

[alicer: Creando un paquete con soluciones analíticas para Walmart Chile](#)

En el equipo de Customer Analytics de Walmart Chile hemos desarrollado un paquete de R llamado `alicer` (inspirados por la experiencia del paquete `Rbnb` de AirBnb¹) con el fin de acelerar la entrega de análisis y productos digitales basados en datos, mejorar la calidad de estos, y facilitar la inducción de nuevos miembros al equipo.

Algunos de los beneficios más importantes que hemos conseguido en nuestro equipo gracias a la incorporación de un paquete de R interno en nuestro flujo de trabajo son:

[Facilitar la conexión y consulta de múltiples fuentes de datos](#)

Nuestro paquete incluye funciones que hacen más rápida y fácil la conexión a distintas fuentes de datos (Data Warehouse, bases de datos transacciones y APIs de servicios como Salesforce). Tenemos listados como dependencias tanto el paquete `DBI`, como los paquetes que proveen de drivers para conectarse a cada una de estas fuentes (`odbc`, `RPostgres`, etc). Además, usamos las funciones `ui_` del paquete `usethis` para orientar al usuario sobre como guardar sus credenciales en `.Renviron` cuando se conecta por primera vez. El mapeo de las credenciales como variables de entorno facilita también escribir scripts que puede ejecutarse de forma no supervisada en contenedores Docker.

Asimismo, se han creado funciones que facilitan las consultas SQL desde R, actuando como wrappers de `DBI::dbGetQuery`, pero integrando además funcionalidades como el poder pasar parámetros o variables hacia las queries (mediante `DBI::sqlInterpolate`), leer queries directamente desde un archivo `.sql`, retornar siempre un output con clase `tibble` y, de forma opcional, “cachear” resultados anteriores en un archivo `.rds` para no tener que ejecutar con tanta frecuencia queries pesadas, pero mantener reproducibilidad de los resultados.

[Facilitar la aplicación de técnicas econométricas de evaluación de impacto](#)

En el equipo hemos recibido múltiples preguntas de negocio respecto de cómo impactan ciertas acciones corporativas en la venta de los locales (por ejemplo, inicio de cobro de estacionamientos, remodelaciones, y cese de entrega de bolsas plásticas) y que pueden abordarse mediante la técnica cuasi-experimental de diferencias en diferencias². La aplicación de esta técnica requiere de varios pasos, entre los que se cuenta, en nuestro caso particular, obtener un dataset de venta por local en un formato determinado, calcular una matriz de correlación de pendientes para identificar locales con tendencias paralelas al local de interés, y luego hacer un análisis de regresión creando variables binarias con ciertos criterios.

Para ampliar el uso de esta técnica y permitir que todos los miembros del equipo puedan aplicarla (incluso quienes no tienen experiencia previa con ella) hemos creado una familia de funciones llamada `evdid_` que ayudan al usuario a completar todas las etapas del análisis, entregando resultados intermedios que permiten realizar análisis derivados y validaciones (por ejemplo, excluir de la regresión a locales que sufrieron un evento exógeno por el cual ya no se cumple el supuesto de tendencias paralelas). También se provee de funciones que toman el resultado de la regresión de diferencias en diferencias y crean automáticamente visualizaciones fáciles de leer para mostrarlas a audiencias del negocio y/o incorporarlas en reportes de R Markdown.

¹ <https://peerj.com/preprints/3182.pdf>

² https://es.wikipedia.org/wiki/Diferencias_en_diferencias

Agilizar la manipulación de datos y los análisis exploratorios

Hemos cargado algunas de las tablas de dimensiones más importantes que existen en nuestro Data Warehouse, tales como el árbol de productos vendidos por Walmart y la lista de locales/tiendas (en ambos casos acompañadas de una gran cantidad de columnas o atributos), dentro del directorio `data/` del paquete.

De esta forma, un analista que ha importado datos en R usando una query SQL a un cierto nivel de agregación (por ejemplo, venta a nivel de local) puede rápidamente obtener métricas a nivel de comuna, región, u otra agrupación, sin tener que modificar su query, gracias a la gramática de `dplyr` en conjunto con estos datasets integrados.

Entrega de reportería ad-hoc mediante informes R Markdown parametrizados

Si bien nuestro equipo llevaba varios años usando Tableau para la entrega de múltiples métricas, existía también una gran carga de trabajo debido a solicitudes de informes ad-hoc que requerían escribir queries SQL manualmente, las cuales iban acompañadas de un flujo de trabajo orientado a Excel y PowerPoint, perdiendo replicabilidad y generando mucho espacio para el error humano.

Ahora hemos podido hacer más eficiente la entrega de esos informes gracias al uso de funciones de R que contienen queries SQL parametrizadas, y que se utilizan dentro de reportes de R Markdown, que quedan disponibles en nuestro GitHub corporativo para usarse de nuevo cuando nos llega una solicitud similar. Los reportes de uso más frecuente están disponibles dentro del mismo paquete como plantillas.

Además, dado que hay miembros de nuestro equipo que prefieren utilizar Python, hemos tomado ventaja del paquete `reticulate` para permitirles a ellos colaborar en estas plantillas, agregando secciones que se ejecutan sin problemas al hacer `knitr` de los reportes.

Inducción a nuevos miembros del equipo, y promoción de mejores prácticas y principios del *tidyverse*

Utilizamos vignettes y la documentación del paquete como formas de educar a nuevos miembros del equipo en el uso de R para resolver problemas concretos de analítica en Walmart. También procuramos que los ejemplos incluidos en la documentación ilustren conceptos como el uso del pipe para combinar funciones, y respeten la guía de estilo del *tidyverse*³.

Asimismo, intentamos que nuestras propias funciones se guíen por el manifiesto de las tidy tools⁴, abrazando el uso de la programación funcional, retornando estructuras de datos simples cuando siempre que sea posible (preferentemente tibbles), aspirando a que todas las funciones realicen muy bien una tarea concreta (en lugar de crear funciones enormes que realicen múltiples tareas a la vez) y que sean concatenables mediante el pipe para realizar tareas más complejas.

³ <https://style.tidyverse.org/>

⁴ <https://cran.r-project.org/web/packages/tidyverse/vignettes/manifesto.html>