



Introducing selfisher: open source software for statistical analyses of fishing gear selectivity

Brooks, Mollie Elizabeth; Melli, Valentina; Savina, Esther Anne Charlotte Marie; Santos, Juan; Millar, Russell B.; O'Neill, Finbarr G; Veiga-Malta, Tiago; Krag, Ludvig Ahm; Feekings, Jordan P.

Published in:
Canadian Journal of Fisheries and Aquatic Sciences

Link to article, DOI:
[10.1139/cjfas-2021-0099](https://doi.org/10.1139/cjfas-2021-0099)

Publication date:
2022

Document Version
Other version

[Link back to DTU Orbit](#)

Citation (APA):
Brooks, M. E., Melli, V., Savina, E. A. C. M., Santos, J., Millar, R. B., O'Neill, F. G., Veiga-Malta, T., Krag, L. A., & Feekings, J. P. (2022). Introducing selfisher: open source software for statistical analyses of fishing gear selectivity. *Canadian Journal of Fisheries and Aquatic Sciences*, 79(8), 1189-1197. <https://doi.org/10.1139/cjfas-2021-0099>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Supplementary material D: Catch comparison analyses of paired hauls of *Nephrops* twin-rigged trawls

7 Dec 2020

This example is based on the data from Melli et al. (2018). An anterior gear modification, namely the counter-herding device FLEXSELECT, was tested in a twin-rig configuration, where two identical bottom trawls were towed in parallel. One trawl was equipped with FLEXSELECT, while the other worked as baseline. The aim of the study was to determine if FLEXSELECT could prevent fish species from entering the trawl in a *Nephrops* (*Nephrops norvegicus*) fishery. Data for haddock *Melanogrammus aeglefinus* were collected for 21 hauls. Of these, 13 were conducted in day-time and 8 in night-time. In each haul and for each trawl, the total length (rounded down to the lower centimetre) of all haddock individuals was recorded.

Preliminaries

```
library(selfisher)
library(plyr) #for aggregating data across hauls
library(ggplot2); theme_set(theme_bw())
library(bbmle) #for AICtab
```

```
## Loading required package: stats4
```

```
library(stats4)
library(splines)
```

Data structure

First, we load the data and check out the variables. Here we can see that all the hauls are contained in one data frame (long format), with each row corresponding to a length class in a given haul. The number of individuals of that length-class caught in each trawl is reported in the columns TEST1 (test trawl) and TEST2 (baseline trawl). The column TIME classifies the haul as day-time (D) or night-time (N).

```
data("comphaddock")
head(comphaddock)
```

```
##   SPECIES HAUL TIME LENGTH TEST1 TEST2
## 1   Had   10   D    5.5     0     0
## 2   Had   10   D    6.5     0     0
## 3   Had   10   D    7.5     0     0
## 4   Had   10   D    8.5     0     0
## 5   Had   10   D    9.5     0     0
## 6   Had   10   D   10.5     0     1
```

To understand if the test trawl caught significantly fewer individuals of a given length-class than the baseline trawl we need to perform a catch comparison analysis (Krag et al., 2015). This analysis estimates the probability of catching an individual of a given length in the test trawl given that it was caught in either trawl.

In addition, the analysis aims at determining if the length-based efficiency of FLEXSELECT presents diel differences, as haddock is known to migrate vertically in the water column during the night.

Transforming data

Before fitting a model in `selfisher`, the following preparatory steps need to be performed:

- 1) Convert counts (i.e. number of individuals caught per length-class) to totals, proportions and ratios;

```
comphaddock = transform(comphaddock,
  total = TEST1 + TEST2,
  prop = TEST1 / (TEST1+TEST2),
  ratio = TEST1 / TEST2
)
```

This step is required because, unlike with other GLM functions for binomial regression, it is not possible in `selfisher` to specify the binomial variable as a two-column response variable, e.g. `cbind(N_TEST1, N_TEST2)`.

- 2) Remove eventual length classes where no individuals were caught;

```
comphaddock = subset(comphaddock, !is.na(prop))
```

- 3) Scale the length. This step is necessary for numerical stability, as a model often used for catch comparison analyses is the polynomial of order four, which requires to raise the length to the 4th power.

```
comphaddock$s1 = scale(comphaddock$LENGTH)
head(comphaddock)
```

```
##   SPECIES HAUL TIME LENGTH TEST1 TEST2 total      prop      ratio      s1
## 6      Had   10   D   10.5     0     1     1 0.00000000 0.0000000 -1.750253
## 9      Had   10   D   13.5     0     5     5 0.00000000 0.0000000 -1.433650
## 10     Had   10   D   14.5     1    10    11 0.09090909 0.1000000 -1.328116
## 11     Had   10   D   15.5     8    15    23 0.34782609 0.5333333 -1.222581
## 12     Had   10   D   16.5    11    11    22 0.50000000 1.0000000 -1.117047
## 13     Had   10   D   17.5     7    21    28 0.25000000 0.3333333 -1.011513
```

Model fitting

The following is a typical model for catch comparison data with multiple paired hauls, which models the proportion of fish in the test versus the baseline trawl (`prop`) as a function of length (`s1`). This is expressed in the `selfisher` function by a two sided formula with the proportion (`prop`) on the left side and fixed and random effects on the right side. Formulas in the `selfisher` package follow the convention of the `lme4` and `glmmTMB` packages.

Since we are interested in determining if there is a length-dependent difference in the efficiency of the Test gear between day-time and night-time, we include in the model `TIME` as an explanatory variable.

```
m1 = selfisher(prop~(s1+I(s1^2)+I(s1^3)+I(s1^4))*TIME, total = total, comphaddock, haul = HAUL)
```

In this example all individuals were length-measured (i.e. there was no subsampling). In case of a subsampled species, an offset or q-ratio needs to be specified in the model. The `selfisher` function takes the `total` number of fish in the test and baseline using a separate argument, `total`. The argument `haul` needs to be specified in order to perform double-bootstrapping as demonstrated below. Otherwise, it could be omitted from the model specification as it doesn't affect the fit. The `haul` argument tells the software how to group the data for resampling in the bootstrapping procedure.

Alternative model

An alternative approach would consist in fitting a spline (Miller, 2013), often preferred to polynomial interpolation because it yields similar results while avoiding Runge's phenomenon (i.e. oscillation at the edges of the length range represented in the data).

```
m2 = selfisher(prop~(bs(s1, df=4))*TIME, total = total, comphaddock, haul=HAUL)
m3 = selfisher(prop~(bs(s1, df=5))*TIME, total = total, comphaddock, haul=HAUL)
```

Model comparison

We can determine which model fits best using the Akaike's Information Criterion (AIC; Akaike, 1974).

```
AICtab(m1, m2, m3)
```

```
##      dAIC df
## m3  0.0 12
## m1  0.4 10
## m2  6.8 10
```

This tells us that `m1` and `m3` show equally good fit (0.4 delta AIC units). According to the parsimony rule, we selected `m1` that is a simpler model.

Predictions

To see how the model fits the data, we need to plot observations and predictions together, keeping them separated by `TIME`.

```
dat_D = comphaddock[ which(comphaddock$TIME=='D'), ]
dat_N = comphaddock[ which(comphaddock$TIME=='N'), ]
newdata1 = data.frame(LENGTH = unique(dat_D$LENGTH), TIME = "D")
newdata2 = data.frame(LENGTH = unique(dat_N$LENGTH), TIME = "N")

newdata = rbind(newdata1, newdata2, deparse.level = 1)

newdata = transform(newdata,
                    s1 = (LENGTH-mean(comphaddock$LENGTH))/sqrt(var(comphaddock$LENGTH)),
```

```

        total = 1,
        HAUL = NA
    )
newdata$prop = predict(m1, newdata = newdata, type = "response")

```

Confidence intervals by double-bootstrapping

We then estimate the 95% Efron Confidence intervals (Efron, 1982), by accounting for within- and between-hauls variation (Millar, 1993). The code below resamples hauls, then resamples fish within hauls, fits the model to the resampled data, then makes predictions from the model onto `newdata`. The type of predictions we want in this case are the catch comparison rates, thus we specify `type="response"`. To read about the `predict` function, type `?predict.selfisher` in the R console.

Windows bootstrapping in parallel

```

library(snow)

ncpus = 4
cl = makeCluster(rep("localhost", ncpus))
clusterExport(cl, "newdata")
bs = bootSel(m1, nsim = 1000, parallel = "snow", cl = cl,
            FUN = function(mod){predict(mod, newdata = newdata, type = "response")})
stopCluster(cl)

```

Code for bootstrapping in Mac and Linux

```

bs = bootSel(m1, nsim = 1000, parallel = "multicore", ncpus = 4,
            FUN = function(mod){predict(mod, newdata = newdata, type = "response")})

```

Then we calculate quantiles across bootstraps for each row of `newdata`.

```

quants = apply(bs$t, 2, quantile, c(0.025, 0.5, 0.975))
newdata[,c("lo", "mid", "hi")] = t(quants)

```

Plotting with CIs

Here, we plot the modelled catch comparison curve with CIs and experimental observations, obtained by aggregating the hauls per TIME.

```

sumdat1 = ddply(dat_D, ~LENGTH+s1, summarize,
              prop = sum(TEST1)/sum(total),
              ratio = sum(TEST1)/sum(TEST2),
              total = sum(total),
              TIME = "D")

sumdat2 = ddply(dat_N, ~LENGTH+s1, summarize,

```

```

prop = sum(TEST1)/sum(total),
ratio = sum(TEST1)/sum(TEST2),
total = sum(total),
TIME = "N")

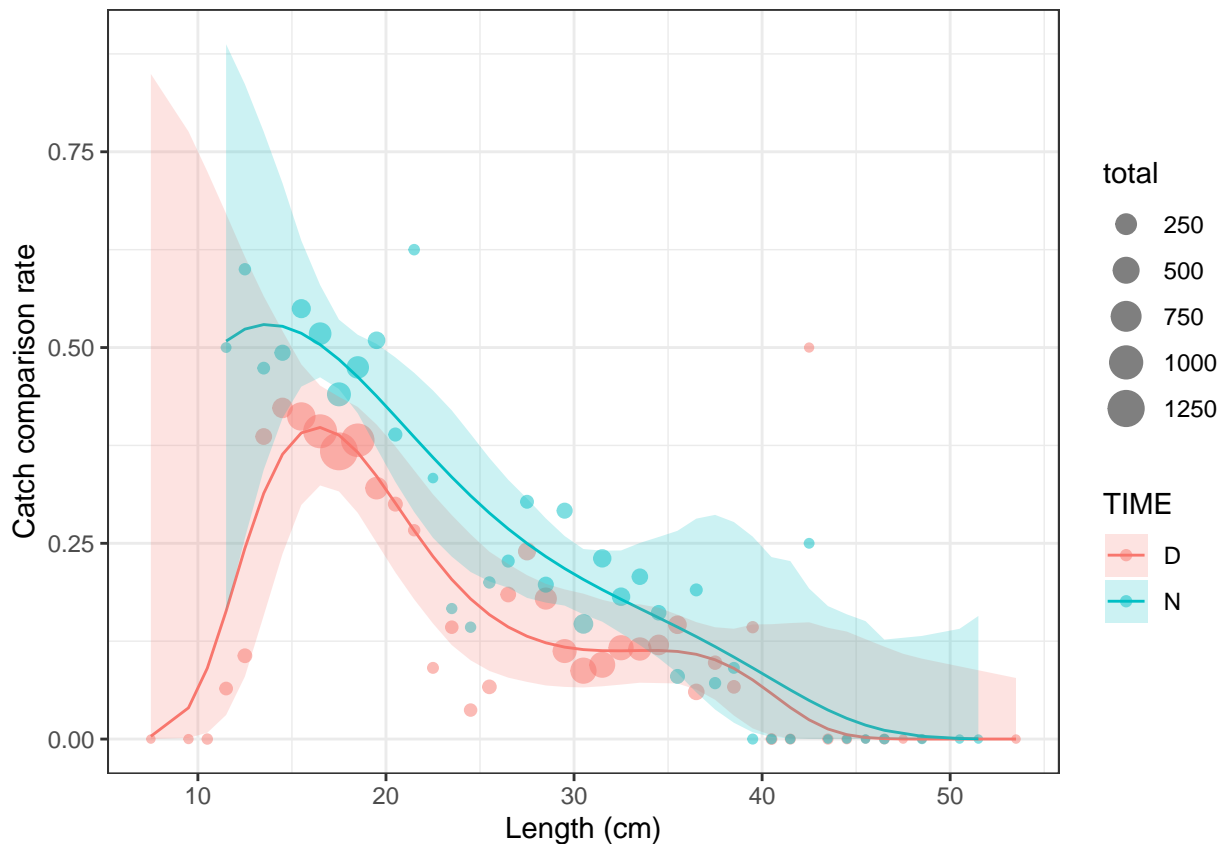
```

```
sumdat = rbind(sumdat1, sumdat2)
```

```

ggplot(sumdat, aes(LENGTH, prop))+geom_point(aes(size = total, col = TIME), alpha = 0.5)+
  geom_line(data = newdata, aes(col = TIME))+
  geom_ribbon(data = newdata, aes(ymin = lo, ymax = hi, fill = TIME), alpha = 0.2)+
  ylab("Catch comparison rate") + xlab("Length (cm)")

```



In accordance to Melli et al. (2018) a significant difference in catch comparison rate between day-time and night-time is found for individuals between 16 and 18 cm, as represented by the lack of overlapping between the CIs.

Melli et al. (2018) argued that, being the difference in a length range that is not usually retained when using a commercial codend, and that commercial fishing operations take place in both day- and night-time conditions, it is of greater interest to estimate the effect of FLEXSELECT with respect to the baseline trawl without the factor TIME.

Therefore, we repeat the steps of the process leaving out the factor TIME.

```

m4 = selfisher(prop~sl+I(sl^2)+I(sl^3)+I(sl^4), total = total, comphaddock, haul = HAUL)
m5 = selfisher(prop~bs(sl, df = 4), total = total, comphaddock, haul = HAUL)

```

```
AICtab(m4, m5)
```

```
##      dAIC df
## m4 0.0  5
## m5 5.1  5
```

Again, we select the polynomial of order 4 and use it to predict the catch comparison rates with CIs, which are then plotted against the overall experimental observations.

Windows code

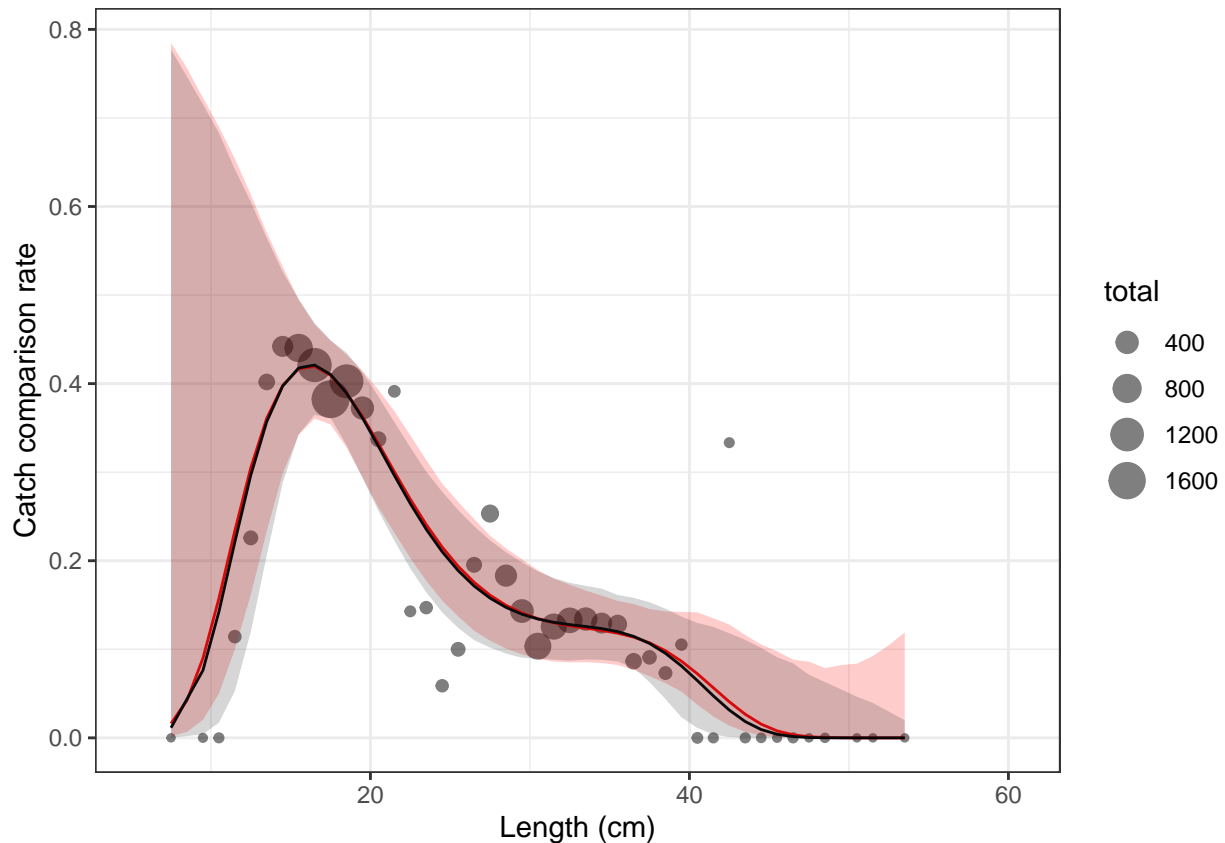
```
ncpus = 4
cl = makeCluster(rep("localhost", ncpus))
clusterExport(cl, "newdata")
bs = bootSel(m4, nsim = 1000, parallel = "snow", cl = cl,
             FUN = function(mod){predict(mod, newdata = newdata, type = "response")})
stopCluster(cl)
```

Mac and Linux code

```
bs = bootSel(m4, nsim = 1000, parallel = "multicore", ncpus = 4,
             FUN = function(mod){predict(mod, newdata = newdata, type = "response")})
```

```
quants = apply(bs$t, 2, quantile, c(0.025, 0.5, 0.975))
newdata[,c("lo", "mid", "hi")] = t(quants)

sumdat = ddply(comphaddock, ~LENGTH+s1, summarize,
              prop = sum(TEST1)/sum(total),
              ratio = sum(TEST1)/sum(TEST2),
              total = sum(total))
```



A graphical comparison to the published results in Melli et al. (2018; in red) shows that the estimated catch comparison curves and relative CIs are very similar.

Catch ratio

To directly quantify the difference in catch between the test and baseline trawls, it is common practice to estimate the catch ratio curve, using the relationship between catch comparison rate (cc) and catch ratio (cr): $cr = cc / (1 - cc)$

First, we obtain predictions for the catch ratio specifying `type="ratio"` in the `predict` function.

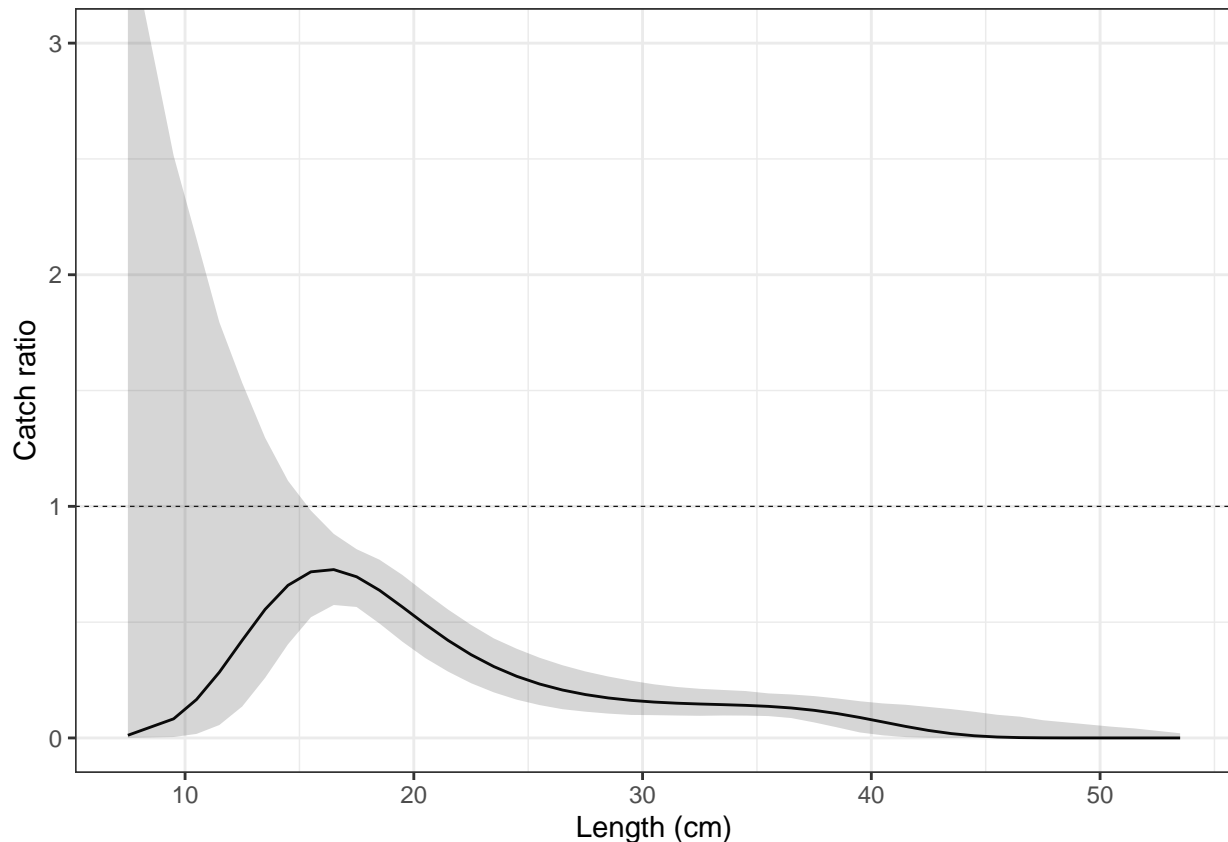
```
newdata$ratio = predict(m4, newdata = newdata, type = "ratio")
```

Second, we apply the relationship between cc and cr to obtain the CIs for the catch ratio curve.

```
bs$cr = bs$t / (1 - bs$t)
CRquants = apply(bs$cr, 2, quantile, c(0.025, 0.5, 0.975))
newdata[,c("cr_lo", "cr_mid", "cr_hi")] = t(CRquants)
```

Catch ratio plot


```
ggplot(sumdat, aes(LENGTH, ratio))+
  geom_line(data = newdata, aes(y = ratio))+
  geom_hline(aes(yintercept = 1), size = 0.2, linetype = "dashed")+
  geom_ribbon(data = newdata, aes(ymin = cr_lo, ymax = cr_hi), alpha = 0.2)+
  ylab("Catch ratio")+ xlab("Length (cm)") +
  coord_cartesian(ylim = c(0, 3))
```



The results show that for individuals above 16 and up to 53 cm, the test gear with FLEXSELECT retained significantly fewer individuals. The effect is length-dependent, with a more pronounced reduction at larger length classes. Considering a minimum conservation reference size of 27cm for haddock in the fishing area (Skagerrak and Kattegat), the reduction of commercial-sized individuals is above 60%.

References

- Akaike, H. (1974). A new look at the statistical model identification. *IEEE T. Automat. Contr.* 19, 716–722.
- Efron, B. (1982). The jackknife, the bootstrap and other resampling plans. In *Society for industrial and applied mathematics (SIAM) Monograph No. 38*, CBSM-NSF.
- Krag, L.A., Herrmann, B., Karlsen, J.D., and Mieske, B. (2015). Species selectivity in different sized toplews trawl designs: Does size matter? *Fish. Res.* 172, 243-249.
- Melli, V., Karlsen, J.D., Feekings, J.P., Herrmann, B., and Krag, L.A. (2018). FLEXSELECT: counterherding device to reduce bycatch in crustacean trawl fisheries. *Canad. J. Fish. Aquat. Sci.* 75, 850–860.
- Millar, R.B. (1993). Incorporation of between-haul variation using bootstrapping and nonparametric estimation of selection curves. *Fish. Bullet.* 91, 564–572.

Miller, T. (2013). A comparison of hierarchical models for relative catch efficiency based on paired-gear data for US Northwest Atlantic fish stocks. *Can. J. Fish. Aquat. Sci.* 70(9), 1306–1316.