

# *The Jack Knife*

*Laurence Kell*

*August 13th, 2014*

## *Introduction*

The jack knife can be used to estimate standard errors and bias for parameter in non-linear estimation and quantities derived from theme.

## *Estimation*

When using the jack knife the  $i^{th}$  observation is omitted in turn from the sample data set and the parameter re-estimated and its mean calculated  $\bar{\theta}_i$ .

$$f(X) \simeq \overline{f^{??}} \equiv \frac{1}{N} \sum_{i=1}^N f_i^J$$

## *Variance*

The variance can is then calculated by

$$\sigma_{f(\bar{x})} = \sqrt{N-1} \sigma_{f^J}$$

where

$$\sigma_{f^J}^2 \equiv \overline{(f^J)^2} - (\overline{f^J})^2$$

The covariance between two parameters  $\alpha$  and  $\beta$  parameters is given by

$$COV_{\alpha,\beta} = \frac{N-1}{N} \sum_{i=1}^N [\theta_i^J - \bar{\theta}^J]_a [\theta_i^J - \bar{\theta}^J]_b$$

where

$$\bar{\theta}_\alpha^J = \frac{1}{N} \sum i = 1 N \theta_J^{i,\alpha}$$

$N-1$  is used as the denominator rather than  $N$  since the population mean is unknown.

## *Bias estimation and correction*

The bias of the estimator, calculated over the entire sample, can be estimated by

$$f(X) \simeq N f(\bar{x}) - (N-1) \bar{f}^J$$

and reduces bias by an order of magnitude from  $O(N^{-1})$  to  $O(N^{-2})$ .

## Simulation

We simulate an object with known properties using the biodyn package then conduct an assessment.

```
library(plyr)
library(biodyn)
library(ggplot2)
library(ggplotFL)

bd=sim()
bd=window(bd,end=49)
```

A proxy for stock abundance is also required to estimate the stock parameters, Therefore an unbiased catch per effort (CPUE) series is generated from mid year stock biomass.

```
cpue=(stock(bd)[,-dims(bd)$year]+stock(bd)[,-1])/2
cpue=rlnorm(1,log(cpue),.2)
```

```
ggplot(cpue)+geom_point(aes(year,data))
```

Before performing an assessment we have to provide initial guesses and bounds for the parameters

```
#set parameters
setParams(bd) =cpue
setControl(bd)=params(bd)
control(bd)[3:4,"phase"]=-1
```

The stock assessment can now be fitted

```
#fit
bdHat=fit(bd,cpue)

plot(biodyns("Estimate"=bdHat,"Actual"=bd))+
  theme(legend.position="bottom")
```

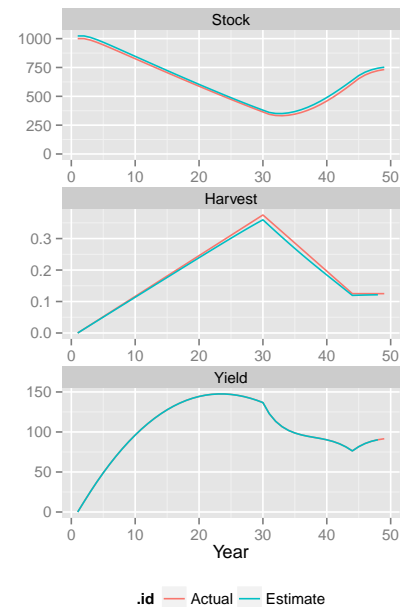
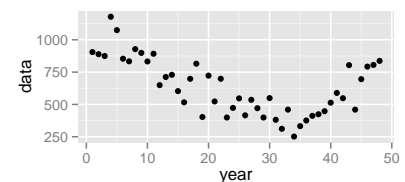


Figure 1: Fitted compared to actual time series

### Jack knife procedure

The CPUE index is jack knifed and the model refitted for each resampled data set, which are held in the iter dim.

```
bdJK=fit(bdHat,jackknife(cpue))
```

The normal plot method can not be used as the iter does not hold samples from a probability distribution. Instead plotJack is used which estimates the error bars.

```
plotJack(bdHat,bdJK)
```

### Estimates

Estimates of the bias and SEs can now be calculated, these can be done for FLQuant and FLPar objects, e.g. for current stock biomass

```
#FLQuant
```

```
library(ggplotFL)
```

```
true=stock(bd)[,49]
```

```
hat =stock(bdHat)[,49]
```

```
jack=stock(bdJK)[,49]
```

```
n =dims(jack)$iter
```

```
mn =apply(jack, 1:5, mean)
```

```
rsdl=sweep(jack,1:5,mn,"-")
```

```
ss =apply(rsdl^2,1:5,sum)
```

```
bias =(n-1)*(hat-mn)
```

```
biasCorrected=n*hat-(n-1)*mn
```

```
se =sqrt(((n-1)/n)*ss)
```

Reference points

```
# FLPar
```

```
true=refpts(bd)
```

```
hat =refpts(bdHat)
```

```
jack=refpts(bdJK)
```

```
n =dims(jack)$iter
```

```
mn =apply(jack, seq(length(dim(jack))-1), mean)
```

```
rsdl=sweep(jack, seq(length(dim(jack))-1), mn,"-")
```



Figure 2: Jack knife error bars

```
ss =apply(rsd1^2,seq(length(dim(jack))-1),sum)
```

```
bias      =(n-1)*(hat-mn)
```

```
biasCorrected=n*hat-(n-1)*mn
```

```
se =sqrt(((n-1)/n)*ss)
```