

Fastbook 07

Visual Analytics

Histogramas y gráficos de densidad



07. Histogramas y gráficos de densidad

En este fastbook vamos a conocer dos gráficos muy similares, orientados a la misma función, pero con detalles clave que hay que entender para sacar el mejor partido de ellos. Se trata de los **histogramas** y **diagramas de densidad**. De nuevo, daremos pinceladas para que entiendas cuándo es un error utilizarlos y en qué momentos pueden ayudarnos a entender lo que nuestros datos esconden.

De nuevo, utilizaremos R. Te recuerdo que lo idóneo es que ‘piques’ el código y pruebes todo lo posible, ¡equivocarse y experimentar es parte del aprendizaje también!

Autor: Daniel Pegalajar Luque

[Histogramas: descripción](#)

[Histogramas en R](#)

[Diagramas de densidad: descripción](#)

[Diagramas de densidad en R](#)

[Conclusiones](#)

[Bibliografía](#)

Histogramas: descripción

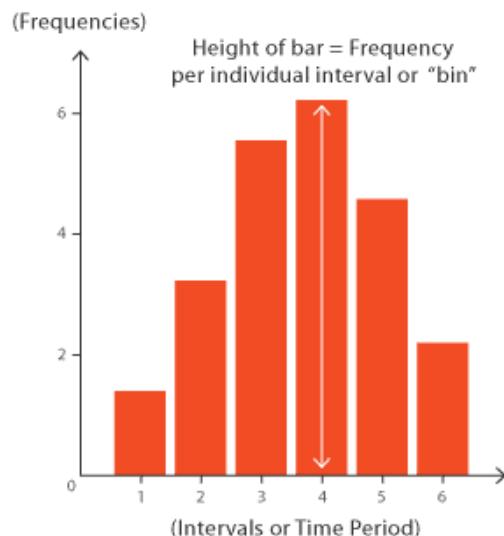
X Edix Educación

Un histograma es una **representación visual** de la **distribución** de una **variable numérica**.

Este tipo de gráficos está enfocado a entender la frecuencia de distribución que se encuentra en los datos de manera subyacente, es decir, la **distribución de probabilidad de una variable continua**.

Más elaborado que un diagrama de cajas y bigotes y con una apariencia similar a los gráficos de barras, hace que mucha gente tienda a confundirlos. Debemos este tipo de gráficos al excelente matemático y bioestadístico inglés, [Karl Pearson](#) (seguro que te suena de otros conceptos como el *Análisis de Componentes Principales* o la prueba χ^2 de Pearson).

Para profundizar en este gráfico veamos un ejemplo de histograma:



Fuente: [DataVizCatalogue](#)

Un histograma es una forma de resumir la información de una variable continua generando cortes o segmentos de esta (conocidos en la mayoría de las librerías gráficas como **bins**) y contando el número de observaciones que caen dentro de cada uno de ellos.

En este tipo de gráficos, **el analista decide el grosor de esos bins, factor clave para la visualización** resultante. Incluso podemos generar *insights* muy diferentes. Ten en cuenta, por tanto, la importancia de este parámetro.

Los histogramas te ayudarán a estimar dónde se concentran los valores de un conjunto de **datos**, en qué zona se sitúan los extremos o cuándo se producen gaps extraños en la distribución o valores inusuales o erróneos. Por si todo lo anterior fuese poco, **te permite estimar visualmente la forma de la distribución de los datos**. Con todas estas ventajas, el histograma se convierte en una poderosa herramienta de análisis y visualización.

Histogramas en R

X Edix Educación

Cómo ya es tradición con R, cargamos algunas opciones para trabajar más ágilmente. En esta ocasión, añadimos el conjunto de datos [midwest](#).

```
# Desactivamos la notación científica. ¿A quién le gusta ver en sus gráficos números como
# 1e25?
options(scipen = 999)

# Cargamos las librerías necesarias para pintar
library(ggplot2) # Nuestra biblia a partir de ahora
library(scales) # Nos ayudará a mejorar el aspecto de nuestros gráficos

library(tidyverse) # Necesario si queremos realizar algún tratamiento en los datos

# Establecemos un tema por defecto para nuestros gráficos
# Personalmente soy fanático de theme_bw(), es el tema clásico 'dark-on-light'
# ggplot ofrece un listado de temas completos que puedes aprovechar. Echa un vistazo:
# https://ggplot2.tidyverse.org/reference/ggtheme.html

# Hay gente que realiza sus propios temas, generando auténticas obras de arte, ¿te atreves?
theme_set(theme_bw())

# Cargamos los datos que vamos a utilizar

data("diamonds", package = "ggplot2")
data("txhousing", package = "ggplot2")
data("midwest", package = "ggplot2")
```

Comencemos con el código más básico, como ocurría con el boxplot: **solo necesitamos una variable continua para poder pintar un histograma**. En este caso usaremos el área de las regiones.

```
## Tu primer histograma en R

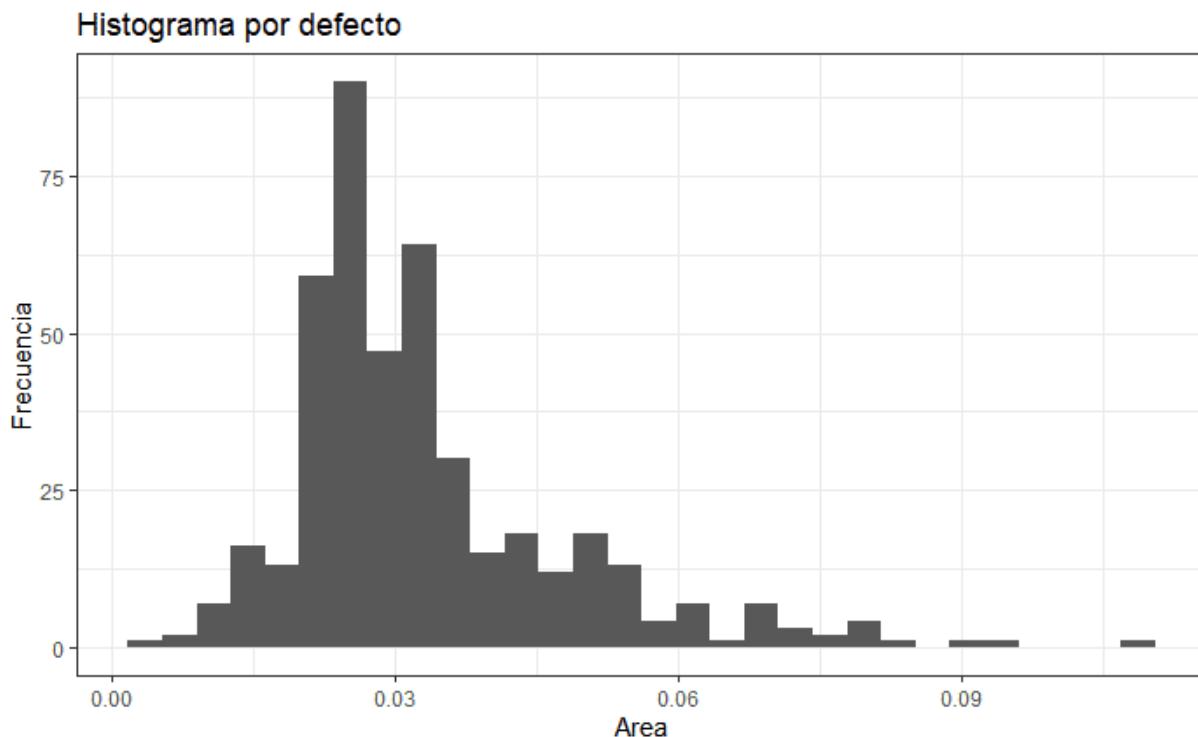
gg ← ggplot(midwest, aes(x = area)) +
  geom_histogram() +
  labs(title = "Histograma por defecto",
       x = "Área",
       y = "Frecuencia")

gg
```

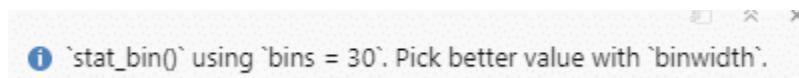
Este conjunto de datos contiene información sobre los condados en varios estados del medio oeste de los Estados Unidos. Como es de esperar, estos condados varían en extensión, por lo que un histograma nos servirá para entender la **distribución de estas áreas geográficas**.

El área se mide en millas cuadradas y con este primer gráfico ya podemos obtener bastante información:

- Los condados van desde las 0,005 hasta las 0,11 millas cuadradas, esto es de 12.950 a los 285.000 metros cuadrados.
- La moda se sitúa en torno al valor 0,025, aunque existe un segundo valor destacado en 0,032.
- A partir de ese núcleo central de condados, la distribución va perdiendo fuerza conforme aumenta el tamaño de estos, llegando a existir algunos valores atípicos e incluso gaps entre valores.



Si has lanzado el histograma siguiendo los pasos del código de arriba, habrás obtenido un aviso o *warning* como este:



Por defecto, `geom_histogram()` **selecciona 30 como los bins a generar**. Como dijimos al inicio, este parámetro es realmente importante y elegir un valor adecuado puede ayudar mucho.

Para seleccionar el tamaño de los bins existen una gran cantidad de reglas, vamos a ver las dos más importantes:

1

Basándose en la regla del pulgar (*rule of thumb*), conocida técnicamente como la **regla de Sturges**:

```
```{r}
sturges ← function(x) {
 n=length(x)
 ceiling(log(n,2))+1
}

sturges(midwest$area)```
[1] 10
```

Podemos encapsular en una función dicha regla para facilitar su uso. Esta función se basa en la fórmula:

$$k = \lceil \log_2 n \rceil + 1$$

Donde k es igual al número de bins. Esta regla no siempre da unos buenos resultados, de hecho, **algunos expertos aconsejan su uso cuando el tamaño del dataset no excede las 200 observaciones** (midwest tiene 437).

2

Otra alternativa es la regla de [Freedman-Diaconis](#), que otorga un ancho de bin en lugar de un **número de segmentos**:

$$h = 2r/n^{1/3}$$

Donde  $r$  es el rango intercuartílico de los datos y  $n$  el número de observaciones.

```
```{r}
fd ← function(x) {
  2 * IQR(x) / length(x)^(1/3)
}

fd(midwest$area)

[1] 0.00369
```

Esta regla es **muy robusta y suele funcionar muy bien en la práctica**.

Si utilizamos los dos casos anteriores, podemos obtener dos gráficos distintos al inicial y que nos puede servir para entender la importancia de este parámetro.

```
k ← sturges(midwest$area)

gg ← ggplot(midwest, aes(x = area)) +
  geom_histogram(bins = k) +
  labs(title = "Histograma con la regla de Sturges",
       x = "Area",
       y = "Frecuencia")

gg

h ← fd(midwest$area)

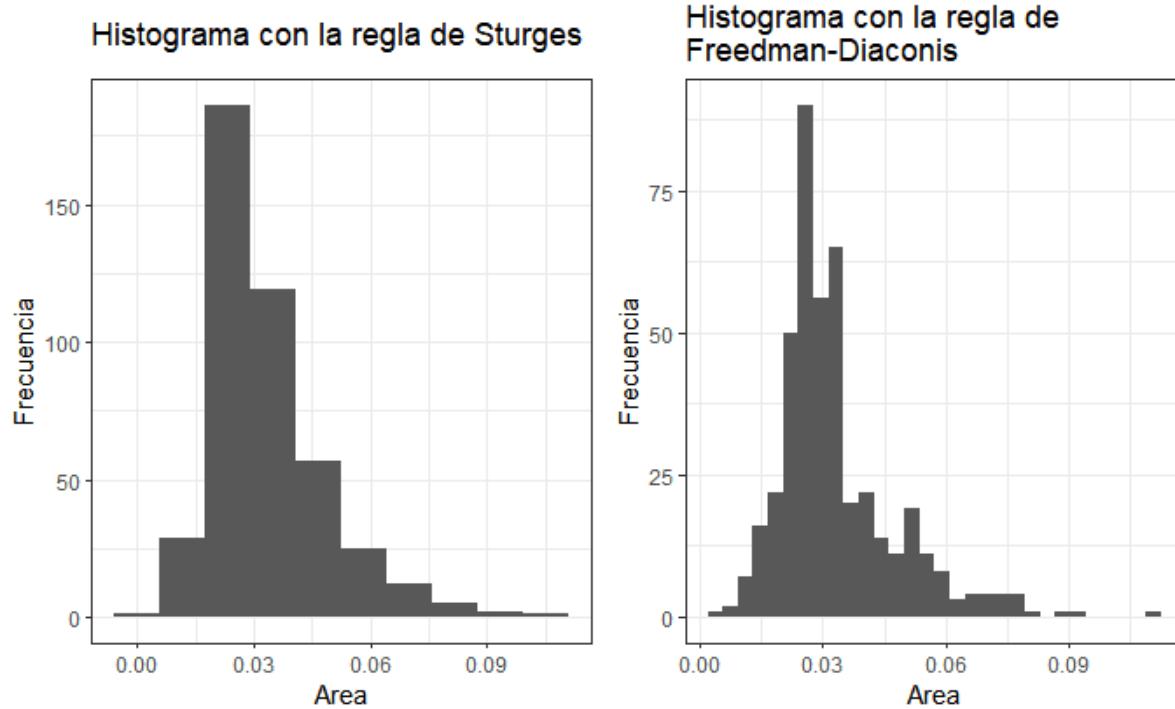
gg2 ← ggplot(midwest, aes(x = area)) +
  geom_histogram(binwidth = h) +
  labs(title = "Histograma con la regla de \nFreedman-Diaconis",
       x = "Area",
       y = "Frecuencia")

gg2
```

Observa la diferencia entre ambos: en uno definimos el número de bins con el argumento bins; en el otro caso, usamos el ancho de bin con el argumento binwidth.

Una vez que tenemos creados ambos gráficos, podemos combinarlos de la siguiente forma:

```
library(patchwork)  
gg + gg2
```



¿Observas las diferencias?

Antes de entrar a comentar estos gráficos, déjame que te enseñe que esta combinación de visualizaciones es gracias a la librería [patchwork](#). Esta librería permite **combinar diferentes visualizaciones de una forma sencilla y con un lenguaje trivial** (por ejemplo, en este caso con un símbolo + se unifican gráficos creados por diferentes vías). ¡Úsala y crea dashboards a tu gusto!

La regla de Freedman-Diaconis es mucho más fina y permite desglosar mejor la distribución, descubriendo esos dos picos que conforman una bimodal. En este caso, Sturges es mucho más tosca y con mayor nivel de agregación.

Como siempre, estas son reglas matemáticas que no tienen por qué adaptarse a tus necesidades. Experimenta con alguno de los dos argumentos anteriores y genera el resultado más apropiado para lo que buscas transmitir.

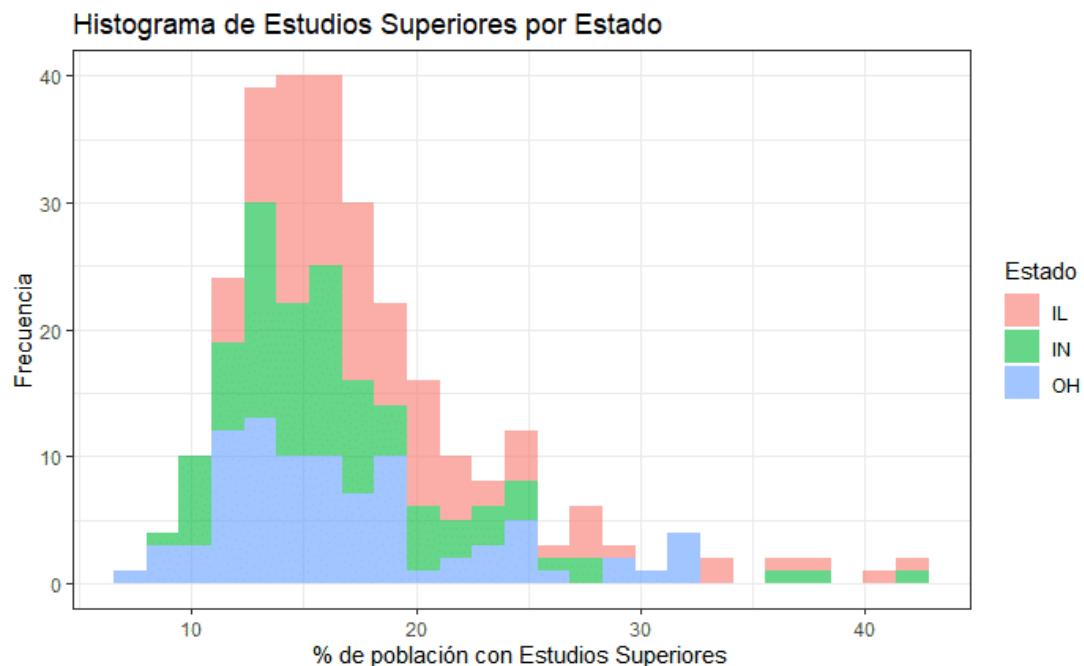
Al igual que en fastbooks anteriores, vamos a aprovechar el potencial de añadir otras dimensiones para extraer nuevos aprendizajes.

Comenzamos seleccionando una serie de estados de los 5 posibles (IL, IN, MI, OH o WI, te los dejo por si quieras probar alternativas). Con nuestra selección fijada, **filtramos el dataset original en base a ello**.

A partir de ahí, **configuramos el mapeo de la información**: vamos a estudiar el % de población con estudios superiores por estado, la primera será nuestra variable X y el estado se encargará de ‘rellenar’ de color cada histograma.

Otorgamos algo de transparencia a cada histograma para visualizarlos mejor y fijamos el número de bins (puedes hacer tus pruebas como deseas en estos dos parámetros).

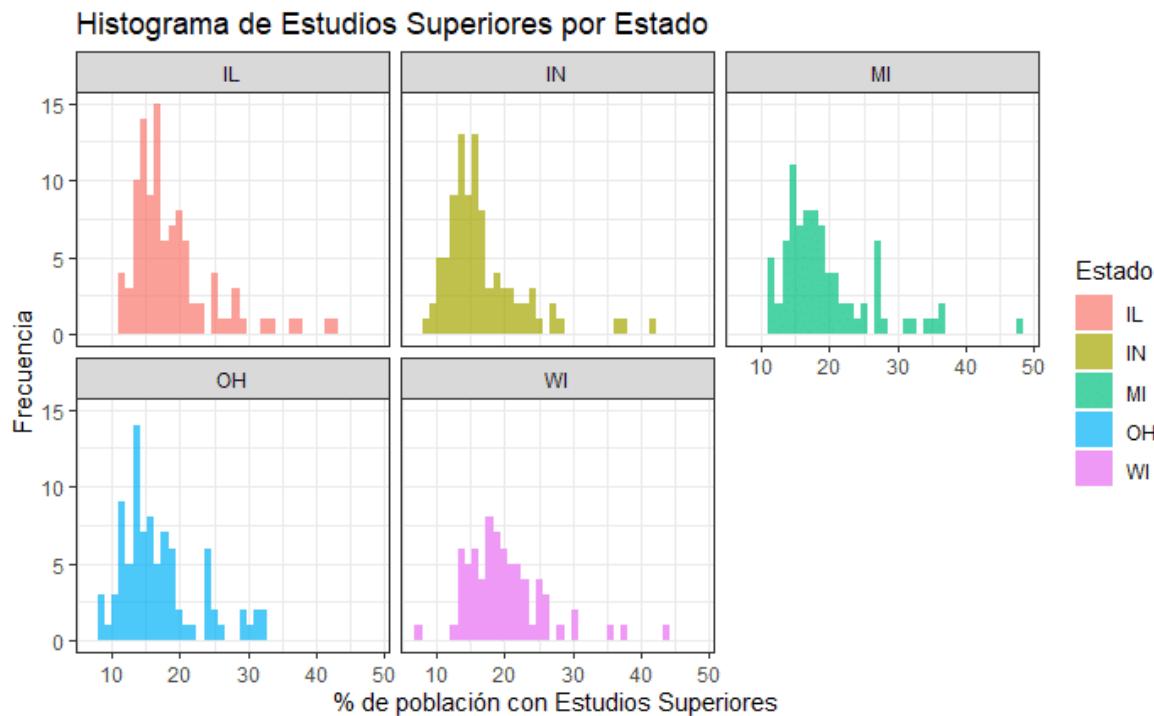
```
## Utilizando las dimensiones a tu favor
estados <- c("OH", "IL", "IN")
gg <- ggplot(midwest %>% filter(state %in% estados), aes(x = percollege, fill = state)) +
  geom_histogram(alpha = 0.6, bins = 25) +
  labs(title = "Histograma de Estudios Superiores por Estado",
       x = "% de población con Estudios Superiores",
       y = "Frecuencia",
       fill = "Estado")
```



Recuerda que anteriores fastbooks aprendiste una herramienta que te permite separar visualizaciones en diferentes paneles y mantener una presentación limpia: los **facets**.

```
## Utilizando las dimensiones a tu favor
gg ← ggplot(midwest, aes(x = percollege, fill = state)) +
  geom_histogram(alpha = 0.7, bins = 40) +
  facet_wrap(~state) +
  labs(title = "Histograma de Estudios Superiores por Estado",
       x = "% de población con Estudios Superiores",
       y = "Frecuencia",
       fill = "Estado")
gg
```

Gracias al uso de facets podemos tener todos los estados a la vez y comparar todas las distribuciones de una vez. Aún así, recuerda que una opción alternativa a esta ejecución sería utilizar **gráficos de violín**.

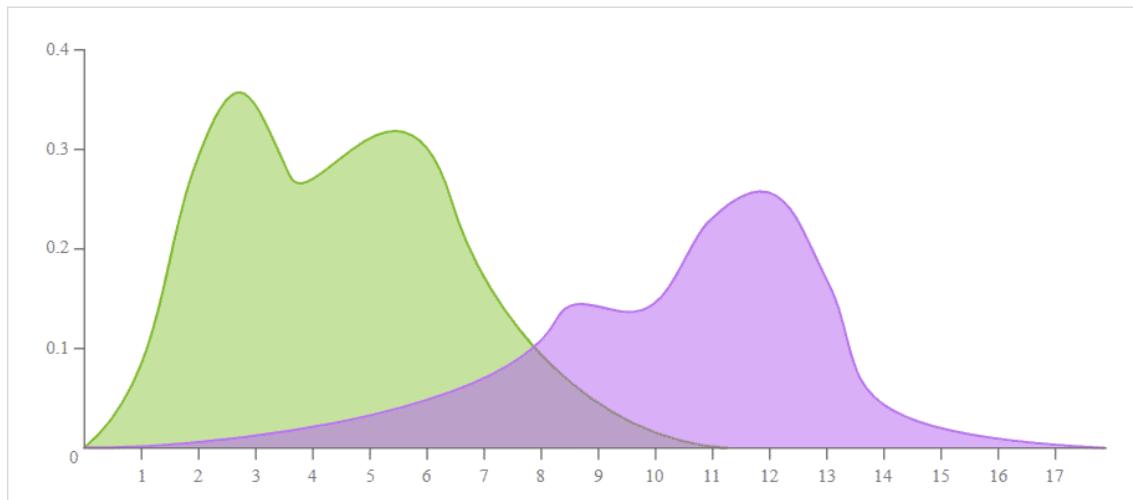


Diagramas de densidad: descripción

X Edix Educación

Los **diagramas de densidad**, también conocidos como **gráficos de núcleo de distribuciones** (en español suena a física nuclear, en inglés es más atractivo: **Kernel Density Plots**), son unos gráficos orientados a visualizar la distribución de los datos en un intervalo continuo o de tiempo.

Como puedes apreciar en la imagen de abajo, son una variación de los histogramas que utilizan un suavizado estadístico para representar los valores. Este suavizado **elimina o disminuye el posible ruido de valores atípicos** y sigue cumpliendo bien su papel de mostrar los ‘picos’ de la distribución de la información.



Fuente: [DataVizCatalogue](#).

El eje X de la visualización representa los valores de la variable a ilustrar y la Y, conforma la estimación de densidad, la distribución de probabilidad basada en los datos observados en dicha variable (probabilidad básica).

Estos gráficos representan una ventaja sobre los histogramas a la hora de mostrar la forma real de la distribución de los datos ya que, a diferencia de los primeros, **no necesitan especificar el tamaño de los bins o segmentos, no se ven afectados por este problema** (ya sabes cómo cambia la perspectiva según la elección de este parámetro).

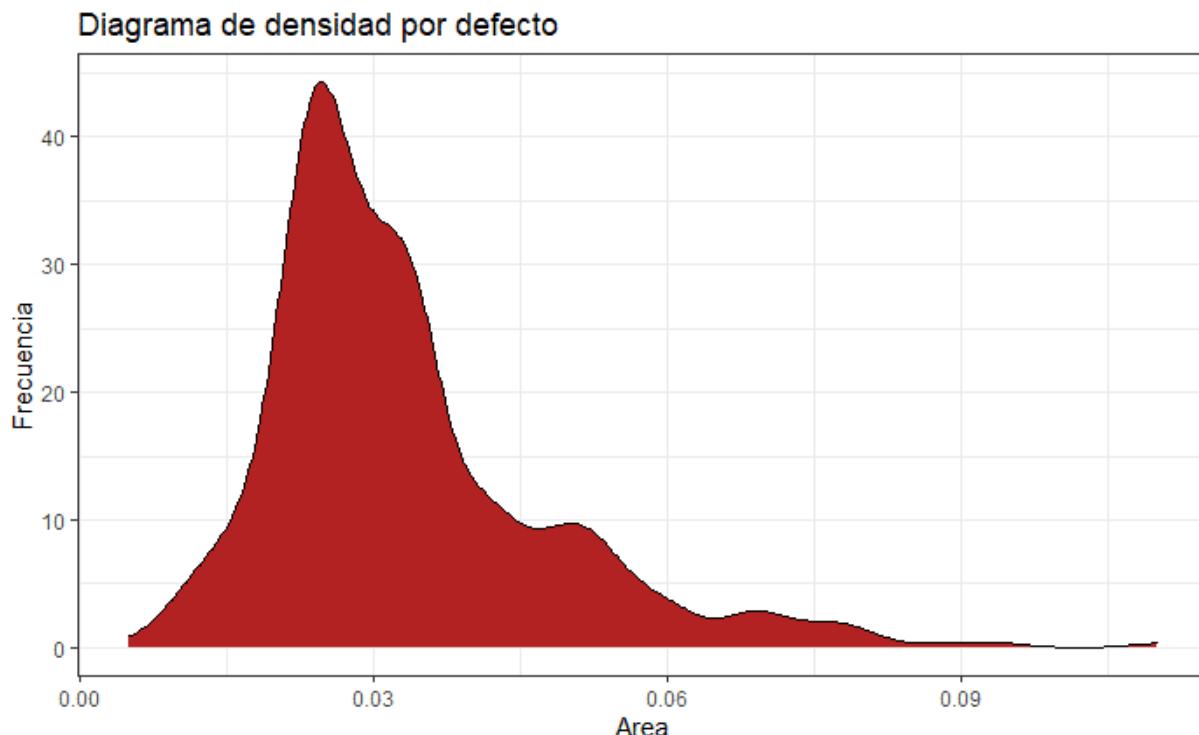
Esta visualización puede usarse en las mismas condiciones que un histograma, por lo tanto, recuerda **no utilizarla cuando vayas a pintar cinco o más diagramas de densidad en la misma visualización**. En ese caso, es mejor que **alternes con los facets o gráficos de violín** para obtener una visualización más limpia o fácil de leer.

Diagramas de densidad en R

X Edix Educación

Continuamos desde el ejemplo anterior, por lo tanto, se da por hecho la carga inicial de datos y librerías para poder practicar. Vamos a seguir trabajando con el conjunto de datos `midwest` para que puedas comparar fácilmente las diferencias entre ambas visualizaciones:

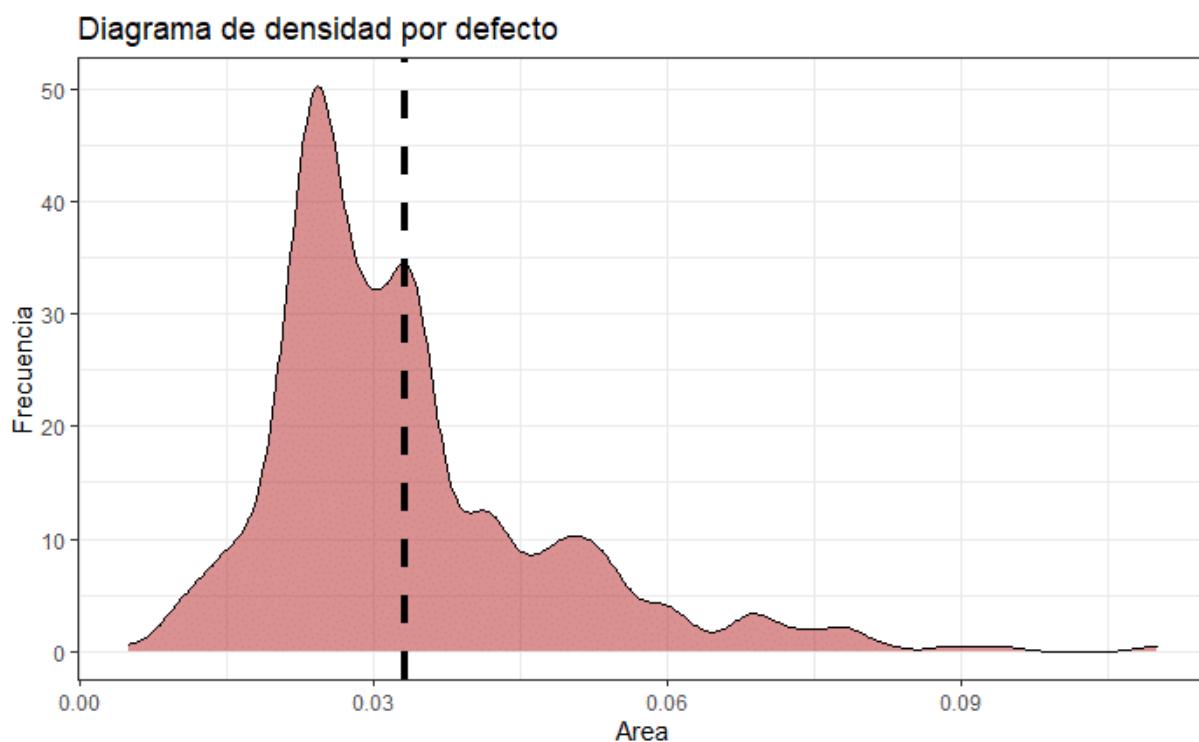
```
## Tu primer diagrama de densidad en R
gg ← ggplot(midwest, aes(x = area)) +
  geom_density(fill = "firebrick") +
  labs(title = "Diagrama de densidad por defecto",
       x = "Area",
       y = "Frecuencia")
gg
```



La función para poder generar diagramas de densidad es el *geom_density()*. Puedes especificar un color de relleno con el argumento *fill* dentro del mismo geom. Si no lo haces, el diagrama estará vacío y solo verás la distribución a través del contorno superior.

Por último, recuerda que el eje que utilices marcará la orientación del diagrama, es decir, en lugar de X usa Y (para este caso no es útil pero no te vendrá mal saberlo).

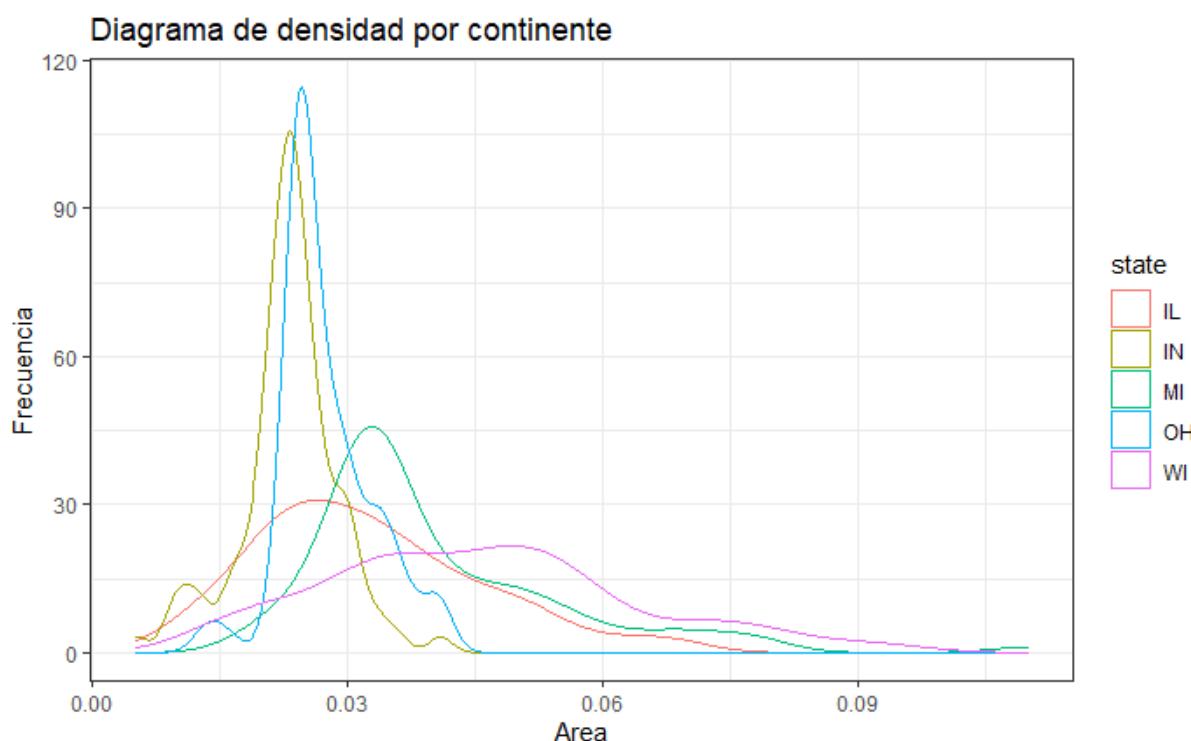
```
## Añadiendo la media
gg <- ggplot(midwest, aes(x = area)) +
  geom_density(fill = "firebrick", alpha = 0.5, adjust = 0.7) +
  geom_vline(aes(xintercept=mean(area)),
             color="black", linetype="dashed", size=1.5) +
  labs(title = "Diagrama de densidad por defecto",
       x = "Area",
       y = "Frecuencia")
gg
```



Podemos explorar más opciones:

- El argumento *alpha* permite añadir transparencia al color seleccionado. Realmente útil cuando se superponen varios diagramas de densidad.
- El argumento *adjust* permite modificar el suavizado de la distribución. Por debajo de 0 captará más ruido procedente de los datos, por encima de 1 aplicará un suavizado superior. Experimenta con diferentes valores para descubrir los efectos.
- Podemos apoyarnos en otros geoms como *geom_vline()* para superponer la media de la distribución. Este tipo de acciones nos permite fraccionar la distribución a placer.

```
## Cortando la distribución por continente
gg ← ggplot(midwest, aes(x = area, colour = state)) +
  geom_density(alpha = 0.8) +
  labs(title = "Diagrama de densidad por continente",
       x = "Area",
       y = "Frecuencia")
gg
```

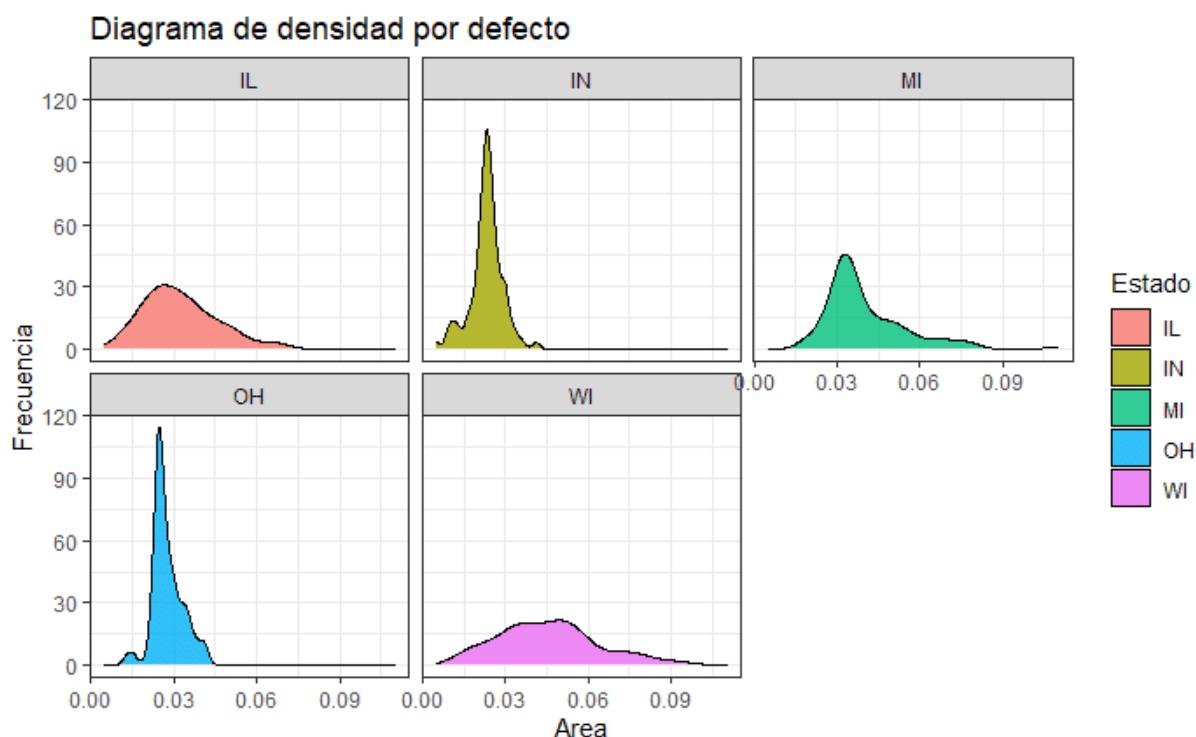


Mediante el uso de argumentos adicionales como `colour` o `fill` podemos usar una variable categórica (estado al que pertenece la observación) para diseccionar a la distribución original en múltiples visualizaciones.

Sin embargo, al igual que con los histogramas, **no se recomienda pintar cinco o más diagramas de densidad en la misma visualización**, ya que puede ser engoroso y dificultar la lectura de información. La imagen anterior es el mejor ejemplo posible, ¿no crees?

De todas formas, ya has aprendido otras alternativas en anteriores fastbooks para lidiar con este problema: los **facets**.

```
gg ← ggplot(midwest, aes(x = area, fill = state)) +  
  geom_density(alpha = 0.8) +  
  facet_wrap(~state) +  
  labs(title = "Diagrama de densidad por defecto",  
       x = "Area",  
       y = "Frecuencia",  
       fill = "Estado")  
  
gg
```

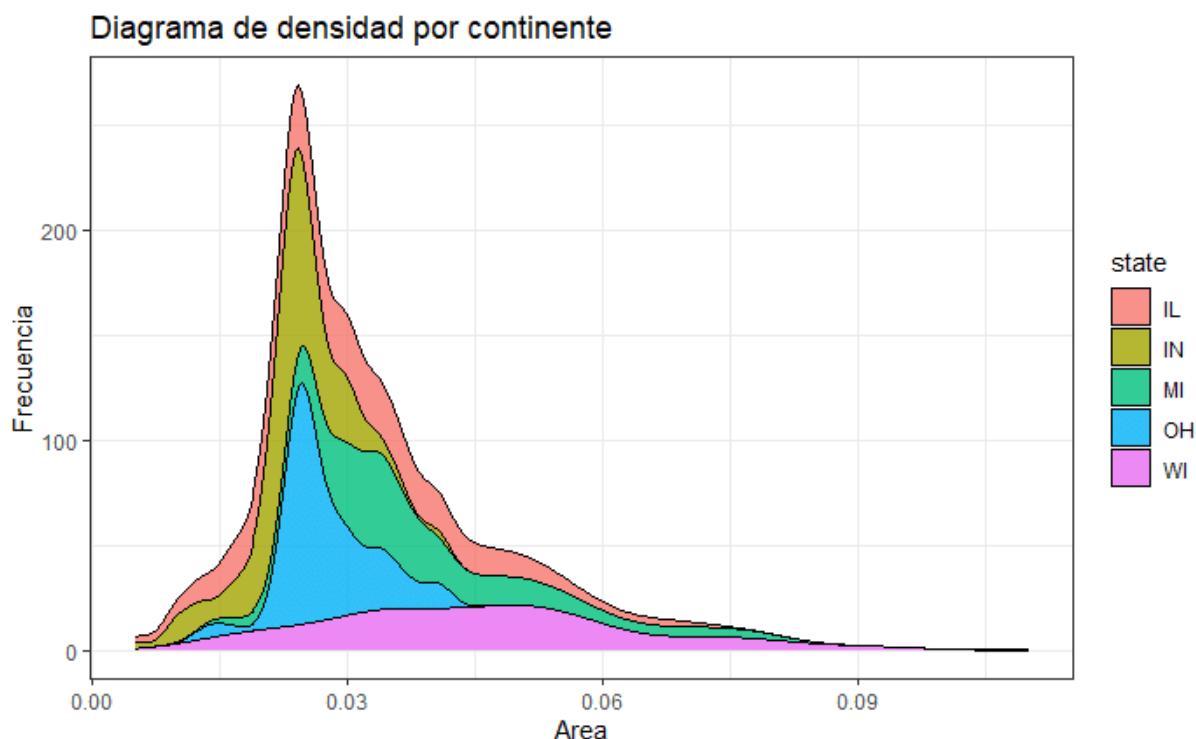


Al romper cada estado en su cuadrícula es mucho más sencillo entender y sacar conclusiones. En este ejemplo, apreciamos que Indiana y Ohio tienen una distribución similar, mientras que Michigan o Wisconsin son los estados que poseen áreas más extensas en cuanto a tamaño.

Existen alternativas que nos proporciona el propio *geom_density()*:

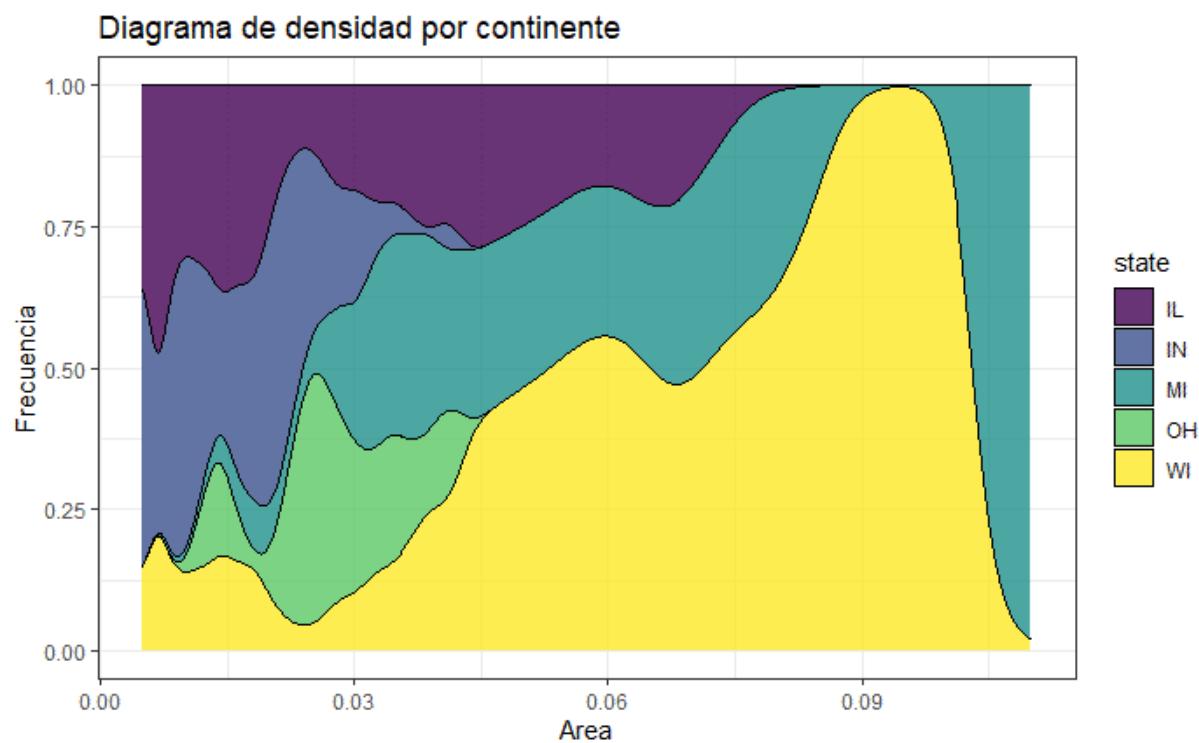
Podemos **stackear** o apilar las diferentes densidades para extraer las conclusiones de una forma alternativa.

```
## Cortando la distribución por continente
gg <- ggplot(midwest, aes(x = area, fill = state)) +
  geom_density(alpha = 0.8, position = "fill") +
  scale_fill_viridis_d() +
  labs(title = "Diagrama de densidad por continente",
       x = "Area",
       y = "Frecuencia")
gg
```



Otra posibilidad es usar el argumento *fill* en lugar de *stack* para obtener una visualización alternativa: comprime toda la densidad entre 0 y 1 y los intervalos posibles que pueda tomar la variable numérica. A partir de este punto, distribuye en función de la variable categórica.

```
## Cortando la distribución por continente  
gg <- ggplot(midwest, aes(x = area, fill = state)) +  
  geom_density(alpha = 0.8, position = "fill") +  
  scale_fill_viridis_d() +  
  labs(title = "Diagrama de densidad por continente",  
       x = "Area",  
       y = "Frecuencia")  
gg
```



¿Qué opción te resulta más fácil de leer?

Conclusiones

X Edix Educación

En este fastbook hemos conocido dos visualizaciones muy similares en apariencia, pero con detalles y claves que las diferencian notablemente: hablamos de los **histogramas** y los **diagramas de densidad**. Estos gráficos **nos permiten entender la distribución de la probabilidad** que subyace detrás de un conjunto de datos, nos ayuda a detectar outliers de una forma rápida y sencilla e, incluso, a visualizar posibles gaps que nos alerten de errores o sucesos a tener en cuenta.

Hemos aprovechado otros conceptos aprendidos en los fastbooks anteriores, por ejemplo: el uso de colores como dimensión adicional o la potencia de los facets para mantener una visualización clara.

Como siempre, te recomiendo encarecidamente que pruebes los resultados en cada opción y no te quedes solo con la lectura del tema. Esta ciencia se aprende con práctica, así que, adelante, intenta picar el código para afianzar lo aprendido.

¿Te atreves a utilizar esta visualización con algún conjunto de datos propio? Prueba y extrae conclusiones.

Bibliografía

X Edix Educación

- [Ggplot2 – Página principal.](#)
- [Data Visualization de Kieran Healy.](#)

¡Enhorabuena! Fastbook superado

edix

Creamos Digital Workers