von Bertalanffy Growth Function

Preliminaries

Load Necessary Packages

```
> library(FSA)      # for filterD(), headtail(), col2rgbt(), vbFuns(), vbStart()
> library(dplyr)    # for mutate(), select()
> library(nlstools) # for nlsBoot()
```

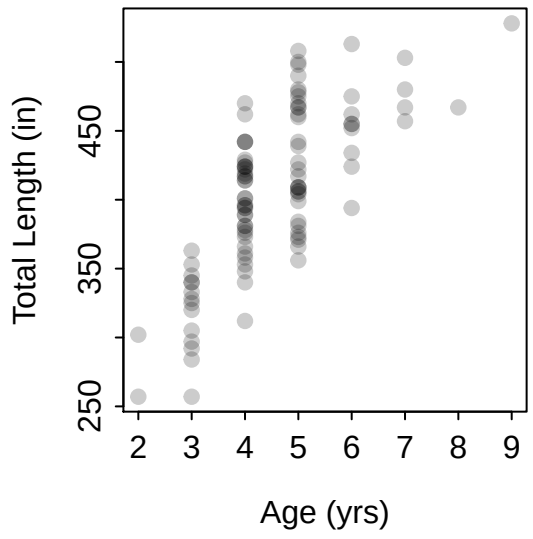
Load Data and Make Some Preparations

```
> # Set your working directory to where your external data files (and scripts) are located.
> setwd("C:/aaaWork/Web/GitHub/RcourseNunavut2016/Handouts")
> dSC <- read.csv("SawyerCo_reduced.csv")
> wae <- filterD(dSC,waterbody=="NELSON LAKE",species=="Walleye",!is.na(len),!is.na(age))
```

```
> xlbl <- "Age (yrs)"
> ylbl <- "Total Length (in)"
> clr1 <- "black"
> clr2 <- col2rgbt(clr1,1/5)
```

Quick Summaries

```
> plot(len~age,data=wae,pch=19,col=clr2,xlab=xlbl,ylab=ylbl)
```



```
> Summarize(len~age,data=wae,digits=1)
Warning: RHS variable was converted to a factor.
  age  n nvalid  mean   sd min    Q1 median    Q3 max percZero
1   2  2      2 279.5 31.8 257 268.2  279.5 290.8 302        0
2   3 14     14 320.1 29.6 257 299.0  326.5 340.0 363        0
3   4 36     36 399.3 34.9 312 377.5  398.5 424.0 470        0
4   5 31     31 430.7 44.9 356 401.5  422.0 468.5 508        0
5   6  9      9 451.6 33.2 394 434.0  455.0 462.0 513        0
6   7  4      4 476.8 19.9 457 464.5  473.5 485.8 503        0
7   8  1      1 467.0   NA 467 467.0  467.0 467.0 467        0
8   9  1      1 528.0   NA 528 528.0  528.0 528.0 528        0
```

Fit Typical VBGF

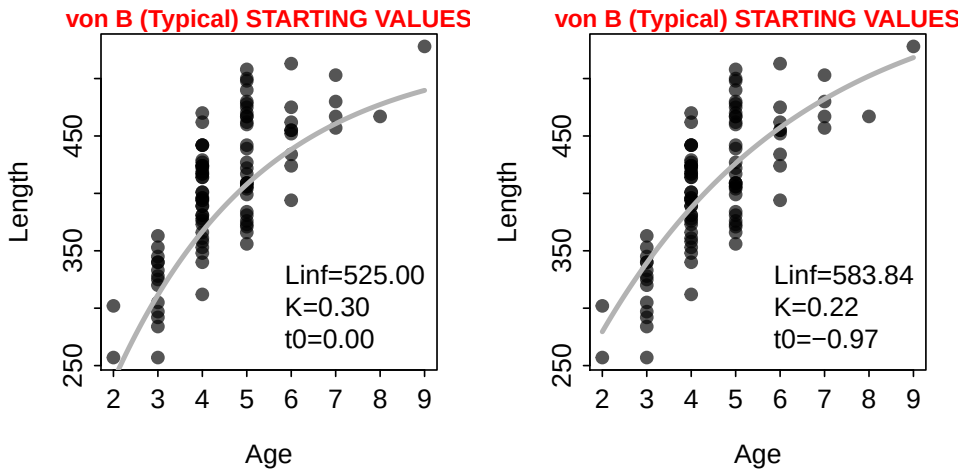
Declare a Function

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

  E[L|t] = Linf*(1-exp(-K*(t-t0)))

  where Linf = asymptotic mean length
          K = exponential rate of approach to Linf
         t0 = 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: 0x043249d4>
```

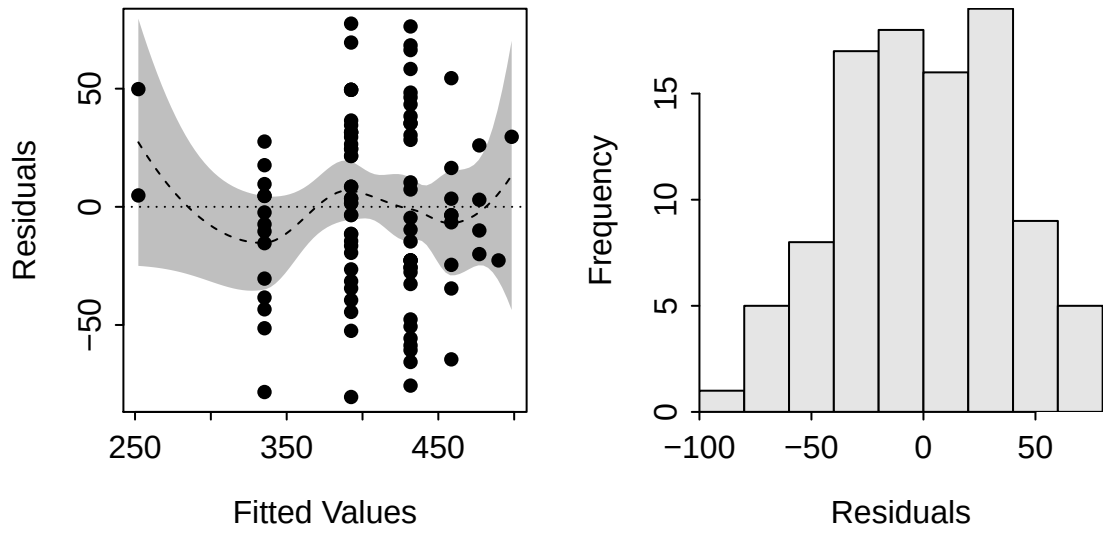
Find Starting Values

```
> # Demos manual generation with plot ... LEFT plot
> svb <- vbStarts(len~age,data=wae,type="Typical",plot=TRUE,fixed=list(Linf=525,K=0.3,t0=0))
> # Demos automatic generation ... RIGHT plot
> svb <- vbStarts(len~age,data=wae,type="Typical",plot=TRUE)
```



Fit the Model

```
> fit1 <- nls(len~vb(age,Linf,K,t0),data=wae,start=svb)
> residPlot(fit1)
```



Summarize the Fit

```
> summary(fit1,correlation=TRUE)

Formula: len ~ vb(age, Linf, K, t0)

Parameters:
     Estimate Std. Error t value Pr(>|t|)
Linf 517.3784    31.8747  16.232  < 2e-16
K      0.3765     0.1035   3.639 0.000445
t0     0.2249     0.5244   0.429 0.669063

Residual standard error: 37.56 on 95 degrees of freedom

Correlation of Parameter Estimates:
   Linf  K
K  -0.95
t0 -0.80  0.94

Number of iterations to convergence: 8
Achieved convergence tolerance: 1.463e-06
```

```
> ( cf <- coef(fit1) )
       Linf           K          t0
517.3783956   0.3764900   0.2248583
```

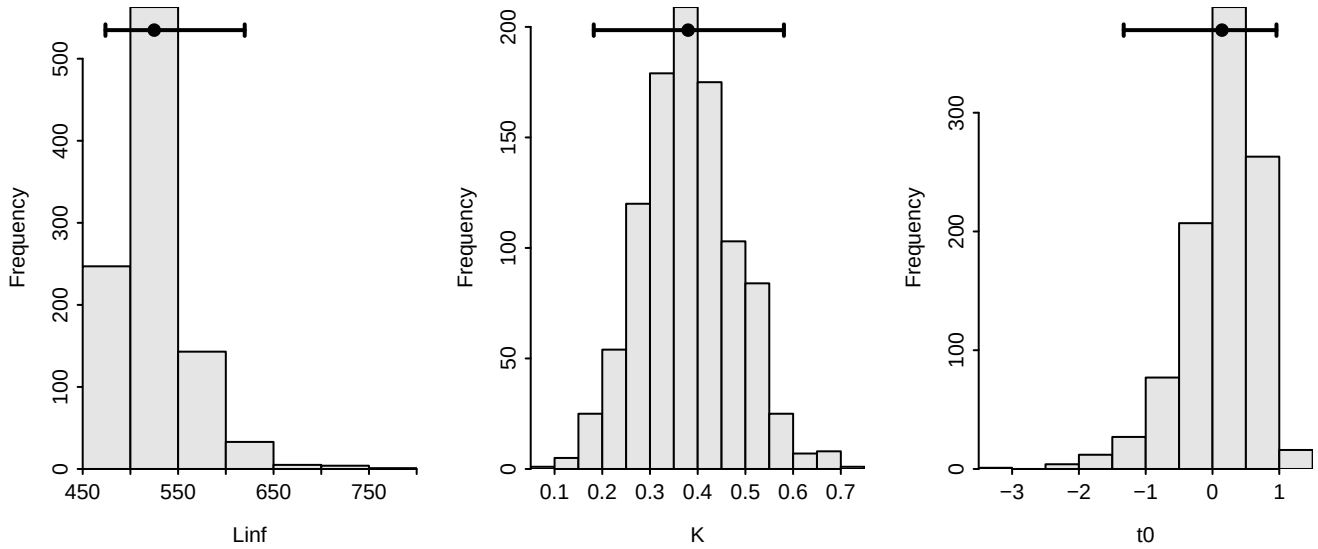
```
> confint(fit1)
Waiting for profiling to be done...
            2.5%       97.5%
Linf 472.3980539 629.8699947
K      0.1831627   0.5889712
t0    -1.3560959   0.9735696
```

```
> boot1 <- nlsBoot(fit1,niter=1000)
Warning in nlsBoot(fit1, niter = 1000): The fit did not converge 4 times during bootstrapping
```

```
> str(boot1)
List of 4
 $ coefboot: num [1:996, 1:3] 584 473 543 532 480 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:3] "Linf" "K" "t0"
 $ rse     : num [1:996] 36.8 36.5 34.6 40 39.3 ...
 $ bootCI  : num [1:3, 1:3] 518.641 0.377 0.236 473.823 0.182 ...
  ..- 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] 524.9511 0.3799 0.142 37.7737 0.0992 ...
  ..- 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,]   583.5195 0.2623084 -0.2765158
[2,]   473.2263 0.5324424  0.6123111
[3,]   543.4278 0.3343795  0.1158536
[994,] 533.8433 0.3469700  0.2342875
[995,] 507.9200 0.4301398  0.4756919
[996,] 480.1709 0.4600134  0.3838285
```

```
> confint(boot1,plot=TRUE,rows=1,cols=3)
         95% LCI     95% UCI
Linf 473.8233134 619.7879874
K      0.1821066   0.5807424
t0    -1.3295883   0.9580000
```



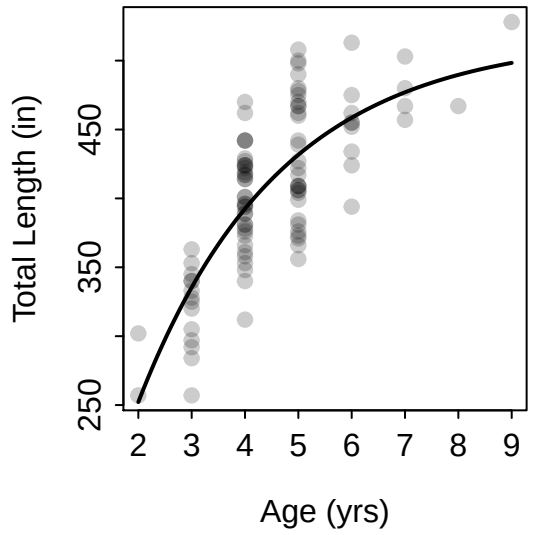
Make Predictions

```
> ageX <- 5
> predict(fit1,data.frame(age=ageX))
[1] 431.6674
```

```
> pv <- apply(boot1$coefboot,MARGIN=1,FUN=vb,t=ageX)
> quantile(pv,c(0.025,0.975))
    2.5%    97.5%
422.7171 440.0841
```

Visualize the Fit

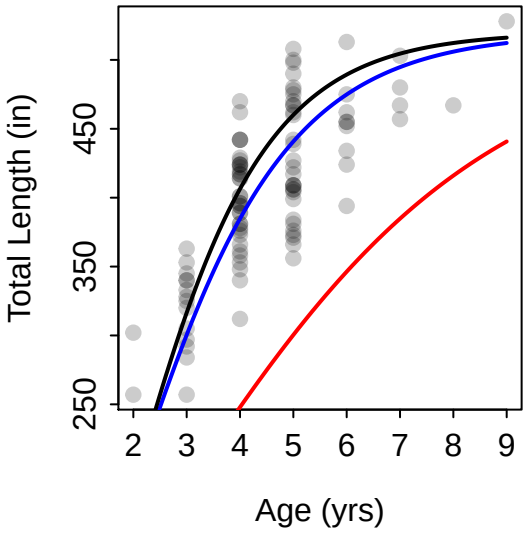
```
> plot(len~age,data=wae,xlab=xlbl,ylab=ylbl,pch=19,col=clr2)
> curve(vb(x,cf),from=2,to=9,n=500,lwd=2,col=clr1,add=TRUE)
```



Fit Gompertz Growth Function

```
> gomp <- GompertzFuns(msg=TRUE)
You have chosen the 'Ricker1'parameterization of the Gompertz function.

  E[L|t] = Linf*exp(-exp(-gi*(t-ti)))

  where Linf = asymptotic mean length
          gi = instantaneous growth rate at the inflection point
          ti = time at the inflection point
```

```
> plot(len~age,data=wae,pch=19,col=clr2,xlab=xlbl,ylab=ylbl)
> curve(gomp(x,Linf=520,gi=0.3,ti=3),from=2,to=9,n=500,lwd=2,add=TRUE,col="red")
> curve(gomp(x,Linf=520,gi=0.6,ti=2),from=2,to=9,n=500,lwd=2,add=TRUE,col="blue")
> curve(gomp(x,Linf=520,gi=0.7,ti=2),from=2,to=9,n=500,lwd=2,add=TRUE,col=clr1)
```




```
> fit2 <- nls(len~gomp(age,Linf,gi,ti),data=wae,start=list(Linf=520,gi=0.7,ti=2))
> AIC(fit1,fit2)
     df      AIC
fit1  4 993.7674
fit2  4 993.3694
```

```
> plot(len~age,data=wae,xlab=xlbl,ylab=ylbl,pch=19,col=clr2,xlim=c(0,10),ylim=c(0,550))
> curve(vb(x,cf),from=0,to=10,n=500,lwd=4,col=clr1,add=TRUE)
> curve(gomp(x,coef(fit2)),from=0,to=10,n=500,lwd=2,col="red",add=TRUE)
> legend("bottomright",c("von Bertalanffy","Gompertz"),col=c("black","red"),lwd=2,bty="n",cex=0.8)
```