

# Package ‘r4fish’

February 2, 2024

**Type** Package

**Title** r4fish analysis

**Version** 1.1.0

**Author** Elmer O. Quispe Salazar

**Maintainer** The package maintainer <qselmer@gmail.com>

**Description** Assessment for Marine Resources Toolkit.

**License** What license is it under?

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.2.3

## R topics documented:

assignerPeruvianGrid . . . . .	2
calculateJuv . . . . .	2
curve.sel . . . . .	3
distCoast . . . . .	3
estimateMode . . . . .	4
g2plotSurvey . . . . .	4
getModeSpecies . . . . .	5
getStockInfo . . . . .	5
getSurveyLmax . . . . .	6
MapPeruGrid . . . . .	6
plot2CompSizeTime . . . . .	7
PlotSimpleFrec2 . . . . .	7
plot_envir . . . . .	8
renderBiometric . . . . .	8
SimpleFrecSp . . . . .	9
subsetSurvey . . . . .	10
validateSp . . . . .	10
<b>Index</b>	<b>12</b>

---

assignerPeruvianGrid	<i>Title</i>
----------------------	--------------

---

**Description**

Title

**Usage**

```
assignerPeruvianGrid(  
  data,  
  xlon = "lon",  
  ylat = "lat",  
  by = "onedegree",  
  metadata = F,  
  save = T,  
  cout = "."  
)
```

**Arguments**

cout

---

calculateJuv	<i>Title</i>
--------------	--------------

---

**Description**

Title

**Usage**

```
calculateJuv(vec, marks, mjuv, abs = T)
```

**Arguments**

abs

---

curve.sel	<i>Title</i>
-----------	--------------

---

**Description**

Title

**Usage**

```
curve.sel(  
  sp = NA,  
  stock = NA,  
  marks = NA,  
  par = c(NA, NA),  
  method = "log3",  
  add.plot = T,  
  add.inv = T  
)
```

**Arguments**

add.inv

---

distCoast	<i>Title</i>
-----------	--------------

---

**Description**

Title

**Usage**

```
distCoast(lon, lat)
```

**Arguments**

lat

---

estimateMode	<i>Estima moda de una frecuencia de tallas</i>
--------------	--

---

### Description

Estima moda de una frecuencia de tallas

### Usage

```
estimateMode(len, freq, tol.n = 150, nmodes = 2, tol.freq = 10)
```

### Arguments

len	vector completo de tallas
freq	vector completo de frecuencias
tol.n	valor de tolerancia del numero de la muestra
nmodes	numero de modas a estimar
tol.freq	valor de tolerancia para ser considerada moda

---

g2plotSurvey	<i>Title</i>
--------------	--------------

---

### Description

Title

### Usage

```
g2plotSurvey(
  dat = dat,
  col.var = "biomasa",
  col.factor = 2,
  factor = NA,
  cols.factor = c("red", "blue"),
  box = "boxplot",
  model = "lm",
  lambda = 1e-04,
  IC.model = TRUE,
  info = TRUE,
  marf = c(0.5, 0.5, 0, 0.5),
  omaf = c(3, 4.5, 2, 2),
  hline = TRUE,
  cline = "white",
  bgf = "white",
  lwdf = 1,
  ltyf = 1,
  pchf = 16,
  cexf = 1.5,
```

```
    unitf = 1,
    saveplot = T,
    outf = "./",
    widthf = 1800,
    heightf = 1200,
    resf = 250,
    labels.f = c("\nverano", "\ninvierno-\nprimavera"),
    labels.y = "Biomasa ton",
    ...
  )
```

Arguments

labels.y

---

getModeSpecies	<i>Title</i>
----------------	--------------

---

Description

Title

Usage

```
getModeSpecies(
  Length = Length,
  sp = "anchoveta",
  src = "fsh",
  nmodes = 2,
  tol = 0,
  savePlot = TRUE,
  dirout = "Outputs/"
)
```

Arguments

dirout

---

getStockInfo	<i>Title</i>
--------------	--------------

---

Description

Title

Usage

```
getStockInfo(sp, data = NULL, ...)
```

Arguments

...

---

getSurveyLmax	<i>Title</i>
---------------	--------------

---

**Description**

Title

**Usage**

```
getSurveyLmax(dat = dat, sp = "anchoveta", src = "pope", col.var = 4:78)
```

**Arguments**

col.var

---

MapPeruGrid	<i>Title</i>
-------------	--------------

---

**Description**

Title

**Usage**

```
MapPeruGrid(
  data,
  colcode = "code",
  colval = "freq",
  typeval = "#",
  xxlim = c(-86, -70),
  yylin = c(-21, -3),
  by = "onedegree",
  gradient = c("yellow", "red"),
  border = NA,
  all.grid = F,
  land.col = "gray90",
  land.border = "gray90",
  legend = T,
  txtleg = "(n)",
  xaxis = 1,
  yaxis = 4,
  portImport = 2,
  save = T,
  cout = "."
)
```

**Arguments**

cout

---

plot2CompSizeTime	<i>Title</i>
-------------------	--------------

---

**Description**

Title

**Usage**

```
plot2CompSizeTime(  
  dat = dat_marks,  
  sp = "anchoveta",  
  stock = "nc",  
  type = "B",  
  ab = NA,  
  factor = NA,  
  cexf = 3,  
  col.in = "gray90",  
  col.bd = "black",  
  marp = c(3, 3, 2, 2),  
  omap = c(1, 2, 1, 1),  
  mgpp = c(1, 0.5, 0),  
  widthFig = 2600,  
  heightFig = 3200,  
  resFig = 380,  
  SavePlot = T,  
  dirout = "Outputs/"  
)
```

**Arguments**

dirout

---

PlotSimpleFrec2	<i>Title</i>
-----------------	--------------

---

**Description**

Title

**Usage**

```
PlotSimpleFrec2(  
  data,  
  sp = sp,  
  stock = stock,  
  col.sp = "red",  
  colset = "navajowhite",  
  cout = ".",
```

```
    save = T,  
    format = ".png",  
    ylim = c(0, 0.5),  
    width = 2625,  
    height = 1750  
  )
```

Arguments

colset

---

plot_envir	<i>Title</i>
------------	--------------

---

Description

Title

Usage

```
plot_envir(  
  what = NA,  
  year.limit = c(1990, 2022),  
  ylim = c(-2, 3.5),  
  magnitude = F,  
  axis.x = T  
)
```

Arguments

axis.x

---

renderBiometric	<i>Crea, unifica archivo biometrico</i>
-----------------	---

---

Description

Crea, unifica archivo biometrico

Usage

```
renderBiometric(  
  cin = "inputs",  
  cout = ".",  
  file = "data.xlsx",  
  encoding = "latin",  
  save = T,  
  ...  
)
```



**Arguments**

cin	vector de caracteres que contienen la ruta de entrada
cout	vector de caracteres que contienen la ruta de salida
file	vector de caracteres que contienen el nombre del archivo
encoding	vector de caracteres para tipo de codificación
save	logico; ¿Se deben guardar el archivo?

**Value**

base de datos biométrico

**Examples**

```
\code{renderBiometric(cin = "input",
cout = NA,
file = "Biometria por especies 03032023.xlsx",
save = T)}
```

---

SimpleFrecSp

---

*Plot: Estructura de tallas para "anchoveta "spTarget" lances*


---

**Description**

Plot: Estructura de tallas para "anchoveta "spTarget" lances

**Usage**

```
SimpleFrecSp(data = x, sp = "anchoveta", stock = "nc", save = T)
```

**Arguments**

data	datos del crucero en formato MF
sp	especie objetivo de la evaluación
stock	stock objetivo de la evaluación
save	valor logico; ¿Se deben guardar la matriz de lances en "csv"?

---

subsetSurvey	<i>Title</i>
--------------	--------------

---

**Description**

Title

**Usage**

```
subsetSurvey(
  dat = dat,
  col.var = c("biomasa", "abundancia"),
  year.lim = c(1996, 2022),
  col.set = "filter",
  set = "ok",
  col.factor = "season",
  col.label = 2,
  col.date = NA,
  col.month = NA,
  col.sems = "semester",
  col.year = "year"
)
```

**Arguments**

col.year

---

validateSp	<i>Identifica errores y valida la base de datos</i>
------------	---

---

**Description**

Identifica errores y valida la base de datos

**Usage**

```
validateSp(
  data = data,
  sp = "anchoveta",
  stock = NA,
  cout = "outputs",
  file = "validateSp"
)
```

**Arguments**

data	datos del crucero en formato MF
sp	especie objetivo de la evaluación
stock	stock objetivo de la evaluación
cout	vector de caracteres que contienen la ruta de salida
file	ector de caracteres que contienen el nombre del archivo

**Examples**

```
\code{validateSp(data = data, sp = "jurel", stock = NA, cout = "outputs",  
file = "document.docx")}
```

# Index

`assignerPeruvianGrid`, [2](#)

`calculateJuv`, [2](#)

`curve.sel`, [3](#)

`distCoast`, [3](#)

`estimateMode`, [4](#)

`g2plotSurvey`, [4](#)

`getModeSpecies`, [5](#)

`getStockInfo`, [5](#)

`getSurveyLmax`, [6](#)

`MapPeruGrid`, [6](#)

`plot2CompSizeTime`, [7](#)

`plot_envir`, [8](#)

`PlotSimpleFrec2`, [7](#)

`renderBiometric`, [8](#)

`SimpleFrecSp`, [9](#)

`subsetSurvey`, [10](#)

`validateSp`, [10](#)