejor que la diagrama de barras. Antes de seguir, entonces, ¿por qué? Pues, el diseño de esta gráfica cumple un par de principios de presentación algo sutiles, pero influyen el impacto de la gráfica. Escoja la opción entre las siguientes que no sería un principio importante que cumple esta gráfica comparado con la diagrama de barras.

- Se favorece presentaciones horizontales
- Mantenga una buena razón de la información proporcionada a la tinta utilizada * No refiera a cero si el valor no es especialmente referente

Esta presentación es mucho mas ligera, ¿no cree? El cambio hasta puntos resulta en un impacto estético más agradable, pero los lineas débiles del fondo habilitan que un lector pueda sacar el valor para una cierta provincia. Opino yo que se gana mucho y se pierde nada.

Si usted quisiera cambiar los puntos de esta gráfica hasta círculos llenados o sólidos, o triángulos o lo que sea, antes de seguir, entonces, ¿cuál argumento ajustaría?

- pch
- las
- mar
- ninguno de ellos

Pues, usted acaba de completar dos lecciones en resumir una variable sola, primero el numero de hijos y hijas, la variable cuantitativa sola, y luego variación en ella por provincia de El Ecuador es decir, un resumen por una variable categorica. Para ella usamos una diagrama de barras y on `dotchart()`, lo que es más parecida a una diagrama de barras que una `diagram` de dispersión.

Tal vez le hubiera dado cuenta que cada vez le dejo más comandos para cumplir, títulos, etiquetas, etcétera. Espero que usted esté experimentando con los comandos, con los diseños, y que muy pronto empezara de jugar con sus propios tramos de datos sea por su propio interés o tal vez en trabajos para otras materias.

Esto es todo para esta lección. No se olvide guardar y anotar pedazos de código en su álbum de recortes digital. La idea es que estos pedazos sean útiles para usted en un futuro, y por eso, sea sabio que no solo pegar el código sino anotarlo también según convenga. Le espero un buen día.