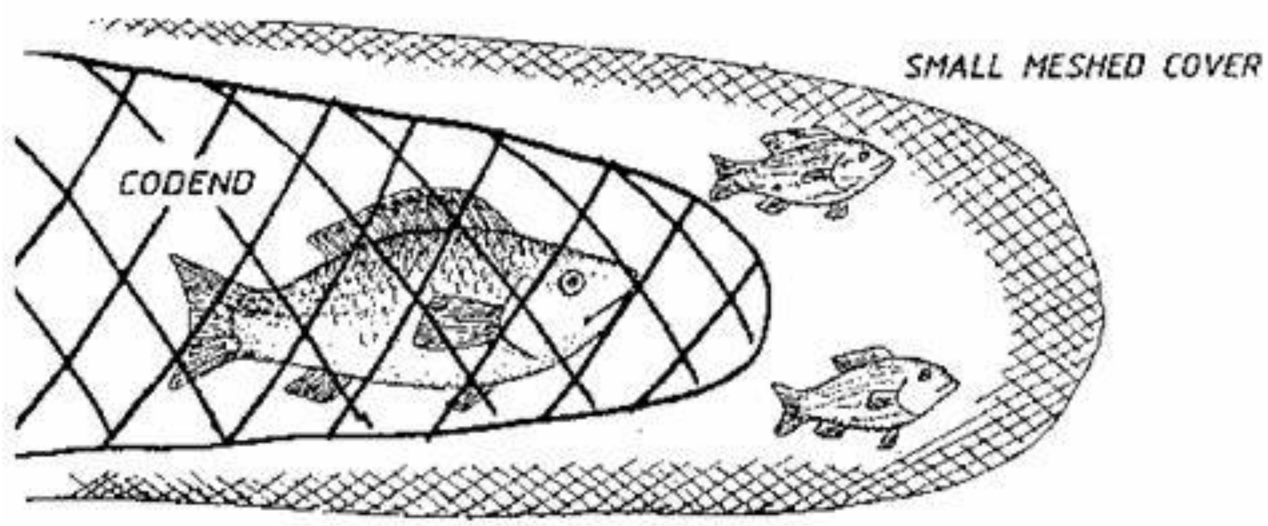
Open source software for statistical models of gear selectivity

Mollie Brooks

in collaboration with
Tiago Veiga-Malta, Ludvig Krag, Jordan Feekings

DTU Aqua

Selectivity model

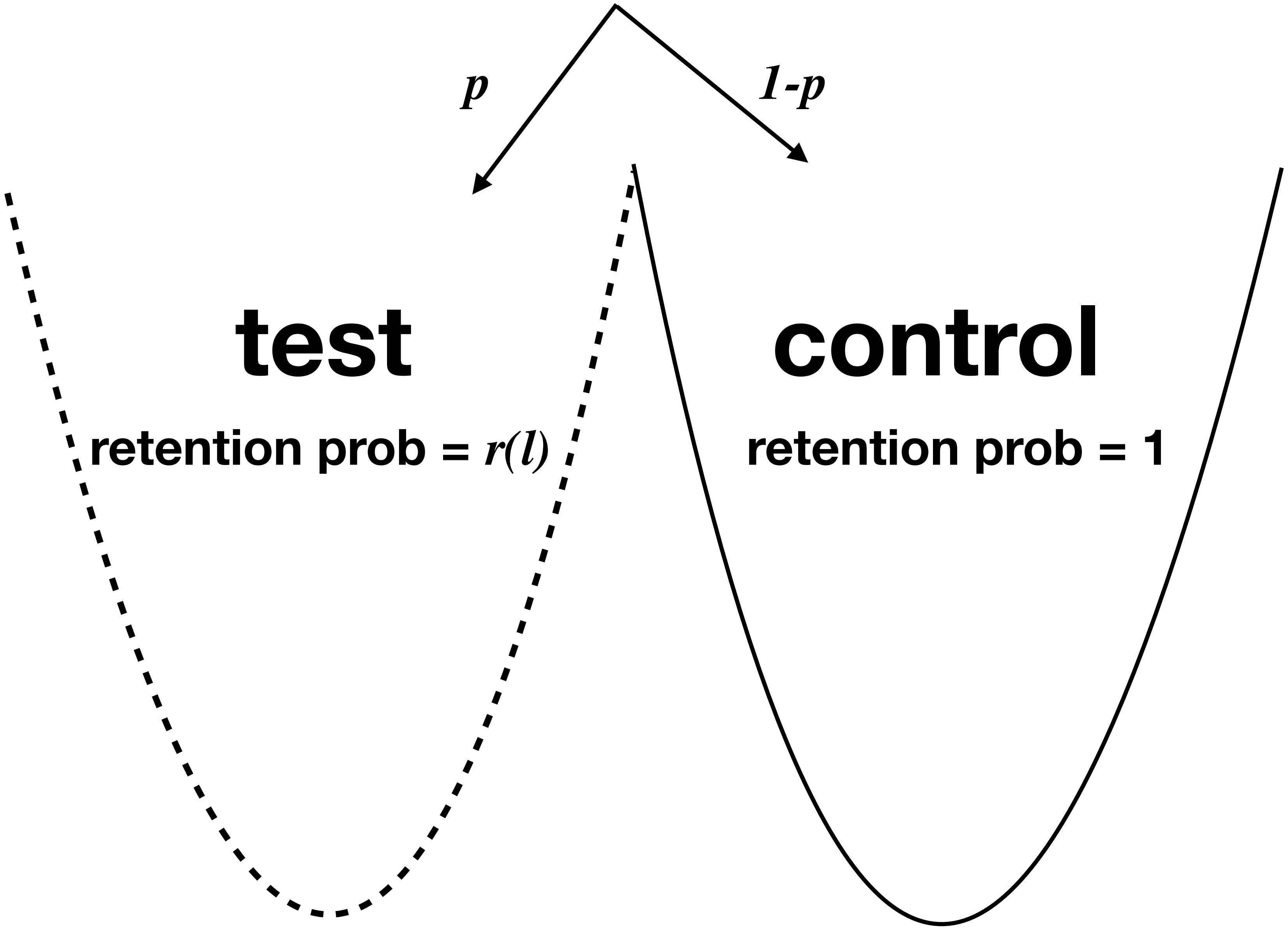


- retention probability as a function of length : $r(l)$
- binomial outcome
- generalized linear model (GLM)
- link function to keep probability between 0 and 1
    - e.g. logit link: $r\ (l) = \exp(a + bl)/(1 + \exp(a + bl))$

(image credit: FAO.org)

Paired gear model

(e.g. twin, trouser, alternate or parallel trawling)



p

1-p

test

retention prob = *r(l)*

control

retention prob = 1

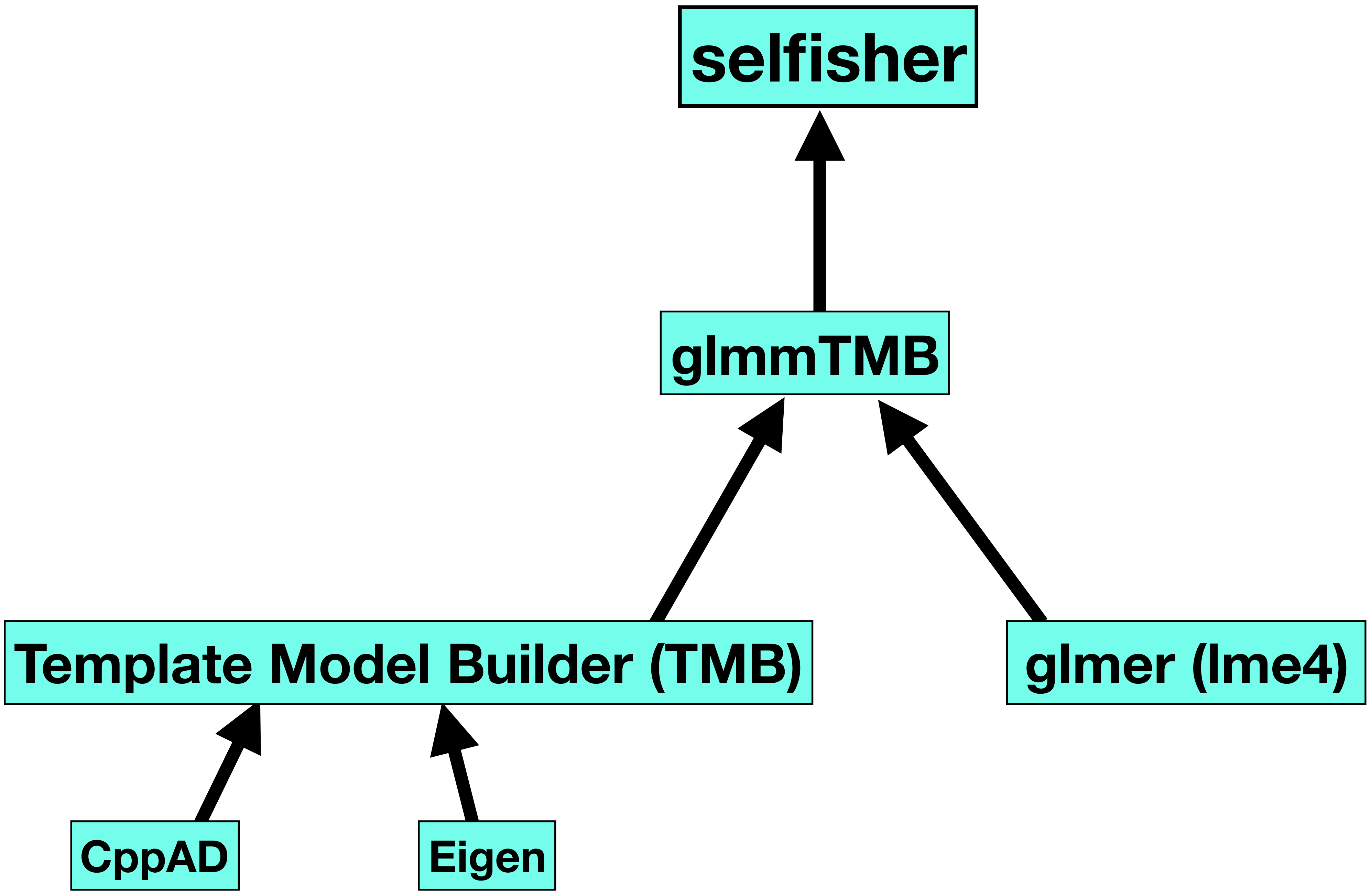
proportion of fish in test gear compared to total fish in test and control:

$$\phi = \frac{pr(l)}{pr(l) + 1 - p}$$

Estimating selectivity

- several software methods exist
- needed something for teaching
- introducing `selfisher`
- SELectivity of FISHERies gear in R

Open-source software pedigree of selfisher



Features of selfisher

- retention model ($\boldsymbol{r}$)
- **r** links: logit, probit, c-log-log, log-log, Gompertz, Richards
- relative fishing power model ($\boldsymbol{p}$)
- fixed and random effects on $\boldsymbol{r}$ and $\boldsymbol{p}$
- offsets
- "double bootstrapping"
- standard regression methods

Fixed effects that could effect selectivity

- with or without stimulation devices
- net age
- total catch weight
- sea state
- substrate qualities

Standard methods

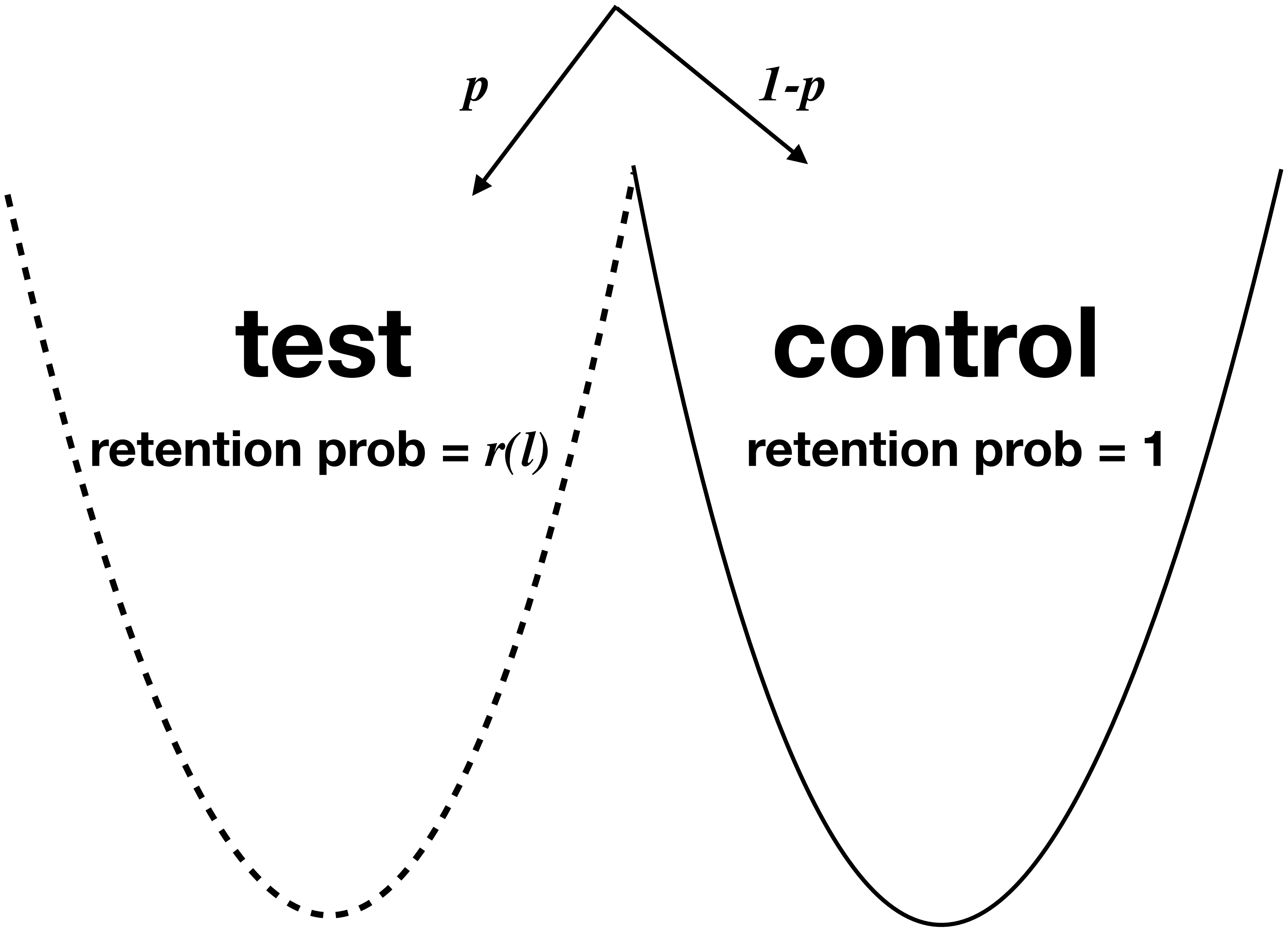
- predict(model, newdata, type)
  - "selection" = r
  - "prob" = p
  - "ratio" = $r/(1-r)$
  - "response" = $pr/(pr+1-p)$
- residuals(model, type)
  - "response"
  - "pearson"
  - "deviance"
- logLik(model)
- AICc(model)

"double bootstrap"

```
resample hauls with replacement

for each haul, for each length class

    assume $\phi_{observed} = k_{observed}/n_{observed}$

    new $\phi_{observed}$ = rbinom(old $\phi_{observed}$, $n_{observed}$)/$n_{observed}$
```

Paired gear model

(e.g. twin, trouser, alternate or parallel trawling)



proportion of fish in test gear compared to total fish in test and control:

$$\phi = \frac{pr(l)}{pr(l) + 1 - p}$$

Model specification details for `psplit=TRUE`

$$
\begin{align}
k &\sim \mathrm{Binom}(\phi, n), \tag{1}\\
\phi &= \frac{pr}{pr + 1 - p}, \tag{2}\\
\mathrm{logit}(r) &= X^{(\mathtt{r})}\beta^{(\mathtt{r})} + Z^{(\mathtt{r})}b^{(\mathtt{r})}, \tag{3}\\
\mathrm{logit}(p) &= X^{(\mathtt{p})}\beta^{(\mathtt{p})} + Z^{(\mathtt{p})}b^{(\mathtt{p})}, \tag{4}\\
& \tag{5}
\end{align}
$$

βs are fixed effect parameters

bs are random effects

Xs and Zs are model matricies

Simple example with psplit

haddock data from Russel Millar's website with his permission

```
data(haddock)
head(haddock)
```

```
##   Lengths nwide nfine
## 1      24     0     1
## 2      25     0     1
## 3      26     0     3
## 4      27     1    14
## 5      28     5    30
## 6      29    19    49
```

```
dat=transform(haddock,
              total=nfine+nwide,
              prop=nwide/(nfine+nwide))
```

Simple example with psplit

assume psplit=0.5

```
m0=selfisher(prop~Lengths, p=~0, psplit=TRUE,
                         total=total, dat)
```

or estimate psplit. . .

```
m1=selfisher(prop~Lengths, p=~1, psplit=TRUE,
                         total=total, dat)
```

Simple example output

compare the models

```r
AICctab(m0, m1)
```

```
##    dAICc df
## m1  0.0  3
## m0 18.6  2
```

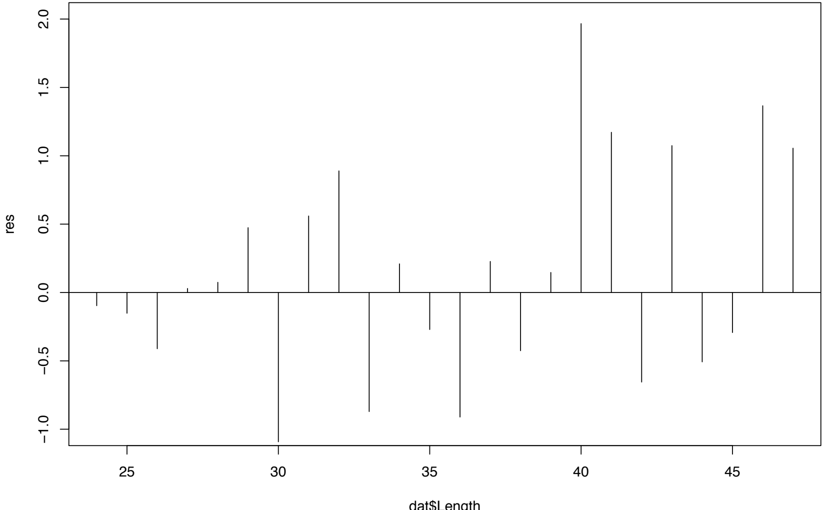
Simple example output

```r
summary(m1)
```

```
##  Family: binomial  ( logit )
## Selectivity formula:          prop ~ Lengths
## Relative fishing power formula:        ~1
## Data: dat
## Total: total
##
##           AIC           BIC        logLik      deviance Pearson.ChiSq
##         100.7         104.3         -47.4          14.8          13.7
##      df.resid
##            21
##
##
## Selectivity model:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -27.6542     5.7488   -4.81 1.51e-06 ***
## Lengths       0.9164     0.1971    4.65 3.32e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Relative fishing power model:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.29346    0.07042   4.167 3.08e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Size at retention probability:
##      p   Lp.Est Lp.Std.Err
## 1 0.25 28.97665  0.2792741
## 2 0.50 30.17542  0.3557115
## 3 0.75 31.37419  0.5549850
##
## Selectivity range (SR):
##   Estimate Std. Error
##  2.3975428  0.5156376
```

Plot residuals

```
res = residuals(m1, type="deviance")
plot(dat$Length, res, type='h', ylim=c(-1,2))
abline(0,0)
```



A multihaul example

```r
data(ccmhtwdat)
ccmhtwdat=subset(ccmhtwdat, total>0)
head(ccmhtwdat, 2)
```

```
##       length haul     type test cover    TW total prop
## 342.1   48.5    4 baseline    2     0 459.8     2    1
## 345.1   51.5    4 baseline    1     0 459.8     1    1
```

```r
tail(ccmhtwdat, 2)
```

```
##        length haul        type test cover  TW total prop
## 2138.2   26.5   22 stimulation    0     1 143     1    0
## 2142.2   30.5   22 stimulation    0     1 143     1    0
```

Mixed model for hypothesis testing

```
mmod=selfisher(prop~length*type +(1|haul), total=total, ccmhtwdat)
summary(mmod)
```

```
##  Family: binomial  ( logit )
## Selectivity formula:          prop ~ length * type + (1 | haul)
## Data: ccmhtwdat
## Total: total
##
##           AIC           BIC        logLik      deviance Pearson.ChiSq
##        1075.2        1097.7        -532.6         507.6         798.2
##      df.resid
##           652
##
## Random effects:
##
## Selectivity model:
##  Groups Name        Variance Std.Dev.
##  haul   (Intercept) 0.5041   0.71
## Number of obs: 657, groups:  haul, 21
##
## Selectivity model:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -9.30422    0.37521 -24.798  < 2e-16 ***
## length                   0.30432    0.01024  29.704  < 2e-16 ***
## typestimulation          3.19766    0.75758   4.221 2.43e-05 ***
## length:typestimulation  -0.22758    0.02569  -8.860  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

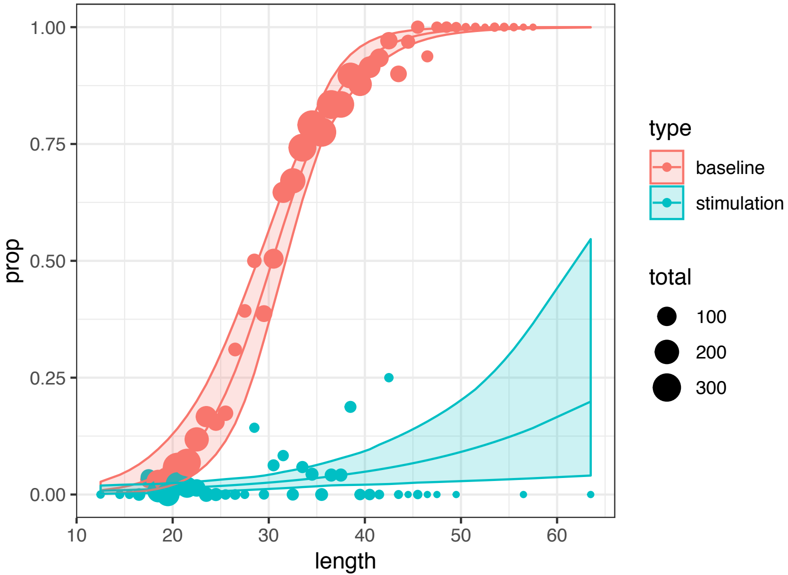
Model and newdata for bootstrapping hauls

```
mod=selfisher(prop~length*type, total=total, haul=haul, ccmhtwda

newdata=expand.grid(length=unique(ccmhtwdat$length),
                    total=1,
                    haul=0,
                    type=c("baseline", "stimulation"))

newdata$prop=predict(mod, newdata=newdata, type="selection")
```

Bootstrap to get CI on predictions

```
bs=bootSel(mod, nsim=100, parallel="multicore", ncpus = 4,
    FUN=function(mod){
        predict(mod, newdata=newdata, type="selection")
        })

quants=apply(bs$t, 2, quantile, c(0.025, 0.5, 0.975))

newdata[,c("lo", "mid", "hi")]=t(quants)
```

Plot bootstrapped predictions



Maybe try other links

```
mmod_loglog=selfisher(prop~length*type +(1|haul),
                total=total, ccmhtwdat, link="loglog")

mmod_richards=selfisher(prop~length*type +(1|haul),
                total=total, ccmhtwdat, link="Richards",
                start = list(betar=c(-10, .2, 1, 0)))

AICctab(mmod, mmod_loglog, mmod_richards)
```

```
##               dAICc df
## mmod_richards  0.0  6
## mmod_loglog    9.8  5
## mmod          26.4  5
```

A catch comparison example with sampling

```r
data("ccmhsampdat")
head(round(ccmhsampdat, 5))
```

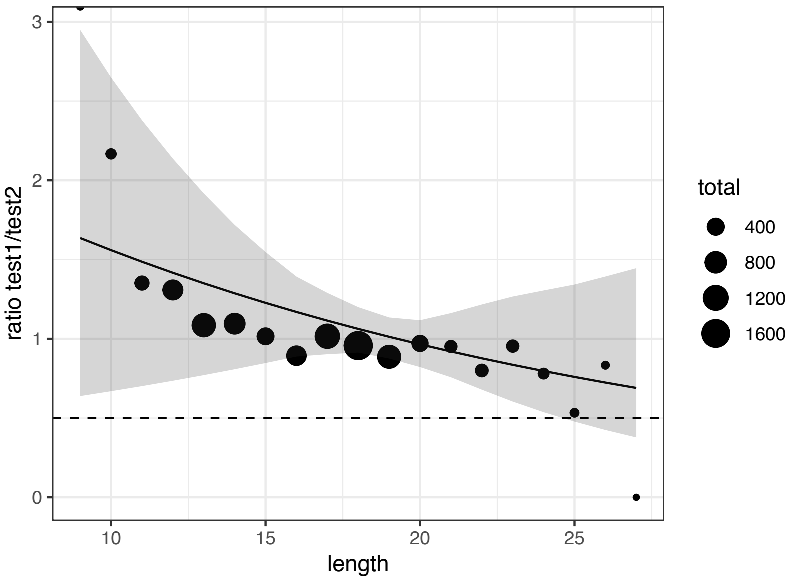
```
##   length test1 test2 sampling_test1 sampling_test2 haul total
## 1      9     1     0        0.00204        0.00197    2     1
## 2     10     0     2        0.00204        0.00197    2     2
## 3     11     4     5        0.00204        0.00197    2     9
## 4     12    32    37        0.00204        0.00197    2    69
## 5     13    56    63        0.00204        0.00197    2   119
## 6     14    39    50        0.00204        0.00197    2    89
```

```r
dat=transform(ccmhsampdat, q=sampling_test1/sampling_test2)

m4=selfisher(prop~length + offset(log(q)),
            haul=haul, total=total, dat)
```

Bootstrap predictions of the ratio

```
newdata=expand.grid(length=unique(dat$length),
                total=1,
                haul=0,
                q=1)

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

bs=bootSel(m4, nsim=100, parallel = "multicore", ncpus = 4,
            FUN=function(mod){
                predict(mod, newdata=newdata, type="ratio")
                })

quants=apply(bs$t, 2, quantile, c(0.025, 0.5, 0.975))

newdata[,c("lo", "mid", "hi")]=t(quants)
```

Plot predictions with observations



Installation and info

selfisher

https://github.com/mebrooks/selfisher

GLMM FAQ

http://bbolker.github.io/mixedmodels-misc/glmmFAQ.html

Future directions

- more testing and documentation
- shiny app (i.e. GUI)
- investigate "dual selection model" possibilities
- courses
- papers

Acknowledgments

- glmmTMB developers, Kasper Kristensen, Ben Bolker
- Funding: European Maritime Fisheries Fund

admb

FAST, ACCURATE, STABLE OPTIMIZATION

FNSNF

FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

European Union
European Maritime and Fisheries Fund