

Fitting a von Bertalanffy Growth Function

Derek H. Ogle, Northland College

20-Aug-2016

Preliminaries

```
> library(FSAdata)           # for TroutBR data
> library(FSA)               # for filterD(), headtail(), col2rgbt(), vbFuns(), vbStart()
> library(nlstools)          # for nlsBoot()
```

Loading the Data and Some Preparations

```
> data(TroutBR)
> str(TroutBR)
'data.frame':   851 obs. of  3 variables:
 $ t1      : int  16 16 17 17 17 17 17 17 17 17 ...
 $ age     : int   4 4 2 3 3 3 3 3 3 4 ...
 $ species: Factor w/ 2 levels "Brown","Rainbow": 1 1 1 1 1 1 1 1 1 1 ...

> rbt <- filterD(TroutBR,species=="Rainbow")
> headtail(rbt)
      t1 age species
1    12  3 Rainbow
2    14  3 Rainbow
3    14  3 Rainbow
625  31  7 Rainbow
626  31  9 Rainbow
627  32  7 Rainbow

> xlbl <- "Age (yrs)"
> ylbl <- "Total Length (in)"
> clr <- col2rgbt("black",0.05)
```

Fit Typical VBGF

Declare a Function

```
> vb <- vbFuns("Typical",msg=TRUE)
You have chosen the 'Typical', 'Traditional', or 'BevertonHolt' parameterization.
```

$$E[L|t] = L_{\infty} * (1 - \exp(-K * (t - t_0)))$$

where L_{∞} = asymptotic mean length

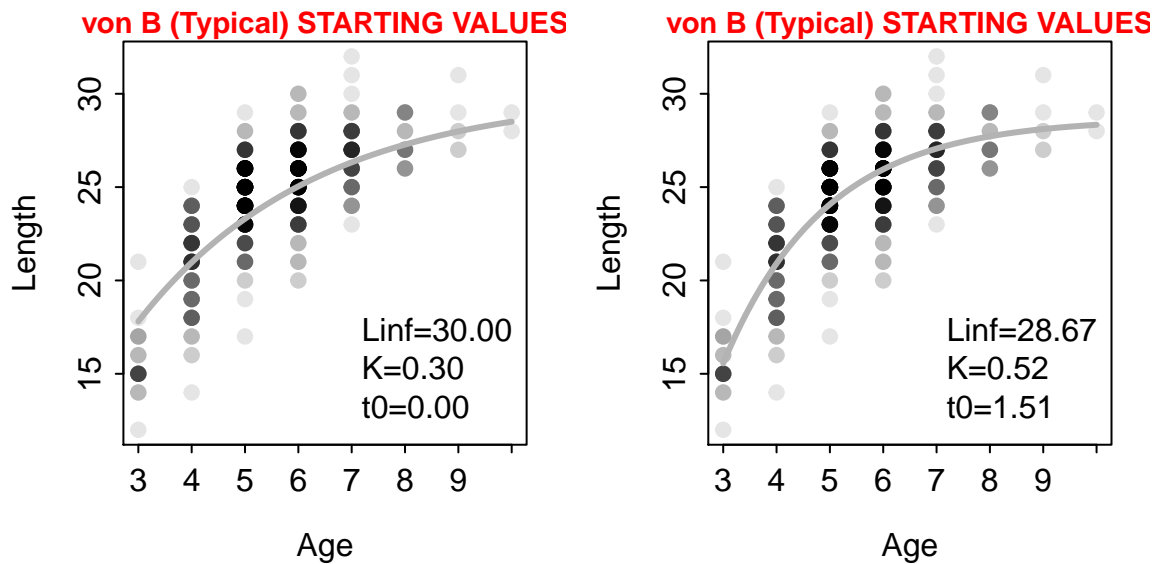
K = exponential rate of approach to L_{∞}

t_0 = the theoretical age when length = 0 (a modeling artifact)

```
> vb
function(t,Linf,K=NULL,t0=NULL) {
  if (length(Linf)==3) { K <- Linf[[2]]
                        t0 <- Linf[[3]]
                        Linf <- Linf[[1]] }
  Linf*(1-exp(-K*(t-t0)))
}
<environment: 0x06c8fea8>
```

Find Starting Values

```
> # Demos manual generation with plot ... LEFT plot
> svb <- vbStarts(tl~age,data=rbt,type="Typical",plot=TRUE,
  fixed=list(Linf=30,K=0.3,t0=0))
> # Demos automatic generation ... RIGHT plot
> svb <- vbStarts(tl~age,data=rbt,type="Typical",plot=TRUE)
```



Fit the Model

```
> fit1 <- nls(tl~vb(age,Linf,K,t0),data=rbt,start=svb)
> summary(fit1,correlation=TRUE)
```

Formula: $tl \sim vb(\text{age}, Linf, K, t0)$

Parameters:

	Estimate	Std. Error	t value	Pr(> t)
Linf	27.71191	0.28383	97.64	<2e-16
K	0.63242	0.04248	14.89	<2e-16
t0	1.71686	0.10159	16.90	<2e-16

Residual standard error: 1.775 on 624 degrees of freedom

Correlation of Parameter Estimates:

	Linf	K
K	-0.91	
t0	-0.71	0.92

Number of iterations to convergence: 3

Achieved convergence tolerance: 9.57e-06

```
> ( cf <- coef(fit1) )
```

	Linf	K	t0
	27.7119083	0.6324231	1.7168636

```
> confint(fit1)
```

	2.5%	97.5%
Linf	27.1916077	28.3279785
K	0.5499956	0.7192266
t0	1.4930214	1.8999245

```

> boot1 <- nlsBoot(fit1,niter=1000)
> str(boot1)
List of 4
 $ coefboot: num [1:1000, 1:3] 27.8 28.2 27.7 27.9 28.1 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:3] "Linf" "K" "t0"
 $ rse      : num [1:1000] 1.85 1.71 1.72 1.83 1.73 ...
 $ bootCI   : num [1:3, 1:3] 27.722 0.629 1.709 27.211 0.554 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:3] "Linf" "K" "t0"
  .. ..$ : chr [1:3] "Median" "2.5%" "97.5%"
 $ estiboot: num [1:3, 1:2] 27.7336 0.6321 1.7101 0.2931 0.0435 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:3] "Linf" "K" "t0"
  .. ..$ : chr [1:2] "Estimate" "Std. error"
 - attr(*, "class")= chr "nlsBoot"

```

```

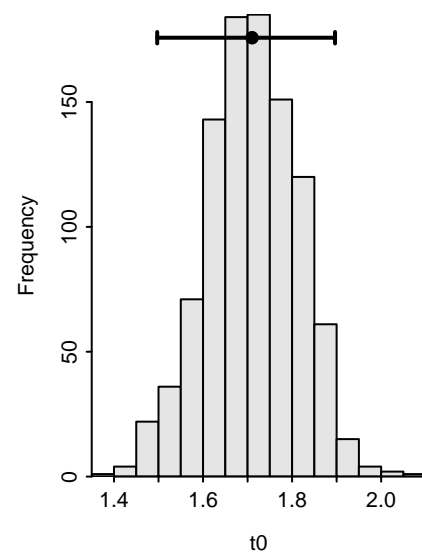
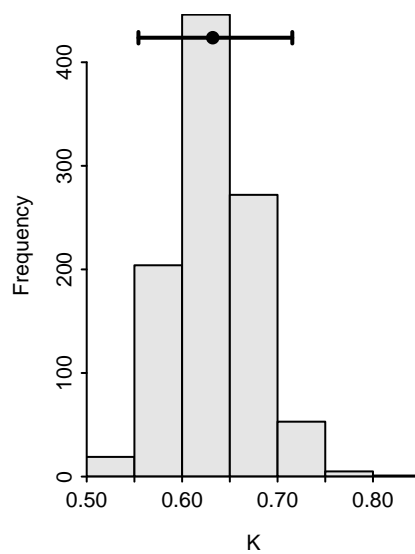
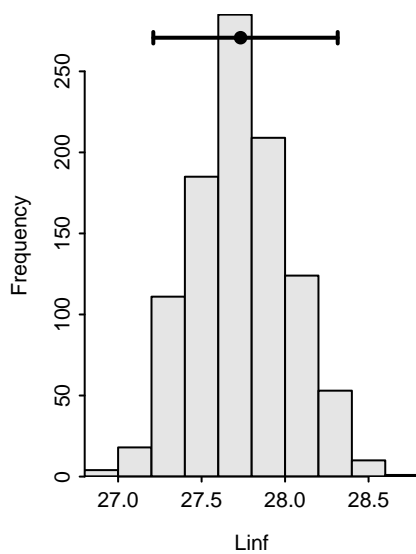
> headtail(boot1$coefboot)
      Linf      K      t0
[1,] 27.82869 0.6217399 1.692651
[2,] 28.23121 0.5665697 1.556142
[3,] 27.67403 0.6252850 1.655133
[998,] 28.36681 0.5885078 1.713334
[999,] 28.30371 0.5468962 1.506789
[1000,] 27.64197 0.6097591 1.591136

```

```

> confint(boot1,plot=TRUE,rows=1,cols=3)
      95% LCI      95% UCI
Linf 27.210856 28.3152377
K     0.554208 0.7154811
t0    1.497038 1.8963883

```



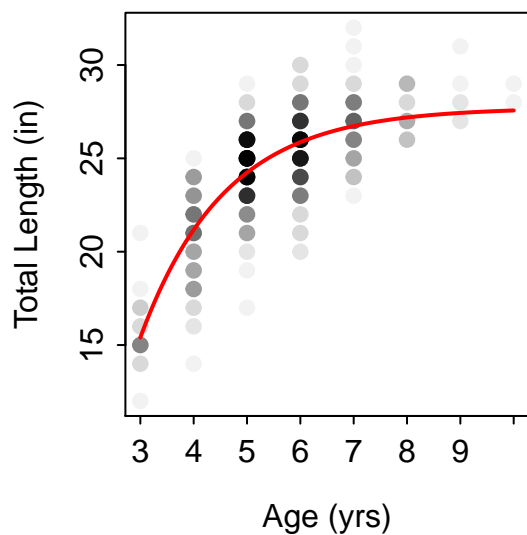
Make Predictions

```
> ageX <- 8
> predict(fit1,data.frame(age=ageX))
[1] 27.19077

> pv <- apply(boot1$coefboot,MARGIN=1,FUN=vb,t=ageX)
> quantile(pv,c(0.025,0.975))
      2.5%      97.5%
26.84727 27.54809
```

Visualize the Fit

```
> plot(tl~age,data=rbt,xlab=xlbl,ylab=ylbl,pch=19,col=clr)
> curve(vb(x,cf),from=3,to=10,n=500,lwd=2,col="red",add=TRUE)
```



```
> residPlot(fit1)
```

