# Projecting biomass dynamics

## Libraries needed for analysis

library(deSolve)

## Setting up the models

### Graham Schaefer

if(m==3){dB<- r*B*(log(Bmax/B)) - C}

### Pella Tomlinson

if(m==4){dB<- (r/p)*B*(1-(B/Bmax)^p) - C}

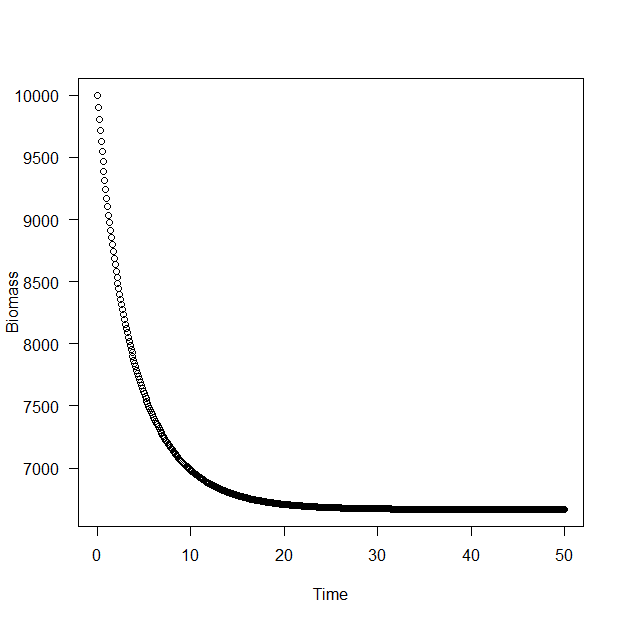
### Fox

if(m==5){dB<- r*B*((Bmax-B)/Bmax) - C}

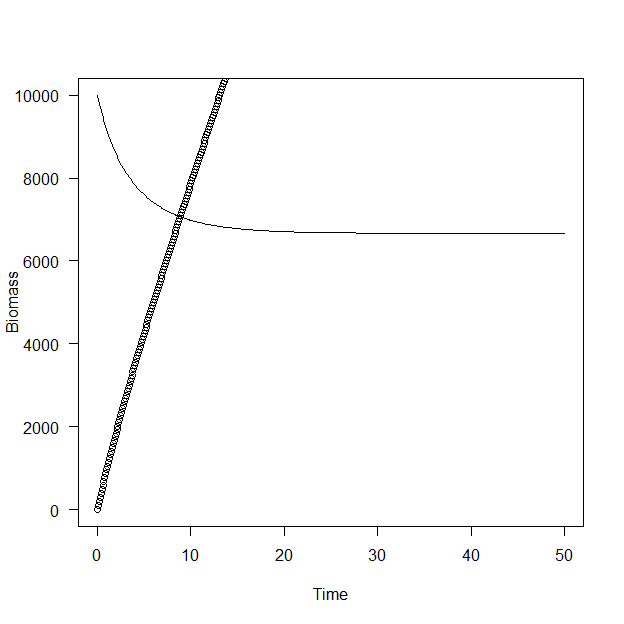
## Differential versus difference model

### Numerically integrating continuous biomass dynamics

ode\_gs<-function(t,x,parms)  
 {  
 # set the state variable   
 B<-x   
   
 # set the parameters  
 r<-parms[1]  
 K<-parms[2]  
 F<-parms[3]  
   
 # the Graham-Schaefer model of biomass dynamics as a   
 # ordinary differential equation  
 dB<- r\*B\*(1-(B/K))-F\*B  
 return(list(dB))  
 }  
parameters<-c(r=0.3,K=10000,F=0.1)  
initial\_biomass<- c(B=10000)  
solution<- ode(  
 y=initial\_biomass,   
 times=seq(0,50,by=0.1),   
 func=ode\_gs,   
 parms=parameters,   
 method="rk4")  
solution<-as.data.frame(solution)  
plot(B~time,solution,ylab="Biomass",xlab="Time",las=1,main="")

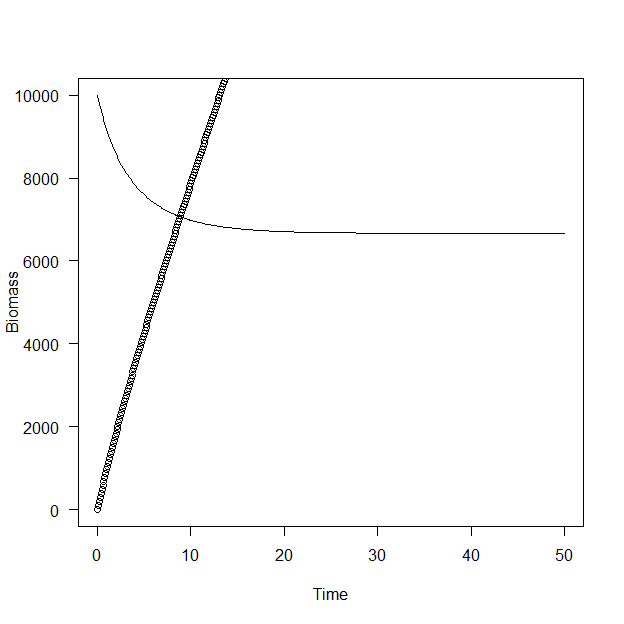


ode\_gs<-function(t,x,parms)  
 {  
 # set the state variable   
 B<-x [1]  
 Y<-x[2]  
 # set the parameters  
 r<-parms[1]  
 K<-parms[2]  
 F<-parms[3]  
   
 # the Graham-Schaefer model of biomass dynamics as a   
 # ordinary differential equation  
 dB<- r\*B\*(1-(B/K))-F\*B  
 # keep track of fishery yield, accumulates over time  
 dY<- F\*B  
 return(list(c(dB,dY)))  
 }  
parameters<-c(r=0.3,K=10000,F=0.1)  
# set initial biomass and yield  
initial\_biomass<- c(B=10000,Y=0)  
solution<- ode(  
 y=initial\_biomass,   
 times=seq(0,50,by=0.1),   
 func=ode\_gs,   
 parms=parameters,   
 method="rk4")  
solution<-as.data.frame(solution)  
plot(B~time,solution,ylab="Biomass",xlab="Time",las=1,main="",type='l',  
 ylim=c(0,10000))  
points(Y~time,solution,ylab="Biomass",xlab="Time",las=1,main="")



Keeping track of annual yield

ode\_gs<-function(t,x,parms)  
 {  
 # set the state variable   
 B<-x [1]  
 Y<-x[2]  
 # set the parameters  
 r<-parms[1]  
 K<-parms[2]  
 F<-parms[3]  
   
 # the Graham-Schaefer model of biomass dynamics as a   
 # ordinary differential equation  
 dB<- r\*B\*(1-(B/K))-F\*B  
 # keep track of fishery yield, accumulates over time  
 dY<- F\*B  
 return(list(c(dB,dY),Y=dY))  
 }  
parameters<-c(r=0.3,K=10000,F=0.1)  
# set initial biomass and yield  
initial\_biomass<- c(B=10000,Y=0)  
solution<- ode(  
 y=initial\_biomass,   
 times=seq(0,50,by=0.1),   
 func=ode\_gs,   
 parms=parameters,   
 method="rk4")  
solution\_cont<-as.data.frame(solution)  
plot(B~time,solution\_cont,ylab="Biomass",xlab="Time",las=1,main="",type='l',  
 ylim=c(0,10000))  
points(Y~time,solution\_cont,ylab="Biomass",xlab="Time",las=1,main="")



## Iterative evaluation of biomass dynamics

Difference model

diff\_gs<-function(t,x,parms)  
 {  
 # set the state variable   
 B<-x   
   
 # set the parameters  
 r<-parms[1]  
 K<-parms[2]  
 F<-parms[3]  
   
 # the Graham-Schaefer model of biomass dynamics as a   
 # difference equation  
 B<-B+r\*B\*(1-(B/K))-F\*B  
 return(list(B))  
 }  
parameters<-c(r=0.3,K=10000,F=0.1)  
parameters[1]<-exp(parameters[1])-1  
parameters[3]<-exp(parameters[3])-1  
  
  
initial\_biomass<- c(B=10000)  
solution<- ode(  
 y=initial\_biomass,   
 times=seq(0,50,by=0.1),   
 func=diff\_gs,   
 parms=parameters,   
 method="iteration")  
solution\_diff<-as.data.frame(solution)  
plot(B~time,solution\_diff,ylab="Biomass",xlab="Time",las=1,main="")  
points(B~time,solution\_cont,col="red")

