NRG Example

Kirstin Holsman

12/13/2017

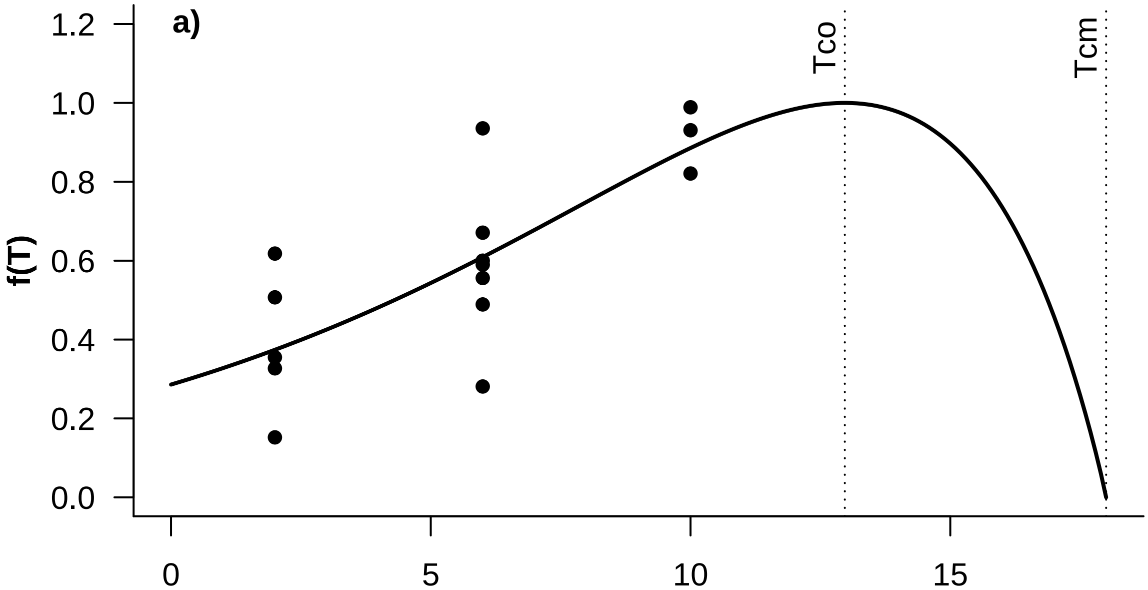
Part 1: Set up

```
rm(list=ls())
graphics.off()
setwd("/Users/kholsman/GitHub/NRG")
## some data for Halibut
  load("data/HalibutC.Rdata")
  #load("Halibut_results.Rdata")
  load("data/alldat4B.Rdata")
  load("data/alldatA3.Rdata")
  load("data/alldat4CD.Rdata")
  source("R/bioE.R")
# Set up some parms
  hal_par<-list(Ceq=2,Req=1,Weq=1,RFR=1,Qox=13560,CA=0.0627,CB=-0.108,Tco=12.9699,
              Tcm=18,QC=3.084,RK4=0.3729,RK1=0.008,Am=0.008,Bact=0.2215,
              logsigma=-9.412,RA=0.0016,RB=-0.1848,QR=0.0644,Trl=12,Trm=NA,
              Tro=0.25,UA=0.0332,UA.sigma=0.082,FA=0.2,SA=0.1181,SA.sigma=0.2574)
  PARMS_USE<-hal_par
  # P..Halibut    Epipelagic  Halibut  4.80 kJ  +/-   0.7 2011 Mar Bio : Seasonal cycles
  # in whole-body proximate composition and energy content of forage fish vary with
  # water depth. Johanna J. Vollenweider • Ron A. Heintz •Lawrence Schaufler • Robert Bradshaw
  HalibutED<-4800  # Halibut energy density
  EpreyUse<-4598.07 # average across sizes
  ebs_data_lowpreyE<-list(W=100,TempC=seq(0,25,.1),Eprey=3400,Epred=HalibutED,indgst=0,diet=0)  #4184
  ebs_data_highpreyE<-list(W=100,TempC=seq(0,25,.1),Eprey=5539.6,Epred=HalibutED,indgst=0,diet=0)  #4
  ebs_data<-list(W=100,TempC=seq(0,25,.1),Eprey=EpreyUse,Epred=HalibutED,indgst=0,diet=0)
```

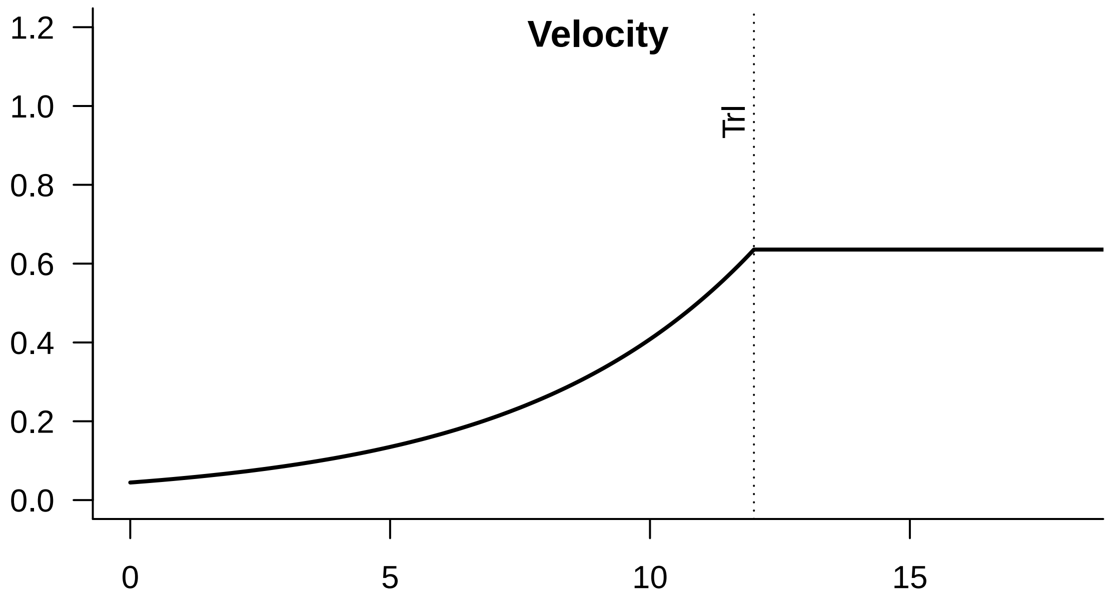
Part 2: Explore main functions

```
# G = (Cmax*f(Tc)*RFR)-(R*Act+SDA+F+U)

# plot the consumption data and f(Tc) functions
  c_data<-ebs_data
  c_data$fTCmodel<-function(TempC){-.5*TempC}
  ft<-fTC_fun(par=PARMS_USE,data=c_data)
# plot the function
  ylimm<-list();ylimm[[1]]<-c(0,1.2)
  plot(ebs_data$TempC,ft,type="l",xlim=c(0,18), ylim=ylimm[[1]],axes=FALSE,ylab="",xlab="",lwd=2)
  points(FTdat,pch=16)
  axis(1);axis(1,at=c(-10,40))
  axis(2,las=2);axis(2,at=c(-10,10))
```

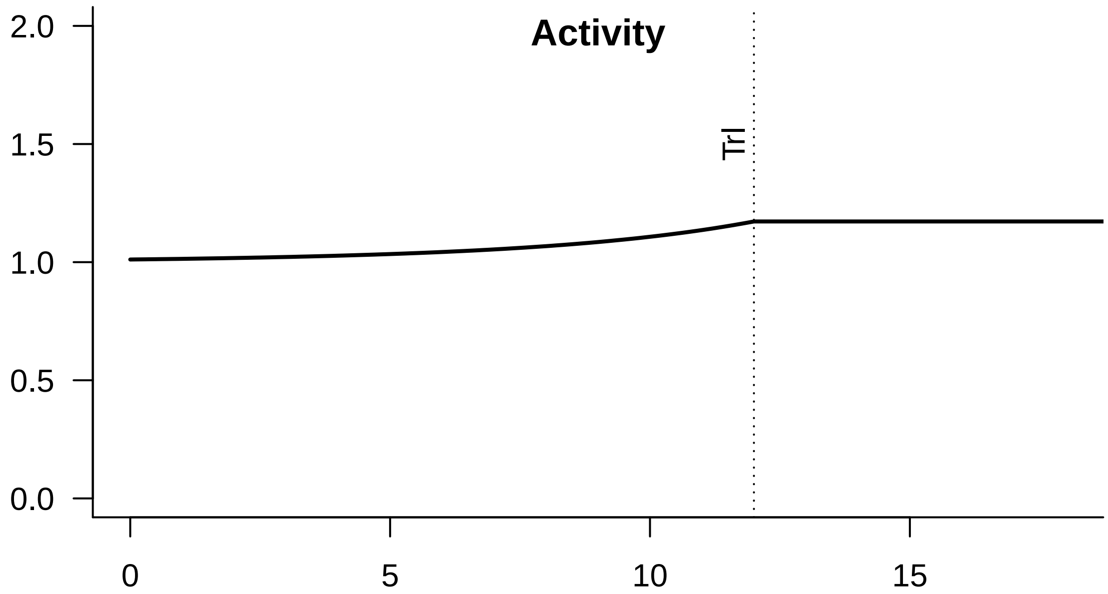
```r
abline(v=PARMS_USE$Tco,lty=3)
abline(v=PARMS_USE$Tcm,lty=3)
abline(v=PARMS_USE$Qc,lty=3)
text(PARMS_USE$Tco-.4,ylimm[[1]][2]*.95,"Tco", srt =90)
text(PARMS_USE$Tcm-.4,ylimm[[1]][2]*.95,"Tcm", srt =90)
text(PARMS_USE$Qc-.4,ylimm[[1]][2]*.95,"Qc", srt =90)
mtext("f(T)",2,outer=FALSE,line=2.5,font=2,cex=1)
text(.3,ylimm[[1]][2],"a)",font=2)
```




```r
# now plot swim velocity functions:
vel_dat<-ebs_data
vel_dat$fitLL<-FALSE
vel_dat$velobs<-NA
veld<-Resp_fun(par=PARMS_USE,data=vel_dat)  # returns Act, fTr,and Vel

plot(ebs_data$TempC,veld$Vel,type="l",xlim=c(0,18), ylim=ylimm[[1]],axes=FALSE,ylab="",xlab="",lwd=
axis(1);axis(1,at=c(-10,40));axis(2,las=2);axis(2,at=c(-10,10))
abline(v=PARMS_USE$Trl,lty=3);text(PARMS_USE$Trl-.4,ylimm[[1]][2]*.8,"Trl", srt =90)
```




```
plot(ebs_data$TempC,veld$Act,type="l",xlim=c(0,18), ylim=c(0,2),axes=FALSE,ylab="",xlab="",lwd=2,ma
axis(1);axis(1,at=c(-10,40));axis(2,las=2);axis(2,at=c(-10,10))
abline(v=PARMS_USE$Trl,lty=3);text(PARMS_USE$Trl-.4,1.5,"Trl", srt =90)
```

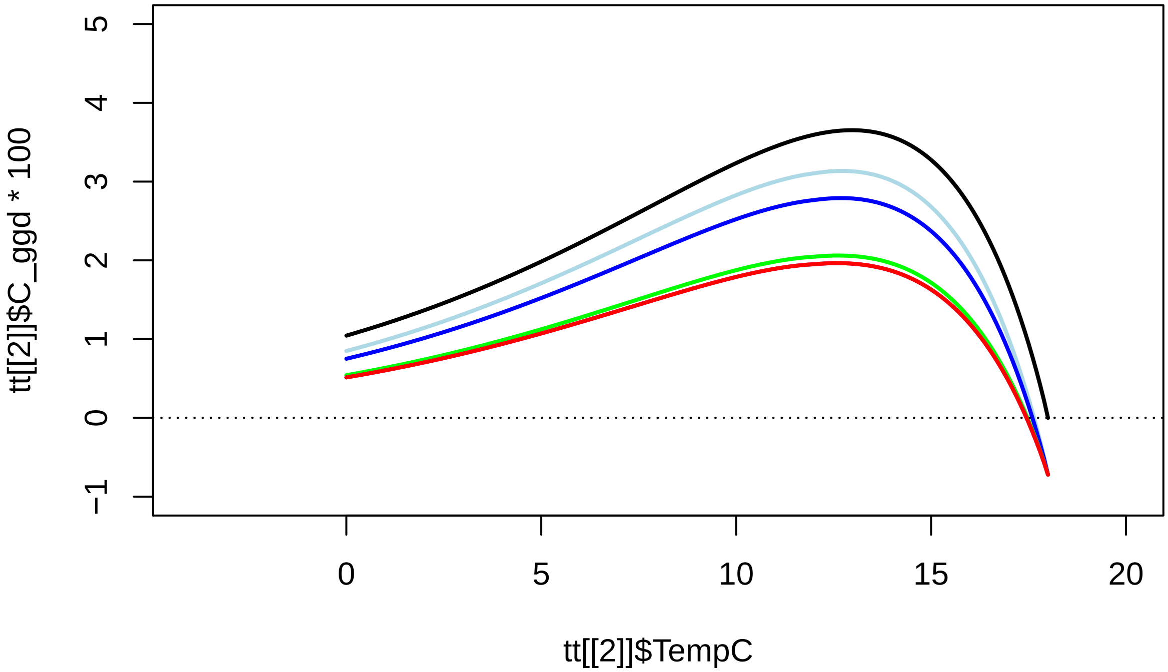



```
# now predict waste functions
  w_dat<-ebs_data
  w_dat$C_in<-10  # can be grams consumed or joules consumed
  Waste_fun(par=PARMS_USE,data=w_dat)
```

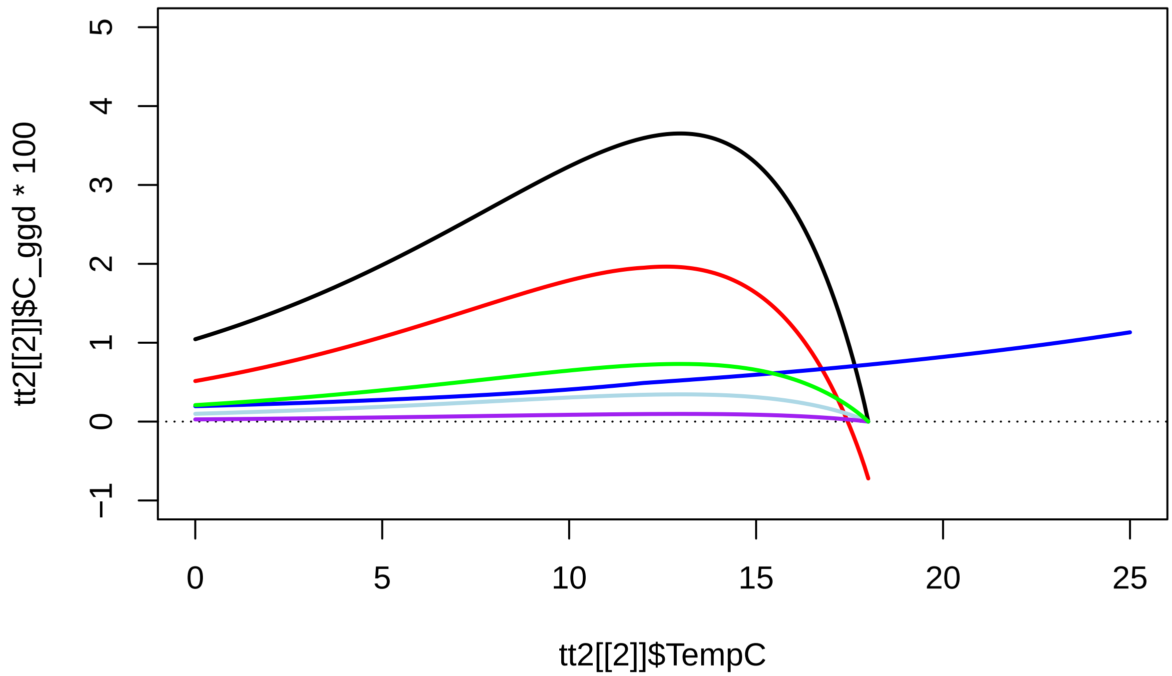
```
## $F
## [1] 2
##
## $U
## [1] 0.2656
```

```
# now put it all together for halibut
  tt<-bioE(par=hal_par,data=ebs_data)
```

```
plot(tt[[2]]$TempC,tt[[2]]$C_ggd*100,type="l",lwd=2,ylim=c(-1,5),xlim=c(-4,20));abline(h=0,lty=3)
lines(tt[[2]]$TempC,tt[[2]]$C_ggd*100-(tt[[2]]$R_ggd*100),type="l",lwd=2,col="lightblue")
lines(tt[[2]]$TempC,tt[[2]]$C_ggd*100-(tt[[2]]$R_ggd+tt[[2]]$SDA_ggd)*100,type="l",lwd=2,col="blue")
lines(tt[[2]]$TempC,tt[[2]]$C_ggd*100-(tt[[2]]$F_ggd+tt[[2]]$R_ggd+tt[[2]]$SDA_ggd)*100,type="l",lw
lines(tt[[2]]$TempC,tt[[2]]$C_ggd*100-(tt[[2]]$U_ggd+tt[[2]]$F_ggd+tt[[2]]$R_ggd+tt[[2]]$SDA_ggd)*1
# the difference is what is avail for growth:
lines(tt[[2]]$TempC,tt[[2]]$G_ggd*100,type="l",lwd=2,col="red",ylim=c(-1,5),xlim=c(-4,20));abline(h
```

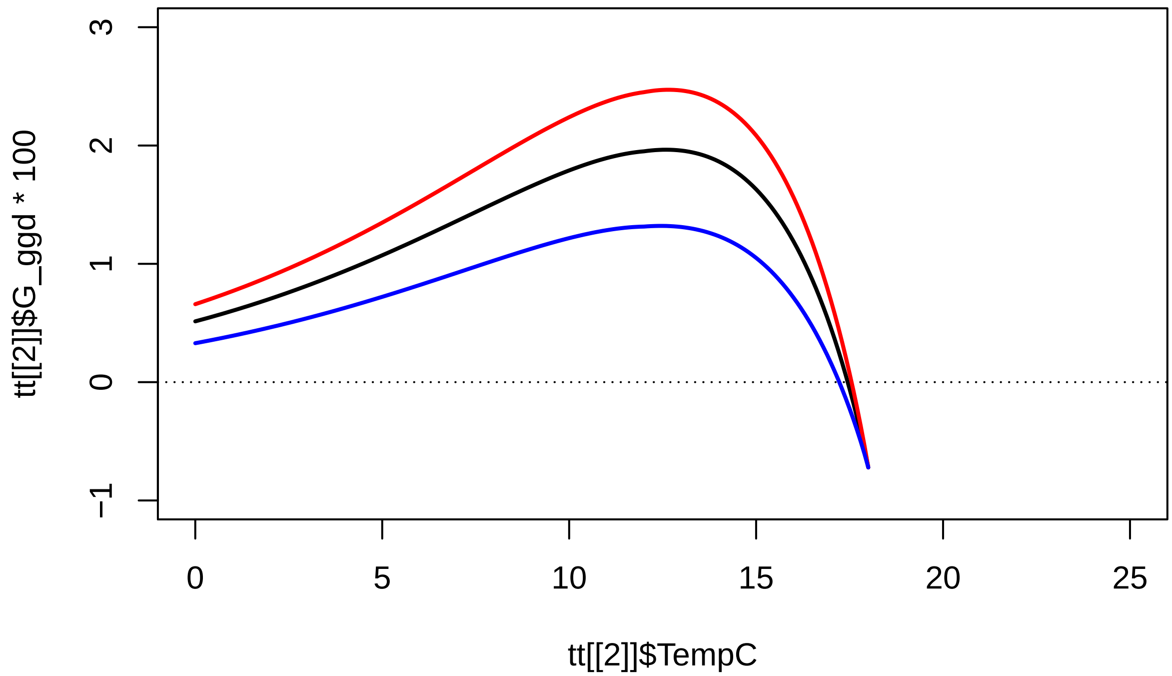



```
tt2<-tt
plot(tt2[[2]]$TempC,tt2[[2]]$C_ggd*100,type="l",lwd=2,ylim=c(-1,5));abline(h=0,lty=3)
lines(tt2[[2]]$TempC,tt2[[2]]$G_ggd*100,type="l",lwd=2,col="red")
lines(tt2[[2]]$TempC,tt2[[2]]$SDA_ggd*100,type="l",lwd=2,col="lightblue")
lines(tt2[[2]]$TempC,tt2[[2]]$R_ggd*100,type="l",lwd=2,col="blue")
lines(tt2[[2]]$TempC,tt2[[2]]$U_ggd*100,type="l",lwd=2,col="purple")
lines(tt2[[2]]$TempC,tt2[[2]]$F_ggd*100,type="l",lwd=2,col="green")
```

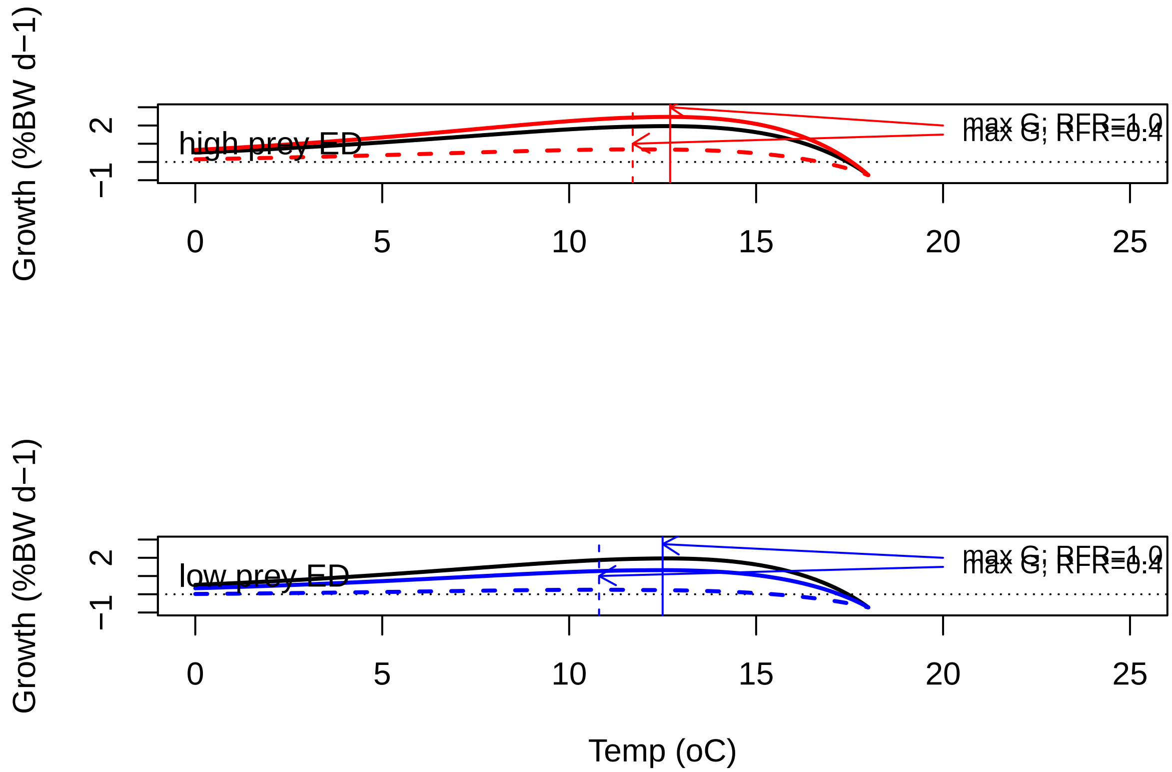


Part 3: Compare effect of changing prey ED and quantity

```r
# now let's compare to high qual prey
  ttH<-bioE(par=hal_par,data=ebs_data_highpreyE)
  ttL<-bioE(par=hal_par,data=ebs_data_lowpreyE)

  plot(tt[[2]]$TempC,tt[[2]]$G_ggd*100,type="l",lwd=2,ylim=c(-1,3));abline(h=0,lty=3)
  lines(ttH[[2]]$TempC,ttH[[2]]$G_ggd*100,col="red",lwd=2)
  lines(ttL[[2]]$TempC,ttL[[2]]$G_ggd*100,col="blue",lwd=2)
```




```r
# now let's compare to lower foraging rate (e.g., less prey)
  RFR2<-0.4
  hal_par2<-hal_par; hal_par2$RFR<-RFR2
  ttH04<-bioE(par=hal_par2,data=ebs_data_highpreyE)
  ttL04<-bioE(par=hal_par2,data=ebs_data_lowpreyE)
  par(mfrow=c(2,1))
  plot(tt[[2]]$TempC,tt[[2]]$G_ggd*100,type="l",lwd=2,ylim=c(-1,3),ylab="Growth (%BW d-1)",xlab="");al
  lines(ttH[[2]]$TempC,ttH[[2]]$G_ggd*100,col="red",lwd=2)
  lines(ttH04[[2]]$TempC,ttH04[[2]]$G_ggd*100,col="red",lwd=2,lty=2)
  abline(v=ttH04[[2]]$TempC[ttH04[[2]]$G_ggd==max(ttH04[[2]]$G_ggd,na.rm=T)],col="red",lty=2)
  abline(v=ttH[[2]]$TempC[ttH[[2]]$G_ggd==max(ttH[[2]]$G_ggd,na.rm=T)],col="red",lty=1)
  arrows(x1=ttH[[2]]$TempC[ttH[[2]]$G_ggd==max(ttH[[2]]$G_ggd,na.rm=T)],x0=20,y1=3,y0=2,col="red",leng
  arrows(x1=ttH04[[2]]$TempC[ttH04[[2]]$G_ggd==max(ttH04[[2]]$G_ggd,na.rm=T)],x0=20,y1=1,y0=1.5,col="r
  text(20,1.5, paste0("max G; RFR=",RFR2),pos=4,cex=.8);  text(20,2, "max G; RFR=1.0",pos=4,cex=.8)
  mtext("high prey ED",side=3,adj=.025,line=-1.5)

  plot(tt[[2]]$TempC,tt[[2]]$G_ggd*100,type="l",lwd=2,ylim=c(-1,3),ylab="Growth (%BW d-1)",xlab="Temp
  lines(ttL[[2]]$TempC,ttL[[2]]$G_ggd*100,col="blue",lwd=2)
  lines(ttL04[[2]]$TempC,ttL04[[2]]$G_ggd*100,col="blue",lwd=2,lty=2)
  abline(v=ttL04[[2]]$TempC[ttL04[[2]]$G_ggd==max(ttL04[[2]]$G_ggd,na.rm=T)],col="blue",lty=2)
  abline(v=ttL[[2]]$TempC[ttL[[2]]$G_ggd==max(ttL[[2]]$G_ggd,na.rm=T)],col="blue",lty=1)
  arrows(x1=ttL[[2]]$TempC[ttL[[2]]$G_ggd==max(ttL[[2]]$G_ggd,na.rm=T)],x0=20,y1=2.75,y0=2,col="blue"
  arrows(x1=ttL04[[2]]$TempC[ttL04[[2]]$G_ggd==max(ttL04[[2]]$G_ggd,na.rm=T)],x0=20,y1=1,y0=1.5,col="b
  text(20,1.5, paste0("max G; RFR=",RFR2),pos=4,cex=.8);  text(20,2, "max G; RFR=1.0",pos=4,cex=.8)
  mtext("low prey ED",side=3,adj=.025,line=-1.5)
```

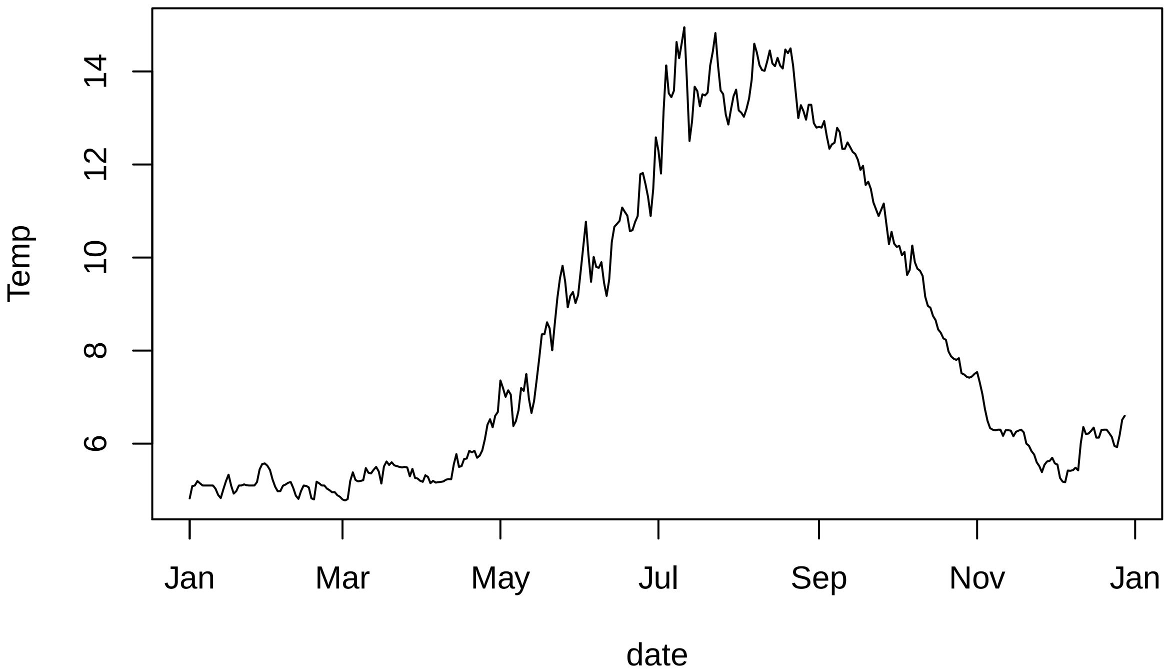


Part 4: Simulate growth over time

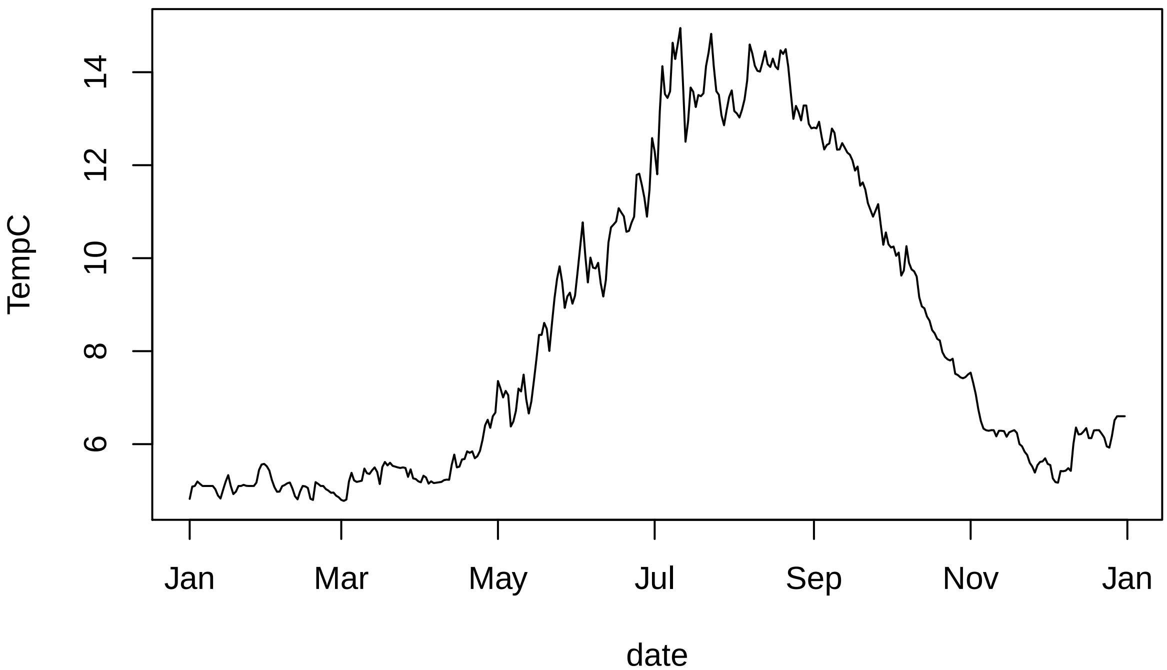
```
# now let's simulate growth overitme
  # load temperature data from A3 area buoy
  yr<-2005
  sub<-all.dat.A3
  sub<-sub[sub$Year==yr,]
  temp<-tapply(sub$Temp,as.character(sub$date),mean,na.rm=T)
  head(temp)
```

```
## 2005-01-01 2005-01-02 2005-01-03 2005-01-04 2005-01-05 2005-01-06
##   4.821739   5.087500   5.100000   5.195833   5.145833   5.100000
```

```
  A3.dat<-data.frame(date=strptime(names(temp), format="%Y-%m-%d"),Temp=temp)
  A3.dat<-A3.dat[order(A3.dat$date),]
  plot(A3.dat,type="l")
```

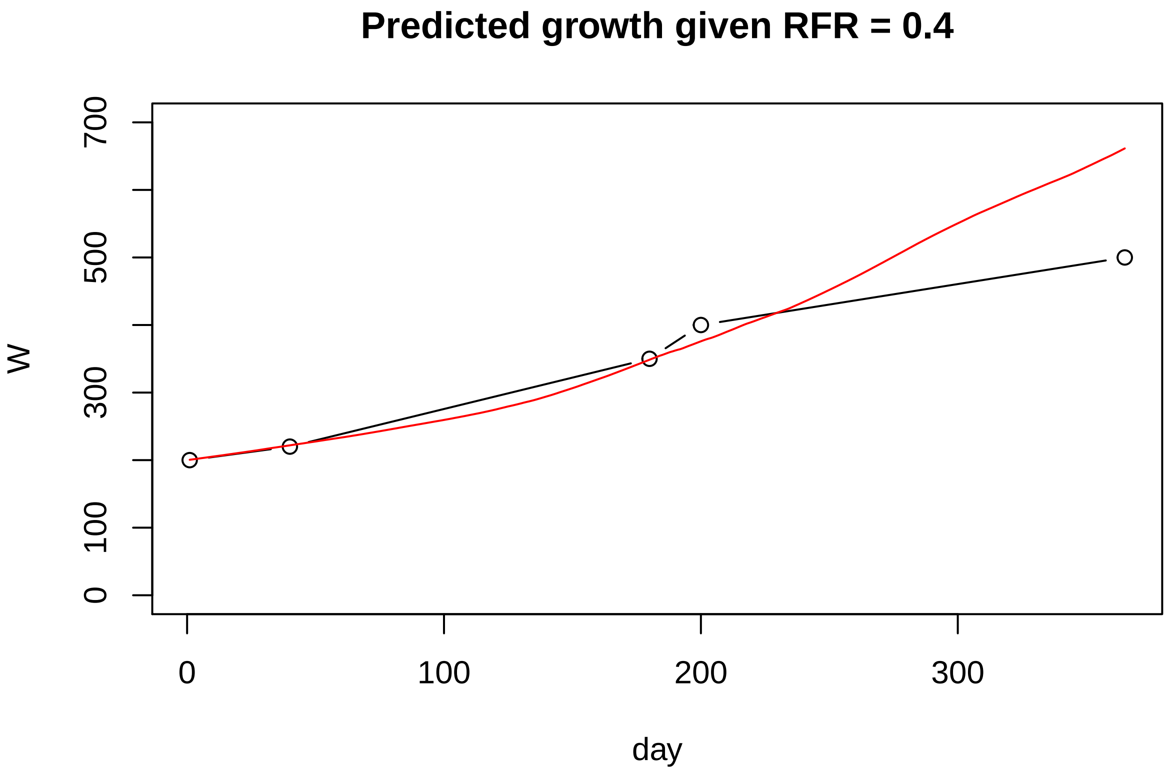



```
Tdat<-data.frame(date=seq.Date(as.Date('2005-01-01'), by = 'day', len = 365),TempC=NA)
Tdat$TempC[format(Tdat$date,"%F")%in%format(A3.dat$date,"%F")]<-A3.dat$Temp
cc<-(which(is.na(Tdat$TempC)))
for(i in 1:length(cc)) Tdat$TempC[cc[i]]<-Tdat$TempC[cc[i]-1]
plot(Tdat,type="l")
```

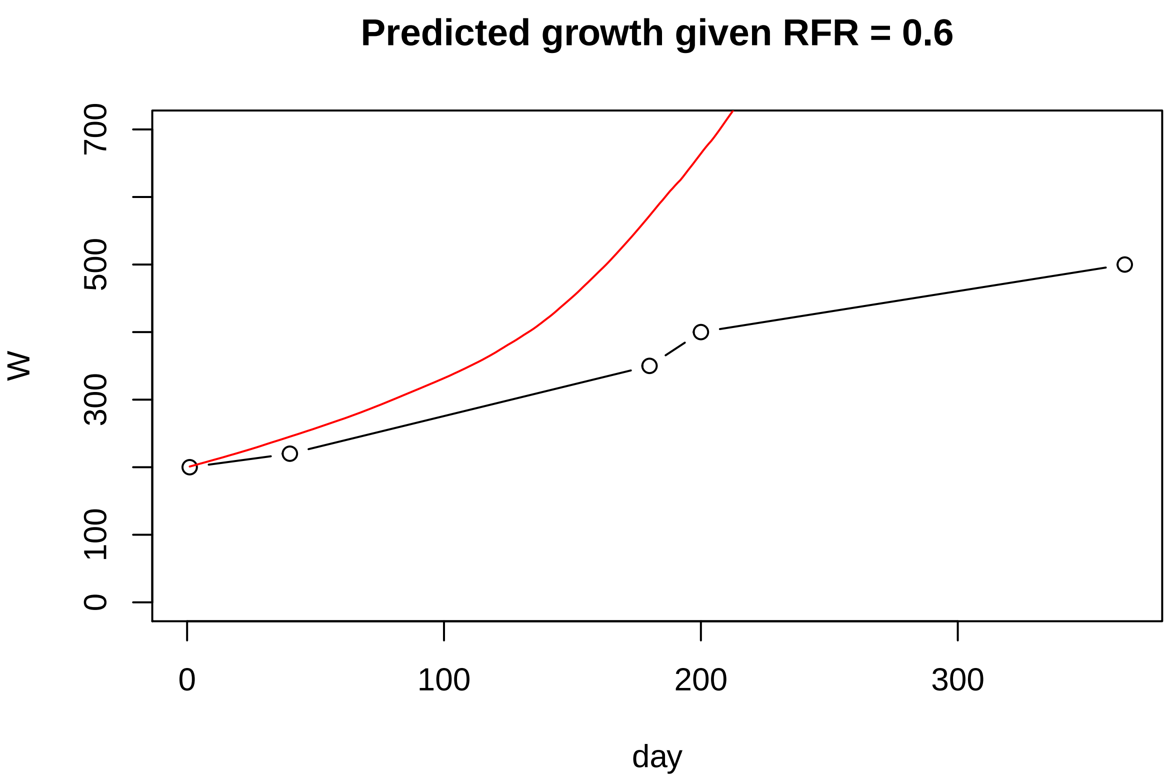



```
nd<-dim(Tdat)[1]  # number of days
Wstart<-200   # weight at the start of the simulation
Wobs<-data.frame(day=c(1,40,180,200,365),W=c(200,220,350,400,500)) # made up dates and weights

par_sim<-hal_par
```

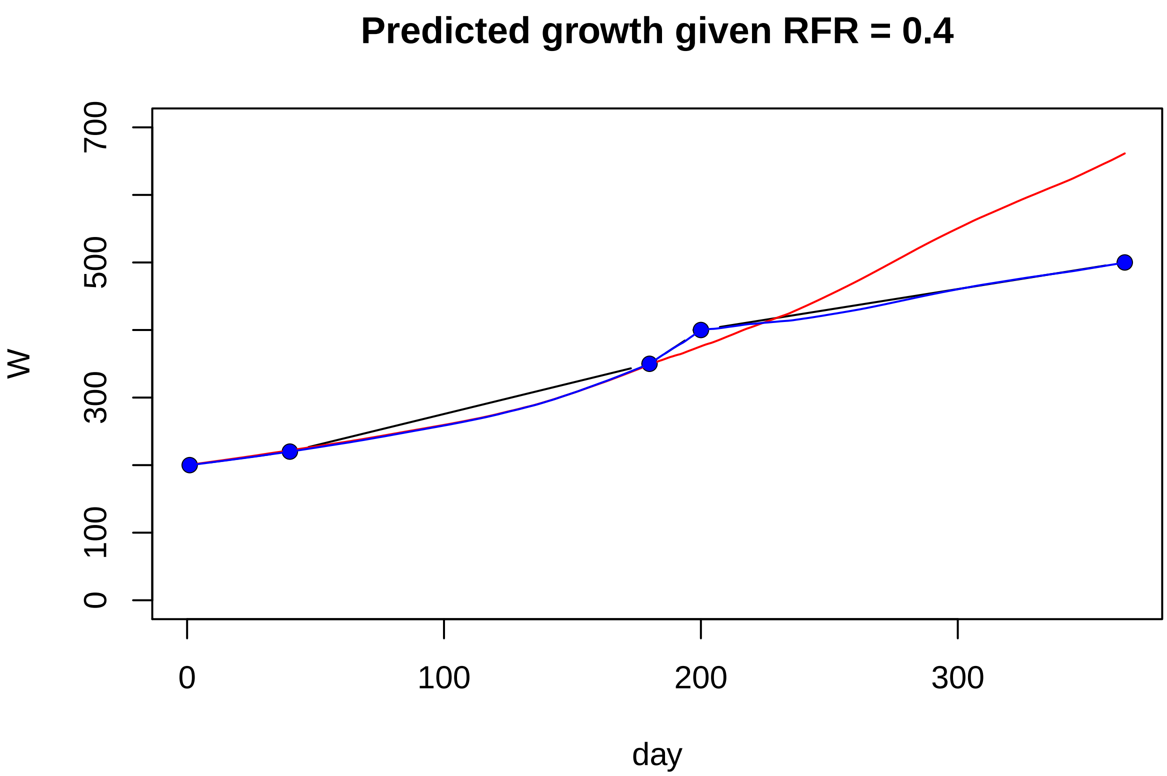
```
sim_dat<-ebs_data
sim_dat$Eprey<-4598.07
sim_dat$Epred<-4800
sim_dat$TempC<-Tdat$TempC
W<-Tdat$TempC*0

sim_W<-function(par=0.4,data,LL=TRUE){
  tmp_par<-data$tmp_par
  tmp_par$RFR<-par[1]
  Wtarget<-data$Wtarget
  tmp_Tdat<-data$tmp_Tdat
  tmp_dat<-data$tmp_dat

  nd<-dim(tmp_Tdat)[1]
  W<-rep(0,nd)
  tt<-bioE(par=tmp_par,data=tmp_dat)
  sim<-data.frame(matrix(0,nd,dim(tt[[2]])[2]))
  colnames(sim)<-names(tt[[2]])
  sim[1,]<-tt[[2]]

  for(d in 1:nd){
    # assign weight at the start of day d to that from the previous day
    tmp_dat$TempC<-tmp_Tdat$TempC[d]
    #if(d==1) tmp_dat$W<-data$Wstart
    if(d>1) tmp_dat$W<-sim$W[d-1]
    tt<-bioE(par=tmp_par,data=tmp_dat)
    sim[d,]<-tt[[2]]
    sim$W[d]<-sim$W[d]+sim$G_ggd[d]*sim$W[d]  # growth in g per d
  }
  What<-sim$W[nd]
  if(LL){
    return( ( (Wtarget)-(What) )^2 )
  }else{
    return(sim)
  }
}

sim_dat$W<-Wobs[1,2] # set W data for the simulation to the first observed weight
subdat<-list(Wtarget=Wobs,tmp_par=par_sim,tmp_Tdat=Tdat,tmp_dat=sim_dat) # create simulation data f

W<-sim_W(par=0.4,data=subdat,LL=F)$W # set Pvalue for whole simulation period, see effect by changi
par(mfrow=c(1,1))
plot(Wobs,type="b",ylim=c(0,700),main=paste0("Predicted growth given RFR = ",0.4))
lines(W,col="red")
```



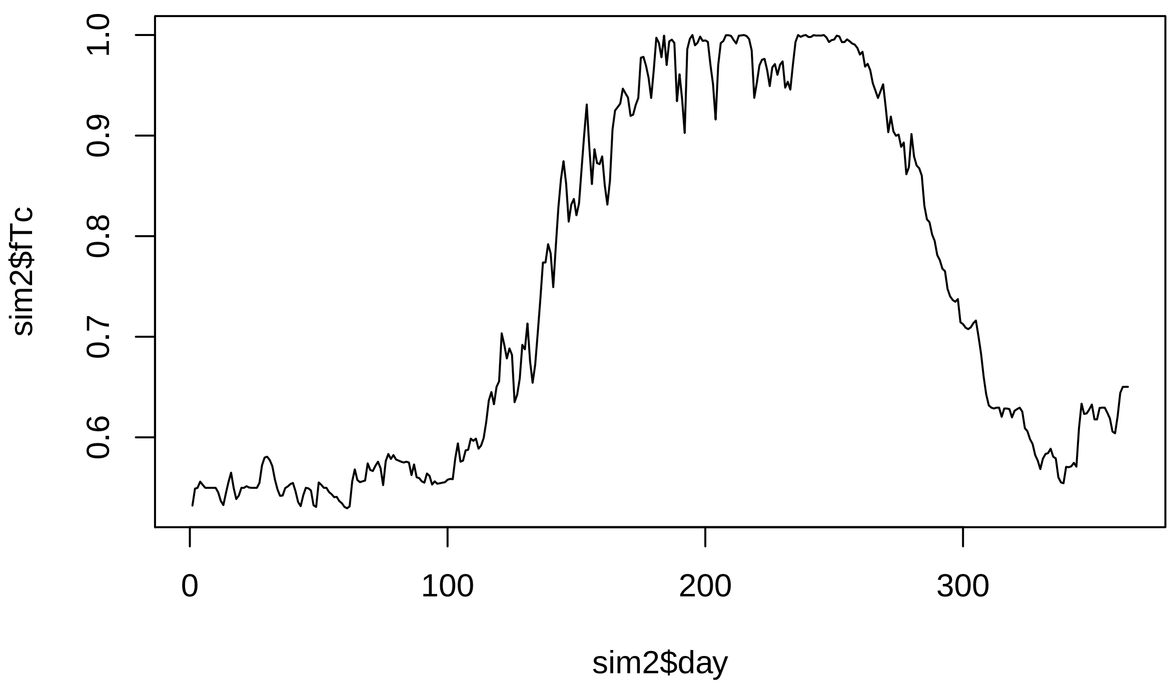

```
W<-sim_W(par=0.6,data=subdat,LL=F)$W # set Pvalue for whole simulation period, see effect by changi
par(mfrow=c(1,1))
plot(Wobs,type="b",ylim=c(0,700),main=paste0("Predicted growth given RFR = ",0.6))
lines(1:365,W,col="red")
```



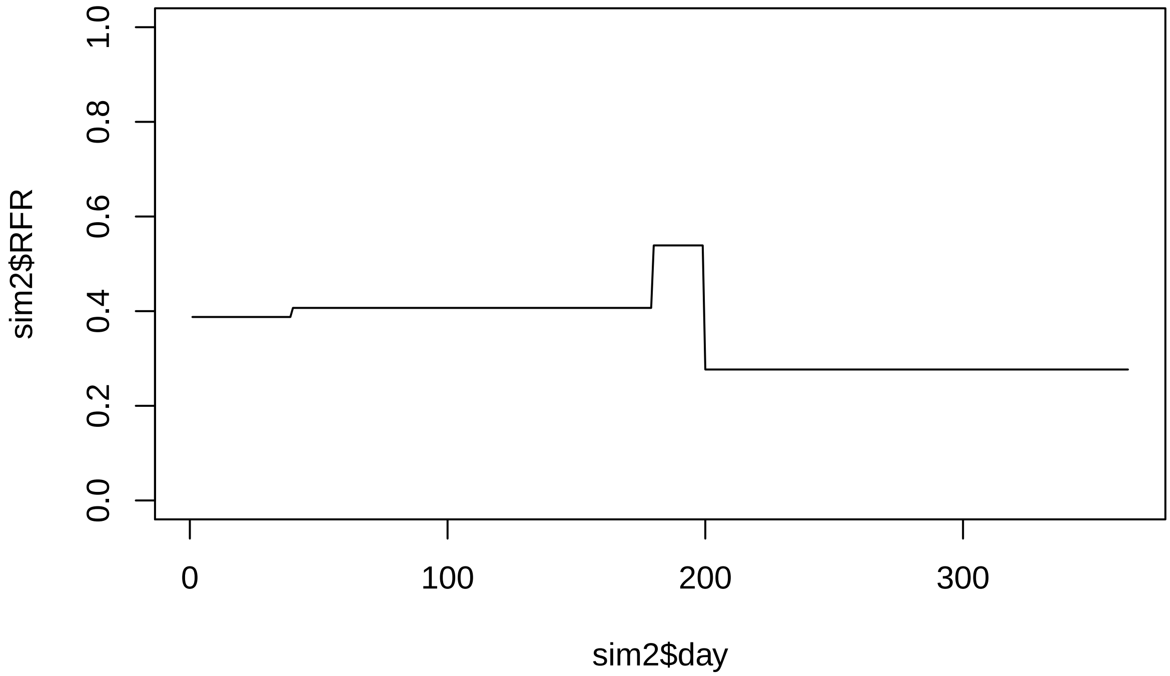
```r
# now let's fit growth to observed growth overtime by adjusting RFR

  outdat<-data.frame(day=Wobs[,1],RFR=Wobs[,1]*0,Wobs=Wobs[,2],What=0)
  sim1<-Tdat
  sim1$What<-0
  sim1$RFR<-0
  sim1$What[1]<-Wobs[1,2]
  nobs<-dim(Wobs)[1]
  outdat$What[1]<-Wobs[1,2]
  sim_dat$W<-Wobs[1,2]

  for(i in 2:nobs){
    subTdat<-Tdat[Wobs$day[i-1]:(Wobs$day[i]-1),]
    sim_dat$W<-outdat$What[i-1]
    subdat<-list(Wtarget=Wobs[i,2],tmp_par=par_sim,tmp_Tdat=subTdat,tmp_dat=sim_dat)
    sim_W(par=.3,data=subdat,LL=TRUE)
    m<- optimize(f=sim_W,lower=0,upper=10,data=subdat,LL=TRUE)
    outdat$RFR[i]<-(m)[1]
    if(i==2)  outdat$RFR[i-1]<-(m)[1]
    sim1[(Wobs$day[i-1]+1):(Wobs$day[i]),]$RFR<-(m)[1]
    if(i==2) tt<-data.frame(day=Wobs$day[i-1]:(Wobs$day[i]-1),sim_W(par=as.numeric(m[1]),data=subdat,
    if(i>2) tt<-rbind(tt,data.frame(day=Wobs$day[i-1]:(Wobs$day[i]-1),sim_W(par=as.numeric(m[1]),data=
    sim1[(Wobs$day[i-1]+1):(Wobs$day[i]),]$What<-tt$W[Wobs$day[i-1]:(Wobs$day[i]-1)]
    outdat$What[i]<-rev(tt$W[Wobs$day[i-1]:(Wobs$day[i]-1)])[1]
  }
  sim2<-tt

  sim_dat$W<-Wobs[1,2] # set W data for the simulation to the first observed weight
  subdat<-list(Wtarget=Wobs,tmp_par=par_sim,tmp_Tdat=Tdat,tmp_dat=sim_dat) # create simulation data f
  plot(Wobs,type="b",ylim=c(0,700),main=paste0("Predicted growth given RFR = ",.4))
  lines(1:365,sim_W(par=0.4,data=subdat,LL=F)$W,col="red")
  lines(1:365,sim1$What,col="blue",type="l")
  points(outdat$day,outdat$What,pch=16,col="blue")
```



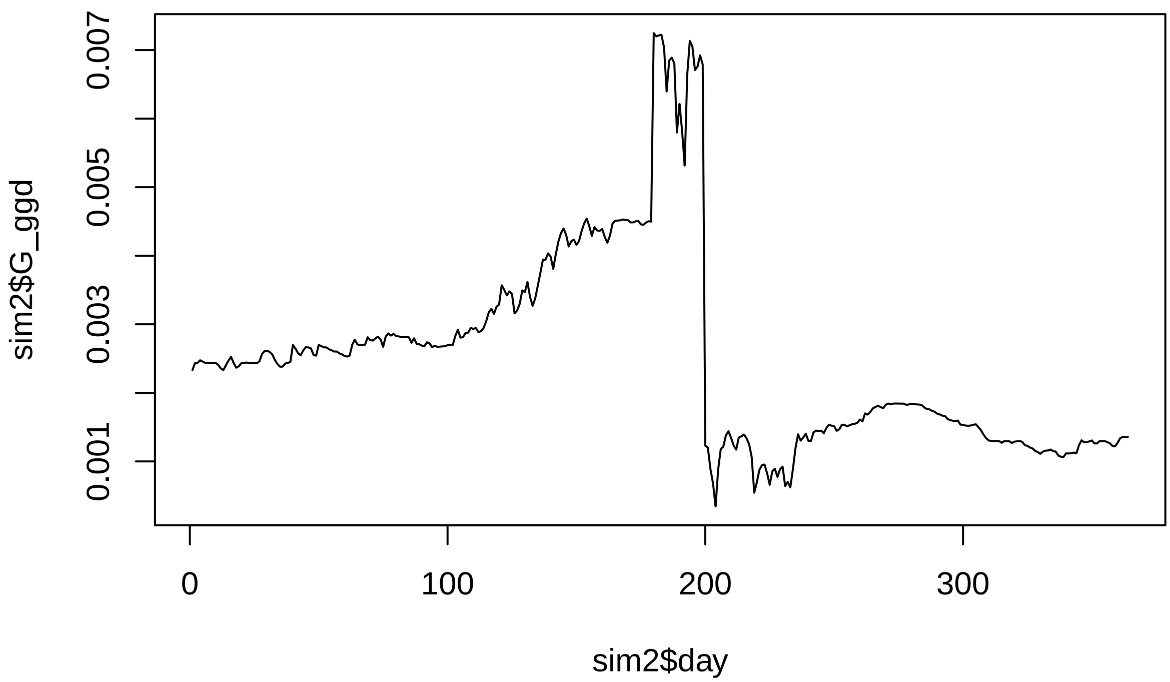

```r
plot(sim2$day,sim2$fTc,type="l")
```




```r
plot(sim2$day,sim2$RFR,type="l",ylim=c(0,1))
```




```r
plot(sim2$day,sim2$G_ggd,type="l")
```



Part 5: Fit parameters

```r
# fit pars to lab data:
    find_ftcpar<-function(par=c(logTco=log(8),logQc=log(2),
                                logsigma=log(.0002)),data=list(c_data=c_data,FTdat=FTdat)){
      FTdat<-data$FTdat
```

```r
    c_data<-data$c_data
    PARMS_USE$Tco<-exp(par[1])
    PARMS_USE$QC<-exp(par[2])
    sigma<-exp(par[3])
    c_data$TempC<-FTdat$TempC
    fThat<-fTC_fun(par=PARMS_USE,data=c_data)
    LL<-dnorm(fThat-FTdat$fTobs,sigma,log=TRUE)
    LLuse<--sum(LL)
    if(is.na(LLuse)){LLuse<-1e6}
    #if(Tcm>35){LLuse<-1e6}
    return(LLuse)
  }
  m<-optim(fn=find_ftcpar,par=c(logTco=log(8),logQc=log(2),
          logsigma=log(.0002)),data=list(c_data=c_data,FTdat=FTdat),
          hessian=TRUE,control=list(maxit=1e6))
  vc <- solve(m$hessian)
  se<-(sqrt(diag(vc)))
  # not working
   #abline(v=exp(m$par[1])+1.95*exp(se[1])); abline(v=exp(m$par[1])-1.95*exp(se[1]))
  # p1<-p2<-PARMS_USE
  # p1$Tco<-as.numeric(exp(m$par[1]+1*se[1]))
  # p1$QC<-as.numeric(exp(m$par[2]+1*se[2]))
  # p2$Tco<-as.numeric(exp(m$par[1]-1*se[1]))
  # p2$QC<-as.numeric(exp(m$par[2]-1*se[2]))
  # lines(ebs_data$TempC,fTC_fun(par=p1,data=c_data))
  # lines(ebs_data$TempC,fTC_fun(par=p2,data=c_data))
  #

  # compare values
  round(exp(m$par),3)
```

```
##   logTco    logQc logsigma
##   12.970    3.084    0.000
```

```r
  hal_par$Tco
```

```
## [1] 12.9699
```

```r
  hal_par$QC
```

```
## [1] 3.084
```