

## GENERANDO GRÁFICOS DEL COVID-19 EN LA REPÚBLICA DOMINICANA USANDO R: EL PAQUETE DRCOVIDPLOTS

FIDEL MORLA<sup>a</sup> Y JOHAN ROSA<sup>b</sup>

Recibido: 1/5/2020 • Aprobado: 31/7/2020

**Cómo citar:** Morla, F., & Rosa, J. (2020). Generando gráficos del COVID-19 en la República Dominicana usando R: el paquete drcovidplots. *Ciencia, Economía y Negocios*, 4(2), 33-50. Doi: <https://doi.org/10.22206/ceyn.2020.v4i2.pp33-50>

### Resumen

*El paquete drcovidplots en R tiene diversas funciones para mostrar datos de forma gráfica del COVID-19 en República Dominicana. La información nacional proviene del Ministerio de Salud Pública y Asistencia Social, mientras que la información a nivel global es extraída de Worldometer<sup>1</sup>.*

**Palabras clave:** COVID19; programación; R-Studio; análisis gráfico.

**Códigos JEL:** C88, Y10, I10.

---

<sup>a</sup> Banco Central de la República Dominicana, Santiago de los Caballeros, República Dominicana.  
Correo-e: [f.morla@bancentral.gov.do](mailto:f.morla@bancentral.gov.do)

<sup>b</sup> Banco Central de la República Dominicana, Santiago de los Caballeros, República Dominicana.  
Correo-e: [j.rosa@bancentral.gov.do](mailto:j.rosa@bancentral.gov.do)

1. <https://www.worldometers.info/>



## GENERATING CHARTS TO ANALYZE COVID-19 IN THE DOMINICAN REPUBLIC USING R: THE DRCOVIDPLOTS PACKAGE

FIDEL MORLA AND JOHAN ROSA

Received: 1/5/2020 • Approved: 31/7/2020

### Abstract

*The R package drcovidplots has various functions to display graphically COVID-19 data in the Dominican Republic. Domestic information is mined from the Ministry of Public Health and Social Assistance, while global information is extracted from Worldometer.*

**Keywords:** COVID19; programming; R-Studio; graphic analysis.

**JEL Codes:** C88, Y10, I10.

## 1. Introducción

En estadística, una representación gráfica es una visualización de datos en forma de diagrama o gráfico. Es decir, representa un conjunto de datos de manera significativa. Además, observar información de forma gráfica puede ayudar a tomar mejores decisiones (Nussbaumer, 2015).

Usualmente, los economistas utilizan el método de representación gráfica ya que es efectivo para mostrar y resumir grandes cantidades de datos, y es útil para mostrar tendencias, patrones y relaciones entre variables. Adicionalmente, se pueden tomar decisiones comparando el valor o el costo, discutiendo algún informe de cómo se presentó una situación específica en el pasado respecto al presente. Considerando que este método facilita la interpretación de los datos, tiene muchas ventajas en los campos de la estadística y la economía. Por esto, los gráficos y cuadros son actualmente componentes fundamentales de la investigación moderna y la presentación de informes.

De esta forma, teniendo en cuenta la situación pandémica que vive el mundo producto del coronavirus y su consecuente enfermedad COVID-19, este documento presenta una herramienta de análisis gráfico con información del Ministerio de Salud Pública y Asistencia Social de la República Dominicana (MSPAS). Este instrumento es el paquete de R llamado drcovidplots, el cual simplifica esta tarea con las funciones que incluye. Solo se necesita un conocimiento básico de programación en R para ejecutar el paquete y reproducir los gráficos.

## 2. Instalación

**La versión de desarrollo de drcovidplots puede instalarse desde GitHub desde el indicador de la consola R con:**

```
install.packages("devtools")
devtools::install_github("fidelmorla/drcovidplots")
library("drcovidplots")
Todas las dependencias faltantes deben instalarse automáticamente.
```

### 3. Uso

#### 3.1 Cargando la información

Para exemplificar el uso de drcovidplots, utilizamos la información disponible en el boletín especial número 91 del MSPAS. Además, debe mencionarse que la información está en archivos con valores separados por comas en la página del repositorio en Github<sup>1</sup>. Por esta razón, es imprescindible tener conexión a internet a la hora de cargar la información con la función que se muestra más abajo.

Para cargar estos datos, se debe correr el siguiente código en la consola de R:

```
load_data_covid_dr()
```

Esta función creará los siguientes objetos:

**Tabla 1.** Objetos creados por la función load\_data\_covid\_dr()

Objeto	Clase del objeto	Contenido
data_cum	data.frame	Información agregada por fecha
data_density		Densidad poblacional por provincia
data_province		Positivos, muertes, recuperados y pruebas por provincia
data_sex		Positivos por sexo al último reporte disponible
data_type		Distribución por tipo de aislamiento o estado de los positivos
East	character	Vector con las abreviaciones para las provincias de la Región Este
Metropolitan		Vector con las abreviaciones para las provincias de la Región Metropolitana
North		Vector con las abreviaciones para las provincias de la Región Norte
South		Vector con las abreviaciones para las provincias de la Región Sur
NE		Vector con la abreviación para la fila de No especificado
rep_actual	integer	Número entero indicando el número del último boletín especial del MSPAS

**Fuente:** elaboración propia.

1 [https://github.com/fidelmorla/drcovidplots/tree/master/csv\\_data](https://github.com/fidelmorla/drcovidplots/tree/master/csv_data)

### 3.2 Funciones

La utilización de las funciones es relativamente sencilla. Todas las funciones tienen los siguientes dos argumentos:

**Tabla 2.** Argumentos más recurrentes en las funciones

Argumento	Descripción
saveplot	Lógico. ¿Debería guardar el objeto ggplot en .GlobalEnv? Predeterminado FALSE.
savepng	Lógico. ¿Debería guardar una versión png del gráfico? Predeterminado FALSE.

**Fuente:** elaboración propia.

Las únicas excepciones son: la función `g_cases_province()`, la cual incluye un argumento adicional `n_province` para determinar el número de provincias con las cuales se generará el gráfico. Esta función representa gráficamente, en orden descendente, las provincias dadas en `n_province`. De forma predeterminada, la función grafica las quince provincias con más casos de COVID-19 partiendo de la información del último boletín del MSPAS.

La otra función que cuenta con otros argumentos es la `g_map_covid()`. Esta cuenta con:

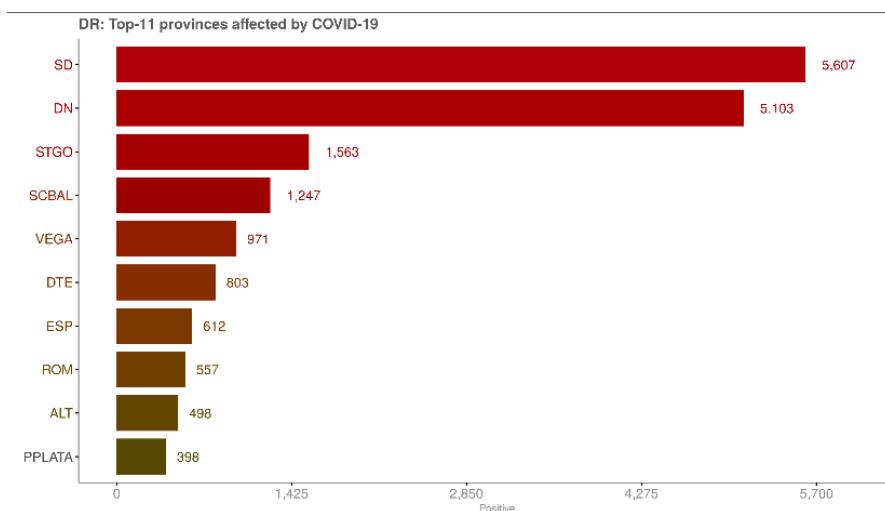
- i) el argumento `date`, para indicar la fecha para introducir en el mapa, predeterminado la última fecha disponible (“latest”),
- ii) el argumento `variable`, para indicar cuál variable utilizar. Tiene tres posibilidades: “Cases”, “Deaths” y “Recovered”,
- iii) el argumento `by_inhabitants`, una variable lógica que indica si crear el mapa teniendo en cuenta la población de cada provincia. De forma predeterminada es “TRUE”.

### 3.3 Ejemplos

Ahora exemplificaremos el uso de algunas funciones. Debe mencionarse que para el siguiente ejercicio se asume que se ejecutaron las funciones descritas en la parte 3.1.

Por ejemplo, utilizaremos la función `g_cases_province()` con dos opciones para el número de provincias, dígase 11 y 22.

```
> g_cases_province(n_province = 11, savepng = TRUE)
```

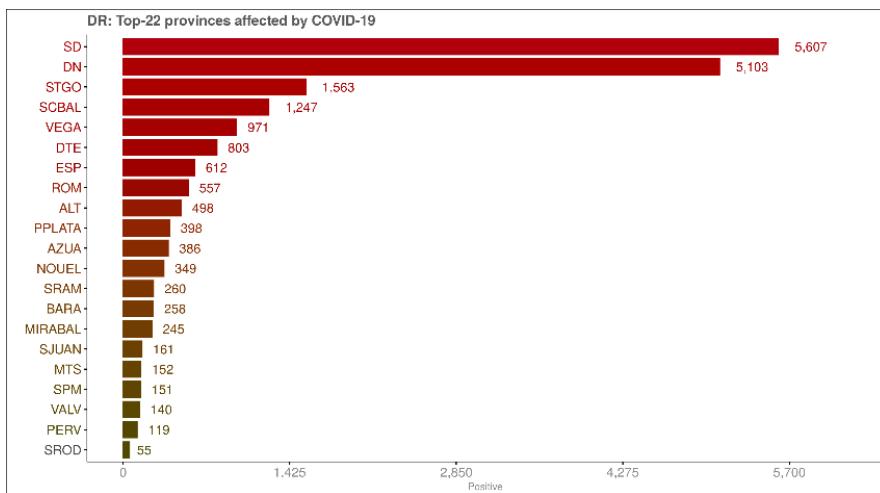


**Figura 1.** Las 11 provincias con mayor cantidad de casos de COVID19 en RD

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico

```
> g_cases_province(n_province = 22, savepng = TRUE)
```



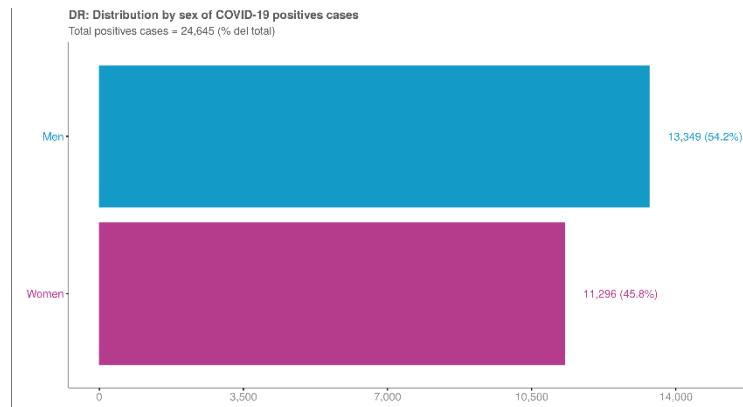
**Figura 2.** Las 22 provincias con mayor cantidad de casos de COVID19 en RD.

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico

Otras funciones:

```
> g_sex(savepng = TRUE)
```



**Figura 3.** Distribución por sexo de los casos positivos de COVID19

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico

> g\_recovered\_province(savepng = TRUE)

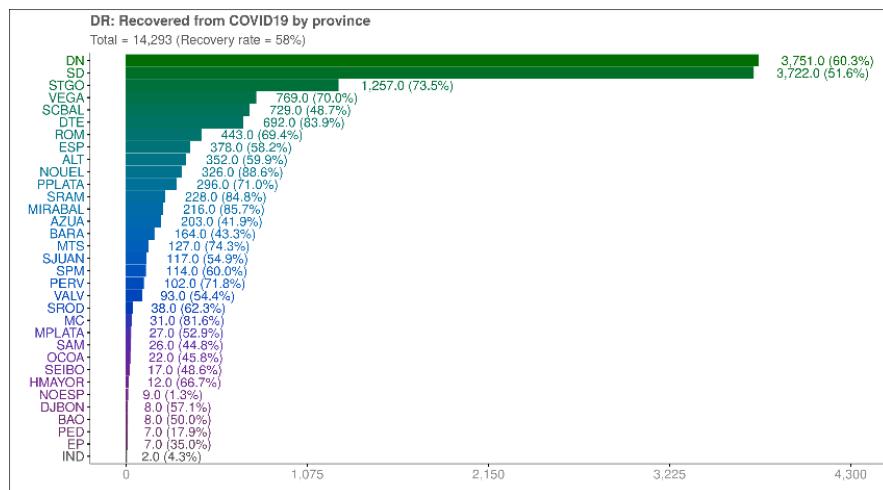


Figura 4. Recuperados de COVID19 por provincia

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico

> g\_death(savepng = TRUE)

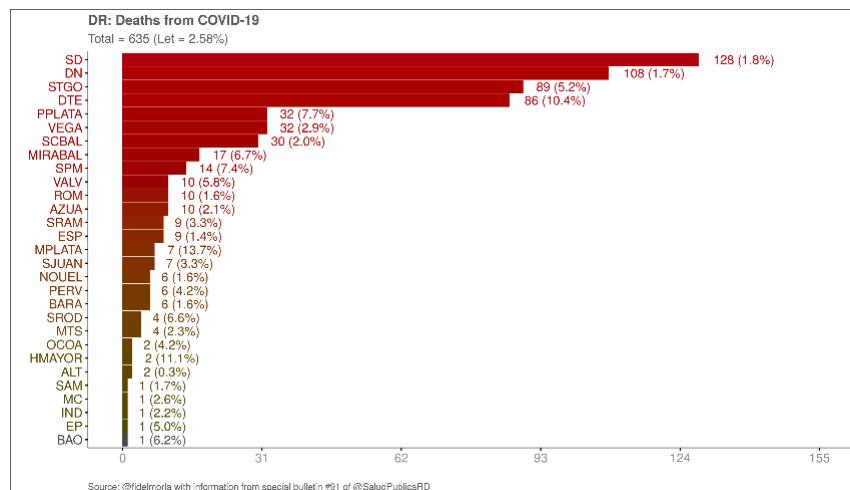
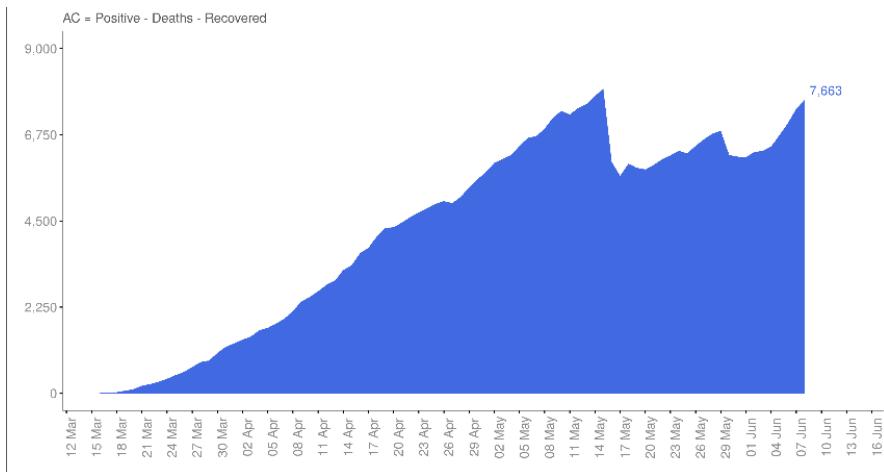


Figura 5. Decesos por COVID19

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico

```
> g_active_cases(savepng = TRUE)
```

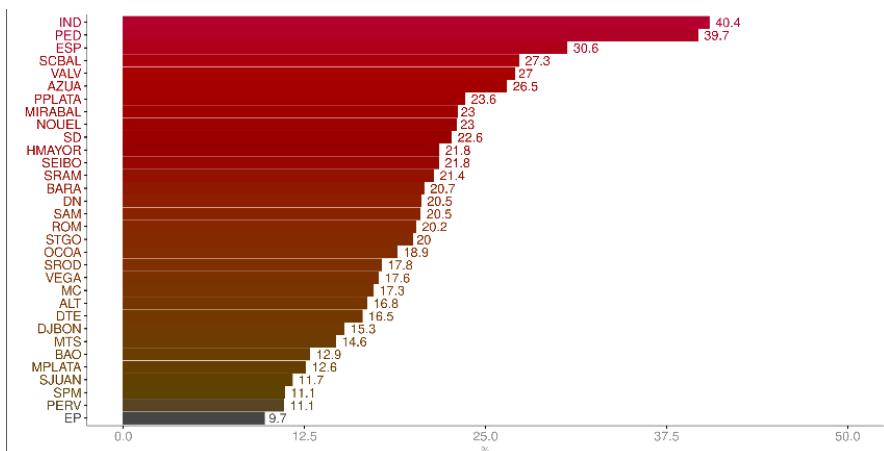


**Figura 6.** Casos activos de COVID19

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico

```
> g_positive_province(savepng = TRUE)
```

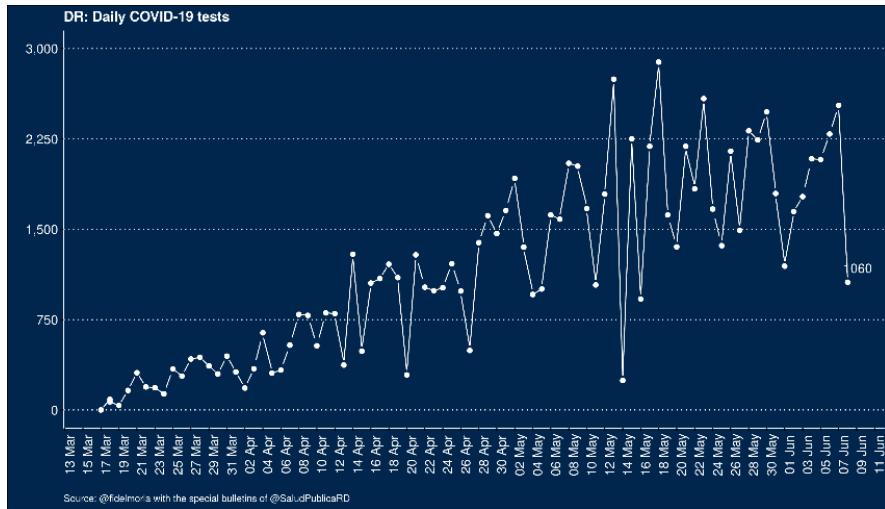


**Figura 7.** Positividad de COVID19 por provincia

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico. La positividad es el ratio entre los casos positivos y el número de pruebas realizadas.

> g\_tests(savepng = TRUE)



**Figura 8.** Pruebas diarias de COVID19

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

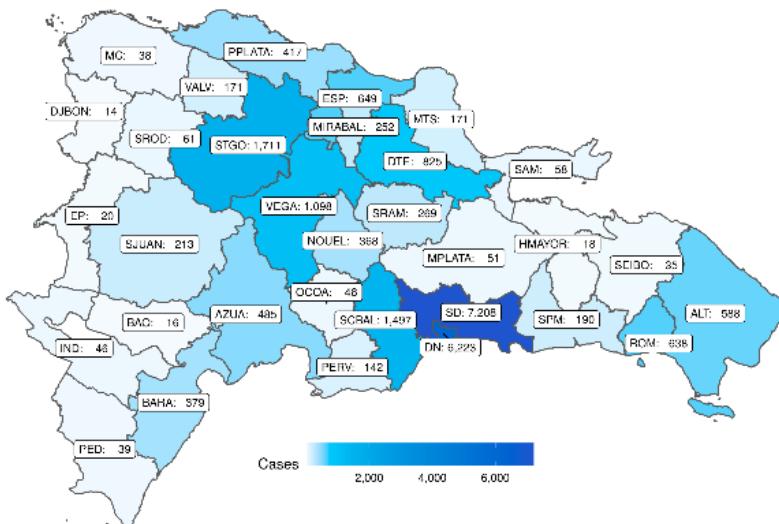
**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico.

Esta última función genera un total de cuatro gráficos, todos concernientes a las pruebas realizadas:

- i. Pruebas diarias
- ii. Pruebas acumuladas
- iii. Pruebas diarias por cada millón de habitantes
- iv. Pruebas acumuladas por cada millón de habitantes

```
> g_map_covid(variable = 'Cases', savepng = TRUE)
```

DR: Cases of COVID19 by province  
Total = 24,645



**Figura 9.** Mapa de casos activos de COVID19 por provincia

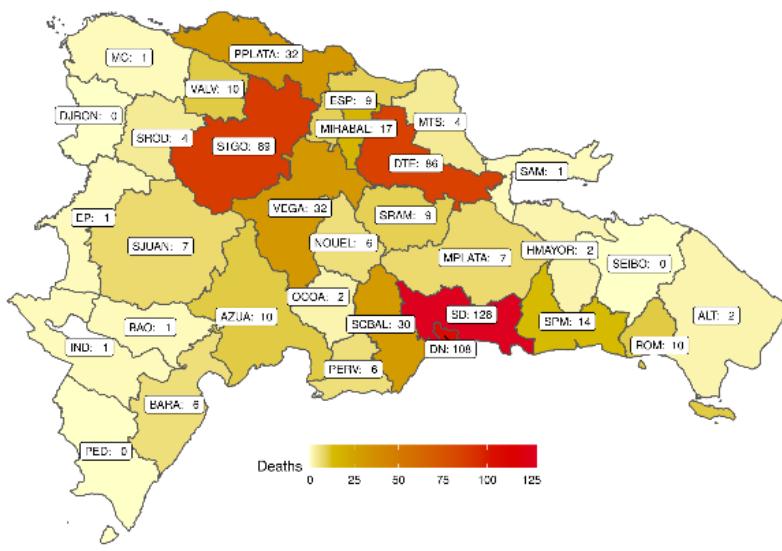
**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico.

```
> g_map_covid(variable = 'Deaths', savepng = TRUE)
```

DR: Deaths by COVID19 by province

Total = 635



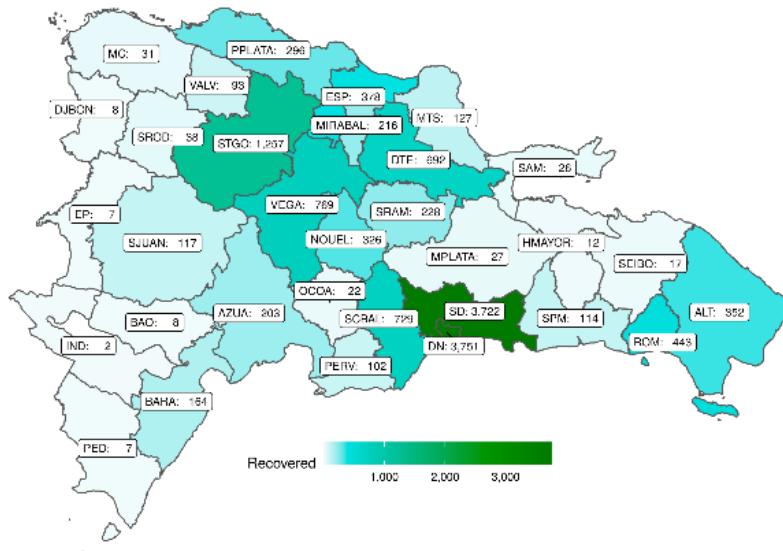
**Figura 10.** Mapa de decesos por COVID19 por provincia

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico

```
> g_map_covid(variable = 'Recovered', savepng = TRUE)
```

DR: Recovered from COVID19 by province  
Total = 14,293



**Figura 10.** Mapa de recuperados de COVID19 por provincia

**Fuente:** Información disponible en el boletín especial del MSPAS número 91.

**Nota.** Se utiliza el argumento de guardado en png en todas las funciones para generar el gráfico

#### 4. Conclusión

Este documento se dedica principalmente a explicar el funcionamiento del paquete drcovidplots, el cual proporciona varias funciones para graficar diferentes variables sobre la situación del COVID19 en República Dominicana. Esta herramienta de análisis gráfico tiene 27 funciones que utilizan información de los boletines especiales del MSPAS. Cabe la posibilidad de que la herramienta continúe evolucionando, a medida que surja más información del COVID19 en República Dominicana. Por tal razón, siempre recomendamos actualizar el paquete ocasionalmente.

Los códigos del paquete están disponibles en el enlace <https://github.com/fidelmorla/drcovidplots>. Además, los usuarios que utilicen el paquete y encuentren problemas o dificultades pueden reportarlas en la página <https://github.com/fidelmorla/drcovidplots/issues>.

## Cita oficial

Para citar el paquete “drcovidplots” en publicaciones, use:

Morla, F. (2020). *drcovidplots: Plots for COVID19 data from Dominican Republic*. R package version 0.1.11. <https://github.com/fidelmorla/drcovidplots>.

## Referencias

- Arnold, J. B. (2019). *ggthemes: Extra Themes, Scales and Geoms for ‘ggplot2’*. R package version 4.2.0. Recuperado de <http://github.com/jrnold/ggthemes>.
- Auguie, B. (2017). *gridExtra: Miscellaneous Functions for “Grid” Graphics*. R package version 2.3.
- Grolemund, G., & Wickham, H. (2011). Dates and Times Made Easy with lubridate. *Journal of Statistical Software*, 40(3), 1-25. Recuperado de <http://www.jstatsoft.org/v40/i03/>.
- Henry, L. & Wickham, H. (2020). *purrr: Functional Programming Tools*. Recuperado de <http://purrr.tidyverse.org>, <https://github.com/tidyverse/purrr>.
- Lin, T., & Robinson, D. (2020). *ggridge: A Grammar of Animated Graphics*. Recuperado de <https://ggridge.com>, <https://github.com/thomasp85/ggridge>.
- McKinnon, S. E. (2020). *lemon: Freshing Up your ‘ggplot2’ Plots*. R package version 0.4.4. Recuperado de <https://github.com/stefanewards/lemon>.
- Milton, S. & Wickham, H. (2014). *magrittr: A Forward-Pipe Operator for R*. R package version 1.5.
- Nussbaumer, C. (2015). *Storytelling with Data: A Data Visualization Guide for Business Professionals*. John Wiley & Sons: New York.

- Ooms, J. (2018). *gifsiki: Highest Quality GIF Encoder*. Recuperado de <https://gif.ski/> (upstream), <https://github.com/r-rust/gifsiki> (devel).
- Pebesma, E. (2018). Simple Features for R: Standardized Support for Spatial Vector Data. *The R Journal*, 10(1), 439-446, doi: <https://doi.org/10.32614/RJ-2018-009>
- R Core Team. (2020). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. Recuperado de <https://www.R-project.org/>.
- Rudis, B., Bolker, B. & Schulz, J. (2017). *ggalt: Extra Coordinate Systems, 'Geoms', Statistical Transformations, Scales and Fonts for 'ggplot2'*. R package version 0.4.0. Recuperado de <https://github.com/hrbrmstr/ggalt>
- Slowikowski, K. (2020). *ggrepel: Automatically Position Non-Overlapping Text Labels with 'ggplot2'*. R package version 0.8.2. Recuperado de [http://github.com/slowkow/ggrepel](https://github.com/slowkow/ggrepel)
- Stengel, D., Calori, G. & Giannoudis, P. (2008). Graphical data presentation. *Injury*, 39(6), 659-665.
- Temple, D. (2020). *XML: Tools for Parsing and Generating XML Within R and S-Plus*. R package version 3.99-0.3. Recuperado de <http://www.omegahat.net/RXML>
- Wickham, H. & Henry, L. (2020). *tidy: Tidy Messy Data*. <https://tidy.tidyverse.org>, Recuperado de <https://github.com/tidyverse/tidy>.
- Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. New York: Springer-Verlag.
- Wickham, H. (2019). *rvest: Easily Harvest (Scrape)*. Recuperado de <http://rvest.tidyverse.org/>, <https://github.com/tidyverse/rvest>.
- Wickham, H. (2019). *stringr: Simple, Consistent Wrappers for Common String Operations*. Recuperado de <http://stringr.tidyverse.org>, <https://github.com/tidyverse/stringr>.
- Wickham, H. (2020). *forcats: Tools for Working with Categorical Variables (Factors)*. Recuperado de <http://forcats.tidyverse.org>, <https://github.com/tidyverse/forcats>.
- Wickham, H., Hester, J., & Ooms, J. (2020). *xml2: Parse XML*. Recuperado de <https://xml2.r-lib.org/>, <https://github.com/r-lib/xml2>.

- Wickham, H., François, R., Henry, L., & Müller, K. (2020). *dplyr: A Grammar of Data Manipulation. R package version 1.0.0*. Recuperado de <https://CRAN.R-project.org/package=dplyr>
- Zeileis, A., Fisher, J.C., Hornik, K., Ihaka, R., McWhite, C.D., Murrell, P., Stauffer, R., & Wilke, C.O. (2019). *colorspace: A Toolbox for Manipulating and Assessing Colors and Palettes*. *arXiv 1903.06490, arXiv.org E-Print Archive*. Recuperado de <http://arxiv.org/abs/1903.06490>.

**Anexo 1.** Lista de funciones disponibles en drcovidplots

Función	Descripción
g_active_cases	Casos activos de COVID19
g_cases_province	Provincias con la mayoría de los casos positivos de COVID19
g_death	Número total de muertes
g_density	Correlación entre los casos de COVID19 y la densidad poblacional
g_evolution_covid	Evolución de casos nuevos y casos totales de COVID19
g_growth_cases	Tasa de crecimiento de casos positivos de COVID19
g_hospital	Personas en aislamiento hospitalario por COVID19
g_incidence	Incidencia de COVID19
g_incidence_region	Tasa de incidencia y letalidad por región de COVID19
g_l2	Tasa de letalidad (L2) de Ghani <i>et al.</i> (2005) de COVID19
g_log_nc	Casos nuevos versus casos totales en logaritmos de COVID19
g_map_covid	Mapa dominicano con casos de COVID-19 por provincia
g_np	Casos negativos y positivos de COVID19
g_np_daily	Casos diarios negativos y positivos de COVID19
g_positive	Positividad nacional del COVID19
g_positive_province	Positividad por provincia de COVID19
g_positive_region	Positividad por región de COVID19
g_recovered	Recuperados de COVID19
g_recovered_daily	Número diario de personas recuperadas de COVID19
g_recovered_growth	Tasa de crecimiento de los recuperados de COVID19
g_recovered_province	Recuperado por provincia de COVID19
g_region	Casos COVID por región
g_sex	Distribución por sexo de casos positivos de COVID19
g_status	Distribución por estado de individuos afectados de COVID19
g_tests	Pruebas de COVID19
g_world	Comparación mundial de la situación de COVID19
load_data_covid_dr	Carga las bases de datos

**Anexo 2.** Lista de paquetes utilizados para la creación de drcovidplots

Paquete	Autor(es)
ggplot2	Wickham, H. (2016)
dplyr	Wickham, H., R. François, L. Henry & K. Müller (2020)
tidyr	Wickham, H. & L. Henry (2020)
purrr	Henry, L. & H. Wickham (2020)
stringr	Wickham, H. (2019)
magrittr	Milton, S. & H. Wickham (2014)
ggridanimate	Lin Pedersen, T. & D. Robinson (2020)
lubridate	Grolemund, G. & H. Wickham (2011)
XML	Temple Lang, D. (2020)
rvest	Wickham, H. (2019)
xml2	Wickham, H., J. Hester & J. Ooms (2020)
grDevices	R Core Team (2020)
ggalt	Rudis, B., B. Bolker & J. Schulz (2017)
ggthemes	Arnold, J.B. (2019)
gridExtra	Auguie, B. (2017)
grid	R Core Team (2020)
forcats	Wickham, H. (2020)
gifski	Ooms, J. (2018)
colorspace	Zeileis, A., J. C. Fisher, K. Hornik, R. Ihaka, C. D. McWhite, P. Murrell, R. Stauffer, C. O. Wilke (2019)
ggrepel	Slowikowski, K. (2020)
lemon	McKinnon, S.E. (2020)
sf	Pebesma, E. (2018)