SELECT package demonstration  
Alternate haul

## Summary

This is a simple case study that revisits the alternate haul data analysed in Wileman et al. (1996, Manual of Methods of Measuring the Selectivity of Towed Fishing Gears, ICES Coop Research Report, No 215).

**This case study demonstrates:**

* The use of data that is provided within the SELECT package.
* Fixing the relative fishing power using the direct comparison experimental design type dtype=dc.

### Data source

The data are originally from Pope et al. (1975, Manual of methods for fish stock assessment Part 111 Selectivity of fishing gear.FA0 Fish. Tech. Pap. (41) Rev. 1.) These are alternative haul catch data of haddock in an 87 mm diamond experimental codend alternating with a 35 mm control gear.

The data are only given as totals summed over hauls, so no exploration of between-haul variability is possible. The data are included with the SELECT package.

#devtools::install\_github("rbmillar/SELECT")  
require(SELECT)  
require(tidyverse)

### Load and inspect the data

data(Pope)  
Pope

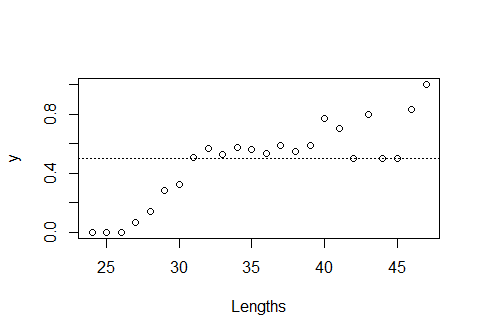
## Lengths nfine nwide  
## 1 24 1 0  
## 2 25 1 0  
## 3 26 3 0  
## 4 27 14 1  
## 5 28 30 5  
## 6 29 49 19  
## 7 30 60 29  
## 8 31 50 51  
## 9 32 70 91  
## 10 33 108 120  
## 11 34 88 118  
## 12 35 84 107  
## 13 36 68 78  
## 14 37 37 52  
## 15 38 33 40  
## 16 39 12 17  
## 17 40 5 17  
## 18 41 6 14  
## 19 42 10 10  
## 20 43 1 4  
## 21 44 6 6  
## 22 45 2 2  
## 23 46 1 5  
## 24 47 0 1

### Define variable names

v.names=c("Lengths","nfine","nwide")

### Produce a plot of catch-share proportions

Pope=transform(Pope,n=nfine+nwide,y=nwide/(nfine+nwide))   
plot(y~Lengths,data=Pope)  
abline(h=0.5,lty=3)



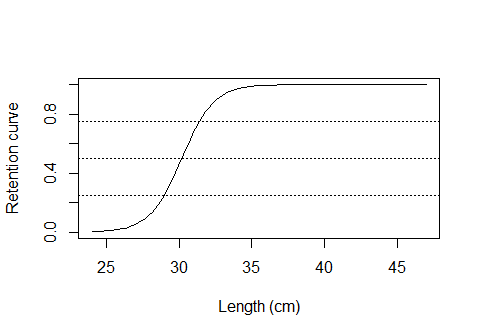
The experimental condend has catch share of 50% or more for all of the larger lengths, indicating that it has higher fishing power than the control.

### Fit logistic selection curve

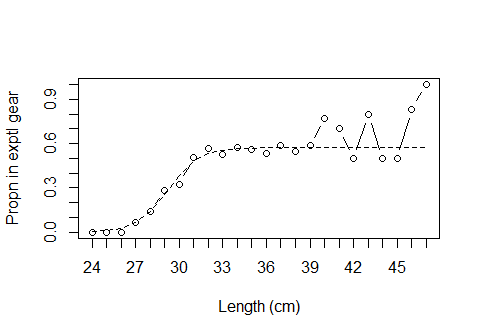
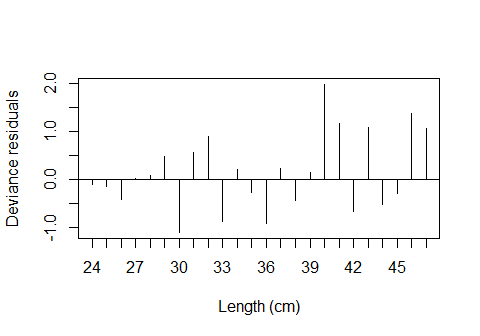
Logist.fit=SELECT(Pope,var.names=v.names,dtype="ph")

##   
## Design specification `ph` is deprecated and  
## has been changed to `ec` (experimental/control)

plot(Logist.fit)



ModelCheck(Logist.fit)



## Model fit:  
## null.l model.l full.l npar AIC   
## -87.53367 -47.37439 -39.95578 3.00000 100.74878   
## GOF:  
## Deviance Pearson.chisq dof Deviance.CF Pearson.CF   
## 14.8372224 13.7358636 21.0000000 0.7065344 0.6540887

Estimates(Logist.fit)

## par raw s.e.  
## L50 30.1767091 0.35505969  
## SR 2.3988249 0.51389589  
## p 0.5728613 0.01721723

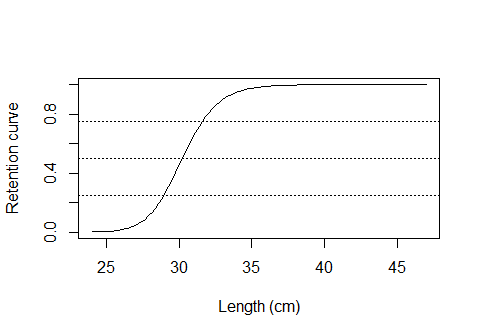
The logistic fit looks good. For completeness we’ll do the Richards fit also.

### Fit Richards curve

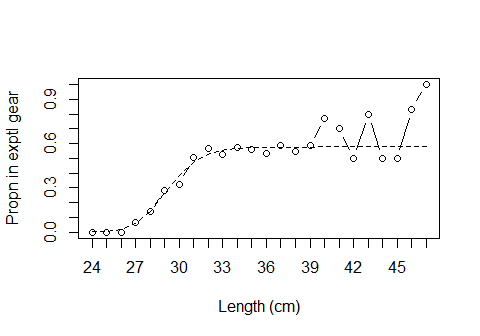
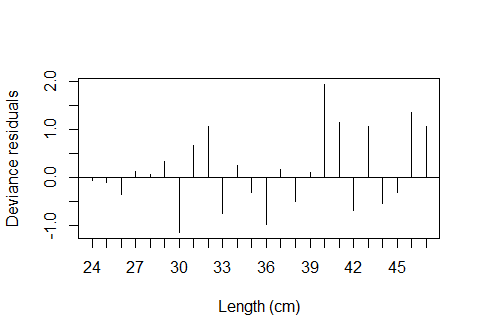
Rich.fit=SELECT(Pope,var.names=v.names,dtype="ph",stype="richards")

##   
## Design specification `ph` is deprecated and  
## has been changed to `ec` (experimental/control)

plot(Rich.fit)



ModelCheck(Rich.fit)



## Model fit:  
## null.l model.l full.l npar AIC   
## -87.53367 -47.43011 -39.95578 4.00000 102.86023   
## GOF:  
## Deviance Pearson.chisq dof Deviance.CF Pearson.CF   
## 14.9486667 13.8936358 20.0000000 0.7474333 0.6946818

Estimates(Rich.fit)

## par raw s.e.  
## L50 30.2099993 0.39203642  
## SR 2.6354642 0.87628498  
## delta 0.6161326 1.17071785  
## p 0.5775485 0.02303817

There is very little difference between the logistic and Richards fits.

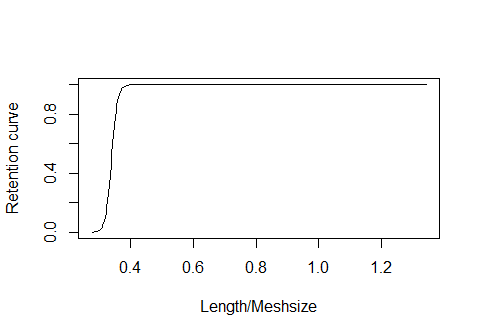
The logistic is preferred, since even without being able to correct for over-dispersion, AIC prefers the logistic. A likelihood-ratio test also would not reject goodness of fit of the logistic.

## Fixing the split parameter

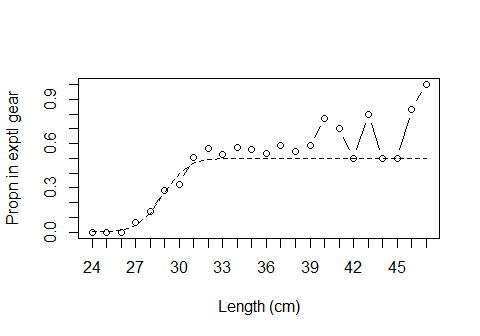
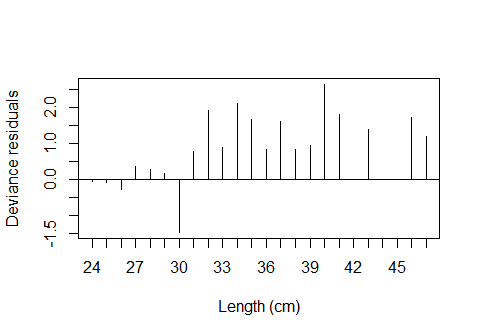
Fixing the value of the split parameter requires specifying the design type as direct comparison, dc. Also, meshsizes must be given.

The default is equal fishing power, i.e., 50/50 split. If unequal then the relative fishing power is specified through a vector of length two that is provided as the rel.power argument. For example rel.power=c(0.4,0.6) means that the first and second mesh sizes have 40% and 60% split, respectively.

#50/50 split  
EqualPower.fit=SELECT(Pope,var.names=v.names,dtype="dc",Meshsize=c(35,87))  
plot(EqualPower.fit)



ModelCheck(EqualPower.fit)

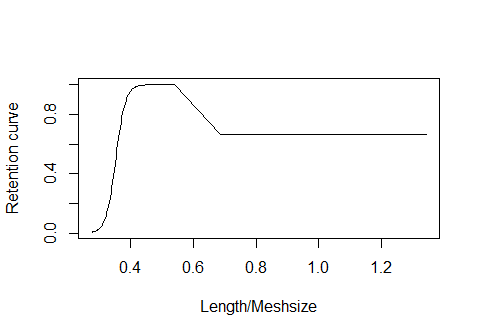


## Model fit:  
## null.l model.l full.l npar AIC   
## -87.53367 -57.97030 -39.95578 2.00000 119.94059   
## GOF:  
## Deviance Pearson.chisq dof Deviance.CF Pearson.CF   
## 36.029032 34.683536 22.000000 1.637683 1.576524

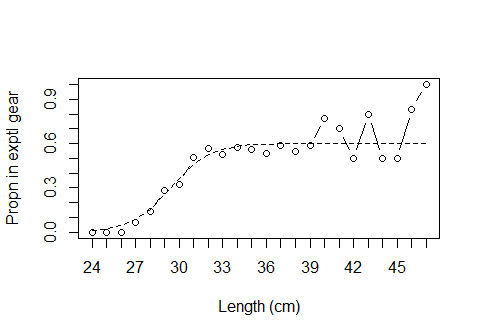
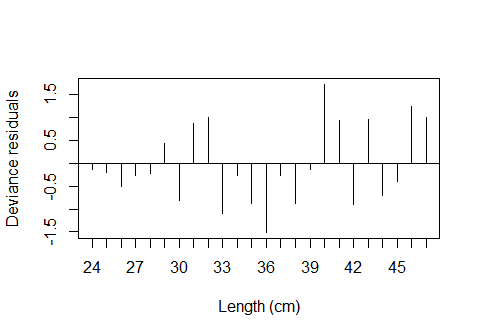
Estimates(EqualPower.fit)

## par raw s.e.  
## L50 29.428845 0.2506689  
## SR 1.780782 0.3385444

#40/60 split  
UnEqualPower.fit=SELECT(Pope,var.names=v.names,dtype="dc",  
 Meshsize=c(35,87), rel.power=c(0.4,0.6))  
plot(UnEqualPower.fit)



ModelCheck(UnEqualPower.fit)



## Model fit:  
## null.l model.l full.l npar AIC   
## -87.53367 -48.45797 -39.95578 2.00000 100.91595   
## GOF:  
## Deviance Pearson.chisq dof Deviance.CF Pearson.CF   
## 17.0043880 16.0557234 22.0000000 0.7729267 0.7298056

Estimates(UnEqualPower.fit)

## par raw s.e.  
## L50 30.614372 0.2912733  
## SR 2.924902 0.5593281

The equal fishing power model is clearly not a good fit.