

# Package ‘Riscam’

June 3, 2012

**Type** Package

**Title** What the package does (short line)

**Version** 1.0

**Date** 2010-11-10

**Author** Who wrote it

**Maintainer** Who to complain to <yourfault@somewhere.net>

**Description** More about what it does (maybe more than one line)

**License** What license is it under?

**LazyLoad** yes

## R topics documented:

Riscam-package . . . . .	1
A . . . . .	2
read.admb . . . . .	3
<b>Index</b>	<b>5</b>

---

Riscam-package	<i>This package contains supporting functions and routines for the stock assessment program iscam</i>
----------------	---

---

## Description

Provides R support for Integrated Statistical Catch Age Model (iscam), includes routines for reading ADMB output and custom graphics and table routines for summarizing model results.

## Details

Package:	Riscam
Type:	Package
Version:	1.0
Date:	2010-11-10
License:	GPL
LazyLoad:	1 yes

~~ An overview of how to use the package, including the most important ~~ ~ functions ~~

### Author(s)

Author: Steven Martell Maintainer: Steve Martell <s.martell@fisheries.ubc.ca>

### See Also

~~ Optional links to other man pages, e.g. ~~ ~ <pkg> ~~

---

A

*Pacific hake results*

---

### Description

Output from the hake assessment that is provided in the iscam users guide

### Usage

data (A)

### Format

The format is: List of 42 \$ f : num -379 \$ nlvec : num [1:6, 1:2] -121.6 30.4 -407.5 55.2 11 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:2] "V1" "V2" \$ ro : num 3.21 \$ rbar : num 0.845 \$ bo : num 3.75 \$ kappa : num 8.8 \$ m : num 0.228 \$ tau : num 1.35 \$ sig : num 0.56 \$ age\_tau2 : num [1:2] 0.387 0.681 \$ ngear : num 2 \$ yr : num [1:33] 1977 1978 1979 1980 1981 ... \$ yrs : num [1:34] 1977 1978 1979 1980 1981 ... \$ iyr : num [1:13] 1977 1980 1983 1986 1989 ... \$ age : num [1:15] 1 2 3 4 5 6 7 8 9 10 ... \$ la : num [1:15] 14.2 24.6 32.1 37.5 41.5 ... \$ wa : num [1:15] 0.0144 0.0743 0.1652 0.2646 0.3574 ... \$ fa : num [1:15] 1.32e-05 1.16e-03 3.58e-02 2.19e-01 3.53e-01 ... \$ log\_sel : num [1:66, 1:16] 1 1 1 1 1 1 1 1 1 1 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:16] "V1" "V2" "V3" "V4" ... \$ vax : num [1:15] 0 0 0 0 0 0 0 0 0 0 ... \$ obs\_ct : num [1:2, 1:33] 0.133 0 0.104 0 0.137 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:33] "V1" "V2" "V3" "V4" ... \$ ct : num [1:2, 1:33] 1.33e-01 3.55e-31 1.04e-01 3.31e-31 1.37e-01 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:33] "V1" "V2" "V3" "V4" ... \$ ft : num [1:2, 1:33] 0.326 0 0.32 NA 0.36 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:33] "V1" "V2" "V3" "V4" ... \$ bt : num [1:34] 1.15 1.06 1.09 1.09 1.34 ... \$ sbt : num [1:34] 1.028 0.926 0.804 0.715 0.935 ... \$ rectype : num 1 \$ rt : num [1:32] 4.346 0.262 0.77 17.201 0.19 ... \$ log\_rt(syr,nyr): num [1:33] -1.123 1.469 -1.339 -0.262 2.845 ... \$ delta : num [1:32] 1.471 -1.303 -0.175 2.977 -1.626 ... \$ it : num [1:13] 1.92 2.12 1.65 2.86 1.24 ... \$ pit : num [1:13] 0.977 0.825 1.504 2.684 3.227 ... \$ epsilon : num [1:13] 0.6728 0.9408 0.0911 0.0624 -0.9579 ... \$ F : num [1:33, 1:15] 0.10063 0.01509 0.00483 0.00547 0.0261 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:15] "V1" "V2" "V3" "V4" ... \$ a\_sage : num [1:2] 2 2 \$ a\_nage : num [1:2] 15 15 \$ A : num [1:46, 1:16] 1977 1978 1979 1980 1981 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:16] "V1" "V2" "V3" "V4" ... \$ Ahat : num [1:46, 1:16] 1977 1978 1979 1980 1981 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:16] "V1" "V2" "V3" "V4" ... \$ N : num [1:34, 1:15] 0.325 4.346 0.262 0.77 17.201 ... ..- attr(\*, "dimnames")=List of 2 .. ..\$ : NULL .. ..\$ : chr [1:15] "V1" "V2" "V3" "V4" ... \$ fmsy : num 0.322 \$ msy : num 0.206 \$ bmsy : num 1.48 \$ fit :List of 10 ..\$ npar : int 174 ..\$ nlogl : num -379 ..\$ maxgrad : num 5.41e-05 ..\$ npar : num 175 ..\$ logDetHess: num 888 ..\$ names : chr [1:175] "theta[1]" "theta[2]" "theta[3]" "theta[4]" ... ..\$ est : num [1:175] 1.167 0.688 -1.478 -0.168 0.294 ... ..\$ std : num [1:175] 0.3258 0.2142 0.0489 0.1187 0.0435 ... ..\$ cor : num [1:175, 1:175]

```
1 -0.238 0.126 0.116 -0.215 ... ..$ cov : num [1:175, 1:175] 0.10614 -0.01659 0.00201 0.00448
-0.00304 ...
```

## Examples

```
data(A)
## maybe str(A) ; plot(A) ...
```

---

read.admb

*Data input*


---

## Description

Reads the parameter files, correlation file, and report file outputs from an ADMB program.

## Usage

```
read.admb(ifile)
read.fit(ifile)
read.rep(fn)
read.psv(fn, nsamples=10000)
```

## Arguments

ifile	the file name (without the extension) of the ADMB program
fn	the file name with the extension, use for read.rep & read.psv
nsamples	the number of posterior samples to read in from the binary *.psv file

## Details

The function read.admb reads the contents of the ADMB output parameter file (\*.par) and the correlation file (\*.cor) as well as the report file (\*.rep). The function read.rep, just reads the report file only.

The read.psv function reads the binary output from the -mcsave command line option in ADMB. The default number of samples (rows) is 10,000. Note that this function is also called from read.admb if, and only if, the \*.psv file exists.

## Author(s)

S. Martell, Anders Nielsen, and the original contribution came from George Watters

## Examples

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

## The function is currently defined as
function(ifile)
{
ret=read.fit(ifile)
```

```
fn=paste(ifile, '.rep', sep='')
A=read.rep(fn)
A$fit=ret

pfn=paste(ifile, '.psv', sep='')
if(file.exists(pfn))
A$mc=read.psv(pfn)

return(A)
}
```

# Index

## \*Topic **ADMB**

`read.admb`, [3](#)

## \*Topic **datasets**

A, [2](#)

## \*Topic **iscam**

`read.admb`, [3](#)

## \*Topic **package**

Riscam-package, [1](#)

<pkg>, [2](#)

A, [2](#)

`read.admb`, [3](#)

`read.fit(read.admb)`, [3](#)

`read.psv(read.admb)`, [3](#)

`read.rep(read.admb)`, [3](#)

Riscam(Riscam-package), [1](#)

Riscam-package, [1](#)