Ggplot: gráficos de alta calidad

Julio César Alonso
Alejandra González

No. 33
Marzo de 2012

Apuntes de Economía

ISSN 1794-029X
No. 33

Editor
Julio César Alonso
jcalonso@icesi.edu.co

Asistente editorial
Andrés Mauricio Arcila V.
amarcila@icesi.edu.co

Gestión Editorial
Departamento de Economía - Universidad Icesi

www.icesi.edu.co
Tel: 5552334 ext: 8398. Fax: 5551441
Calle 18 # 122-135 Cali, Valle del Cauca, Colombia

Ggplot: gráficos de alta calidad

Julio César Alonso[*]
Alejandra González [**]
CIENFI - Departamento de Economía - Universidad Icesi
Cali - Colombia

9 de noviembre de 2012

Resumen

Este documento presenta una breve introducción a diferentes tipos de gráficos y discute qué tipos de gráfico emplear de acuerdo a la información que se quiera presentar. Está dirigido a personas interesadas en la presentación y realización, de datos y gráficos, respectivamente en el paquete ggplot2. Se supone un conocimiento previo en el manejo de R.
Palabras claves: Gráficos, ggplot2.

Abstract

This document presents an introdution to different kind of graphics and discusses which kind could be used depending on the information available. The document is appropiate for readers interested in data pensentation and graphics management using R and ggplot2 llibrary. We supose a previusly knowledge of R enviroment.
Key Words: Graphics, ggplot2, R-project.

[*]Director del centro de investigación en economía y finanzas (CIENFI) y director académico de la maestría en Economía de la Universidad Icesi

[**]Asisitente de investigación del CIENFI y estudiante de la maestría en Economía de la Universidad Icesi.

Objetivos de Aprendizaje

Al finalizar la lectura de este documento se espera que el lector esté en capacidad de:

- Escoger el tipo de gráfico apropiado dependiendo de la clase de variable con que se cuenta y el tipo de relación o evolución que se desea presentar.
- Emplear el paquete ggplot de R para graficar un conjunto de datos.

1. Introducción

En la actualidad la cantidad de información disponible en línea es numerosa y está creciendo a ritmos impresionantes. La disponibilidad de información pone al alcance de la mano de todos información pertinente para la toma de decisiones. Pero mucha información puede convertirse en un problema y abrumar al analista si no se sistematiza. Saber organizarla y presentarla es de gran ayuda e importancia a la hora de tomar decisiones. En este orden de ideas, esta información será más clara en la medida que se utilicen las herramientas indicadas para hacerlo.

El uso de gráficos se remonta a mucho tiempo atrás, pero sólo desde el siglo XVIII re-emergió esta herramienta. De acuerdo con Beniger y Robyin (1978) esta forma para presentar resultados respondió a las necesidades de: i.) la organización dentro de un espacio, ii.) la comparación entre variables discretas y continuas, y iii.) la distribución de variables continuas y una comparación entre continuas y discretas. Actualmente, con la ayuda de la tecnología, los gráficos se han convertido en una herramienta que se emplea a diario.

Así, el uso de puntos, líneas, colores entre otros instrumentos permite exponer información de forma mas estilizada y fácil de entender para los usuarios. De esta manera, más que sustituir las tablas estadísticas, los gráficos se han convertido en una de las más simples y poderosas formas para presentar información.

Al utilizar los gráficos, como herramienta para presentar información y resultados, es posible que la información sea fácilmente manipulada. Por tanto, resulta importante que esta información sea presentada con claridad, precisión y eficiencia. En este orden de ideas, Tufte (2001) propone que los gráficos deberían: i.) mostrar los datos, ii.) inducir al usuario a ver la información relevante, iii.) evitar distorsionar lo que los datos dicen, iv.) presentar muchos datos en un espacio pequeño, v.) permitir comparación entre gráficos (comparables), vi.) servir a un propósito claro y vii.) encontrarse integrados con las descripciones estadísticas y con el conjunto de datos.

En este documento se busca mostrar una forma rápida, fácil y estilizada de presentar información gráficamente empleando el paquete ggplot de R. Primero, se discute sobre la clasificación de las variables que se desean graficar. Segundo, se realiza una introducción al paquete ggplot2. Finalmente, se presentan ideas de la elección de un buen gráfico de acuerdo a lo sugerido por Abela (2008).

2. Clasificación de las variables

De acuerdo con Newbold, Carlson y Thorne (2008) existen dos métodos para clasificar las variables. Un método se refiere a la cantidad y al tipo de información que genera la variable. El segundo método se refiere a clasificar las varaibles de acuerdo a cómo son medidas (por niveles de medición).

Considerando el primer método, los datos pueden ser categóricos o numéricos. Las variables categóricas son variables cuyos resultados no se miden ni se cuentan; por ejemplo, cumple o no cumple y hombre o mujer. Por otro lado, las variables numéricas son respuestas observadas que corresponden a valores numéricos; éstas se subdividen en discretas y continuas. Las variables numéricas discretas, en general, toma un número finito de valores y en la mayoría de los casos proviene de un proceso de conteo; por ejemplo, el número de estudiantes que perdió un curso y el número de personas registradas en una base de datos. Las variables numéricas contínuas toma cualquier valor de un intervalo y generalmente proviene de procesos de medición, por ejemplo la estatura y el tiempo.

El segundo método permite clasificar los datos en cualitativos y cuantitativos. Los datos cualitativos pueden corresponder a niveles de medición nominales u ordinales; mientras que los datos cuantitativos pueden corresponder a niveles de medición de intervalos ó razones.

Los niveles de medición nominales y ordinales clasifican las variables en categorías que son mutuamente excluyentes y colectivamente exhaustivas, es decir, los individuos deben aparecer sólo en una categoría.

Los niveles de medición de intervalo y de razón clasifican los datos en un rango o escala, que para el primer caso es arbitraria pero para el segundo no. Cada valor de la escala corresponde a una categoría. Estas categorías son mutuamente excluyentes y colectivamente exhaustiva, siguen un orden lógico que corresponde a la magnitud de la escala asociada a la característica y las diferencias entre las categorías corresponde a diferencias entre las mediciones. En el caso de los niveles de intervalo, el cero no implica carencia de la característica, por ejemplo la temperatura, contrario a lo que ocurre con los de razón, donde el cero sí corresponde a una carencia de la característica, por ejemplo la distancia.

Por otro lado, la variables nominales no tienen un orden lógico, como por

ejemplo el género; pero los ordinales si siguen un orden lógico que corresponde a una característica, por ejemplo una escala de satisfacción.

3. Una introducción a ggplot

ggplot es un paquete de datos, creado por Hadley Wickham y Winston Chang, que se ejecuta en el software libre R. La principal característica de este paquete es ofrecer una forma fácil y estilizada de crear gráficos.

Para ejecutar este paquete es necesario tener instalado R[1] (Disponible en: http://r-project.org). Después de tener instalado R, es necesario instalar el paquete:

```
> install.packages("ggplot2")
```

Una vez instalado, ggplot podremos ejecutarlo cuando lo deseamos empleando el siguiente código en la línea de comando de R:

```
> library("ggplot2")
```

Es importante tener en cuenta que después de haber instalado por primera vez este paquete no será necesario volver a instalarlo, ejecutarlo será suficiente.

Otro paquete que puede ayudar al momento de hacer gráficos es "directlabels". Este permite llamar sólo ciertas filas o columnas de la base de datos. Este paquete se puede instalar por medio del siguiente código:

```
> install.packages("directlabels")
```

De igual manera que con cualquier otro paquete en R, éste se podrá ejecutar de la siguiente forma:

```
> library("directlabels")
```

3.1. Estructura básica

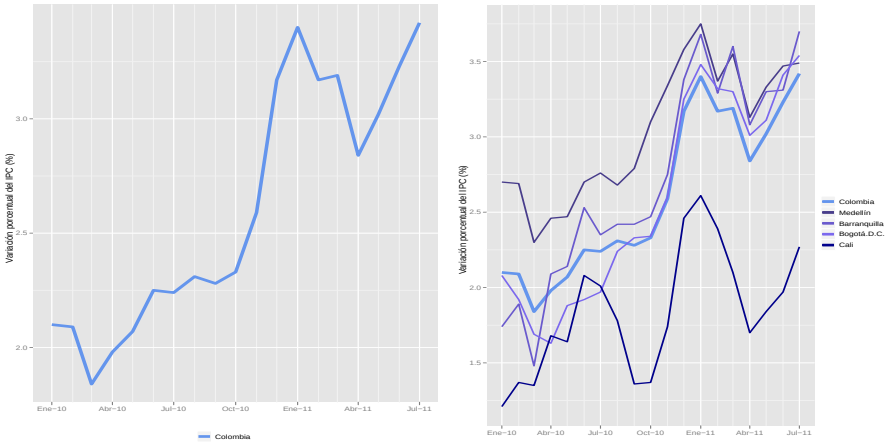
Para graficar empleando el paquete ggplot debemos emplear una serie de comandos que tienen una estructura estándar, y requieren en casi todos los casos cuatro argumentos:

[1] Los gráficos presentados en este documento se realizaron en la versión de R 2.13 y como se verá más adelante algunos en la 2.12.

- El nombre del "data frame" de donde se van a obtener los datos.
- Los datos (variable o variables) que se van a gráficar en el eje x.
- Los datos (variable o variables) que se van a gráficar en el eje y.
- El tipo de gráfico que se va a utilizar.

Al emplear este paquete es posible utilizar dos funciones: qplot y ggplot. La principal diferencia entre estas dos funciones es que con la segunda se pueden presentar diferentes datos y gráficos en un mismo plano cartesiano; mientra que con la primera sólo se puede generar un gráfico por plano cartesiano. En la Figura 1 se observa un ejemplo donde se presenta esta diferencia. En el gráfico del panel izquierdo se incluye una gráfica con qplot y en el derecho una con ggplot. En el ejemplo, la segunda gráfica muestra la misma información de la primera, y además, la de otras líneas que presentan la tendencia, en el mismo periodo, pero para diferentes ítems (las ciudades).

Figura 1: **Diferencia entre qplot y ggplot**



Para emplear la función qplot tendremos que suministrar cuatro argumentos. La estructura de la función corresponde a:

```
> qplot(x,y,data=" ",geom=" ")
```

donde,

- *x:* son los datos que se grafican en el eje horizontal.

- *y:* son los datos que se grafican en el eje vertical (Existen casos donde no se requiere esta información como es el caso de los histogramas).

- *data:* entre las comillas va el nombre de la base de datos (data frame) que se va a utilizar.

- *geom:* entre las comillas se escribe el tipo de gráfico que se desea utilizar, en caso de no colocarlo el paquete asume que la persona quiere un gráfico de dispersión.

Además de estos argumentos el paquete nos permite agregar ciertas características para mejorar la apariencia de los gráficos, dentro de esas características encontraremos el color, la opacidad, los títulos, entre otros. (Ver cuadro 1 para un listado más completo)

Cuadro 1: **Opciones de la función qplot**

Argumento	**Características**	**Resultado**
geom	histogram	Histograma
	line	Gráficos de líneas
	boxplot	Gráfico de caja
	density	Gráfico de densidad
colour	I(blue)	Color de borde azul
fill	I(blue)	Color de relleno azul
alpha	I(cualquier número entre 0 y 1)	1 completamente opaco
		y 0 completamente transparente
main	Nombre	Título del gráfico
xlab	Nombre	Título del eje x
ylim	Nombre	Límites del eje y

Ejemplo: qplot(x,y,data="",geom="", alpha=I(),xlab="", ylim="")

La estructura básica de la función ggplot implica típicamente 4 argumentos. Esta corresponde a:

ggplot(data,aes(x)) + geom_*(aes(y=" "))

donde

- *data:* Es el nombre de la base de datos que se va a utilizar para el gráfico.

- *x:* Es el nombre de la columna (de la base de datos especificada en el paso anterior) que contiene los datos que se van a gráficar en el eje horizontal.

- *y:* entre las comillas se escribe el nombre de la columna que contiene los datos que se van a graficar en el eje horizontal (este argumento se emplea sólo en caso de ser necesario).

- **:* en lugar del asterisco se determina el tipo de gráfico que se quiere realizar. Algunos tipos de gráficos son: histogramas, de densidad, líneas, barras, entre otros.

Igual que con la función qplot, para la función ggplot existen diferentes características que se pueden agregar para mejorar la apariencia de los gráficos. En el cuadro 2 presentamos algunas de estas posibilidades.

Cuadro 2: **Opciones de la función ggplot**

Argumento	**Característica**	**Resultado**
*	_hist	Histograma
	_line	Gráficos de líneas
	_point	Gráfico de dispersión
	_bar	Gráfico de barras
colour	=“blue”	Color de borde azul
fill	=“blue”	Color de relleno azul
+opts=()	=(tittle=“título del gráfico”,	Título del gráfico
	legend.position=“none”)	Oculta la legenda
+labs=()	=(x=“título eje x”,	Título eje x
	y=“título eje y”)	Título eje y
+ylim	(límite inferior,límite superior)	Límites del eje y

Ejemplo: ggplot(data,aes(x)) + geom_*(aes(y=“”,colour=“”)) + labs(x=“”)

Como ya habiamos mencionado, la principal diferencia de ggplot con qplot es que con la primera podemos agregar diferentes gráficos en un mismo plano cartesiano. En este caso esto lo podemos hacer de una manera muy sencilla. Por ejemplo, si queremos tener un gráfico que tenga puntos y líneas el código será:

ggplot(data,aes(x))+geom_point(aes(y=“ ”))+geom_line(y=“ ”)

Una de las características importantes a tener en cuenta en estos gráficos es que la variable que se gráfica en el eje horizontal debe ser la misma para los gráficos que se quieran presentar en el mismo espacio, mientras que la variable que se gráfica en el eje vertical si puede cambiar.

4. Escogiendo el gráfico adecuado

Utilizar un gráfico u otro para mostrar datos puede hacer una gran diferencia. Elegir el tipo de gráfico no es una tarea que se debe tomar a la ligera. El tipo de gráfico depende de lo que se quiere destacar de la información. Siguiendo a Abela (2008); podemos clasificar los gráficos más comunes de acuerdo con su finalidad (Ver figura 2). En el momento de decidir que gráfico utilizar es pertinente pensar en que es lo que deseamos representar:

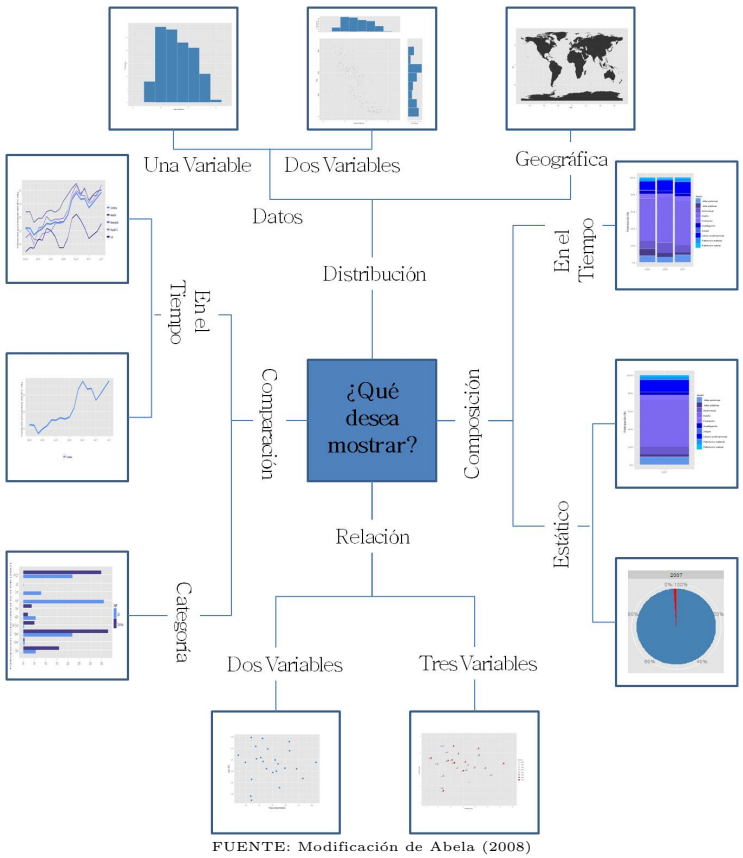
- Relación entre variables.
- Composición de los datos.
- Comparaciones.
- La distribución de los datos.

Pero también es importante considerar el tipo y la cantidad de variables que se va a graficar. Así dependiendo de si los datos corresponden a realizaciones de variables categóricas, numéricas o una combinación de estas, algunos gráficos serán mejores que otros (Ver Figura 3).

Cuadro 3: **Tipos de gráficos de acuerdo con el tipo de variable**

¿Qué desea mostrar?	Nombre del Gráfico	Univariado		Multivariado	
		numérica	Categórica	Numérica	Ambas
Composición	Gráfico de torta		X		
	Barra de porcentaje		X		
	Columnas apiladas			X	
Distribución	Mapas	X	X		
	Histogramas y diagrama de dispersión			X	
	Histograma			X	
Comparación	Gráfico de líneas				X
	Gráfico de una línea				X
	Gráfico de barras		X		
Relación	Diagrama de dispersión			X	
	Gráfico de burbujas			X	

Figura 2: **Tipos de gráficos cuando se gráfican variables categóricas y numéricas**



FUENTE: Modificación de Abela (2008)

En el caso en que buscamos mostrar la relación existente entre variables numéricas, el tipo de gráfico utilizado dependerá de la cantidad de variables que queremos relacionar. El gráfico de dispersión es preferible cuando tenemos dos variables, mientras que el gráfico de burbujas es la mejor opción cuando vamos relacionar tres variables. En este tipo de gráfico el tamaño de la burbuja representa a la tercer variable.

Para presentar la forma en que una categoría se puede descomponer en diferentes sub-categorías debemos decidir si es en un periodo o en varios. Los gráficos de torta y las barras de porcentaje son los mas indicados para mostrar descomposición en un sólo periodo; mientras que las columnas apiladas de porcentaje permiten mostrar cómo esta conformado determinado ítem y cómo ha sido su evolución a través del tiempo.

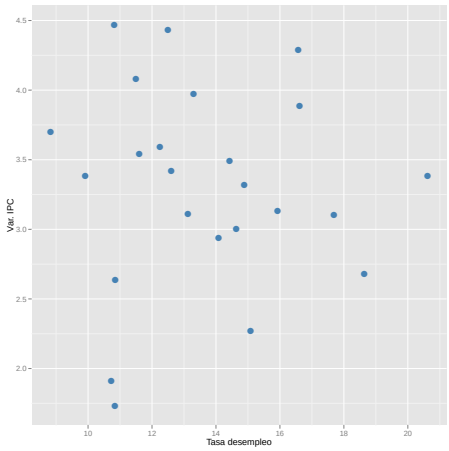
Cuando buscamos presentar comparaciones de los datos necesitamos distinguir nuevamente a través de cuántos periodos deseamos hacer las comparaciones. Los gráficos de barras son los mejores al comparar categorías en un sólo periodo; pero, para comparar a través de diferentes momentos en el tiempo lo más indicado son los gráficos de una sola línea (compara las diferentes realizaciones de una misma categoría a través del tiempo) y los de muchas líneas (compara entre diferentes categorías a través del tiempo).

Finalmente, al presentar la distribución de los datos debemos determinar si queremos saber sólo la distribución de una variable o la de varias variables. En la primera situación los histogramas son mejores; pero en el caso que tengamos dos variables, los gráficos de puntos (o diagramas de dispersión) son preferidos. De igual forma, existen situaciones donde deseamos mostrar la distribución geográfica, en esos casos es pertienente emplear mapas.

4.1. Graficando relaciones con ggplot

Al presentar la relación entre dos variables numéricas, un diagrama de dispersión puede ser el gráfico más indicado. En cada eje se gráfica una de las variables de interés lo cual permite observar la relación que puede existir entre éstas. Por ejemplo, en la figura 3 podemos ver la relación entre la inflación y la tasa de desempleo para diferentes ciudades en un periodo determinado.

Figura 3: **Diagrama de dispersión**



FUENTE: Banco de la República y DANE

El gráfico de la figura 3 se obtiene por medio de las siguientes líneas de código:

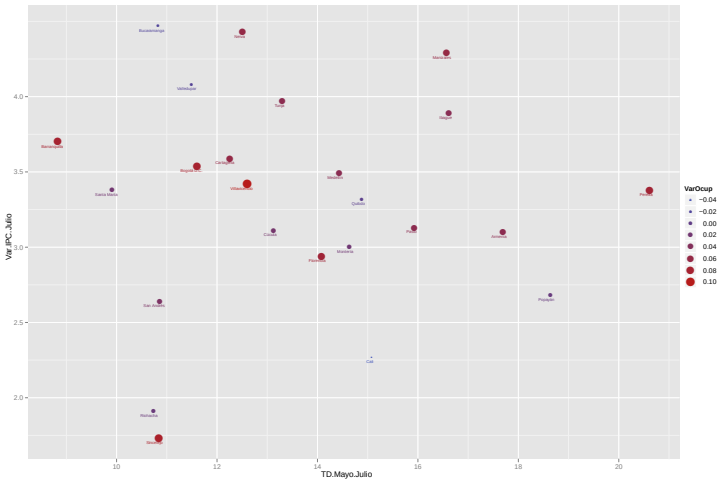
```
> IPCOCU<-read.csv("IPCOCU.csv", sep=";", header=TRUE)
> ggplot(IPCOCU, aes(TD.Mayo.Julio, Var.IPC.Julio)) +
+     geom_point(colour = "steelblue") +
+     labs(x = "Tasa desempleo", y = "Var. IPC")
```

Explicación del código que genera la Figura 3

La base de datos que utilizamos para graficar este diagrama de dispersión es IPCOCU, que previamente fue cargada del archivo IPCOCU.csv con la función "read.csv". Esta base de datos incluye cuatro columnas, la primera corresponde al nombre de las 24 princiaples ciudades de Colombia, la segunda son los datos de la variación del IPC (Var.IPC.Julio), la tercera es la tasa de desempleo (TD.Mayo.Julio) y la última la variación de personas ocupadas (VarOcup).
El eje X de la figura 3 es la tasa de desempleo y el eje Y la variación del IPC. En la función "geom_point" se determina el tipo de gráfico que se realizará, en este caso es uno de puntos, y que el color de estos será "steelblue". Finalmente, los títulos de los ejes se establecen con la función "labs".

Pero no siempre vamos a relacionar dos variables, cuando queremos entender la relación entre tres variables numéricas es recomendable utilizar los gráficos de burbujas. En este tipo de gráficos, además de mostrar una variable en cada eje, se agrega una tercer variable que se medirá dependiendo del tamaño de cada punto del diagrama de dispersión, es importante considerar que esta tercer variable debe ser numérica continua. Por ejemplo, además de graficar la relación entre la tasa de desempleo y la inflación para varias ciudades en un determinado momento, en la figura 4 es posible mostrar cuánto varió la población de ocupados, de tal forma que los puntos más grandes son los que muestran una mayor variación en la población ocupada.

Figura 4: **Gráfico de burbujas**



FUENTE: Banco de la República y DANE

```
> qplot(TD.Mayo.Julio, Var.IPC.Julio, data = IPCOCU, size = VarOcup,
+     colour = VarOcup, xlim = c(8, 22), ylim = c(1, 5)) +
+     geom_text(aes(label = Ciudad),size = 2, hjust = 0.75, vjust = 2)
```

Explicación del código que genera la Figura 4

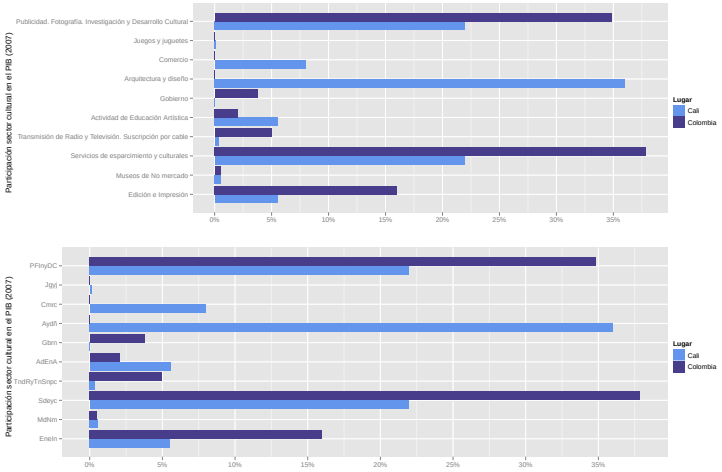
Para este gráfico empleamos nuevamente la base de datos IPCOCU, el eje X e Y nuevamente corresponde a la tasa de desempleo y a la variación del IPC, respectivamente. Como observamos en el código que genera la figura 4 estamos empleando la función qplot. Esta función esta predeterminada para que cuando no definamos el tipo de gráfico, como en este caso, el resultado sea uno de puntos. Pero esta figura se diferencia de la anterior porque en este caso estamos incluyendo una tercer variable. La variable VarOcup, es una variable númerica continua que determina el tamaño y el color de los puntos, así entre más grande el punto y más rojo, es porque esta variable toma un mayor valor. En este gráfico también establecemos que el eje X empieza en 8 y termina en 22, y que el Y va desde 1 hasta 5. Finalmente, usamos la función geom_text, donde determinamos que cada punto va a tener el nombre de la ciudad a que corresponda, el tamaño de estos letreros los establecemos con "size", y la posición de estos la determinamos con "hjust" y con "vjust".

4.2. Graficando comparaciones con ggplot

En ciertas ocasiones necesitamos graficar variables que no son continuas. En especial es común que deseemos hacer comparaciones en el comportamientro

entre diferentes componentes de la misma variable, ya sea durante un momento determinado o a través del tiempo.
Un diagrama de barras es pertinente utilizarlo cuando tenemos diferentes ítems por variable o simplemente cuando tenemos varias variables categóricas. Cada barra representa un ítem diferente y su longitud representa la frecuencia (que puede ser absoluta o relativa; es decir la cantidad de veces que aparece un ítem o el porcentaje sobre el total de todos los ítems), la cantidad o el porcentaje de cada categoría. Un ejemplo de este tipo se muestra en la figura 5, la participación de los sectores culturales en el PIB de la cultura en Cali y en Colombia, para un año determinado.

Figura 5: **Gráfico de barras**



FUENTE: Alonso, Gallego y Rios (2010)

```
> ggplot(Barchart, aes(Medio, Cali, fill = Lugar)) + geom_bar(position = "dodge")+
+     coord_flip() + scale_y_continuous(formatter = "percent") +
+     scale_fill_manual(values = c("cornflowerblue", "darkslateblue")) +
+     labs(x = "Participación sector cultural en el PIB (2007)",
+         y = "")
```

Explicación del código que genera la Figura 5-Panel superior

Para esta figura utilizamos los datos del archivo Barchart.csv, el cual incluye tres columnas: la primera corresponde a un subsector cultural (Medio), la segunda se refiere a si esa participación corresponde a Cali o a Colombia (Lugar) y la tercera muestra la participación, en el PIB de Cali y de Colombia, de los subsectores culturales (Cali). El código que genera esta figura tiene como base la función ggplot. El eje X de esa figura corresponde al subsector cultural y el Y a la participación de este en el PIB de Cali y de Colombia. Para distinguir si esta participación es la municipal o la nacional, se utiliza el comando "fill", que da un color diferente a la participación dependiendo del lugar, este color se determina en la función "scale_fill_manual". La función "geom_bar" define que este código va a generar un gráfico de barras y al utilizar el comando "position = "dodge" se separan las barras, para que estas no queden apiladas. "coord_flip()" por su parte hace que el eje X e Y se intercalen y las barras queden en sentido horizontal y no vertical. Para este caso, también establecemos que los datos del eje Y corresponden a porcentajes (formatter = "percent"), es importante tener en cuenta que esto es posible porque la variable representada en este eje es una variable numérica continua. Finalmente, la función "labs" se emplea para establecer los nombres de los ejes.

```
> ggplot(Barchart, aes(Medio, Cali, fill = Lugar)) + coord_flip() +
+     geom_bar(position = "dodge") + scale_y_continuous(formatter = "percent") +
+     scale_fill_manual(values = c("cornflowerblue", "darkslateblue")) +
+     labs(x = "Participación sector cultural en el PIB (2007)",
+         y = "") + scale_x_discrete(formatter = "abbreviate")
```

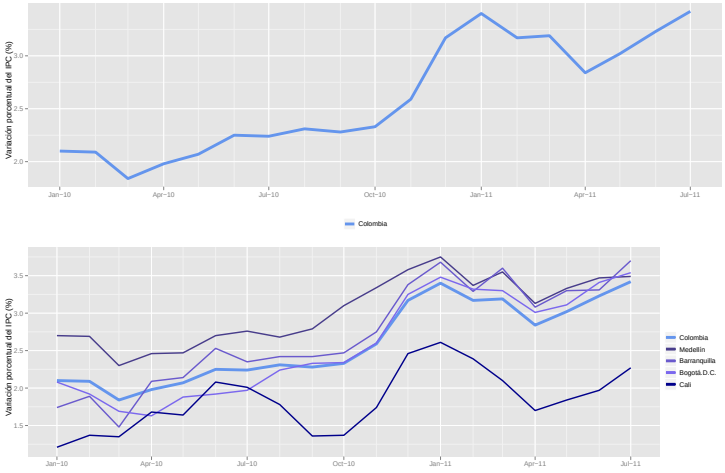
Explicación del código que genera la Figura 5- Panel inferior

La única diferencia entre el código que genera la figura del panel superior y la del panel inferiorz es que se utiliza la función "scale_x_discrete()". Dado que el nombre de los subsectores culturales es muy largo, ggplot permite abreviarlos automaticamente, esto utilizando la siguiente línea de código: "scale_x_discrete(formatter = "abbreviate")". En el caso de esta figura es importante considerar que definir el título del eje Y como " " significa que ese eje no tendrá nombre.

Por otro lado, cuando las comparaciones las queremos hacer en diferentes periodos es apropiado utilizar gráficos de líneas, donde además de mostrar la evolución de una variable en un periodo, se puede comparar la evolución de esta y otras variables durante el mismo tiempo. Cada línea representa una categoría diferente y muestra la evolución de esa determinada variable en el tiempo. Ejemplo de lo anterior sería la evolución de variables macroeconómicas como la

inflación, durante un periodo determinado (Ver Figura 6). Es importante tener en cuenta que al graficar estas variables se coloca en el eje X el tiempo y en el eje Y la variable de interés.

Figura 6: **Gráfico de Líneas**



FUENTE: Banco de la República

```
> ggplot(IPC, aes(as.Date(Mes))) + geom_line(aes(y = Total.IPC,
+     colour = "Colombia"), size = 2) +
+     labs(x = "", y = "Variación porcentual delIPC (%)", colour = "") +
+     scale_colour_manual(values = c("cornflowerblue")) +
+     opts(legend.position = "bottom")
```

Explicación del código que genera la Figura 6- Panel superior

Los datos empleados para este gráfico se encuentran en el archivo IPC.csv. Esta base de datos contine seis columnas, la primera la fecha (Mes) a la que corresponden los datos, la segunda a los datos de la variación del IPC de Colombia (Total.IPC), y de la tercera a la sexta se encuentran los datos de la varaiación del IPC para Medellín, Barranquilla, Bogotá y Cali. En el eje X graficamos la fecha, para la cual se utiliza el comando "as.Date", es decir se le dice al paquete que esa es una fecha que debe organizar cronologicamente. La función "geom_line" determina que estamos haciendo un gráfico de líneas. En el eje Y representamos los datos de la variación porcentual del IPC de Colombia. El comando "colour" en este caso define que la línea que represente esos datos va a tener el color que se establezca en la función "scale_colour_manual" y que va a ser de tamaño 2. Los títulos los definimos con la función "labs" y la posición de la leyenda la establecemos con el comando "legend.position".

```
> ggplot(IPC, aes(as.Date(Mes))) + geom_line(aes(y = Total.IPC,
+     colour = "Colombia"), size = 2) + geom_line(aes(y = Medellin,
+     colour = "Medellin"), size = 1) + geom_line(aes(y = Barranquilla,
+     colour = "Barranquilla"), size = 1) + geom_line(aes(y = Bogota.D.C.,
+     colour = "Bogota.D.C."), size = 1) + geom_line(aes(y = Cali,
+     colour = "Cali"), size = 1) + labs(x = "",
+     y = "Variacion porcentual del IPC (%)", colour = "") +
+     scale_colour_manual(values = c("cornflowerblue",
+     "darkslateblue", "slateblue", "mediumslateblue", "darkblue"))
```

Explicación del código que genera la Figura 6-Panel inferior

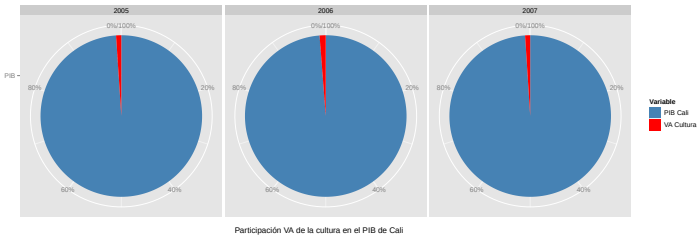
La principal diferencia entre el código empleado para la Figura del panel superior y esta, es que en este representamos diferentes ítems en el mismo plano, por lo que aparecen diferentes líneas "+geom_line".

4.3. Graficando composiciones con ggplot

Estos gráficos permiten mostrar cómo los datos se encuentran subclasificados. La forma de presentarlos dependerá si esta composición cambia a través del tiempo o si es estática. Cuando pensamos en la composición de los datos en un momento del tiempo es recomendable emplear, por ejemplo, gráficos de torta, columnas de porcentaje por subcomponentes, entre otros gráficos.
Un gráfico de pastel es un círculo que se divide para presentar la proporción de cada categoría en el total. La participación del PIB cultural de Cali en el PIB total de esta misma ciudad, en un momento determinado, es un ejemplo de este

tipo de gráficos (Ver Figura 7).

Figura 7: **Gráfico de torta**



FUENTE:Alonso, Gallego y Rios

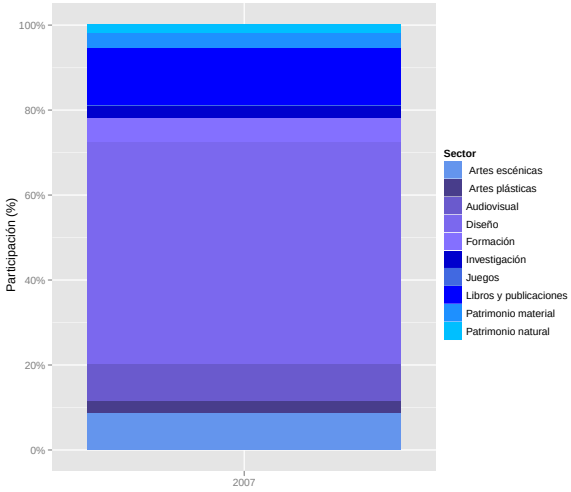
```
> ggplot(PPIB, aes(x = "PIB", y = ParticipacionPIBCali, fill = Variable)) +
+     geom_bar(width = 1) + coord_polar(theta = "y") + facet_wrap(~Ano,
+     nrow = 1) + scale_y_continuous(formatter = "percent") +
+     scale_fill_manual(values =c("steelblue","red")) +
+     labs(x = "Participacion VA de la cultura en el PIB de Cali",y = "")
```

Explicación del código que genera la Figura 7

Los datos utilizados para este gráfico se encuentran en el archivo PPIB.csv. Esta base de datos contiene tres columnas, la primera es el Año, la segunda determina si el dato del lado derecho corresponde al PIB total de Cali o sólo a la parte cultural (Variable) y la tercera corresponde a la participación (ParticipaciónPIBCali). En este caso es importante definir el eje Y pues es el que tiene los datos que nos interesa graficar. El código empleado en este caso es similar al del gráfico de barras "geom_bar"; pero adicional a esta función se emplea "coord_polar" donde establecemos que lo que los datos que vamos a graficar son los del eje Y y que estos los representaremos en un gráfico de torta. Dado que son varios los años que graficamos, emplearemos "facet_wrap" para que ggplot separe los datos por Años y por tanto se generen tantos gráficos como años existan en la base de datos y los organice en una sola fila. El color de relleno del gráfico cambiará dependiendo de la cantidad de ítems que se definan en la columna de la base de datos llamada Variable, que pare este caso son solo 2 ("steelblue","red").

Las columnas de porcentaje por subcomponente, así como el gráfico de torta, muestran cómo está conformado el total de determinada categoría en un momento del tiempo. Por ejemplo, en la figura 8 graficamos la participación en el PIB cultural de Cali de cada subsector cultural y durante un periodo determinado.

Figura 8: **Columna de Porcentaje**



FUENTE:Alonso, Gallego y Rios

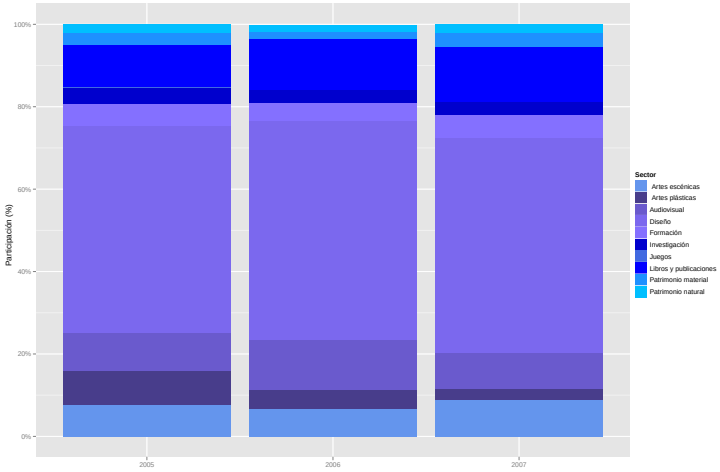
```
> año2007<-which(Stacked[,2]==2007)
> Saño2007<-Stacked[año2007,]
> ggplot(Sano2007, aes(x = as.character(Ano), y = Participacion,
+     fill = Sector)) + geom_bar() + scale_y_continuous(formatter = "percent") +
+     scale_fill_manual(values = c("cornflowerblue", "darkslateblue",
+         "slateblue", "mediumslateblue", "lightslateblue", "mediumblue",
+         "royalblue", "blue", "dodgerblue", "deepskyblue")) +
+     labs(x = "", y = "Participacion (%)")
```

Explicación del código que genera la Figura 8

Para este gráfico empleamos el archivo Stacked.csv. Esta base de datos contiene tres columnas, en la primera aparece el nombre de los subsectores culturales (Sector), en la segunda el Año y en la tercera la participación del sector en el PIB cultural de Cali (Participacion). Considerando el gráfico que presentamos es necesario modificar la base de datos, de esa forma con las primeras dos líneas del código se crea un "data frame" donde se alamcenan sólo los datos que corresponden al año 2007. Este gráfico de barras por componente, en el eje X ubica el Año y en el Y la participación de cada componente en el PIB. El comando "fill" lo utilizamos para que cada participación aparezca de un color diferente, el cual se define posteriomente con la función "scale_fill_manual". La función "geom_bar" determina que el gráfico es de barras. Dado que las participaciones estan dadas en decimales, empleamos la función "scale_y_continuous" donde establecemos que estos datos apareceran como porcentaje en la Figura. Finalmente la función "labs" la utilizamos para establecer los nombres de los ejes.

Los anteriores gráficos nos permiten mostrar la composición de una variable determinada en un momento del tiempo. Pero en ocasiones buscamos ver cómo ha cambiado la estructura a través de diferentes periodos. Cuando surge esta necesidad es posible que hagamos columnas de porcentaje en las que mostremos por ejemplo la forma en que los diferentes subsectores del sector cultural de Cali participan en el total del sector, esto durante 3 años diferentes.

Figura 9: **Columnas apiladas**



FUENTE:Alonso, Gallego y Rios

```
> ggplot(Stacked, aes(Ano, Participacion, fill = Sector)) +
+     geom_bar(stat = "identity") +scale_y_continuous(formatter = "percent") +
+     scale_fill_manual(values = c("cornflowerblue","darkslateblue",
+     "slateblue", "mediumslateblue", "lightslateblue","mediumblue", "royalblue",
+     "blue", "dodgerblue", "deepskyblue")) +labs(x = "", y = "Participación (%)")
```

Explicación del código que genera la Figura 9

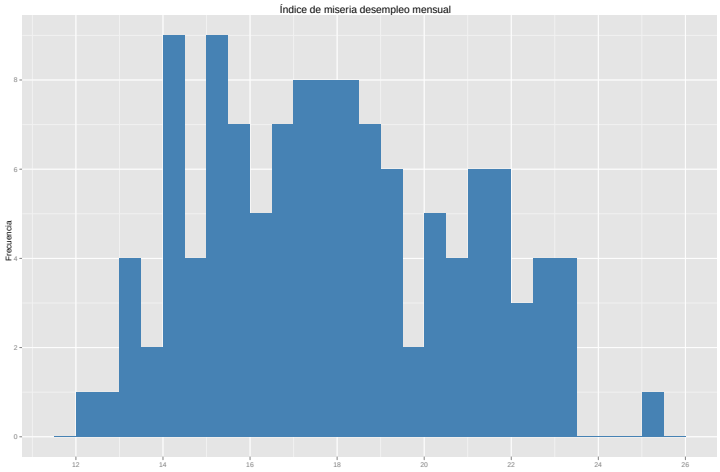
El código empleado para este gráfico es similar al utilizado para la Figura 8. La diferencia es que para este caso se utilizó la base de datos "Stacked" completa, por tanto ya no solo aparece una barra sino tres.

4.4. Graficando distribuciones con ggplot

Los anteriores gráficos son utilizados tradicionalmente cuando se tienen pocos datos, pero cuando la muestra a graficar es muy grande resulta útil mostrar los datos organizados en grupos.
Uno de los gráficos recomendados para mostrar estas agrupaciones es el histograma. Este gráfico de barras presenta datos numéricos agrupados, de tal forma que cada barra representa la frecuencia o el porcentaje de cada grupo. Dado que se trata de variables continuas no hay brecha entre cada barra del histograma y en el eje X se gráfica la variable de interés mientras en el eje Y se representa la frecuencia o porcentaje, de esta forma el resultado final nos permite hacernos una idea de la forma en que se distribuye la variable que se representa en el eje X.

Existen múltiples ejemplos de variables que podemos representar en histogramas, por ejemplo en la figura 10 observamos un histograma donde se agrupa por rangos el índice de miseria.

Figura 10: **Histograma**



FUENTE: Calculos CIENFI basados en datos del Banco de la República y del DANE

```
> ggplot(IMeIGBC, aes(Indice.de.Miseria.Desempleo.mensual))
+     + geom_histogram(fill = "steelblue",binwidth = 2, colour = "white") +
+     xlim(10, 27) + labs(x = "Índice de Miseria",y = "Frecuencia")
```

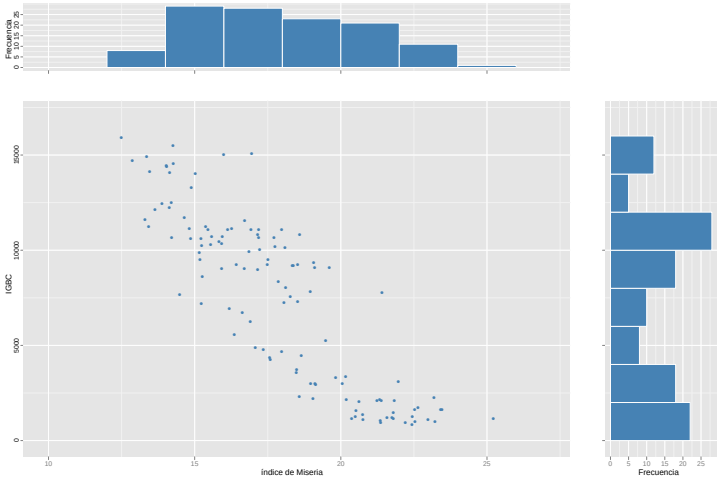
Explicación del código que genera la Figura 10

En el caso de un histograma sólo son necesarios los datos del eje X pues el eje Y es la frecuencia con que los datos de X aparezca. La base de datos empleada en este caso es la correspondiente al archivo IMeIGBC.csv. Este "data frame" esta conformado por 4 columnas, la primera es la Fecha, la segunda es el índice de miseria considerando el desempleo mensual (Indice.de.Miseria.Desempleo.mensual), la tercera es el indice de miseria considerando el desempleo timestre móvil (Indice.de.Miseria.Desempleo.Tri.movil) y al cuarta es el IGBC. El tipo de gráfico lo definimos con la función "geom_histogram", donde establecemos que el color de fondo de las barras será "steelblue", el de la líneas de las barras será "white" y el ancho entre intervalos a gráficar es 2. El eje X esta entre 10 y 27 y los títulos de los ejes los establecemos con la función "labs".

Pero además de conocer la distribución de una variable específica en ocasiones tenemos la necesidad de relacionar estas variables. Por ejemplo además de

mostrar la distribución del IGBC y del índice de miseria a través de histogramas, podemos presentar la relación entre estas dos variables numéricas (Ver Figura 11).

Figura 11: **Gráfico de distribución**



FUENTE:Calculos CIENFI basados en datos del Banco de la República y del DANE

```
> IM <- ggplot(IMeIGBC, aes(Indice.de.Miseria.Desempleo.mensual)) +
+     geom_histogram(fill = "steelblue", binwidth = 2, colour = "white") +
+     opts(axis.text.x = NULL, axis.text.y = theme_text(angle = 90)) +
+     xlim(10, 27) + labs(x = "", y = "Frecuencia")
> IGBC1 <- ggplot(IMeIGBC, aes(IGBC)) + geom_histogram(fill = "steelblue",
+     binwidth = 2000, colour = "white") + opts(axis.text.y = NULL) +
+     xlim(0, 17000) + coord_flip() + labs(x = "", y = "Frecuencia")
> point <- ggplot(IMeIGBC, aes(Indice.de.Miseria.Desempleo.mensual,
+     IGBC)) + geom_point(colour = "steelblue") + xlim(10, 27) +
+     ylim(0, 17000) + opts(axis.text.y = theme_text(angle = 90)) +
+     labs(x = "Ãndice de Miseria", y = "IGBC")
> grid.newpage()
> pushViewport(viewport(layout = grid.layout(5, 5)))
> vplayout <- function(x, y) viewport(layout.pos.row = x, layout.pos.col = y)
> print(IM, vp = vplayout(1, 1:4))
> print(IGBC1, vp = vplayout(2:5, 5))
> print(point, vp = vplayout(2:5, 1:4))
```

Explicación del código que genera la Figura 11

En este gráfico se presentan en un mismo panel tres gráficos diferentes. El código utilizado genera un panel en blanco "grid.newpage()". "pushViewport", permite definir como se va a dividir este panel. Finalmente, la posición donde se va a ubicar cada gráfico la definimos con "print(IM, vp = vplayout(1, 1:4))", que para este caso indica que el gráfico IM estará ubicado en la fila 1 y entre las columnas 1 hasta la 4.

Finalmente, existen situaciones donde queremos realizar comparaciones geográficas, en ese caso un mapa sería lo más adecuado. La última parte de este documento esta dedicada a mostrar la forma en que se pueden realizar mapas en ggplot2.

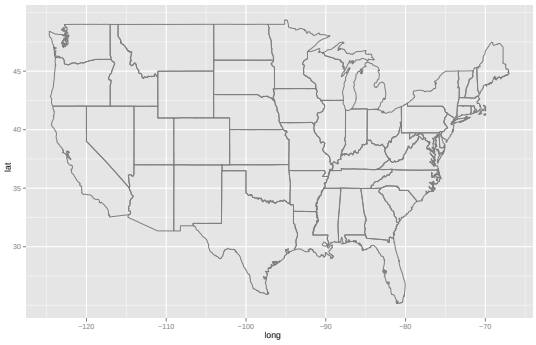
Primero, es necesario que quien esté interesado en hacer algún mapa tenga instalado en su computador R 2.12 (No todas las versiones posteriores son compatibles). Segundo, es necesario definir los bordes del lugar que se desea representar en los mapas. De acuerdo con Wickham (2009) al ejecutar el paquete "Maps" se encuentran disponibles, en R, los siguientes mapas:

Cuadro 4: **Mapas disponibles en ggplot2**

Párea Geografica	**Nombre del mapa**
Francia	france
Italia	italy
Nueva Zelanda	nz
USA a nivel de condado	county
USA a nivel de estado	state
Limites de USA	usa
Mundo	world

Partiendo de lo anterior, el mapa donde se muestra cada uno de los Estados de Estados Unidos será:

Figura 12: **Estados Unidos**



ggplot(us.cities, aes(long,lat))+borders("state")

Donde us.cities es una base de datos que esta disponible en R:

Cuadro 5: **Ciudades de Estados Unidos**

name	country.etc	pop	lat	long	capital
Abilene TX	TX	113888	32.45	-99.74	0
Akron OH	OH	206634	41.08	-81.52	0
Alameda CA	CA	70069	37.77	-122.26	0
Albany GA	GA	75510	31.58	-84.18	0
Albany NY	NY	93576	42.67	-73.80	2
Albany OR	OR	45535	44.62	-123.09	0

Para ejecutar esta base de datos es necesario escribir el siguiente código: data(us.cities)

Dado que existe limitación respecto a los mapas y bases de datos disponibles en R, a continuación se ofrece una alternativa. Por ejemplo para hacer un mapamundi es posible obtener los datos de http://www.mappinghacks.com/data/. En esta Página web encontrará un archivo que contiene la información que se presenta en el Cuadro 5 para todos los países del mundo.

Cuadro 6: **Países**

Fips	Iso2	Iso3	Un	Name	Area	Pop2005	Region	Subregion	Lon	Lat
AC	AG	ATG	28	Antigua and Barbuda	44	83039	19	29	61.783	17.078
AG	DZ	DZA	12	Algeria	238174	32854159	2	15	2.632	28.163
AJ	AZ	AZE	31	Azerbaijan	8260	8352021	142	145	47.395	40.430
AL	AL	ALB	8	Albania	2740	3153731	150	39	20.068	41.143
AM	AM	ARM	51	Armenia	2820	3017661	142	145	44.563	40.534
AO	AO	AGO	24	Angola	124670	16095214	2	17	17.544	12.296

FUENTE: http://www.mappinghacks.com/data/

Con esta información y después de abrir R 2.12 se deben cargar los siguientes paquetes:

```
> library("sp")
> library("maps")
```

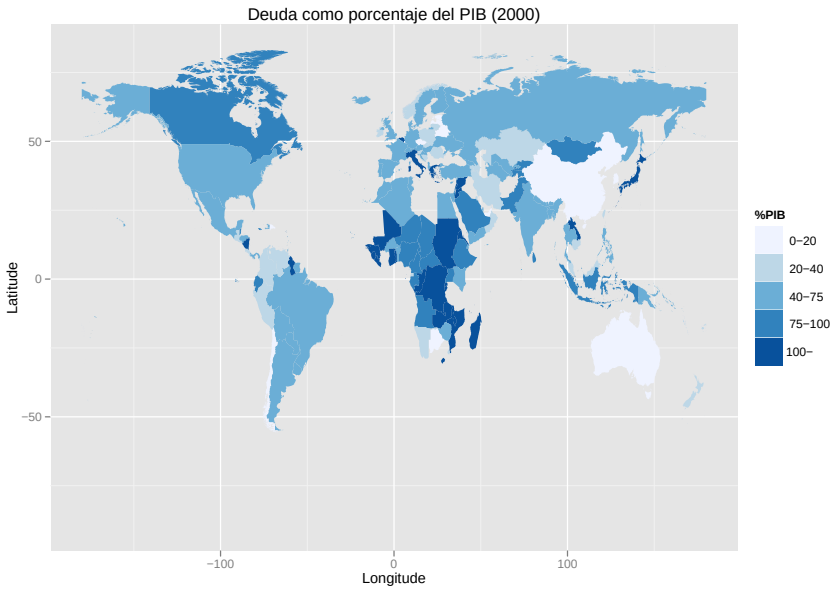
```
> library("rgdal")
> library("maptools")
> library("rgeos")
```

Pa graficar la deuda pública como por centaje del PIB para los diferentes países que se encuentran en la base de datos se puede ejecutar el siguiente código:

```
> world.map <- readOGR(dsn="Ubicación archivo TM_WORLD_BORDERS-0.3",
+     layer="TM_WORLD_BORDERS-0.3")
> world.ggmap <- fortify(world.map, region = "NAME")
> mapa1<-read.csv("2000N.csv", sep = ";", dec = ".")
> names(mapa1) <- c("id",  "Deuda")
> mapa1$id <- tolower(mapa1$id)
> world.ggmap$id <- tolower(world.ggmap$id)
> world.ggmape <- merge(world.ggmap, mapa1, by = "id", all = TRUE)
> world.ggmape <- world.ggmape[order(world.ggmape$order), ]
> world.plot <- ggplot(data = world.ggmape, aes(x = long, y = lat,
+     group = group))
> mapa2000 <-world.plot + geom_polygon(aes(fill = Deuda )) +
+     labs(x = "Longitude", y = "Latitude", fill = "%PIB") +
+     opts(title = "Deuda como porcentaje del PIB (2000)") +
+     scale_fill_brewer(type="seq")
```

y el resultado es:

Figura 13: **Mapamundi**



Explicación del código que genera la Figura 13

Para realizar este gráfico es necesario primero cargar el archivo donde se encuentra la información relacionada con la ubicación geográfica de los países, esto se realiza en las líneas 1-4. Luego se organizan los datos de forma conveniente para facilitar la realización del mapa (Líneas 5-8). En las líneas 9-10 se determina los datos que se van a graficar y en las restantes líneas se específica el tipo de gráfico, los titulos de los ejes y el color.

5. Comentarios Finales

La visualización de información cada vez más se ha convertido en una necesidad para entender las grandes cantidades de información disponibles. En este documento presentamos una breve introducción al paquete ggplot de R, paquete que permite producir gráficos de gran calidad de manera rápida y sencilla. Antes de terminar, es importante anotar que la flexibilidad de este paquete es muy grande y lo descrito en este tutorial solo muestra las aplicaciones más sencillas de este paquete. Éste brinda mucha más opciones que pueden ser estudiadas en detalle en la documentación del paquete o en los numerosos blogs que existen sobre el tema.

Referencias

A. Abela. *ADVANCED PRESENTATIONS BY DESIGN*. Pfeiffer, 2008.

J. C. Alonso, A. I. Gallego, y A. M. Rios. *Industrias Culturales de Santiago de Cali: caracterización y cuentas económicas*. 2010.

J. R. Beniger y D. L. Robyn. Quantitative graphics in statistics: A brief history. 1978.

N. Paul, C. William, y B. Thorne. *Estadística para administración y economía*. Pearson Educación S.A., 2008.

E. R. Tufte. *The visual display of quantitative information*. Graphics Press LLC, 2001.

H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. 2009.