



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 B: Paired gear analysis of codend selectivity dependent on mesh size

19 Oct 2021

Case study background

This example deals with brown shrimp selectivity data from Santos et al. (2018). The brown shrimp beam-trawl fishery is one of the most important fisheries in the Southern North Sea. Despite its relevance, this is also one of the least regulated fisheries in European waters. Concerns regarding the size selection of the fishery motivated the German research project *CRANNET* (2013-2015), which assessed brown shrimp size selection of commercially used and alternative codend designs. Codend selectivity data was collected during experimental fishing trials using the paired gear method (Millar and Walsh, 1992; Wileman, 1996). The experimental method consisted of fishing with two identical beam trawls, simultaneously and in parallel on the same shrimp population. One of the trawls mounted a small-mesh (11 mm) control codend with very limited selectivity (assumed to be non-selective) on the range of shrimp lengths available for the trawl. The second trawl mounted, one at a time, a total of 33 different test codend designs varying in mesh size and mesh type were tested.

This case study draws on a subset of the *CRANNET* data to demonstrate the use and functionality of `selfisher` in selectivity analysis based on paired gear data. It uses a subset of the catch data, the 87 hauls during which 13 diamond-mesh codends varying in mesh size (ranging from 19.1 mm to 36.3 mm) were tested. Additional information relative to fishing conditions and catch characteristics were recorded at haul level.

Preliminaries

```
library(selfisher)
library(xtable)
library(bbmle)
library(ggplot2); theme_set(theme_bw())
```

Data structure

We can read in the data that is distributed with the package and examine it.

```
data(pairedshrimp)
head(pairedshrimp)
```

##	length	ms	cw	seast	test	control	sampling_test	sampling_control	haul
## 1	23	20.188	12.074	4	0	1	0.232	0.182	1
## 2	24	20.188	12.074	4	1	1	0.232	0.182	1
## 3	26	20.188	12.074	4	0	1	0.232	0.182	1
## 4	28	20.188	12.074	4	0	2	0.232	0.182	1
## 5	29	20.188	12.074	4	0	3	0.232	0.182	1
## 6	30	20.188	12.074	4	2	1	0.232	0.182	1

Here we can see the column names of the data. There is a column for `haul` because multiple hauls h , $i = 1, 2, \dots, 87$ are contained in the same data object. Therefore the information presented in the remaining columns was collected at haul level. The column `length` contains observed (total) length classes l of brown shrimp (mm), `ms` contains the measured mesh size (mm) of the test codend, `cw` is the total catch weight (kg) collected in the test codend, `seast` is the state of the sea recorded during towing (Beaufort scale). Columns `test` and `control` contain the numbers of shrimps of length l sampled from each of the paired codends, while `sampling_test` and `sampling_control` are the associated sampling fractions.

Further data details

The covariate catch weight (`cw`) consists of a punctual, quantitative value of the total catch biomass observed in the test codend (Kg) per haul. The dataset analysed here includes a total of 87 hauls. Therefore, `cw` has 87 unique catch weight measurements. Catch sampling was carried out for each codend separately. Brown shrimp catches were sorted from the bulk of the catch and weighted. A random subsample of brown shrimp was collected from the total catch and frozen for later length measurement in the laboratory. The sampling fractions of shrimp (`sampling_test` and `sampling_control`) are therefore calculated as the ratio of the subsample weight to the total weight caught. Once in the laboratory, the subsamples were thawed and placed on a plate to be photographed. The total length of each individual was obtained by digital image analysis. Total lengths were rounded to the half millimetre below, to provide count data of the number of shrimp in each half millimetre-width length groups.

The state of the sea during the fishing activities was automatically measured using the vessel facilities. Measurements were done based on the ordinal Douglas sea state scale (0-9). The covariate `seast` is the average of the vessel's sea state measurements within haul. Therefore, as in the case of `cw` covariate, the `seast` covariate has 87 unique values.

Data transformation

In `selfisher`, it is necessary to model the binomial response as a proportion and total because unlike many other methods for binomial GLMs, the underlying code does not accept a two column response variable. Therefore, we transform the data to create these new columns. We also create a column for `qratio` (i.e. q_h below) which is the ratio of the sampling fraction of the test over the control gear. If the data contained raising factors instead, then `qratio` would be the raising factor of the control gear over the test gear.

```

pairedshrimp = transform(pairedshrimp, total = test + control,
                          qratio = sampling_test / sampling_control,
                          prop = test / ( test + control )
)

```

Selectivity analysis

The selectivity analysis based on paired gear data is usually done with the model introduced by Millar and Walsh (1992):

$$\phi(l) = \frac{p * r(l)}{(1.0 - p) + p * r(l)}$$

In the equation above, $\phi(l)$ expresses the probability that a shrimp of length l in the total catch of the paired gear was caught by the test gear. This probability is related to a sequence of two fishing events. The first event is controlled by the probability that a shrimp entering the paired gear did it through the test trawl (relative fishing power), expressed as the length-independent split parameter p . Conditioned on the probability of entering the test trawl, the second event is the length-dependent retention probability of the test codend $r(l)$. Retention probability is usually (but not exclusively) modelled with the logistic function, and summarized by two selectivity parameters L_{50} (length of 50% retention probability) and SR (difference in the lengths of 75% and 25% retention). Further details of this model and extensions involving alternatives to the logistic function can be found in Millar and Walsh (1992) and Wileman (1996).

In Santos *et al.* (2018), the effect of codend design as well as other variables describing catch and operational characteristics of the hauls were assessed using the so-called Fryer method (Fryer, 1991). The Fryer method is carried out in two steps. In the first step the parameters l_{50} , SR , p and associated Hessian-based covariance-variance matrix of individual hauls are estimated. The estimates become the data used in the second step, where the effect of the measured explanatory variables (fixed effects) on codend selectivity is estimated. Such estimations are obtained using the EM-algorithm, which allows quantifying the strength of the fixed effects in the presence of between-haul variation.

The Fryer method was developed at a time before suitable generalized mixed modeling software such as **selfisher** was available. The Fryer method suffers from small sample bias in the fits to individual haul data. In contrast, **selfisher** enables quantifying and testing the effect of explanatory variables in a single step and directly on the measured catch data.

The methods in **selfisher** generalize the original selectivity model introduced by Millar and Walsh (1992), by allowing multiple fixed and random effects to simultaneously model relative fishing efficiency and the selectivity of the test codend:

$$\phi_{h,l} = \frac{q_h p_h r_{h,l}}{(1.0 - p_h) + q_h p_h r_{h,l}}$$

In the equation above, $q_h = \frac{s_{t,h}}{s_{c,h}}$ is the ratio of the fraction of brown shrimp sampled in the test gear to the fraction sampled in the control gear, p_h is the relative fishing power of the test gear in haul h (split parameter), where $p_h = \text{logit}^{-1}(\mu + u_h)$ and u_h is a random effect $u_h \sim N(0, \sigma_u^2)$ accounting

for haul-specific random variation from the mean value on the logit scale μ . For simplicity in this example, we assume a logit link in the retention model and therefore the retention probability model for haul h and length l is $r_{h,l} = \text{logit}^{-1}(y_{h,l}) = \frac{\exp(y_{h,l})}{1+\exp(y_{h,l})}$. This expresses the haul-specific probability for a fish entering in the test gear to be retained where $y_{h,l}$ is the expectation on the link scale combining fixed- and random-effects potentially influencing retention probability of the test codend. Four models varying in the structure of $y_{h,l}$ are initially considered:

$$\begin{aligned} \text{model 1 : } & y_{h,l} = \beta_0 + a_h + (\beta_1 + \tau_h) * l \\ \text{model 2 : } & y_{h,l} = \beta_0 + a_h + (\beta_1 + \tau_h) * l + \beta_2 * ms_h \\ \text{model 3 : } & y_{h,l} = \beta_0 + a_h + (\beta_1 + \tau_h) * l + \beta_2 * ms_h + \beta_3 * l * ms_h \\ \text{model 4 : } & y_{h,l} = \beta_0 + a_h + (\beta_1 + \tau_h) * l / ms_h \end{aligned}$$

All models listed above include an intercept β_0 which expresses the baseline shrimp retention log-odds when all additional covariates (including shrimp length) are set to 0. The coefficient a_h is a random effect $a_h \sim N(0, \sigma_a^2)$ accounting for haul-specific random variation of the intercept. Model 1 assumes that codend retention can be exclusively described by shrimp length. In models 1, 2, and 3, β_1 is the slope quantifying the rate of increment in retention probability (on the link scale) due to increments in the length of shrimp. The slope of size selection curves can vary significantly and randomly between hauls, even while keeping the design characteristics of the codend constant (Fryer, 1991). The coefficient $\tau_h \sim N(0, \sigma_\tau^2)$ is introduced to account for haul-specific random variation of the baseline β_1 value. Model 2 extends model 1 by adding mesh size (ms) as a fixed effect that varies by haul: ms_h . Model 2 assumes that retention probability is a result of separate effects of shrimp length and mesh size. With the addition of an interaction term of mesh size and shrimp length (β_3), in model 3 it is assumed that mesh size influences both the position and slope of the size selection curve.

The tested diamond-mesh codends were made of the same netting material, same length, and the number of meshes in circumference were reduced proportionally to increments in mesh size. Therefore, we assume that the meshes of the tested codends present the same geometry during fishing. Based on the principle of geometrical similarity (Baranov, 1948; Millar and Holst, 1997; Revill and Holst, 2004) it would be a reasonable *a priori* assumption that the selection curves will vary systematically with mesh size. Model 4 is built upon the hypothesis that variation in size selection can be explained by the principle of geometrical similarity, implying that the size selection varies proportionally to mesh size (i.e. doubling the mesh size implies doubling the values of l_{50} and SR of the selection curve)(Millar and Holst, 1997).

Model fitting with `selfisher`

In this section, the models described above are fit using a function named `selfisher` within the `selfisher` package. Formulas in these models follow the convention of the `lme4` and `glmmTMB` packages. The function takes a formula for the retention model `rformula` as the first argument. This is a two sided model with the proportion on the left side and fixed and random effects on the right side. For example, in model 1 (`m1`) below, `prop ~ length + (length | haul)` says that the proportion (`prop`) of fish retained in the test gear depends on `length` and that the intercept and slope vary randomly by haul. It also takes an argument for the relative fishing power model (`pformula`) which is a one-sided formula, e.g. `pformula =~(1 | haul)`; this says that relative fishing power should be estimated and can vary randomly by haul. If instead, we wanted to fix the split at 0.5, then we could specify `pformula =~0`. To tell the function that this is a paired gear

model with one non-selective gear, we use the argument `psplit = TRUE`. The function also takes as arguments the names of the columns for the `total` and `qratio` within the data frame specified by the argument `data`.

One optional argument appears in this example: the `start` argument. It tells the function what starting values to use in maximum likelihood estimation. See `?selfisher` for the full flexibility of how to specify starting values, but here we only give starting values for the retention model. To get good starting values for the retention model's intercept (β_0 above) and coefficient on `length` (β_1 in models 1, 2, and 3 above), we use the `inits` function which takes guesses for l_{50} and SR as its arguments (30 and 8 respectively here). The coefficient β_1 has a different meaning in model 4, but the starting values work as supported by plots below, so it's not a problem. Models fit with TMB (as in `selfisher`) are usually robust to starting values, but due to the complexity of paired gear models, they sometimes get stuck in local minima during maximum likelihood estimation. In this case study, the length of the `start` vector must equal the number of fixed effects coefficients in the retention model, i.e. the β s in the equations above, so in `m2` and `m3` we combine guesses for β_0 and β_1 with zeros for the other coefficients

As described above, model 4 (`m4`) assumes geometric similarity. To include `length/ms` in a model in R, it is necessary to tell the formula interface to use the term "as is" by putting an `I()` around it.

```
m1 = selfisher( prop ~ length + ( length | haul ),
                total = total, psplit = TRUE, pformula =~( 1 | haul ),
                qratio = qratio, data = pairedshrimp, start = c(inits(30,8))
              )

m2 = selfisher( prop ~ length + ms + ( length | haul ),
                total = total, psplit = TRUE, pformula =~( 1 | haul ),
                qratio = qratio, data = pairedshrimp, start = c(inits(30,8),0)
              )

m3 = selfisher( prop ~ length * ms + ( length | haul ),
                total = total, psplit = TRUE, pformula =~( 1 | haul ),
                qratio = qratio, data = pairedshrimp, start = c(inits(30,8),0,0)
              )

m4 = selfisher( prop ~ I( length/ms ) + ( I( length/ms ) | haul ),
                total = total, psplit = TRUE, pformula =~( 1 | haul ),
                qratio = qratio, data = pairedshrimp, start = c(inits(30,8))
              )
```

Model selection

Retention is a mechanical process that can be largely explained by the relationship between the mesh characteristics and the morphology of the species being selected. Of the above models, model 4 is the only one that has a mechanistic justification, namely geometric similarity. In terms of model selection we *a priori* propose model 4, taking the view that strong evidence against model 4 is required to prefer an alternative. There are several thousand observations in the data, and consequently Akaike Information Criterion will tend to choose the most complicated model (Heinze

et al., 2018). In contrast, the Bayesian Information Criterion, BIC, includes a stronger penalty for the number of parameters than AIC and therefore it tends to select simpler models than AIC. We can use functions from the `bbmle` package to create either an AIC or BIC table of the models.

```
AICtab(m1, m2, m3, m4)
```

```
##      dAIC df
## m4  0.0  7
## m3  0.2  9
## m2  7.5  8
## m1 96.6  7
```

```
BICtab(m1, m2, m3, m4)
```

```
##      dBIC df
## m4  0.0  7
## m3 13.1  9
## m2 13.9  8
## m1 96.6  7
```

Both AIC and BIC rank model 4 on top, as was the *a priori* expectation.

Extended models

Model 4 is now extended by adding other covariates in the data frame that could potentially influence size selection, such as `cw` and `seast`. The following models are fitted following the same procedure as for models 1 to 4:

```
m5 = selfisher( prop ~ I( length/ms )+ cw + ( I( length/ms ) | haul ),
                total = total, psplit = TRUE, pformula =~( 1 | haul ),
                qratio= qratio, data = pairedshrimp, start = c(inits(30,8),0)
                )

m6 = selfisher( prop ~ I( length/ms ) + seast + ( I( length/ms ) | haul ),
                total = total, psplit = TRUE, pformula =~( 1 | haul ),
                qratio= qratio, data = pairedshrimp, start = c(inits(30,8),0)
                )

m7 = selfisher( prop ~ I( length/ms ) + cw + seast + ( I( length/ms ) | haul ),
                total = total, psplit = TRUE, pformula =~( 1 | haul ),
                qratio = qratio, data = pairedshrimp, start = c(inits(30,8),0,0)
                )

m8 = selfisher( prop ~ I( length/ms ) + cw * seast +( I( length/ms ) | haul ),
                total = total, psplit=TRUE, pformula =~( 1 | haul ),
                qratio = qratio, data = pairedshrimp, start = c(inits(30,8),0,0,0)
                )
```

Then we can compare all the models.

```
BICtab(m1, m2, m3, m4, m5, m6, m7, m8)
```

```
##      dBIC df
## m4   0.0  7
## m5   2.4  8
## m6   8.3  8
## m7   8.6  9
## m8  13.0 10
## m3  13.1  9
## m2  13.9  8
## m1  96.6  7
```

This table shows that none of the individual or combined effects associated with `cw` and `seast` were strong enough to be included in the most parsimonious model. Therefore model 4 is selected in this case study for further assessments.

Simple results

Consistent with other model procedures implemented in R, a summary of model results and fit statistics can be obtained via `summary(model.object)`. Before inspecting the size selectivity results provided by model 4, the estimated relative fishing efficiency of the test gear is presented. By default, `selfisher()` summary shows the estimated split parameter p on the logit scale, but this might be updated in a new version. An inverse logit transformation is needed to obtain the fishing power $p \in [0, 1]$:

```
round(boot::inv.logit(confint(m4, level=0.95, component="p")[1,c( 1, 3, 2 )]),3)
```

```
##      2.5 % Estimate    97.5 %
##      0.484      0.503      0.522
```

The split parameter estimated by model 4 is $p = 0.503$ with 95% confidence interval (0.484–0.522), very close to the value estimated in the original study ($p = 0.492$ (0.472–0.512)). This includes the reference value of 0.5, from which we conclude that there is no significant evidence against equal catch efficiency of the test and control gears.

Goodness-of-fit

The mixed models presented above are fitted at haul level. Therefore it is reasonable to assess the goodness-of-fit of these models on individual haul data. Due to the large dataset used, a random sample of 12 hauls are picked to demonstrate how well the $\phi(l)$ curves estimated by model 4 fit to the data. We can use the function `predict` to examine the estimated retention curve, ie. the model's “response” variable. See `?predict.selfisher` for details of how to use this function including the different types of predictions available.

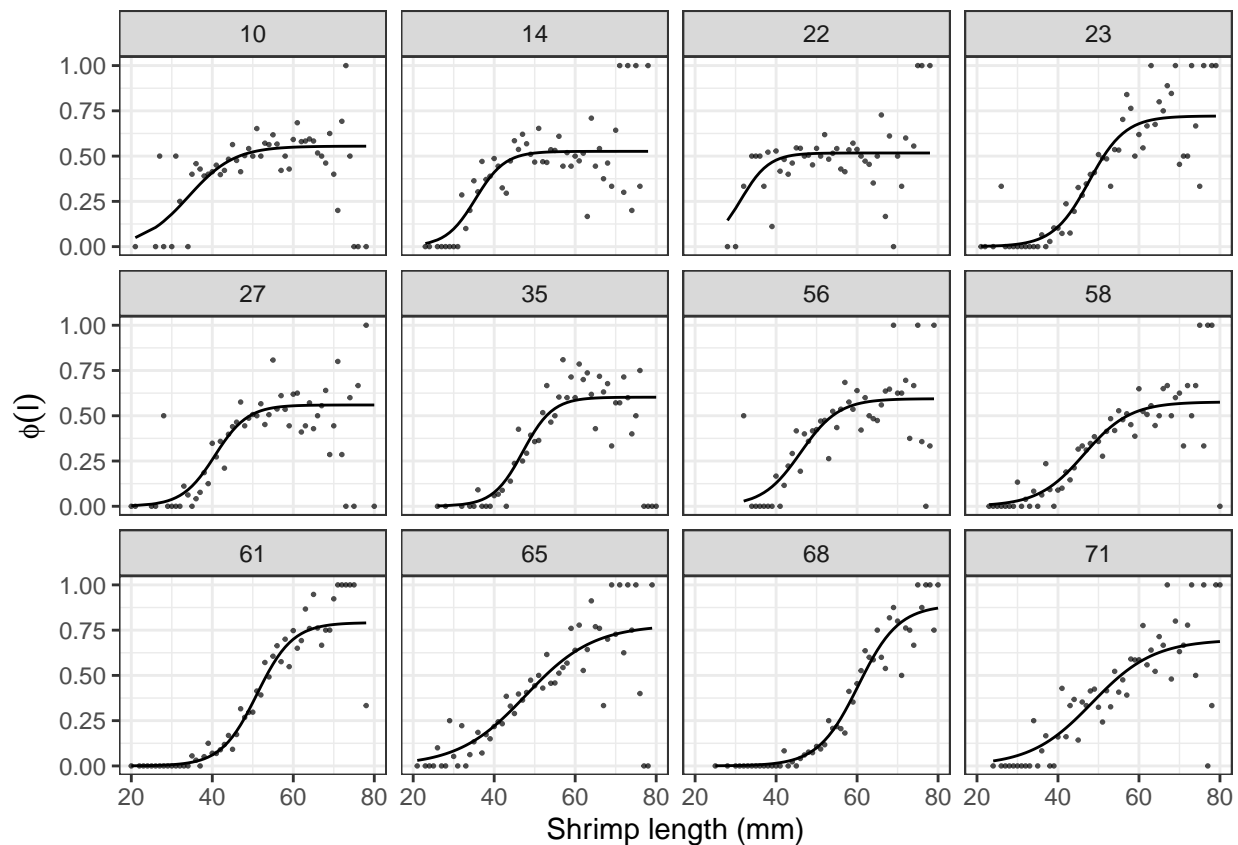

```

pairedshrimp$hat_phi = predict( m4, type = "response" )

set.seed( 999 )

ggplot(pairedshrimp[pairedshrimp$haul%in%sample(unique(pairedshrimp$haul),12),],
  aes(x=length,y=prop))+
  ylab( eval( parse( text = paste0("expression(", "phi", "(l)") ) ) )+
  xlab( "Shrimp length (mm)" )+
  xlim( 20, 80 )+
  geom_point( alpha = .7, size = 0.3 )+
  geom_line( aes( x = length, y = hat_phi ) )+
  facet_wrap( ~ haul, ncol = 4, nrow = 3)+
  theme_bw()

```



The figure above reveals that the fitted $\phi(l)$ curves describe well the trends and variability of the data at individual haul level.

Population-average predictions

Mixed modelling is a formal procedure that takes into account specific details of the data collection enabling sound hypothesis testing on fixed effects and model selection. However, fitting mixed models can be a computational-intensive task. Moreover, the researcher is also typically interested

in obtaining average selectivity predictions, as these are relevant to the selectivity applied to the fishery. It is therefore recommended to refit the best candidate model leaving out the random effects. Bootstrapping can then be used to obtain valid standard errors and confidence intervals for estimated quantities such as l_{50} and SR .

```
m4_fe = selfisher( prop ~ I( length/ms ), total = total,
                  psplit = TRUE, pformula =~1, haul = haul,
                  qratio = qratio, data = pairedshrimp, start = c( inits( 30, 8 ) ) )
```

A summary of model coefficients describing codend retention can be obtained using standard procedures in R

```
coef( summary( m4_fe ) )$r
```

```
##           Estimate Std. Error  z value Pr(>|z|)
## (Intercept) -9.231103 0.09420144 -97.99322     0
## I(length/ms)  5.143515 0.06443302  79.82732     0
```

Bootstrapping to get confidence intervals on population-average l_{50} and SR

Selectivity statistics l_{50} and SR are often obtained by simple calculations involving model coefficients β_0 and β_1 , or for simple models in `selfisher`, the function `L50SR(model.object)` can calculate them. However, standard calculations need to be updated when using multiple fixed effects to describe codend retention. Details on how to calculate l_{50} and SR from the models considered in this case study can be found in Table 1 below.

To obtain a bootstrap distribution of the selectivity parameters estimated by model 4, first we generate a bootstrap distribution of model coefficients using `bootSel()`, as follows. The `bootSel()` function applies the user-defined function `FUN` to each refit model; here we define `FUN` so that it returns the fixed effect (`fixef`) coefficients of the retention model (`$r`). This is the code to perform the computations in parallel on Linux or Mac computers, but see the other case studies for how to do it in Windows.

```
bootpars_m4_fe = bootSel( m4_fe, nsim = 1000,
                        parallel = "multicore", ncpus = 4,
                        FUN = function( x ){fixef( x )$r }
                        )
```

Selectivity statistics l_{50} and SR are then calculated from each set of coefficients in the bootstrap distribution generated above. Finally, the resulting bootstrap distribution is used to obtain 95% percentile confidence intervals of l_{50} and SR .

```
# extract bootstrap distribution from bootSel object
bootpars = bootpars_m4_fe$t

# create a grid of mesh sizes within the experimental range for predictions
```

```

ms = seq( from = 20, to = 35, by = .5 )

# calculate bootstrap distribution of l50 based on model coefficients
## l50 = -m * beta_0 / beta_1 (see table 3)
L50_boot = apply( bootpars, 1, function( x ){ -ms * x[1] / x[2] } )

# get Efron confidence intervals for l50
L50_ci = t( apply( L50_boot, 1, quantile, c ( 0.025, 0.5, 0.975 ) ) )

# create a data frame for plotting
L50_df = data.frame( mesh_size = ms )

L50_df [, c("lower_limit", "median", "upper_limit" )] = L50_ci

# calculate bootstrap distribution of SR based on model coefficients
## SR = m*log(9) / beta_1 (see table 3)
SR_boot = apply( bootpars, 1, function( x ){ ms * log(9) / x[2] } )

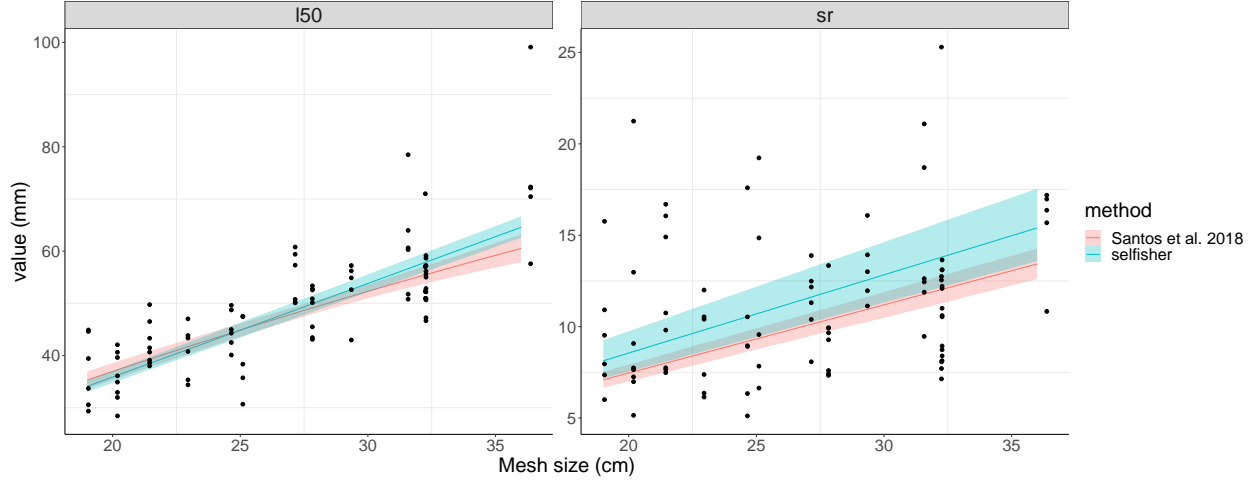
# get Efron confidence intervals for SR
SR_ci = t( apply( SR_boot, 1, quantile, c( 0.025, 0.5, 0.975 ) ) )

# create a data frame for plotting SR
SR_df = data.frame( mesh_size = ms )

SR_df[, c( "lower_limit", "median", "upper_limit" )] = SR_ci

```

The following figure compares the predictions of l_{50} and SR estimated by model 4, with those from Santos et al. (2018). The black points in the figure represent values of l_{50} and SR estimated at haul level, and used in the original study as input data. Average predictions for l_{50} and estimation uncertainty (expressed in the amplitude of the confidence band) by model 4 are equivalent to those from the original study. Model 4 predicted higher values of SR , with larger uncertainty than Santos et al. (2018). This is a plausible result considering the different model structures applied, and the large variation of the by-haul estimates. Moreover, there is not statistical evidence to reject the possibility that the true value of SR could fall within the continue region of confidence bands overlap.



model	l50	SR
1	$\frac{-\alpha}{\beta_1}$	$\frac{\log(9)}{\beta_1}$
2	$\frac{-(\alpha + \beta_2 * ms)}{\beta_1}$	$\frac{\log(9)}{\beta_1}$
3	$\frac{-(\alpha + \beta_2 * ms)}{\beta_1 + \beta_3 * ms}$	$\frac{\log(9)}{\beta_1 + \beta_3 * ms}$
4	$\frac{-\alpha * ms}{\beta_1}$	$\frac{\log(9) * ms}{\beta_1}$
5-8	$\frac{-(\alpha + \sum_{i=2}^p \beta_i * x_i) * ms}{\beta_1}$	$\frac{\log(9) * ms}{\beta_1}$

Table 1: Calculations to obtain selectivity parameters from population-average models 1-8 (fitted leaving out the random effects). Note that the last row for models 5-8 is the extension of model 4 to include additional covariates $x_i, i = 2, \dots, p$.

References

- Baranov, F. 1948. Theory and assessment of fishing gear. Pishchepromisdat, Moscow.
- Fryer, R. 1991. A model of between-haul variation in selectivity. ICES Journal of Marine Science, 48: 281–290. Oxford University Press.
- Heinze, G., Wallisch, C., and Dunkler, D. 2018. Variable selection—a review and recommendations for the practicing statistician. Biometrical Journal, 60: 431–449. Wiley Online Library.

- Millar, R., and Walsh, S. 1992. Analysis of trawl selectivity studies with an application to trouser trawls. *Fisheries Research*, 13: 205–220. Elsevier.
- Millar, R. B., and Holst, R. 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science*, 54: 471–477. Oxford University Press.
- Revill, A. S., and Holst, R. 2004. Reducing discards of north sea brown shrimp (c. Crangon) by trawl modification. *Fisheries Research*, 68: 113–122. Elsevier.
- Santos, J., Herrmann, B., Stepputtis, D., Günther, C., Limmer, B., Mieske, B., Schultz, S., *et al.* 2018. Predictive framework for codend size selection of brown shrimp (crangon crangon) in the north sea beam-trawl fishery. *PloS one*, 13: e0200464. Public Library of Science San Francisco, CA USA.
- Wileman, D. A. 1996. Manual of methods of measuring the selectivity of towed fishing gears. *ICES Coop. Res. Rep.*, 215: 1–126.