

# Clase 06 Manipulación de datos con dplyr

## Diplomado en Análisis de datos con R para la Acuicultura

Dr. José A. Gallardo | [jose.gallardo@pucv.cl](mailto:jose.gallardo@pucv.cl) | Pontificia  
Universidad Católica de Valparaíso

10 October 2021

# PLAN DE LA CLASE

## 1.- Introducción

- ▶ ¿Para qué manipular datos?
- ▶ Librería dplyr: Tuberías.
- ▶ Librería dplyr: Comandos clave.

## 2). Práctica con R y Rstudio cloud.

- ▶ Realizar manipulación de datos con dplyr.
- ▶ Realizar gráficas avanzadas con ggplot2.
- ▶ Elaborar un reporte dinámico en formato pdf con Rmarkdown.

# MANIPULACIÓN DE DATOS

## ¿Para qué manipular datos?

- ▶ Para dar el formato adecuado a nuestro set de datos previo al análisis estadístico.
- ▶ Para hacerlos más legibles y organizados.
- ▶ Etapa clave para una correcta visualización de datos.

## Ejemplos de tareas comunes durante esta etapa:

- ▶ Filtrar datos por categorías.
- ▶ Remover datos o imputar datos faltantes.
- ▶ Agrupar datos por algún criterio.
- ▶ Seleccionar y resumir variables.
- ▶ Generar variables derivadas a partir de variables existentes.

# LIBRERÍA DPLYR: FUNCIONES CLAVE

La librería **dplyr** posee varias funciones que permiten manipular data.frames de forma ágil e intuitiva.

## Funciones claves:

**select()**: Permite extraer o seleccionar variables/columnas específicas de un data.frame.

**mutate()**: Permite calcular nuevas variables “derivadas”. Util para calcular proporciones, tasas.

**filter()**: Para filtrar desde una tabla de datos un subconjunto de filas. Ej. solo un nivel de de un factor, observaciones que cumplen algún criterio (ej.  $> 20$ ).

**group\_by()**: Permite agrupar filas con base a los niveles de alguna variable o factor.

## LIBRERÍA DPLYR: EL OPERADOR PIPE (TUBERÍA).

**dplyr** usa el operador pipe `%>%` como una tubería para enlazar un data.frame con una o más funciones.

```
x <- rnorm(5)
y <- rnorm(5)
dat <- data.frame(x,y)
dat %>% max
```

```
## [1] 1.989735
```

```
dat %>% arrange(y)
```

```
##           x           y
## 1 -0.52834972 -0.07111831
## 2 -0.08157893  0.03155888
## 3  0.14927082  0.22745319
## 4  0.32750915  1.38660288
## 5  1.16733240  1.98973475
```

# ESTUDIO DE CASO: MUESTREO DE PECES

**Objeto: peces**

| Pez | Especie | Sexo   | Peso | Parásitos |
|-----|---------|--------|------|-----------|
| 1   | A       | Hembra | 174  | 0         |
| 2   | A       | Hembra | 155  | 2         |
| 3   | A       | Hembra | 131  | 25        |
| 4   | B       | Macho  | 163  | 8         |
| 5   | B       | Macho  | 103  | 33        |
| 6   | B       | Hembra | 138  | 15        |
| 7   | C       | Hembra | 135  | 5         |
| 8   | C       | Macho  | 138  | 20        |
| 9   | C       | Hembra | 135  | 45        |

# FUNCIÓN SELECT()

```
select(peces, Especie, Sexo)
```

```
## # A tibble: 9 x 2
```

```
##   Especie Sexo
```

```
##   <chr>   <chr>
```

```
## 1 A      Hembra
```

```
## 2 A      Hembra
```

```
## 3 A      Hembra
```

```
## 4 B      Macho
```

```
## 5 B      Macho
```

```
## 6 B      Hembra
```

```
## 7 C      Hembra
```

```
## 8 C      Macho
```

```
## 9 C      Hembra
```

# FUNCIÓN SELECT() CON PIPE

```
peces %>% select(Especie, Sexo)
```

```
## # A tibble: 9 x 2
```

```
##   Especie Sexo
```

```
##   <chr>   <chr>
```

```
## 1 A      Hembra
```

```
## 2 A      Hembra
```

```
## 3 A      Hembra
```

```
## 4 B      Macho
```

```
## 5 B      Macho
```

```
## 6 B      Hembra
```

```
## 7 C      Hembra
```

```
## 8 C      Macho
```

```
## 9 C      Hembra
```



# FUNCIÓN FILTER() CON PIPE

```
peces %>% filter(Sexo == "Macho")
```

```
## # A tibble: 3 x 5
```

| ##   | Pez   | Especie | Sexo  | Peso  | Parasitos |
|------|-------|---------|-------|-------|-----------|
| ##   | <dbl> | <chr>   | <chr> | <dbl> | <dbl>     |
| ## 1 | 4     | B       | Macho | 163   | 8         |
| ## 2 | 5     | B       | Macho | 103   | 33        |
| ## 3 | 8     | C       | Macho | 138   | 20        |

# MÚLTIPLES FUNCIONES Y TUBERÍAS

```
peces %>% select(Especie, Sexo, Peso) %>%  
  filter(Sexo == "Macho")
```

```
## # A tibble: 3 x 3  
##   Especie Sexo  Peso  
##   <chr>   <chr> <dbl>  
## 1 B      Macho  163  
## 2 B      Macho  103  
## 3 C      Macho  138
```

# FUNCIÓN SUMMARIZE()

```
peces %>% select(Especie, Sexo, Peso, Parasitos) %>%  
  summarize(Minimo_Peso = min(Peso),  
            Minimo_Parasitos = min(Parasitos))
```

```
## # A tibble: 1 x 2  
##   Minimo_Peso Minimo_Parasitos  
##         <dbl>         <dbl>  
## 1         103             0
```

## FUNCIÓN SUMMARIZE() + GROUP\_BY()

```
peces %>% group_by(Especie) %>%  
  summarize(n = n(),  
            Minimo_Peso = max(Peso),  
            Promedio_peso= mean(Peso))
```

```
## # A tibble: 3 x 4  
##   Especie      n Minimo_Peso Promedio_peso  
##   <chr>    <int>      <dbl>      <dbl>  
## 1 A         3        174        153.  
## 2 B         3        163        135.  
## 3 C         3        138        136
```

# FUNCIÓN MUTATE()

```
peces %>% select(Especie, Peso, Parasitos) %>%  
  mutate(Densidad_parasitos = Parasitos/Peso)
```

```
## # A tibble: 9 x 4  
##   Especie  Peso Parasitos Densidad_parasitos  
##   <chr>   <dbl>   <dbl>         <dbl>  
## 1 A      174      0          0  
## 2 A      155      2        0.0129  
## 3 A      131     25        0.191  
## 4 B      163      8        0.0491  
## 5 B      103     33        0.320  
## 6 B      138     15        0.109  
## 7 C      135      5        0.0370  
## 8 C      138     20        0.145  
## 9 C      135     45        0.333
```

# PRÁCTICA ANÁLISIS DE DATOS

1.- Guía de trabajo Rmarkdown disponible en drive.

## **Clase 06**

2.- La tarea se realiza en Rstudio.cloud. **Clase 06 - Manipular datos con dplyr**

# RESUMEN DE LA CLASE

- ▶ Manipulamos datos con dplyr.
- ▶ Aplicamos tuberías con pipe `%>%`.
- ▶ Comunicamos un análisis exploratorio de datos de forma efectiva.