

# Package ‘R4MFCL’

September 13, 2013

**Title** R functions for MULTIFAN-CL

**Version** 0.2

**Author** Simon Hoyle, Pierre Kleiber, Shelton Harley, Nick Davies, and Adam Langley.

**Description** Functions for automating the running and analysis of MULTIFAN-CL stock assessments, by manipulating the input files, and analyzing and plotting the output files. R4MFCL is a collection of utility functions for stock assessments using the model MULTIFAN-CL (Fournier et al 1998; [www.multifan-cl.org](http://www.multifan-cl.org)). There are several groups of R4MFCL functions: 1) input and output functions, for reading MULTIFAN-CL files into R objects and writing them back out as text files in the form that MULTIFAN-CL accepts as input. 2) data manipulation functions, for editing and restructuring the input objects. 3) plotting functions, for producing plots and maps from the result objects. 4) information functions, for comparing objects and giving information about, for example, flag settings.

**Maintainer** Simon Hoyle <[simon.hoyle@gmail.com](mailto:simon.hoyle@gmail.com)> and Sam McKechnie <[samm@spc.int](mailto:samm@spc.int)>

**License** GPL-2

**Date** 2013-09-10

**Suggests** maps, mapdata, mapproj

## R topics documented:

|                            |    |
|----------------------------|----|
| add.catch.frq . . . . .    | 4  |
| add.cpue.frq . . . . .     | 4  |
| add.flag . . . . .         | 5  |
| carry.effort.frq . . . . . | 6  |
| change.fishflag . . . . .  | 6  |
| change.flag . . . . .      | 7  |
| change.negflag . . . . .   | 7  |
| change.size.frq . . . . .  | 8  |
| change_data . . . . .      | 8  |
| check.eff.devs . . . . .   | 9  |
| check_flag_value . . . . . | 9  |
| clean.lfdata . . . . .     | 10 |
| compare.ce.frq . . . . .   | 10 |

|  |    |
|--|----|
| compare.frq . . . . .                    | 11 |
| compare.size.frq . . . . .               | 11 |
| compare_par_flags . . . . .              | 12 |
| condor.go . . . . .                      | 13 |
| condor.go2 . . . . .                     | 14 |
| convert.frq.ver6 . . . . .               | 14 |
| copy.condor.files . . . . .              | 15 |
| create.missing.ce . . . . .              | 15 |
| crit.fishery.summary . . . . .           | 16 |
| crit.summary . . . . .                   | 16 |
| datfromstr . . . . .                     | 17 |
| do.critical.calcs . . . . .              | 17 |
| doit.rm_flag . . . . .                   | 18 |
| effortcreep . . . . .                    | 19 |
| fix_growth . . . . .                     | 19 |
| frq.change.nint . . . . .                | 20 |
| frq.remove.size.or.weight.data . . . . . | 20 |
| get.critical.age . . . . .               | 21 |
| get.length.output . . . . .              | 21 |
| get.outcomes . . . . .                   | 22 |
| get.tag.structure . . . . .              | 23 |
| get.weight.output . . . . .              | 23 |
| labels_store . . . . .                   | 24 |
| load.LFdata . . . . .                    | 24 |
| make.projection.betyft.frq . . . . .     | 25 |
| map_all_pacific . . . . .                | 26 |
| merge.frq . . . . .                      | 26 |
| merge.tag . . . . .                      | 27 |
| merge_tag_objs . . . . .                 | 28 |
| pack.fisheries.frq . . . . .             | 28 |
| plot.base.comparison . . . . .           | 29 |
| plot.biomass . . . . .                   | 29 |
| plot.biomass.combined . . . . .          | 30 |
| plot.F.time . . . . .                    | 31 |
| plot.fishery.impact.r . . . . .          | 31 |
| plot.Kobe . . . . .                      | 32 |
| plot.Kobe.template.bw . . . . .          | 33 |
| plot.Kobe.template.col . . . . .         | 33 |
| plot.mfcl.betyft09 . . . . .             | 34 |
| plot.nofishing . . . . .                 | 34 |
| plot.nofishing.combined . . . . .        | 35 |
| plot.pacific.alb . . . . .               | 35 |
| plot.pacific.skj . . . . .               | 36 |
| plot.pacific.species . . . . .           | 36 |
| plot.pacific.WCPFC . . . . .             | 37 |
| plot.pacific.yft . . . . .               | 38 |
| plot.recruitment . . . . .               | 38 |
| plot.recruitment.combined . . . . .      | 39 |
| plot_cpue_cv_frq . . . . .               | 40 |
| ppath . . . . .                          | 40 |
| R4MFCL . . . . .                         | 41 |
| read.catchrep . . . . .                  | 41 |

|                              |    |
|------------------------------|----|
| read.ests                    | 42 |
| read.fit                     | 42 |
| read.frq                     | 43 |
| read.impact                  | 43 |
| read.ini                     | 44 |
| read.par                     | 44 |
| read.rep                     | 45 |
| read.tag                     | 45 |
| read.var                     | 46 |
| read_nmd.frq                 | 46 |
| reconstruct.frq.ce           | 47 |
| reconstruct.frq.size         | 48 |
| region_single_frq            | 48 |
| region_single_ini            | 49 |
| region_single_tag            | 49 |
| regroup_fishery_grps.doitall | 50 |
| rename.fisheries.doitall     | 51 |
| rename.fisheries.frq         | 51 |
| rename.fisheries.tag         | 52 |
| rename.fishery_grps.doitall  | 52 |
| retro.frq                    | 53 |
| retro.tag                    | 54 |
| rm_fisheries.doitall         | 54 |
| rm_fisheries.frq             | 55 |
| rm_fisheries.tag             | 55 |
| rm_fishflag                  | 56 |
| rm_flag.doitall              | 56 |
| run.profile                  | 57 |
| seas.flag                    | 58 |
| seas.frq                     | 58 |
| seas.tag                     | 59 |
| setup.cpue                   | 59 |
| setup.effcreep               | 60 |
| setup.growth                 | 61 |
| setup.growth.offsets         | 61 |
| setup.idphcatch              | 62 |
| setup.lensel                 | 63 |
| setup.LFwt                   | 63 |
| setup.M                      | 64 |
| setup.pscatch                | 65 |
| setup.startyr                | 65 |
| setup.steepness              | 66 |
| setup.timesplit              | 67 |
| sort.frq                     | 67 |
| start_year.frq               | 68 |
| steepness.doit               | 68 |
| summarise.size.frq.bet       | 69 |
| tag_grps_rm                  | 70 |
| timesplit.doitall            | 70 |
| timesplit.frq                | 71 |
| timesplit.tag                | 71 |
| varfromstr                   | 72 |

|                         |    |
|-------------------------|----|
| write.frq . . . . .     | 73 |
| write.ini . . . . .     | 73 |
| write.par . . . . .     | 74 |
| write.tag . . . . .     | 74 |
| write_nmd.frq . . . . . | 75 |

## Index 76

---

|               |                         |
|---------------|-------------------------|
| add.catch.frq | <i>add.catch.frq.Rd</i> |
|---------------|-------------------------|

---

### Description

Used in sensitivity analyses, this function replaces the catch column relating to the fisheries specified - while accounting for the possibility that the number of rows or order differ between the sensitivity runs. The script assumes that the sensitivity \*.frq will contain either: only those rows which are to be modified; or all rows to be modified.

### Usage

```
add.catch.frq(frq, filepath, fshries)
```

### Arguments

|          |  |
|----------|--|
| frq      | The frq file object  |
| filepath | The path and filename of the frq file with the replacement fishery data. |
| fshries  | The id numbers of the fisheries to be edited                             |

### Author(s)

Nick Davies

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|              |                        |
|--------------|------------------------|
| add.cpue.frq | <i>add.cpue.frq.Rd</i> |
|--------------|------------------------|

---

### Description

Replaces the nominal effort in the original .FRQ file with stanadrdised effort based on the CPUE index Flexible to handle either sort of frq file and you have the choice to include the cv. Puts in -1 for effort first to make sure we account for missing values of CPUE

### Usage

```
add.cpue.frq(CPUE.file = "P:/yft/2009/Data Preparation/CPUE/indices/yft_JPstd_R1
```

**Arguments**

CPUE.file  
data  
fishery  
add.cv

**Author(s)**

Shelton Harley

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

add.flag

*add.flag.Rd*

---

**Description**

Adds a flag to the doitall object.

**Usage**

```
add.flag(doitall, flagtype, flagnum, newval, phase)
```

**Arguments**

doitall  
flagtype  
flagnum  
newval  
phase

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
carry.effort.frq    carry.effort.frq.Rd
```

---

### Description

Replaces the effort in the last year with effort in the previous year and sets catch to -1.

### Usage

```
carry.effort.frq(data = out.data, fishery = 1, last = 2008)
```

### Arguments

```
data
fishery
last
```

### Author(s)

Shelton Harley and Nick Davies

### Examples

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
change.fishflag    change.fishflag.Rd
```

---

### Usage

```
change.fishflag(a, fisheries, flagnum, newvals)
```

### Arguments

```
a
fisheries
flagnum
newvals
```

### Author(s)

Simon Hoyle

### Examples

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

`change.flag`*change.flag.Rd*

---

**Usage**

```
change.flag(doitall, flagtype, flagnum, newval)
```

**Arguments**

```
doitall  
flagtype  
flagnum  
newval
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

`change.negflag`*change.negflag.Rd*

---

**Usage**

```
change.negflag(doitall, flagtype, flagnum, newval)
```

**Arguments**

```
doitall  
flagtype  
flagnum  
newval
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

|                 |                           |
|-----------------|---------------------------|
| change.size.frq | <i>change.size.frq.Rd</i> |
|-----------------|---------------------------|

---

**Usage**

```
change.size.frq(ver = 6, data = data, FISH = 1, LF.FILE = "P:/yft/2009/Data Prep
```

**Arguments**

```
ver
data
FISH
LF.FILE
WT.FILE
```

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|             |                       |
|-------------|-----------------------|
| change_data | <i>change_data.Rd</i> |
|-------------|-----------------------|

---

**Usage**

```
change_data(obj, searchtext, xlines, newline)
```

**Arguments**

```
obj
searchtext
xlines
newline
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```



---

|                |                          |
|----------------|--------------------------|
| check.eff.devs | <i>check.eff.devs.Rd</i> |
|----------------|--------------------------|

---

**Usage**

```
check.eff.devs(parfile, repfile, frqfile, parlim = 5.9)
```

**Arguments**

```
parfile  
repfile  
frqfile  
parlim
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

|                  |                            |
|------------------|----------------------------|
| check_flag_value | <i>check_flag_value.Rd</i> |
|------------------|----------------------------|

---

**Usage**

```
check_flag_value(parname, flagtype, flagnums, fishery = NA, flaglist = T)
```

**Arguments**

```
parname  
flagtype  
flagnums  
fishery  
flaglist
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

|              |                        |
|--------------|------------------------|
| clean.lfdata | <i>clean.lfdata.Rd</i> |
|--------------|------------------------|

---

**Usage**

```
clean.lfdata(infrq)
```

**Arguments**

```
infrq
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                |                          |
|----------------|--------------------------|
| compare.ce.frq | <i>compare.ce.frq.Rd</i> |
|----------------|--------------------------|

---

**Usage**

```
compare.ce.frq(file1, file2, fm, plotname, fdesc = "")
```

**Arguments**

```
file1
file2
fm
plotname
fdesc
```

**Author(s)**

Adam Langley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|             |                       |
|-------------|-----------------------|
| compare.frq | <i>compare.frq.Rd</i> |
|-------------|-----------------------|

---

**Usage**

```
compare.frq(file1, file2, fm = "all", plotname, fdesc = "", lwd=2, what=rep(TRUE, 3))
```

**Arguments**

```
file1
file2
fm
plotname
fdesc
lwd
what
```

**Author(s)**

Adam Langley and Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

|                  |                            |
|------------------|----------------------------|
| compare.size.frq | <i>compare.size.frq.Rd</i> |
|------------------|----------------------------|

---

**Usage**

```
compare.size.frq(frq1, frq2, fishery = 5, wt=T, prefix = "_", doyears, fdesc="", summary)
```

**Arguments**

```
frq1
frq2
fishery
wt
prefix
doyears
fdesc
summary
```

**Author(s)**

Nick Davies

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

compare\_par\_flags    *compare\_par\_flags.Rd*

---

**Description**

Compares the flags in two par files and reports differences.

**Usage**

```
compare_par_flags(par1, par2, flaglist = T)
```

**Arguments**

par1  
par2  
flaglist

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

`condor.go`*condor.go.Rd*

---

## Description

Used to compile MULTIFAN-CL files and submit a job to condor.

## Usage

```
condor.go(run.dir, frq.obj, tag.obj, doitall.obj, ini.obj, sub.obj, species = "a
```

## Arguments

```
run.dir
frq.obj
tag.obj
doitall.obj
ini.obj
sub.obj
species
condor_f
par.obj
run_now
fixpermissions
```

## Author(s)

Simon Hoyle and Pierre Kleiber

## Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

condor.go2

*condor.go2.Rd*


---

**Description**

Used to compile MULTIFAN-CL files and submit a job to condor.

**Usage**

```
condor.go2(run.dir, frq.obj, tag.obj, doitall.obj, ini.obj, sub.obj = suball, sp
```

**Arguments**

```
run.dir
frq.obj
tag.obj
doitall.obj
ini.obj
sub.obj
species
condor_f
par.obj
run_now
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

convert.frq.ver6

*Convert.frq.ver6.Rd*


---

**Description**

Converts a frq file frm vesion 5 to version 6.

**Usage**

```
convert.frq.ver6(a)
```

**Arguments**

a

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

copy.condor.files    *copy.condor.files.Rd*

---

**Usage**

```
copy.condor.files(rundir, condor.files = "./condor.files/")
```

**Arguments**

rundir  
condor.files

**Author(s)**

Simon Hoyle and Pierre Kleiber

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

create.missing.ce    *create.missing.ce.Rd*

---

**Usage**

```
create.missing.ce(data = data, yr = 2008, termfish)
```

**Arguments**

data  
yr  
termfish

**Author(s)**

Shelton Harley

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
crit.fishery.summary
      crit.fishery.summary.Rd
```

---

**Description**

Takes the output from do.critical.calcs and gets the key reference points

**Usage**

```
crit.fishery.summary(crit)
```

**Arguments**

```
crit
```

**Author(s)**

Shelton Harley

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
crit.summary      crit.summary.Rd
```

---

**Description**

Takes the output from do.critical.calcs and gets the key reference points

**Usage**

```
crit.summary(crit, years)
```



**Arguments**

crit  
years

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|            |                      |
|------------|----------------------|
| datfromstr | <i>datfromstr.Rd</i> |
|------------|----------------------|

---

**Usage**

```
datfromstr(datstring)
```

**Arguments**

datstring

**Author(s)**

Pierre Kleiber

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                   |                             |
|-------------------|-----------------------------|
| do.critical.calcs | <i>do.critical.calcs.Rd</i> |
|-------------------|-----------------------------|

---

**Description**

Uses the dimensioning stuff provided in the rep file and the mean lengths and weights at age, and takes the fishery specific catch at age from the ests file

**Usage**

```
do.critical.calcs(repfile = "P:/yft/2007/BaseYFT/yftfinal2007.rep", ests = "P:/y
```

**Arguments**

repfile  
ests

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

doit.rm\_flag

*doit.rm\_flag.Rd*

---

**Usage**

```
doit.rm_flag(a, flagtype, flag, value)
```

**Arguments**

a  
flagtype  
flag  
value

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

effortcreep*effortcreep.Rd*

---

**Description**

Adjusts effort in a specified fishery at a consistent rate through time.

**Usage**

```
effortcreep(frq.obj, fisheries, creep)
```

**Arguments**

```
frq.obj  
fisheries  
creep
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

fix\_growth*fix\_growth.Rd*

---

**Usage**

```
fix_growth(a)
```

**Arguments**

```
a
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
frq.change.nint      frq.change.nint.Rd
```

---

**Usage**

```
frq.change.nint(in.frq, add.lfint, add.wfint)
```

**Arguments**

```
in.frq
add.lfint
add.wfint
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

```
frq.remove.size.or.weight.data
      frq.remove.size.or.weight.data.Rd
```

---

**Description**

Provide a matrix (exclude) containing the following columns: Year | Month | week | fishery, and the function will remove the size and/or weight observations depending on the T/F flags.

**Usage**

```
frq.remove.size.or.weight.data(data = test.data, exclude = exclude, size = T, weight = F)
```

**Arguments**

```
data
exclude
size
weight
```

**Author(s)**

Shelton Harley

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
get.critical.age    get.critical.age.Rd
```

---

**Description**

Calculates the age (and associated length, and weight) where the weight of a cohort is maximised.

**Usage**

```
get.critical.age(data = Base.rep)
```

**Arguments**

data                      A .rep object

**Author(s)**

Shelton Harley

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
get.length.output    get.length.output.Rd
```

---

**Description**

Not for standard use, and now may be obsolete. Adjusts the weight data to focus on the areas with most of the catch.

**Usage**

```
get.length.output(REGION = 1, DIR = "P:/yft/2009/Data Preparation/size data/")
```

**Arguments**

REGION  
DIR

**Author(s)**

Adam Langley

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

get.outcomes

*get.outcomes.Rd*

---

**Description**

Extracts management information from result files

**Usage**

```
get.outcomes(file.rep, file.par, nofish = T)
```

**Arguments**

file.rep  
file.par  
nofish

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

`get.tag.structure` *get.tag.structure.Rd*

---

**Description**

Creates an object holding tag result information from the tag report file

**Usage**

```
get.tag.structure(tagrepfile="temporary_tag_report",tagfile="skj.tag",year1=1972
```

**Arguments**

```
tagrepfile
tagfile
year1
```

**Author(s)**

Pierre Kleiber

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

`get.weight.output` *get.weight.output.Rd*

---

**Description**

Not for standard use, and now may be obsolete. Adjusts the weight data to focus on the areas with most of the catch.

**Usage**

```
get.weight.output(REGION = 1, DIR = "P:/yft/2009/Data Preparation/size data/")
```

**Arguments**

```
REGION
DIR
```

**Author(s)**

Adam Langley

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|              |                        |
|--------------|------------------------|
| labels_store | <i>labels_store.Rd</i> |
|--------------|------------------------|

---

**Description**

Loads the labels.tmp file into an object.

**Usage**

```
labels_store(labelfile = basecase.labels)
```

**Arguments**

labelfile

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|             |                       |
|-------------|-----------------------|
| load.LFdata | <i>load.LFdata.Rd</i> |
|-------------|-----------------------|

---

**Description**

Loads length frequency data from a database via ODBC and labels it with region. These data can then be processed into MFCL format, via another function. Mainly included as an example.

**Usage**

```
load.LFdata(species = "ALB", gear = "L")
```

**Arguments**

species

gear



**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

```
make.projection.betyft.frq  
      make.projection.betyft.frq.Rd
```

---

**Description**

Formats MULTIFAN-CL frq file for projections. Has been made obsolete by a collection of more complex projection functions.

**Usage**

```
make.projection.betyft.frq(frq.in = base.frq, fish = 1:24, years = 10)
```

**Arguments**

```
frq.in  
fish  
years
```

**Author(s)**

Shelton Harley

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

|                 |                           |
|-----------------|---------------------------|
| map_all_pacific | <i>map_all_pacific.Rd</i> |
|-----------------|---------------------------|

---

### Description

Draw a map with dimensions as specified, adding the EEZ boundaries.

### Usage

```
map_all_pacific(plot_title = "", lims = c(100, 300, -45, 45), eezfile = "L:/alb/
```

### Arguments

```
plot_title
lims
eezfile
```

### Author(s)

Adam Langley and Simon Hoyle

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

|           |                     |
|-----------|---------------------|
| merge.frq | <i>merge.frq.Rd</i> |
|-----------|---------------------|

---

### Description

Combine catch, effort and size frequency data when merging fisheries together. Currently set up for the albacore assessment and needs adapting.

### Usage

```
merge.frq(frq.obj, oldf, newf, mergelf = FALSE)
```

### Arguments

```
frq.obj
oldf
newf
mergelf
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

merge.tag

*merge.tag.Rd*

---

**Description**

Change the fishery numbers for tag recoveries in a tag object.

**Usage**

```
merge.tag(tag.obj, oldf, newf)
```

**Arguments**

tag.obj

oldf

newf

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

|                |                          |
|----------------|--------------------------|
| merge_tag_objs | <i>merge_tag_objs.Rd</i> |
|----------------|--------------------------|

---

**Description**

Combine two tag objects into one.

**Usage**

```
merge_tag_objs(obj1, obj2, relgrps)
```

**Arguments**

```
obj1
obj2
relgrps
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                    |                              |
|--------------------|------------------------------|
| pack.fisheries.frq | <i>pack.fisheries.frq.Rd</i> |
|--------------------|------------------------------|

---

**Description**

Remove gaps between fishery numbers

**Usage**

```
pack.fisheries.frq(frq.obj)
```

**Arguments**

```
frq.obj
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
plot.base.comparison
      plot.base.comparison.Rd
```

---

**Description**

Plot F/FMSY against B/BMSY

**Usage**

```
plot.base.comparison(baseres, labs)
```

**Arguments**

```
baseres
labs
```

**Author(s)**

Adam Langley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

```
plot.biomass      plot.biomass.Rd
```

---

**Description**

Plots biomass by region and then combined Option to add on CI as a polygoon

**Usage**

```
plot.biomass(plotdir = "H:/rmfcl/test/figs/", plotrep = test, varfile = NULL, ty
```

**Arguments**

```
plotdir
plotrep
varfile
type
plotname
plottype
```

**Author(s)**

Pierre Kleiber and Shelton Harley

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
plot.biomass.combined  
plot.biomass.combined.Rd
```

---

**Description**

Plots biomass combined across all regions Option to add on CI as a polygon

**Usage**

```
plot.biomass.combined(plotdir = "H:/rmfcl/test/figs/", plotrep = test, varfile =
```

**Arguments**

```
plotdir  
plotrep  
varfile  
type  
plotname  
plottype
```

**Author(s)**

Pierre Kleiber

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

|             |                       |
|-------------|-----------------------|
| plot.F.time | <i>plot.F.time.Rd</i> |
|-------------|-----------------------|

---

**Description**

Plots annual F by year for adults and juveniles (as defined by the maturity ogive in the \*.ini file)

**Usage**

```
plot.F.time(plotdir = "H:/rmfcl/test/figs/", plotrep = "C:/assessments/alb/2008/
```

**Arguments**

```
plotdir
plotrep
inifile
plotname
plotype
COL
```

**Author(s)**

Pierre Kleiber

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                       |                                 |
|-----------------------|---------------------------------|
| plot.fishery.impact.r | <i>plot.fishery.impact.r.Rd</i> |
|-----------------------|---------------------------------|

---

**Description**

Does the fishery impact plot by taking the output straight from the plot.rep files You need to give the file names for the input files

**Usage**

```
plot.fishery.impact.r(plotdir = "H:/rmfcl/test/figs/", type = "Total", plotrep =
```

**Arguments**

plotdir  
 type  
 plotrep  
 impnames  
 plotname  
 plottype  
 COL

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

plot.Kobe

*plot.Kobe.Rd*

---

**Description**

Take MULTIFAN-CL results and produce a Kobe plot

**Usage**

```
plot.Kobe(plotdir = "S:/OFP Publications/Tuna Fishery Assessment Report/2007/Fig
```

**Arguments**

plotdir  
 plotrep  
 type  
 plotname  
 plottype  
 COL

**Author(s)**

Adam Langley and Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```



---

```
plot.Kobe.template.bw  
plot.Kobe.template.bw.Rd
```

---

**Description**

Make the template for a Kobe plot without colour

**Usage**

```
plot.Kobe.template.bw (Type)
```

**Arguments**

Type

**Author(s)**

Adam Langley and Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
plot.Kobe.template.col  
plot.Kobe.template.col.Rd
```

---

**Description**

Make the template for a Kobe plot

**Usage**

```
plot.Kobe.template.col (Type)
```

**Arguments**

Type

**Author(s)**

Adam Langley and Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

plot.mfcl.betyft09 *plot.mfcl.betyft09.Rd*

---

**Usage**

```
plot.mfcl.betyft09(lims = c(100, 260, -45, 45))
```

**Arguments**

lims

**Author(s)**

Adam Langley and Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

plot.nofishing *plot.nofishing.Rd*

---

**Usage**

```
plot.nofishing(plotdir = "H:/rmfcl/test/figs/", plotrep = testq0, type = "SSB",
```

**Arguments**

plotdir  
plotrep  
type  
plotname  
plottype  
COL

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
plot.nofishing.combined  
plot.nofishing.combined.Rd
```

---

**Description**

Plot the nofishing plots.

**Usage**

```
plot.nofishing.combined(plotdir = "H:/rmfcl/test/figs/", plotrep = testq0, type
```

**Arguments**

```
plotdir  
plotrep  
type  
plotname  
plottype  
COL
```

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

```
plot.pacific.alb    plot.pacific.alb.Rd
```

---

**Description**

Plots the Pacific and includes boundaries for the albacore tuna model.

**Usage**

```
plot.pacific.alb(plot_title = "", eez_dir = "I:/assessments/Pop dy modeling/MFCL
```

**Arguments**

```
plot_title  
eez_dir  
plot_eez
```

**Author(s)**

Adam Langley and Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

plot.pacific.skj     *plot.pacific.skj.Rd*

---

**Description**

Plots the Pacific and includes boundaries for the skipjack tuna model.

**Usage**

```
plot.pacific.skj(plot_title = "")
```

**Arguments**

plot\_title

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

plot.pacific.species  
                          *plot.pacific.species.Rd*

---

**Description**

Plots the Pacific and includes boundaries for the specified model.

**Usage**

```
plot.pacific.species(plot_title = "", uselims = NA, add.WCPFC = F, add.EPO = F,
```

**Arguments**

```
plot_title
uselims
add.WCPFC
add.EPO
sp
add.EEZ
eez_file
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

`plot.pacific.WCPFC` *plot.pacific.WCPFC.Rd*

---

**Description**

Plots the Pacific and includes boundaries for the yellowfin tuna model.

**Usage**

```
plot.pacific.WCPFC(plot_title = "", lims = c(100, 260, -45, 45))
```

**Arguments**

```
plot_title
lims
```

**Author(s)**

Adam Langley and Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
plot.pacific.yft      plot.pacific.yft.Rd
```

---

### Description

Plots the Pacific and includes boundaries for the yellowfin tuna model.

### Usage

```
plot.pacific.yft(plot_title = "", lims = c(100, 260, -45, 45), add.WCPFC = F)
```

### Arguments

```
plot_title
lims
add.WCPFC
```

### Author(s)

Adam Langley and Simon Hoyle

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
plot.recruitment      plot.recruitment.Rd
```

---

### Description

Plot recruitment by region and then combined Option to add on CI as a polygon for combined R only

### Usage

```
plot.recruitment(plotdir = "H:/rmfcl/test/figs/", plotrep = test, varfile = NULL)
```

### Arguments

```
plotdir
plotrep
varfile
plotname
plottype
```

**Author(s)**

Pierre Kleiber

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
plot.recruitment.combined  
plot.recruitment.combined.Rd
```

---

**Description**

Plots biomass combined across all regions Option to add on CI as a polygon

**Usage**

```
plot.recruitment.combined(plotdir = "H:/rmfcl/test/figs/", plotrep = test, varfi
```

**Arguments**

```
plotdir  
plotrep  
varfile  
plotname  
plottype
```

**Author(s)**

Pierre Kleiber

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

plot\_cpue\_cv\_frq     *plot\_cpue\_cv\_frq.Rd*

---

**Description**

Takes a version 6 frq file and par file and plots the CPUE and CVs for the chosen fisheries. Currently the fisheries need to have effort wts.

**Usage**

```
plot_cpue_cv_frq(frq, parf, fisheries)
```

**Arguments**

```
frq
parf
fisheries
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

ppath                     *ppath.Rd*

---

**Description**

Joins parts of a file path together without fussing with "/" signs.

**Usage**

```
ppath(p1,p2)
```

**Arguments**

```
p1
p2
```

**Author(s)**

Pierre Kleiber



**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

R4MFCL

*R4MFCL-package.Rd***Description**

R4MFCL is a collection of utility functions for stock assessments using the model MULTIFAN-CL (Fournier et al 1998; [www.multifan-cl.org](http://www.multifan-cl.org)). There are several groups of R4MFCL functions: - input and output functions, for reading MULTIFAN-CL files into R objects and writing them back out as text files in the form that MULTIFAN-CL accepts as input. - data manipulation functions, for editing and restructuring the input objects - plotting functions, for producing plots and maps from the result objects - information functions, for comparing objects and giving information about, for example, flag settings.

read.catchrep

*read.catchrep.Rd***Description**

Reads the catch.rep result file into an object.

**Usage**

```
read.catchrep(catchrep.file)
```

**Arguments**

```
catchrep.file
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read.ests

*read.ests.Rd*


---

### Description

Load the ests.rep file into an object.

### Usage

```
read.ests(rep.obj, ests = "C:/assessments/alb/2008/6_area/28.splitgr3/ests.rep",
```

### Arguments

```
rep.obj
ests
x
```

### Author(s)

Simon Hoyle

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read.fit

*read.fit.Rd*


---

### Description

Loads the observed and expected size frequency from the \*.fit file by fishery and time period.

### Usage

```
read.fit(fit.file)
```

### Arguments

```
fit.file
```

### Author(s)

Simon Hoyle

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read.frq

*read.frq.Rd*


---

**Description**

Reads in the frq file into a frq object for either version 4 or 6+.

**Usage**

```
read.frq(frq.file, frq.title = "", ntop = 0, fishdefs = NA)
```

**Arguments**

```
frq.file
```

```
frq.title
```

```
ntop
```

```
fishdefs
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

read.impact

*read.impact.Rd*


---

**Description**

Reads the \*.rep files from various impact runs into their own objects, and names them.

**Usage**

```
read.impact(impdir = "H:/rmfcl/test/", impnames = c("ll", "psassoc", "psunassoc")
```

**Arguments**

```
impdir
```

```
impnames
```

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read.ini

*read.ini.Rd*


---

**Description**

Reads the \*.ini data input file into an object.

**Usage**

```
read.ini(ini.file)
```

**Arguments**

```
ini.file
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read.par

*read.par.Rd*


---

**Description**

Reads the \*.par output and input MULTIFAN-CL parameter file into an object.

**Usage**

```
read.par(par.file)
```

**Arguments**

```
par.file
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read.rep

*read.rep.Rd*


---

**Description**

Reads the rep file, which contains most of the important results, into an object.

**Usage**

```
read.rep(rep.file)
```

**Arguments**

```
rep.file
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read.tag

*read.tag.Rd*


---

**Description**

Reads the \*.tag data input file into an object.

**Usage**

```
read.tag(tagfile)
```

**Arguments**

```
tagfile
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read.var

*read.var.Rd*


---

**Description**

Reads the \*.var result file into an object.

**Usage**

```
read.var(var.file)
```

**Arguments**

```
var.file
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

read\_nmd.frq

*read\_nmd.frq.Rd*


---

**Description**

Reads in the frq file into a frq object for either version 4 or 6+.

**Usage**

```
read_nmd.frq(frq.file, frq.title = "", ntop = 0, fishdefs = NA)
```

**Arguments**

```
frq.file
```

```
frq.title
```

```
ntop
```

```
fishdefs
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

reconstruct.frq.ce *reconstruct.frq.ce.Rd*

---

**Description**

Replaces the nominal effort in the original .FRQ file with stanadrdised effort based on a CPUE index. Not generalised - specific to bigeye 2008.

**Usage**

```
reconstruct.frq.ce(CPUE.file = "X:/yft/2009/Data Preparation/CPUE/indices/yft_JP
```

**Arguments**

```
CPUE.file  
data  
fishery
```

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
reconstruct.frq.size
      reconstruct.frq.size.Rd
```

---

**Description**

Pull in new size and weight frequency data and rebuild the frq object. Not generalized - specific to WCPO bigeye.

**Usage**

```
reconstruct.frq.size(data = data, FISH = 1, LF.FILE = "P:/yft/2009/Data Preparat
```

**Arguments**

```
data
FISH
LF.FILE
WT.FILE
```

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

```
region_single_frq  region_single_frq.Rd
```

---

**Description**

Change a frq object to a single region, removing all fisheries outside that region

**Usage**

```
region_single_frq(frq, region)
```

**Arguments**

```
frq
region
```

**Author(s)**

Simon Hoyle



**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

region\_single\_ini    *region\_single\_ini.Rd*

---

**Description**

Change an ini object to a single region, removing all fisheries outside that region

**Usage**

```
region_single_ini(ini)
```

**Arguments**

ini

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

region\_single\_tag    *region\_single\_tag.Rd*

---

**Description**

Change a tag object to a single region, removing all fisheries outside that region

**Usage**

```
region_single_tag(tag, region, keepfish)
```

**Arguments**

tag  
region  
keepfish

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
regroup_fishery_grps.doitall  
      regroup_fishery_grps.doitall.Rd
```

---

**Description**

Regroup all the fisheries in the vector *f* to the groups in the vector *newgrps* for the specified flag.

**Usage**

```
regroup_fishery_grps.doitall(doitall, f, flag, newgrps)
```

**Arguments**

```
doitall  
f  
flag  
newgrps
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
rename.fisheries.doitall  
rename.fisheries.doitall.Rd
```

---

**Description**

Rename all the fisheries in the vector oldfs to the numbers in the vector newfs.

**Usage**

```
rename.fisheries.doitall(doitall, oldfs, newfs)
```

**Arguments**

```
doitall  
oldfs  
newfs
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

```
rename.fisheries.frq  
rename.fisheries.frq.Rd
```

---

**Description**

Rename all the fisheries in the vector oldfish to the numbers in the vector newfish.

**Usage**

```
rename.fisheries.frq(frq.obj, oldfish, newfish)
```

**Arguments**

```
frq.obj  
oldfish  
newfish
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
rename.fisheries.tag
      rename.fisheries.tag.Rd
```

---

**Description**

Rename the fisheries in oldfish to the fishery numbers in newfish.

**Usage**

```
rename.fisheries.tag(tag.obj, oldfish, newfish)
```

**Arguments**

```
tag.obj
oldfish
newfish
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
rename.fishery.grps.doitall
      rename.fishery.grps.doitall.Rd
```

---

**Description**

Rename all the fisheries in the vector oldfs to the numbers in the vector newfs, for the specified flag.

**Usage**

```
rename.fishery.grps.doitall(doitall, oldfs, newfs, flag, keep = T, newgrps = c(0
```

**Arguments**

```
doitall
oldfs
newfs
flag
keep
newgrps
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

retro.frq

*retro.frq.Rd*


---

**Description**

Set a frq object up fo a retrospective analysis. Need some more testing.

**Usage**

```
retro.frq(frq.obj, retro.tag.obj = NA)
```

**Arguments**

```
frq.obj
retro.tag.obj
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|           |                     |
|-----------|---------------------|
| retro.tag | <i>retro.tag.Rd</i> |
|-----------|---------------------|

---

**Description**

Set a tag object up fo a retrospective analysis. Need some more testing.

**Usage**

```
retro.tag(tag.obj, yr)
```

**Arguments**

```
tag.obj  
yr
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

|                      |                                |
|----------------------|--------------------------------|
| rm_fisheries.doitall | <i>rm_fisheries.doitall.Rd</i> |
|----------------------|--------------------------------|

---

**Description**

Removes all flags for specified fisheries from the doitall.

**Usage**

```
rm_fisheries.doitall(a, rmfisheries)
```

**Arguments**

```
a  
rmfisheries
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

rm\_fisheries.frq    *rm\_fisheries.frq.Rd*

---

**Description**

Removes all catch and effort in specific fisheries.

**Usage**

```
rm_fisheries.frq(frq.obj, fishery)
```

**Arguments**

```
frq.obj  
fishery
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

rm\_fisheries.tag    *rm\_fisheries.tag.Rd*

---

**Description**

Removes all recoveries in specified fisheries from a tag object.

**Usage**

```
rm_fisheries.tag(tag.obj, fisheries)
```

**Arguments**

```
tag.obj  
fisheries
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                          |                       |
|--------------------------|-----------------------|
| <code>rm_fishflag</code> | <i>rm_fishflag.Rd</i> |
|--------------------------|-----------------------|

---

**Description**

Removes all occurrences of changes to a specified fish flag from the doitall.

**Usage**

```
rm_fishflag(doitall, flag)
```

**Arguments**

```
doitall
flag
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                              |                           |
|------------------------------|---------------------------|
| <code>rm_flag.doitall</code> | <i>rm_flag.doitall.Rd</i> |
|------------------------------|---------------------------|

---

**Description**

Removes all occurrences of changes to a specified flag from the doitall.

**Usage**

```
rm_flag.doitall(a, flagtype, flag, value)
```



**Arguments**

a  
flagtype  
flag  
value

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

run.profile

*run.profile.Rd*

---

**Description**

Run a likelihood profile analysis on a stock assessment. Needs to be generalized and tested.

**Usage**

```
run.profile(rundir, rungrp, startpar = NA, ptype = "Fmult", target, nsteps = 300)
```

**Arguments**

rundir  
rungrp  
startpar  
ptype  
target  
nsteps  
penalty

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do  help(data=index)  for the standard data sets.
```

---

seas.flag

*seas.flag.Rd*


---

**Description**

Change the doitall object so that the specified fishery is made seasonal.

**Usage**

```
seas.flag(a, fishery, flagnum, seasf.list)
```

**Arguments**

```
a
fishery
flagnum
seasf.list
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

seas.frq

*seas.frq.Rd*


---

**Description**

Change the frq object so that the specified fishery is made seasonal.

**Usage**

```
seas.frq(frq.obj, seas.fish)
```

**Arguments**

```
frq.obj
seas.fish
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

seas.tag

*seas.tag.Rd*


---

**Description**

Change the tag object so that the specified fishery is made seasonal.

**Usage**

```
seas.tag(tag.obj, fishlist)
```

**Arguments**

```
tag.obj
fishlist
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

setup.cpue

*setup.cpue.Rd*


---

**Description**

Replace particular CPUE series with other values, which are supplied. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.cpue(rungrp, sourcedir, cpue, spp)
```

**Arguments**

```

rungrp
sourcedir
cpue
spp

```

**Author(s)**

Simon Hoyle

**Examples**

```

##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.

```

---

|                |                          |
|----------------|--------------------------|
| setup.ffcreeep | <i>setup.ffcreeep.Rd</i> |
|----------------|--------------------------|

---

**Description**

Adjust the effort in specified fisheries to adjust for a steady increase in fishing power at a specified rate. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.ffcreeep(rungrp, creeprate)
```

**Arguments**

```

rungrp
creeprate

```

**Author(s)**

Simon Hoyle

**Examples**

```

##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.

```

---

|              |                        |
|--------------|------------------------|
| setup.growth | <i>setup.growth.Rd</i> |
|--------------|------------------------|

---

**Description**

Change the growth parameters to the values supplied in VBopt. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.growth(rungrp, VBopt)
```

**Arguments**

```
rungrp
VBopt
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                      |                                |
|----------------------|--------------------------------|
| setup.growth.offsets | <i>setup.growth.offsets.Rd</i> |
|----------------------|--------------------------------|

---

**Description**

Modifies the growth offsets to the specified values, and turns on their use and estimation in a specified phase. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.growth.offsets(rungrp, ageclasses, penwt, phase, tog)
```

**Arguments**

```
rungrp
ageclasses
penwt
phase
tog
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                 |                           |
|-----------------|---------------------------|
| setup.idphcatch | <i>setup.idphcatch.Rd</i> |
|-----------------|---------------------------|

---

**Description**

Replace the catches in Indonesia Phillippines fisheries with values supplied in a folder with prefix 'idph'. Needs modification to be more general. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.idphcatch(rungrp, sourcedir, idph, spp)
```

**Arguments**

```
rungrp
sourcedir
idph
spp
```

**Author(s)**

Nick Davies

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

setup.lensel

*setup.lensel.Rd*


---

**Description**

Change selectivity to fully length-based in the specified fisheries. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.lensel(rungrp, fisheries, tog)
```

**Arguments**

```
rungrp
fisheries
tog
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

setup.LFwt

*setup.LFwt.Rd*


---

**Description**

Change the likelihood weight on the length frequencies to the specified value in specified fisheries, defaulting to all fisheries. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.LFwt(rungrp, newLFwt)
```

**Arguments**

```
rungrp
newLFwt
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

setup.M

*setup.M.Rd*

---

**Description**

Change the starting value of mean natural mortality in the ini file, and turn off M estimation. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.M(rungrp, newM)
```

**Arguments**

rungrp

newM

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```



---

`setup.pscatch`*setup.pscatch.Rd*

---

**Description**

Replace the catches in purse seine fisheries (2011 WCPO bigeye) with values supplied in a folder with prefix 'PScatch'. Needs modification to be more general. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.pscatch(rungrp, sourcedir, PScatch, spp)
```

**Arguments**

```
rungrp
sourcedir
PScatch
spp
```

**Author(s)**

Nick Davies

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do  help(data=index)  for the standard data sets.
```

---

`setup.startyr`*setup.startyr.Rd*

---

**Description**

Changes the start year of the assessment. Runs start\_year.frq. Currently doesn't change the tag file. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.startyr(rungrp, newstartyr)
```

**Arguments**

```
rungrp
newstartyr
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

|                 |                           |
|-----------------|---------------------------|
| setup.steepness | <i>setup.steepness.Rd</i> |
|-----------------|---------------------------|

---

**Description**

Changes the fixed value of steepness in the assessment by editing the doitall file. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.steepness(rungrp, newsteep)
```

**Arguments**

rungrp  
newsteep

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

|                 |                           |
|-----------------|---------------------------|
| setup.timesplit | <i>setup.timesplit.Rd</i> |
|-----------------|---------------------------|

---

**Description**

Modifies the assessment files to include a time split, defined by the parameter splitx. The setup files are generally used in structural sensitivity analyses. They modify an object that contains all the MULTIFAN-CL input files.

**Usage**

```
setup.timesplit(rungrp, splitx, storefish)
```

**Arguments**

```
rungrp
splitx
storefish
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|          |                    |
|----------|--------------------|
| sort.frq | <i>sort.frq.Rd</i> |
|----------|--------------------|

---

**Description**

Sorts the data in a frq file with the fisheries and times in ascending order.

**Usage**

```
sort.frq(frq.obj)
```

**Arguments**

```
frq.obj
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
start_year.frq      start_year.frq.Rd
```

---

**Description**

Change the starting year of the assessment by removing all frq data before that time and changing the start year parameter.

**Usage**

```
start_year.frq(frq.obj, start_yr, halfyr = F)
```

**Arguments**

```
frq.obj
start_yr
halfyr
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
steepness.doit      steepness.doit.Rd
```

---

**Description**

Inserts a new line "recruitmentConstraints 01.par ###" after PHASE 1. This allows steepness to be fixed at a chosen level.

**Usage**

```
steepness.doit(doitall, new.steepness, add_header = T, gap = 2)
```

**Arguments**

```
doitall
new.steepness
```

```
add_header
gap
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

```
summarise.size.frq.bet
      summarise.size.frq.bet.Rd
```

---

**Description**

Takes two frq files from the bigeye assessment and compares the length and weight data on an annual basis for the range of years Just does the last 15 years at the moment. Specific for the 2009 BET assessment, and included as an example. .

**Usage**

```
summarise.size.frq.bet(frql, fishery = 5)
```

**Arguments**

```
frql
fishery
```

**Author(s)**

Shelton Harley

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|             |                       |
|-------------|-----------------------|
| tag_grps_rm | <i>tag_grps_rm.Rd</i> |
|-------------|-----------------------|

---

**Description**

Remove the specified tag groups from the tag object.

**Usage**

```
tag_grps_rm(tag.obj, keep)
```

**Arguments**

```
tag.obj
keep
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|                   |                             |
|-------------------|-----------------------------|
| timesplit.doitall | <i>timesplit.doitall.Rd</i> |
|-------------------|-----------------------------|

---

**Description**

Changes a doitall file to account for time splits. Time splits occur when a fishery is broken up into several fisheries by time, with dates and new fishery codes specified in the 'fishsplit' parameter.

**Usage**

```
timesplit.doitall(doitall, fishsplit, qsplit = T)
```

**Arguments**

```
doitall
fishsplit
qsplit
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|               |                         |
|---------------|-------------------------|
| timesplit.frq | <i>timesplit.frq.Rd</i> |
|---------------|-------------------------|

---

**Description**

Changes a frq file to account for time splits. Time splits occur when a fishery is broken up into several fisheries by time, with dates and new fishery codes specified in the 'fishsplit' parameter.

**Usage**

```
timesplit.frq(frq.obj, divyrs, div.fish)
```

**Arguments**

```
frq.obj
divyrs
div.fish
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

|               |                         |
|---------------|-------------------------|
| timesplit.tag | <i>timesplit.tag.Rd</i> |
|---------------|-------------------------|

---

**Description**

Changes a tag file to account for time splits. Time splits occur when a fishery is broken up into several fisheries by time, with dates and new fishery codes specified in the 'fishsplit' parameter.

**Usage**

```
timesplit.tag(tag.obj, fishsplit)
```

**Arguments**

```
tag.obj
fishsplit
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

varfromstr

---

*varfromstr.Rd*


---

**Usage**

```
varfromstr(datstring, cols = c(2:3))
```

**Arguments**

```
datstring
cols
```

**Author(s)**

Pierre Kleiber

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```



---

`write.frq`*write.frq.Rd*

---

**Description**

Writes out the frq file (catch and effort, size frequency and model structure).

**Usage**

```
write.frq(frqfile, frq.obj)
```

**Arguments**

`frqfile`

`frq.obj`

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

`write.ini`*write.ini.Rd*

---

**Description**

Writes out the ini file in MULTIFAN-CL inoput format, from an ini object.

**Usage**

```
write.ini(ini.file, ini.obj, old.format=FALSE)
```

**Arguments**

`ini.file`

`ini.obj`

`old.format`

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

write.par

*write.par.Rd*


---

**Description**

Writes out the par file from a par object.

**Usage**

```
write.par(par.file, par.obj)
```

**Arguments**

```
par.file
par.obj
```

**Author(s)**

Simon Hoyle

**Examples**

```
##----- Should be DIRECTLY executable !! -----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
```

---

write.tag

*write.tag.Rd*


---

**Description**

Writes a tag object out into a \*.tag text file for input to MULTIFAN-CL.

**Usage**

```
write.tag(tagfile, tag.obj)
```

**Arguments**

```
tagfile
tag.obj
```

**Author(s)**

Simon Hoyle

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

---

|               |                         |
|---------------|-------------------------|
| write_nmd.frq | <i>write_nmd.frq.Rd</i> |
|---------------|-------------------------|

---

**Description**

An alternative write.frq function - writes out the frq file (catch and effort, size frequency and model structure)

**Usage**

```
write_nmd.frq(new.frq, frq.obj)
```

**Arguments**

```
new.frq  
frq.obj
```

**Author(s)**

Nick Davies

**Examples**

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.
```

# Index

## \*Topic \textasciitildekw1

add.catch.frq, [4](#)  
add.cpue.frq, [4](#)  
add.flag, [5](#)  
carry.effort.frq, [6](#)  
change.fishflag, [6](#)  
change.flag, [7](#)  
change.negflag, [7](#)  
change.size.frq, [8](#)  
change\_data, [8](#)  
check.eff.devs, [9](#)  
check\_flag\_value, [9](#)  
clean.lfdata, [10](#)  
compare.ce.frq, [10](#)  
compare.frq, [11](#)  
compare.size.frq, [11](#)  
compare\_par\_flags, [12](#)  
condor.go, [13](#)  
condor.go2, [14](#)  
convert.frq.ver6, [14](#)  
copy.condor.files, [15](#)  
create.missing.ce, [15](#)  
crit.fishery.summary, [16](#)  
crit.summary, [16](#)  
datfromstr, [17](#)  
do.critical.calcs, [17](#)  
doit.rm\_flag, [18](#)  
effortcreep, [19](#)  
fix\_growth, [19](#)  
frq.change.nint, [20](#)  
frq.remove.size.or.weight.data,  
    [20](#)  
get.critical.age, [21](#)  
get.length.output, [21](#)  
get.outcomes, [22](#)  
get.tag.structure, [23](#)  
get.weight.output, [23](#)  
labels\_store, [24](#)  
load.LFdata, [24](#)  
make.projection.betyft.frq,  
    [25](#)  
map\_all\_pacific, [26](#)  
merge.frq, [26](#)  
merge.tag, [27](#)  
merge\_tag\_objs, [28](#)  
pack.fisheries.frq, [28](#)  
plot.base.comparison, [29](#)  
plot.biomass, [29](#)  
plot.biomass.combined, [30](#)  
plot.F.time, [31](#)  
plot.fishery.impact.r, [31](#)  
plot.Kobe, [32](#)  
plot.Kobe.template.bw, [33](#)  
plot.Kobe.template.col, [33](#)  
plot.mfcl.betyft09, [34](#)  
plot.nofishing, [34](#)  
plot.nofishing.combined, [35](#)  
plot.pacific.alb, [35](#)  
plot.pacific.skj, [36](#)  
plot.pacific.species, [36](#)  
plot.pacific.WCPFC, [37](#)  
plot.pacific.yft, [38](#)  
plot.recruitment, [38](#)  
plot.recruitment.combined, [39](#)  
plot\_cpue\_cv\_frq, [40](#)  
ppath, [40](#)  
read.catchrep, [41](#)  
read.ests, [42](#)  
read.fit, [42](#)  
read.frq, [43](#)  
read.impact, [43](#)  
read.ini, [44](#)  
read.par, [44](#)  
read.rep, [45](#)  
read.tag, [45](#)  
read.var, [46](#)  
read\_nmd.frq, [46](#)  
reconstruct.frq.ce, [47](#)  
reconstruct.frq.size, [48](#)  
region\_single\_frq, [48](#)  
region\_single\_ini, [49](#)  
region\_single\_tag, [49](#)  
regroup\_fishery\_grps.doitall,  
    [50](#)  
rename.fisheries.doitall, [51](#)  
rename.fisheries.frq, [51](#)

- rename.fisheries.tag, 52
- rename.fishery.grps.doitall, 52
- retro.frq, 53
- retro.tag, 54
- rm\_fisheries.doitall, 54
- rm\_fisheries.frq, 55
- rm\_fisheries.tag, 55
- rm\_fishflag, 56
- rm\_flag.doitall, 56
- run.profile, 57
- seas.flag, 58
- seas.frq, 58
- seas.tag, 59
- setup.cpue, 59
- setup.effcreep, 60
- setup.growth, 61
- setup.growth.offsets, 61
- setup.idphcatch, 62
- setup.lensel, 63
- setup.LFwt, 63
- setup.M, 64
- setup.pscatch, 65
- setup.startyr, 65
- setup.steeptness, 66
- setup.timesplit, 67
- sort.frq, 67
- start\_year.frq, 68
- steeptness.doit, 68
- summarise.size.frq.bet, 69
- tag\_grps\_rm, 70
- timesplit.doitall, 70
- timesplit.frq, 71
- timesplit.tag, 71
- varfromstr, 72
- write.frq, 73
- write.ini, 73
- write.par, 74
- write.tag, 74
- write\_nmd.frq, 75
- \*Topic \textasciitildekw2**
  - add.catch.frq, 4
  - add.cpue.frq, 4
  - add.flag, 5
  - carry.effort.frq, 6
  - change.fishflag, 6
  - change.flag, 7
  - change.negflag, 7
  - change.size.frq, 8
  - change\_data, 8
  - check.eff.devs, 9
  - check\_flag\_value, 9
  - clean.lfdata, 10
  - compare.ce.frq, 10
  - compare.frq, 11
  - compare.size.frq, 11
  - compare\_par\_flags, 12
  - condor.go, 13
  - condor.go2, 14
  - convert.frq.ver6, 14
  - copy.condor.files, 15
  - create.missing.ce, 15
  - crit.fishery.summary, 16
  - crit.summary, 16
  - datfromstr, 17
  - do.critical.calcs, 17
  - doit.rm\_flag, 18
  - effortcreep, 19
  - fix\_growth, 19
  - frq.change.nint, 20
  - frq.remove.size.or.weight.data, 20
  - get.critical.age, 21
  - get.length.output, 21
  - get.outcomes, 22
  - get.tag.structure, 23
  - get.weight.output, 23
  - labels\_store, 24
  - load.LFdata, 24
  - make.projection.betyft.frq, 25
  - map\_all\_pacific, 26
  - merge.frq, 26
  - merge.tag, 27
  - merge\_tag\_objs, 28
  - pack.fisheries.frq, 28
  - plot.base.comparison, 29
  - plot.biomass, 29
  - plot.biomass.combined, 30
  - plot.F.time, 31
  - plot.fishery.impact.r, 31
  - plot.Kobe, 32
  - plot.Kobe.template.bw, 33
  - plot.Kobe.template.col, 33
  - plot.mfcl.betyft09, 34
  - plot.nofishing, 34
  - plot.nofishing.combined, 35
  - plot.pacific.alb, 35
  - plot.pacific.skj, 36
  - plot.pacific.species, 36
  - plot.pacific.WCPFC, 37
  - plot.pacific.yft, 38
  - plot.recruitment, 38
  - plot.recruitment.combined, 39

- plot\_cpue\_cv\_frq, 40
- ppath, 40
- read.catchrep, 41
- read.ests, 42
- read.fit, 42
- read.frq, 43
- read.impact, 43
- read.ini, 44
- read.par, 44
- read.rep, 45
- read.tag, 45
- read.var, 46
- read\_nmd.frq, 46
- reconstruct.frq.ce, 47
- reconstruct.frq.size, 48
- region\_single\_frq, 48
- region\_single\_ini, 49
- region\_single\_tag, 49
- regroup\_fishery\_grps.doitall, 50
- rename.fisheries.doitall, 51
- rename.fisheries.frq, 51
- rename.fisheries.tag, 52
- rename.fishery\_grps.doitall, 52
- retro.frq, 53
- retro.tag, 54
- rm\_fisheries.doitall, 54
- rm\_fisheries.frq, 55
- rm\_fisheries.tag, 55
- rm\_fishflag, 56
- rm\_flag.doitall, 56
- run.profile, 57
- seas.flag, 58
- seas.frq, 58
- seas.tag, 59
- setup.cpue, 59
- setup.effcreep, 60
- setup.growth, 61
- setup.growth.offsets, 61
- setup.idphcatch, 62
- setup.lensel, 63
- setup.LFwt, 63
- setup.M, 64
- setup.pscatch, 65
- setup.startyr, 65
- setup.steepness, 66
- setup.timesplit, 67
- sort.frq, 67
- start\_year.frq, 68
- steepness.doit, 68
- summarise.size.frq.bet, 69
- tag\_grps\_rm, 70
- timesplit.doitall, 70
- timesplit.frq, 71
- timesplit.tag, 71
- varfromstr, 72
- write.frq, 73
- write.ini, 73
- write.par, 74
- write.tag, 74
- write\_nmd.frq, 75
- add.catch.frq, 4
- add.cpue.frq, 4
- add.flag, 5
- carry.effort.frq, 6
- change.fishflag, 6
- change.flag, 7
- change.negflag, 7
- change.size.frq, 8
- change\_data, 8
- check.eff.devs, 9
- check\_flag\_value, 9
- clean.lfdata, 10
- compare.ce.frq, 10
- compare.frq, 11
- compare.size.frq, 11
- compare\_par\_flags, 12
- condor.go, 13
- condor.go2, 14
- convert.frq.ver6, 14
- copy.condor.files, 15
- create.missing.ce, 15
- crit.fishery.summary, 16
- crit.summary, 16
- datfromstr, 17
- do.critical.calcs, 17
- doit.rm\_flag, 18
- effortcreep, 19
- fix\_growth, 19
- frq.change.nint, 20
- frq.remove.size.or.weight.data, 20
- get.critical.age, 21
- get.length.output, 21
- get.outcomes, 22
- get.tag.structure, 23
- get.weight.output, 23
- labels\_store, 24

load.LFdata, 24  
 make.projection.betyft.frq, 25  
 map\_all\_pacific, 26  
 merge.frq, 26  
 merge.tag, 27  
 merge\_tag\_objs, 28  
  
 pack.fisheries.frq, 28  
 plot.base.comparison, 29  
 plot.biomass, 29  
 plot.biomass.combined, 30  
 plot.F.time, 31  
 plot.fishery.impact.r, 31  
 plot.Kobe, 32  
 plot.Kobe.template.bw, 33  
 plot.Kobe.template.col, 33  
 plot.mfcl.betyft09, 34  
 plot.nofishing, 34  
 plot.nofishing.combined, 35  
 plot.pacific.alb, 35  
 plot.pacific.skj, 36  
 plot.pacific.species, 36  
 plot.pacific.WCPFC, 37  
 plot.pacific.yft, 38  
 plot.recruitment, 38  
 plot.recruitment.combined, 39  
 plot\_cpue\_cv\_frq, 40  
 ppath, 40  
  
 R4MFCL, 41  
 read.catchrep, 41  
 read.ests, 42  
 read.fit, 42  
 read.frq, 43  
 read.impact, 43  
 read.ini, 44  
 read.par, 44  
 read.rep, 45  
 read.tag, 45  
 read.var, 46  
 read\_nmd.frq, 46  
 reconstruct.frq.ce, 47  
 reconstruct.frq.size, 48  
 region\_single\_frq, 48  
 region\_single\_ini, 49  
 region\_single\_tag, 49  
 regroup\_fishery\_grps.doitall, 50  
 rename.fisheries.doitall, 51  
 rename.fisheries.frq, 51  
 rename.fisheries.tag, 52  
 rename.fishery\_grps.doitall, 52  
 retro.frq, 53  
 retro.tag, 54  
 rm\_fisheries.doitall, 54  
 rm\_fisheries.frq, 55  
 rm\_fisheries.tag, 55  
 rm\_fishflag, 56  
 rm\_flag.doitall, 56  
 run.profile, 57  
  
 seas.flag, 58  
 seas.frq, 58  
 seas.tag, 59  
 setup.cpue, 59  
 setup.effcreep, 60  
 setup.growth, 61  
 setup.growth.offsets, 61  
 setup.idphcatch, 62  
 setup.lensel, 63  
 setup.LFwt, 63  
 setup.M, 64  
 setup.pscatch, 65  
 setup.startyr, 65  
 setup.steepness, 66  
 setup.timesplit, 67  
 sort.frq, 67  
 start\_year.frq, 68  
 steepness.doit, 68  
 summarise.size.frq.bet, 69  
  
 tag\_grps\_rm, 70  
 timesplit.doitall, 70  
 timesplit.frq, 71  
 timesplit.tag, 71  
  
 varfromstr, 72  
  
 write.frq, 73  
 write.ini, 73  
 write.par, 74  
 write.tag, 74  
 write\_nmd.frq, 75