

R en la generación de informes automáticos a través de la API de Google Analytics

Miguel Boubeta, Eva María González
María Elena Naranjo, José Manuel Pérez



25 de Octubre 2018
Santiago de Compostela

Índice

- 1 Introducción
- 2 Generación de informes con R
- 3 Conclusiones

Section 1 | Introducción

Introducción

Sobre Improving Metrics ...

- ▶ Trabajamos con nuestros clientes en más de 50 países.

ZARA



Bershka

Stradivarius

PULL&BEAR

ZARA
HOME

OYSHO

UTERQÛE

REALES
SEGUROS

luckia

Pepe Jeans
LONDON

HACKETT
LONDON

janssen | PHARMACEUTICAL COMPANIES
in collaboration

randstad

campus

ABANCA

- ▶ La analítica digital como herramienta de negocio.



Introducción

Servicios analítica digital

1 Analítica web y móvil:

- ▶ Definición de indicadores de rendimiento (KPIs).
- ▶ Volumetrías de tráfico y análisis de contenidos.
- ▶ Canales de adquisición (modelos de atribución y contribución).
- ▶ Análisis de contrataciones e hitos (conversión).
- ▶ Perfilado y segmentación de audiencias.
- ▶ Análisis multiplataforma/multidispositivo.

2 Analítica de entornos sociales:

- ▶ Clasificación de las emociones y del sentimiento.
- ▶ Análisis de las comunicaciones en entornos sociales.
- ▶ Gestión de crisis: análisis de incidencias en entornos sociales.

Introducción

Google Analytics: Herramienta de Google que ofrece información de tráfico de sitios web.

Ventajas: Se pueden obtener informes de:

- ▶ Seguimiento de usuarios exclusivos.
- ▶ Rendimiento del segmento de usuarios.
- ▶ Análisis de las diferentes campañas de marketing online.
- ▶ Sesiones por fuentes de tráfico.
- ▶ Tasas de rebote.
- ▶ Duración de las sesiones, contenidos visitados, conversiones.

Inconvenientes:

- ▶ Rigidez.
- ▶ Aplicación de muestreo.
- ▶ Se malgasta tiempo para ejecutar informes similares.

Section 2 | Generación de informes con R

El paquete googleAnalyticsR

El paquete *googleAnalyticsR* permite obtener datos de la API de Google Analytics.

```
google_analytics(viewId, date_range = NULL, metrics = NULL,  
  dimensions = NULL, filtersExpression = NULL,  
  segments = NULL, anti_sample = TRUE)
```

Observaciones:

- ▶ *viewId* no admite un vector.
- ▶ Puede arrojar errores por alcanzar los parámetros máximos permitidos o por autenticación.

Procesado de datos

Lectura de los datos:

- ▶ *fread* del paquete *data.table*.
- ▶ *read.xlsx* del paquete *xlsx*.

Procesado de fechas (*lubridate*). Se han creado funciones para calcular:

- ▶ Días naturales.
- ▶ Días comerciales.

Procesado de strings:

- ▶ *grep*, *grepl*, *regexpr*, *gregexpr* y *regexecdel* paquete base.
- ▶ Paquete *stringr*.

Procesado de los datos (*data.table*):

- ▶ *merge* para la unión de data frames.
- ▶ Agregaciones por variables categóricas.

```
dtAggr <- dt[, .(sessions = sum(sessions)),  
              by = .(deviceCategory)]
```

Formato

| | A | B |
|----|--------|-------|
| 1 | Tit | a |
| 2 | Blanco | a |
| 3 | Tit1 | a |
| 4 | h1 | a |
| 5 | h2 | a |
| 6 | h3 | a |
| 7 | col1 | 0 |
| 8 | col4 | +0% |
| 9 | cab1 | a |
| 10 | cab2 | a |
| 11 | Tot1 | a |
| 12 | col2 | 0 |
| 13 | col3 | 0,00% |
| 14 | . | . |

Figure 1: Estilos.

Formato del informe (xlsx):

- ▶ Añadir un data frame.

```
addDataFrame(x, sheet, ...)
```

- ▶ Concatenar celdas.

```
addMergedRegion(sheet, startRow, endRow, startColumn, endColumn)
```

- ▶ Establecer los estilos.

```
darEstilo = function (sheet, estilos, rows, cols, nombre){
```

```
  cells = getCells(getRows(sheet, rows), cols)
```

```
  for (i in 1:length(cells)){
```

```
    cells[[i]]$setCellStyle(obtEstilo(estilos, nombre))
```

```
  }
```

```
}
```

Formato final

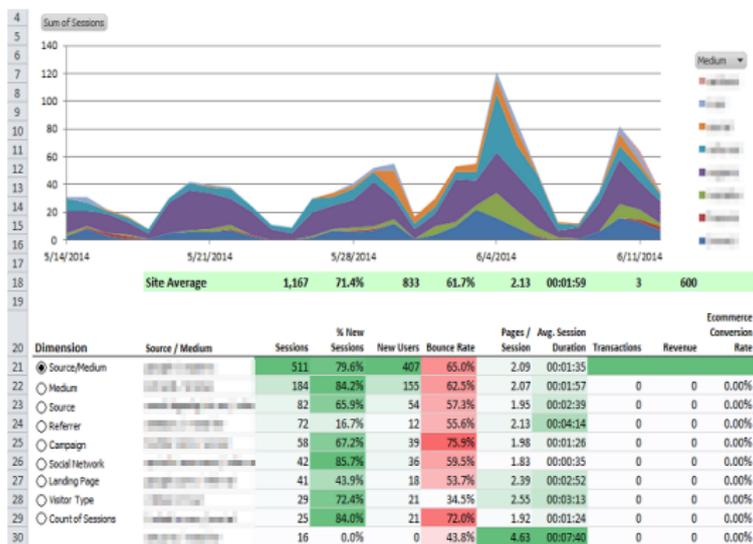
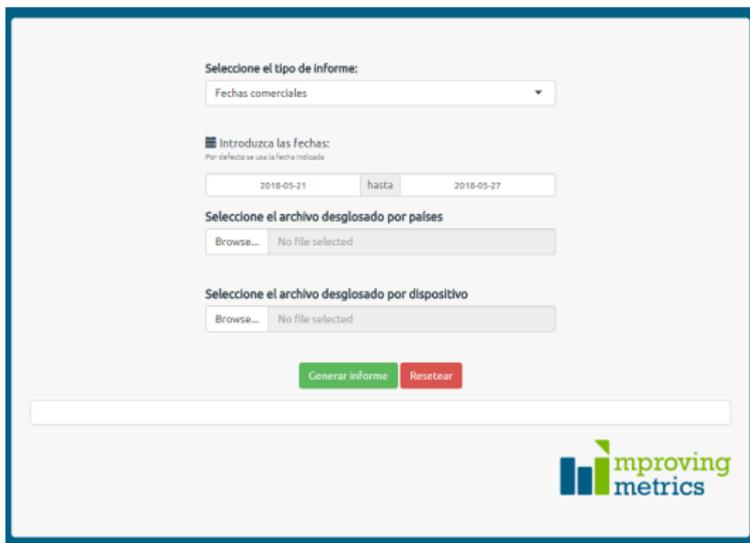


Figure 2: Ejemplo de informe.

Automatización

Aplicación shiny



The screenshot shows a web application interface with the following elements:

- Selección del tipo de informe:** A dropdown menu with "Fechas comerciales" selected.
- Introduzca las fechas:** A section with a calendar icon and the text "Por defecto se usa la fecha indicada". It contains two date input fields: "2018-05-21" and "2018-05-27", separated by the word "hasta".
- Selección del archivo desglosado por países:** A "Browse..." button next to the text "No file selected".
- Selección del archivo desglosado por dispositivo:** A "Browse..." button next to the text "No file selected".
- Buttons:** A green "Generar informe" button and a red "Resetear" button.
- Footer:** A white horizontal bar and the "improving metrics" logo (three blue bars of increasing height and the text "improving metrics").

Figure 3: Ejemplo de aplicación shiny.

Automatización

Aplicación shiny

- ▶ Control de versiones Git.
- ▶ Ejecutable de la aplicación shiny.
 - i) R portable.
 - ii) Google Chrome portable (opcional).
 - iii) Archivo *.vbs*.

Randomize

```
CreateObject("Wscript.Shell").Run  
"R-Portable\App\R-Portable\bin\R.exe  
CMD BATCH --vanilla --slave runShinyApp.R"  
& " " & RND & " ", 0, False
```

- iv) Archivo *.R*.

```
options(encoding = 'UTF-8')  
.libPaths("./R-Portable/App/R-Portable/library")  
shiny::runApp("./App-D/", launch.browser=TRUE)
```

Automatización

Tareas programadas

- ▶ taskscheduleR (Windows), cronR (Linux).

```
cmd <- cron_rscript(rscript = "script.R")

## Every day.
cron_add(cmd, frequency='daily', at='8:00', id='id')

## Get all the jobs.
# cron_ls()
```

Automatización

Descarga de archivos adjuntos:

método *AdvancedSearch* de la librería *RDOMClient*.

Envío del informe por mail (*mailR*):

```
send.mail(from = mailAttach.dt$user,  
         to = "miguelbm@improvingmetrics.com",  
         subject = "asunto",  
         body = "cuerpo del mail",  
         html = TRUE,  
         smtp = list(host.name = "smtp.office365.com",  
                    port = 587,  
                    user.name = mailAttach.dt$user,  
                    passwd = mailAttach.dt$password,  
                    tls = TRUE),  
         authenticate = TRUE,  
         attach.files = "Informe.xlsx")  
)
```

Section 3 | Conclusiones

Conclusiones

- ▶ Se usa el software R para generar informes.
- ▶ Se mejora la función `google_analytics` para conectarse a la API de Google Analytics.
- ▶ Se crean aplicaciones shiny o tareas programadas para la generación de informes automatizados.

R en la generación de informes automáticos a través de la API de Google Analytics

Miguel Boubeta, Eva María González
María Elena Naranjo, José Manuel Pérez



25 de Octubre 2018
Santiago de Compostela