

# Ejemplo reporte pdf con Rmarkdown

Diplomado en Análisis de datos con R para la Acuicultura

Dr. José A. Gallardo | Pontificia Universidad Católica de Valparaíso | jose.gallardo@pucv.cl

02 October 2021

## Metadata

Este script contiene toda la información necesaria para generar un reporte dinámico en formato pdf. Al inicio de este documento usted puede observar la **metadata** en formato **YAML**. Note que hemos agregado un *subtítulo*, un formato de fecha dinámica que establece la fecha del día que se imprime el reporte y el formato pdf con el comando **output**.

## Markdown

Este texto que usted lee ahora se denomina markdown y puede modificarlo a voluntad cuando escriba un reporte, agregando todo tipo de formatos, así como tablas, listas, párrafos o referencias.

### Título con diferente tamaño

En markdown # indica un título.

Para reducir el tamaño de los título agregando más #

```
# Título
## Título
### Título
```

## Titulo

### Título

#### Título

### Ejemplo uso de viñetas o listas numeradas

Puede agregar listas como lo hace habitualmente en word.

- Elemento 1
  - Elemento 2
  - Elemento 3
- 
- Elemento 1
  - Elemento 2

- Elemento 3

### Ejemplo escribir en negrita o cursiva

Puede dar formato de negrita o cursiva usando uno o dos **\*\*** a cada lado de la palabra.

```
**Esto va en negrita**
*Esto va en cursiva*
```

**Esto va en negrita**

*Esto va en cursiva*

### Comando saltos de página

Puedes dar saltos de página para dar formato a tu reporte con el siguiente comando

```
\newpage
```

### Ejemplo para crear tablas

Puede crear tablas usando barras espaciadoras y guiones como en el siguiente ejemplo

```
| Variable A | Variable B | Variable C |
| :-- | :--: | --: |
| A1 | B1 | C1 |
| A2 | B2 | C2 |
```

| Variable A | Variable B | Variable C |
|------------|------------|------------|
| A1         | B1         | C1         |
| A2         | B2         | C2         |

### Bloques de código o chunk

La principal diferencia de Markdown con Rmarkdown es obviamente la posibilidad de incluir códigos de R para analizar datos. Esto se consigue agregando bloques de códigos como el que se muestra a continuación. Observe con detención que el bloque de códigos comienza y termina con 3 tildes invertidas “`````” y que al inicio hay una `r` entre dos `{ }` lo que indica al software que es un código de **R**. Éste primer bloque es para configurar el resto de los bloques de código, por eso se ha incluido un nombre **setup**. En este primer **chunk** puede ver el llamado a ejecutar la librería **knitr**.

Cuando trabajes con bloques de códigos para analizar tus datos, también puedes agregar el nombre del bloque del código, en este caso usaremos de ejemplo el set de datos **BOD**.

Mientras que para ejecutar un bloque de código, debes presionar **la punta de flecha verde o play** que se observa en la parte superior derecha del mismo bloque de códigos.

```
# Este bloque de códigos se llama resumen BOD.
summary(BOD)
```

```
##           Time           demand
##  Min.      :1.000   Min.      : 8.30
##  1st Qu.:2.250   1st Qu.:11.62
##  Median :3.500   Median :15.80
##  Mean    :3.667   Mean    :14.83
##  3rd Qu.:4.750   3rd Qu.:18.25
##  Max.    :7.000   Max.    :19.80
```

También puedes incluir y dar nombre a las gráficas en el mismo bloque de código usando el comando `fig.cap=""`.

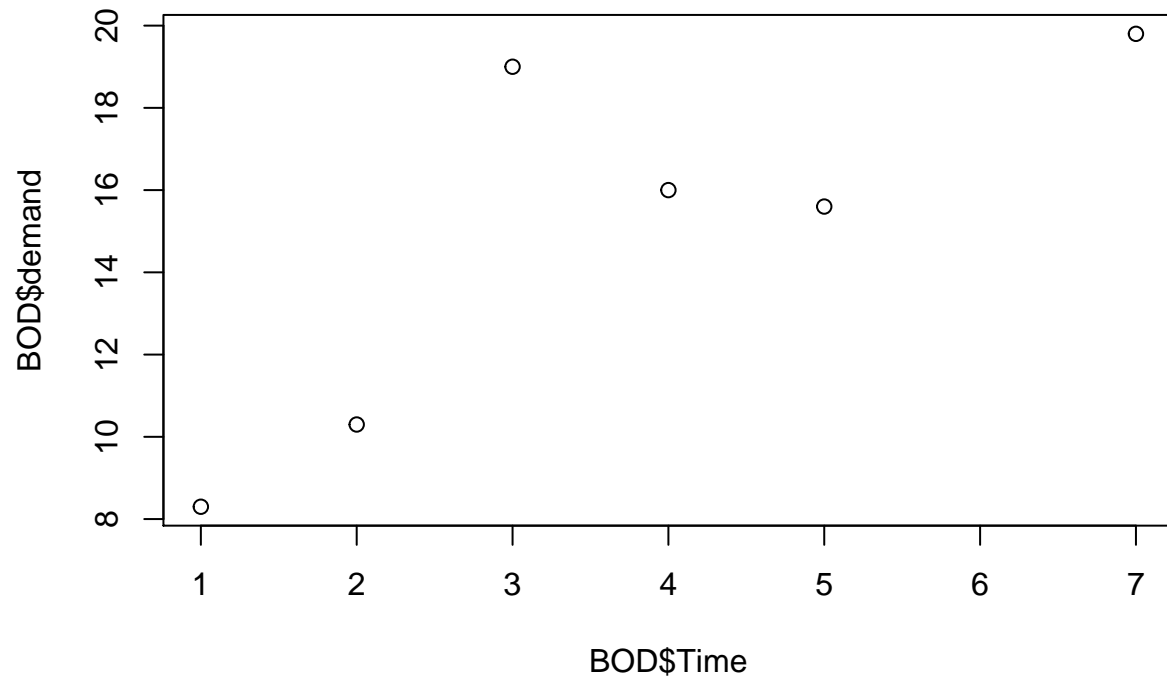


Figure 1: Demanda biológica de oxígeno en función del tiempo

## Librerías de R

Una de las fortalezas de R es que existen miles de librerías para realizar análisis de datos. Algunas están pre-instaladas pero otras no.

A continuación, se presentan algunos comandos para revisar las librerías que están instaladas y habilitadas para trabajar.

```
# Este comando permite revisar que librerías están "habilitadas"
# y listas para usar en tu ambiente de trabajo.
search()
```

```
## [1] ".GlobalEnv"      "package:stats"    "package:graphics"
## [4] "package:grDevices" "package:utils"    "package:datasets"
## [7] "package:methods"  "Autoloads"        "package:base"
```

### Repositorios clave para la búsqueda avanzada de librerías.

| Repositorio                  | Descripción   |
|------------------------------|---|
| <b>CRAN Packages By Name</b> | CRAN Packages By Name permite buscar y acceder a las más de 17.000 librerías de <b>R</b> por su nombre para instalar de forma individual.   |
| <b>CRAN Task Views</b>       | Task views entrega una guía para realizar tareas en distintas disciplinas científicas agrupando diferentes librerías por tema, se instalan de forma agrupada.   |
| <b>Rseek</b>                 | Rseek es una herramienta que permite buscar librerías por palabras clave; tanto dentro como por fuera del repositorio <b>CRAN</b> . Ésta aplicación fue desarrollada por Sasha Goodman de la Stanford university. |
| <b>Bioconductor</b>          | Bioconductor es el gold standards para trabajar en análisis de datos genómicos  |

### Imprimir el reporte

Finalmente, para poder generar un reporte debes presionar el **triángulo negro** que está junto a la palabra **Knit** en la parte superior de este documento. Si presionas **Knit to pdf**, esto indicará a la librería **knitr** que debe leer el presente documento, interpretar y ejecutar el metadata, el texto de bajo nivel y los diferentes bloques de código para elaborar un reporte pdf.

Para finalizar, puedes generar dos reportes adicionales uno en formato html y otro en formato word. Compare los documentos e identifique sus diferencias.

Para más detalles de como trabajar con Rmarkdown investigue el siguiente link.