# # Organización de informes rutinarios

{#organizing-routine-reports}

Esta página cubre el paquete **reportfactory**, que es un complemento para el uso de R Markdown para los informes.

En situaciones en las que se ejecutan reportes de forma rutinaria (diariamente, semanalmente, etc.), facilita la compilación de múltiples archivos R Markdown y la organización de sus resultados. En esencia, proporciona una "fábrica" desde la que se pueden ejecutar los informes R Markdown, obtener automáticamente carpetas con fecha y hora para los resultados, y tener un control de versiones "ligero".

**reportfactory** es uno de los paquetes desarrollados por RECON (R Epidemics Consortium). Aquí está su [sitio web](https://www.repidemicsconsortium.org/) y [Github](https://github.com/reconverse).

## Preparación

### Cargar paquetes

Desde RStudio, instale la última versión del paquete **reportfactory** desde Github.

Puedes hacerlo a través del paquete **pacman** con p\_load\_current\_gh() que forzará la instalación de la última versión desde Github. Proporcione la cadena de caracteres "reconverse/reportfactory", que especifica la organización de Github (reconverse) y el repositorio (reportfactory). También puede utilizar install\_github() del paquete **remotes**, como alternativa.

## Nueva fábrica

Para crear una nueva fábrica, ejecuta la función new\_factory(). Esto creará una nueva carpeta de proyecto R autocontenida. Por defecto:

* La fábrica se añadirá a tu directorio de trabajo
* El nombre del proyecto R de la fábrica se llamará "nueva\_fábrica.Rproj"
* Tu sesión de RStudio se "trasladará" a este proyecto R

Mirando dentro de la fábrica, se puede ver que las subcarpetas y algunos archivos se crearon automáticamente.

* La carpeta report\_sources contendrá sus scripts R Markdown, que generan sus informes
* La carpeta de resultados contendrá los resultados del informe (por ejemplo, HTML, Word, PDF, etc.)
* La carpeta de scripts puede utilizarse para almacenar otros scripts de R (por ejemplo, los que se originan en sus scripts de Rmd)
* La carpeta de datos puede utilizarse para guardar sus datos (se incluyen las subcarpetas "raw" y "clean")
* Un archivo .here, para que puedas utilizar el paquete **here** para llamar a los archivos de las subcarpetas por su relación con esta carpeta raíz (véase la página de [proyectos de R](#r-projects) para más detalles)
* Se ha creado un archivo gitignore en caso de que se vincule este proyecto R a un repositorio de Github (ver [Control de versiones y colaboración con Github])
* Un archivo README vacío, para si usas un repositorio de Github

**ATENCIÓN:** dependiendo de la configuración de tu ordenador, los archivos como ".here" pueden existir pero ser invisibles.

De los ajustes por defecto, a continuación hay varios que puede querer ajustar dentro del comando new\_factory():

* factory = - Proporcionar un nombre para la carpeta de fábrica (por defecto es "new\_factory")
* path = - Designa una ruta de archivo para la nueva fábrica (por defecto es el directorio de trabajo)
* report\_sources = Proporcione un nombre alternativo para la subcarpeta que contiene los scripts R Markdown (por defecto es "report\_sources")
* outputs = Proporcione un nombre alternativo para la carpeta que contiene los resultados del informe (por defecto es "outputs")

Véase ?new\_factory para una lista completa de los argumentos.

Cuando se crea la nueva fábrica, tu sesión de R se transfiere al nuevo proyecto R, por lo que debe cargar de nuevo el paquete **reportfactory**.

Ahora puede ejecutar el comando factory\_overview() para ver la estructura interna (todas las carpetas y archivos) de la fábrica.

El siguiente "árbol" de las carpetas y archivos de la fábrica se imprime en la consola de R. Observe que en la carpeta "data" hay subcarpetas para los datos "raw" y "clean", y datos CSV de ejemplo. También hay "example\_report.Rmd" en la carpeta "report\_sources".

## Crear un informe

Desde el proyecto R de fábrica, cree un informe R Markdown como lo haría normalmente, y guárdelo en la carpeta "report\_sources". Consulta la página de R Markdown para obtener instrucciones. A modo de ejemplo, hemos añadido lo siguiente a la fábrica:

* Un nuevo script de R markdown titulado "daily\_sitrep.Rmd", guardado dentro de la carpeta "report\_sources".
* Datos para el informe ("linelist\_cleaned.rds"), guardados en la subcarpeta "clean" dentro de la carpeta "data"

Podemos ver usando factory\_overview() nuestro R Markdown en la carpeta "report\_sources" y el archivo de datos en la carpeta de datos "clean" (resaltado):

A continuación se muestra una captura de pantalla del comienzo del R Markdown "daily\_sitrep.Rmd". Puedes ver que el formato de salida está configurado para ser HTML, a través de la salida de la cabecera YAML: html\_document.

En este sencillo script, hay comandos para:

* Cargar los paquetes necesarios
* Importe los datos del listado utilizando una ruta de archivo del paquete **aquí** (lea más en la página sobre [importación y exportación](#import-and-export))
* Imprime una tabla resumen de los casos, y la exporta con export() como un archivo .csv
* Imprimir una epicurva, y exportarla con ggsave() como un archivo .png

Puedes revisar sólo la lista de informes R Markdown en la carpeta "report\_sources" con este comando:

## Compilar

En una fábrica de informes, "compilar" un informe R Markdown significa que se ejecutará el script .Rmd y se producirá la salida (como se especifica en el script YAML, por ejemplo, como HTML, Word, PDF, etc.).

La fábrica creará automáticamente una carpeta con fecha y hora para las salidas en la carpeta "outputs".

El propio informe y cualquier archivo exportado producido por el script (por ejemplo, csv, png, xlsx) se guardarán en esta carpeta. Además, el propio script Rmd se guardará en esta carpeta, para que tenga un registro de esa versión del script.

Esto contrasta con el comportamiento normal de un R Markdown "tejido", que guarda las salidas en la ubicación del script Rmd. Este comportamiento por defecto puede resultar en carpetas abarrotadas y desordenadas. El objetivo de la fábrica es mejorar la organización cuando uno necesita ejecutar informes con frecuencia.

### Compilar por nombre

Puedes compilar un informe específico ejecutando compile\_reports() y proporcionando el nombre del script Rmd (sin la extensión .Rmd) a reports =. Para simplificar, puede omitir reports = y simplemente escribir el nombre R Markdown entre comillas, como se indica a continuación.

Este comando compilaría sólo el informe "daily\_sitrep.Rmd", guardando el informe HTML, y las exportaciones de la tabla .csv y la epicurva .png en una subcarpeta con fecha y hora específicas del informe, dentro de la carpeta "outputs".

Ten en cuenta que si elige proporcionar la extensión .Rmd, debe escribir correctamente la extensión tal y como se guarda en el nombre del archivo (.rmd vs. .Rmd).

Ten en cuenta también que, al compilar, es posible que aparezcan temporalmente varios archivos en la carpeta "report\_sources", pero pronto desaparecerán al ser transferidos a la carpeta "outputs" correcta.

### Compilación por número

También puede especificar el script Rmd a compilar proporcionando un número o vector de números a reports =. Los números deben alinearse con el orden en que aparecen los informes cuando se ejecuta list\_reports().

### Compilar todos los

Puedes compilar todos los informes R Markdown en la carpeta "report\_sources" estableciendo el argumento reports = a TRUE.

### Compilar desde la subcarpeta

Puedes añadir subcarpetas a la carpeta "report\_sources". Para ejecutar un informe R Markdown desde una subcarpeta, simplemente proporcione el nombre de la carpeta a subcarpeta =. A continuación se muestra un ejemplo de código para compilar un informe Rmd que vive en una subcarpeta de "report\_sources".

Puedes compilar todos los informes Rmd dentro de una subcarpeta proporcionando el nombre de la subcarpeta a reports =, con una barra al final, como se indica a continuación.

### Parametrización

Como se indicó en la página sobre [Informes con R Markdown](#reports-with-r-markdown), puede ejecutar informes con parámetros especificados. Puedes pasar estos parámetros como una lista a compile\_reports() a través del argumento params =. Por ejemplo, en este informe ficticio hay tres parámetros proporcionados a los informes de R Markdown.

### Utilizar un "archivo de ejecución"

Si tiene varios informes que ejecutar, considere la posibilidad de crear un script de R que contenga todos los comandos compile\_reports(). Un usuario puede simplemente ejecutar todos los comandos en este script de R y todos los informes se compilarán. Puedes guardar este "archivo de ejecución" en la carpeta "scripts".

## Salidas

Después de haber compilado los informes unas cuantas veces, la carpeta "outputs" podría tener este aspecto (los resaltados se han añadido para mayor claridad):

* Dentro de "outputs", se han creado subcarpetas para cada informe Rmd
* Dentro de ellas, se han creado otras subcarpetas para cada compilación única
  + Están marcados con fecha y hora ("2021-04-23\_T11-07-36" significa 23 de abril de 2021 a las 11:07:36)
  + Puedes editar el formato de la fecha/hora. Ver compilación de informes
* Dentro de cada carpeta compilada de fecha/hora, se almacena el resultado del informe (por ejemplo, HTML, PDF, Word) junto con el script Rmd (¡control de versiones!) y cualquier otro archivo exportado (por ejemplo, table.csv, epidemic\_curve.png)

Esta es una vista dentro de una de las carpetas con fecha/hora, para el informe "daily\_sitrep". La ruta del archivo está resaltada en amarillo para enfatizar.

Por último, a continuación se muestra una captura de pantalla de la salida del informe HTML.

Puedes utilizar list\_outputs() para revisar una lista de las salidas.

## Varios

### Tejido

Si lo desea, puede "tejer" uno de sus informes R Markdown clicando el botón "Knit". Si hace esto, como por defecto, las salidas aparecerán en la carpeta donde se guarda el Rmd - la carpeta "report\_sources". En versiones anteriores de **reportfactory**, tener cualquier archivo que no sea Rmd en "report\_sources" impediría la compilación, pero esto ya no es así. Puedes ejecutar compile\_reports() y no se producirá ningún error.

### Scripts

Te animamos a utilizar la carpeta "scripts" para almacenar "archivos de ejecución" o scripts .R que se originan en tus scripts .Rmd. Consulta la página sobre [R Markdown](#reports-with-r-markdown) para obtener consejos sobre cómo estructurar tu código en varios archivos.

### Extras

* Con **reportfactory**, puede utilizar la función list\_deps() para listar todos los paquetes requeridos en todos los informes de toda la fábrica.
* Hay un paquete de acompañamiento en desarrollo llamado **rfextras** que ofrece más funciones de ayuda para asistirle en la construcción de informes, tales como:
  + load\_scripts() - carga todos los scripts .R en una carpeta determinada (la carpeta "scripts" por defecto)
  + find\_latest() - encuentra la última versión de un archivo (por ejemplo, el último conjunto de datos)

## Recursos

Consulta la [página de Github del](https://github.com/reconverse/reportfactory) paquete **reportfactory**

Consulta la [página de Github del](https://github.com/reconhub/rfextras) paquete **rfextras**