Main text figures: The benefits and consequences of predator-mediated mutualisms

5 May 2021

Figure 2: Timeseries of changing predator abundances.

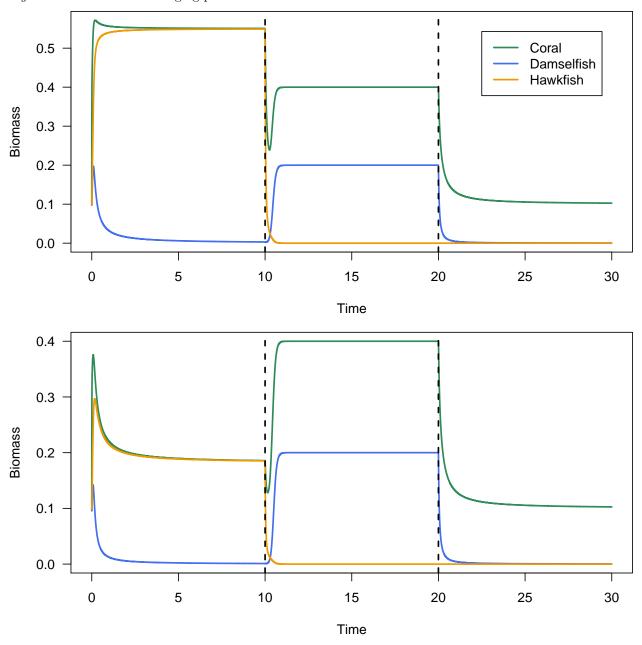


Figure 3: Heatmaps of coral biomass as a function of predators and parameters. Simulations varying alpha and predator abundance:

```
### Chunk is time-consuming to run. Un-comment to run; otherwise this code relies on saved data from a
# phi <- 2
# epsilon <- 0.5
\# pmax \leftarrow rho_D*(1-epsilon*k_H/k_D)
\# alphaset \leftarrow seq(from = 0, to = 1, length.out = 105)
# Pset <- seq(from = 0, to = pmax, length.out = 100)
\# cstar \leftarrow matrix(rep(NaN, length(alphaset)*length(Pset)), nrow = length(alphaset), ncol = length(Pset))
# dstar <- cstar
# hstar <- cstar
# t_end <- 365*2*10;
# tset \leftarrow seq(from=0, to = t_end, length.out = 10*t_end)
# for(k in 1:length(alphaset)){
# alpha <- alphaset[k]</pre>
# for(j in 1:length(Pset)){
# p <- Pset[j]
# d.simu1 <- NaN*tset; d.simu1[1] <- 0.5;
# h.simu1 <- d.simu1;
# c.simu1 <- d.simu1;
# for(i in 2:length(tset)){
   dt \leftarrow tset[i]-tset[i-1]
   d \leftarrow d.simu1[i-1]
#
   h \leftarrow h.simu1[i-1]
#
    c \leftarrow c.simu1[i-1]
   dc \leftarrow ((c+d+epsilon*h)*(1-c)-mu*c)*dt
#
   dd \leftarrow (rho_D*d*(k_D*c-d-h)/(k_D*c)-d*p)*dt
#
   dh \leftarrow (rho_H*h*(k_H*c-h-alpha*d)/(k_H*c)-phi*h*p)*dt
#
    c.simu1[i] \leftarrow c+dc
#
    d.simu1[i] \leftarrow d+dd
#
    h.simu1[i] \leftarrow h+dh
# }
    dstar[k,j] <- d.simu1[length(tset)];</pre>
#
#
   hstar[k,j] \leftarrow h.simu1[length(tset)];
   cstar[k,j] <- c.simu1[length(tset)];</pre>
# }}
#
#
\# dstar_a \leftarrow dstar
# hstar_a <- hstar
# cstar_a <- cstar
```

Simulations varying phi and predator abundance:

```
### Chunk is time-consuming to run. Un-comment to run; otherwise this code relies on saved data from a property alpha <- 0.5
# epsilon <- 0.5
#
# pmax <- rho_D*(1-epsilon*k_H/k_D)</pre>
```

```
# phiset <- seq(from = 1, to = 10, length.out = 105)
# Pset <- seq(from = 0, to = pmax, length.out = 100)
# cstar <- matrix(rep(NaN, length(phiset)*length(Pset)), nrow = length(phiset), ncol = length(Pset))
# dstar <- cstar
# hstar <- cstar
# t_end <- 365*2*10;
# tset \leftarrow seq(from=0, to = t_end, length.out = 10*t_end)
# for(k in 1:length(phiset)){
# phi <- phiset[k]</pre>
# for(j in 1:length(Pset)){
# p <- Pset[j]
# d.simu1 <- NaN*tset; d.simu1[1] <- 0.5;
# h.simu1 <- d.simu1;
# c.simu1 <- d.simu1;
# for(i in 2:length(tset)){
  dt \leftarrow tset[i]-tset[i-1]
  d \leftarrow d.simu1[i-1]
#
   h \leftarrow h.simu1[i-1]
#
   c \leftarrow c.simu1[i-1]
#
  dc \leftarrow ((c+d+epsilon*h)*(1-c)-mu*c)*dt
   dd \leftarrow (rho_D*d*(k_D*c-d-h)/(k_D*c)-d*p)*dt
#
#
   dh \leftarrow (rho_H*h*(k_H*c-h-alpha*d)/(k_H*c)-phi*h*p)*dt
#
   c.simu1[i] \leftarrow c+dc
#
   d.simu1[i] \leftarrow d+dd
#
    h.simu1[i] \leftarrow h+dh
# }
#
   dstar[k,j] <- d.simu1[length(tset)];</pre>
#
   hstar[k,j] <- h.simu1[length(tset)];</pre>
#
    cstar[k,j] <- c.simu1[length(tset)];</pre>
# }}
#
#
# dstar_p <- dstar
# hstar_p <- hstar</pre>
# cstar_p <- cstar</pre>
```

Simulations varying epsilon and predator abundance:

Chunk is time-consuming to run. Un-comment to run; otherwise this code relies on saved data from a
alpha <- 0.5
phi <- 2
#
pmax <- rho_D*(1-0*k_H/k_D)</pre>

```
# epsilonset <- seq(from = 0, to = 1, length.out = 105)
# Pset <- seq(from = 0, to = pmax, length.out = 100)
#</pre>
```

 $\#\ cstar <-\ matrix(rep(NaN, length(epsilonset)*length(Pset)), nrow = length(epsilonset), ncol = length(Pset)$

```
# dstar <- cstar
# hstar <- cstar
# t_end <- 365*2*10;
\# tset <- seq(from=0, to = t_end, length.out = 20*t_end)
# for(k in 1:length(epsilonset)){
# epsilon <- epsilonset[k]</pre>
# for(j in 1:length(Pset)){
# p <- Pset[j]
# d.simu1 <- NaN*tset; d.simu1[1] <- 0.5;
# h.simu1 <- d.simu1;
# c.simu1 <- d.simu1;
# for(i in 2:length(tset)){
# dt <- tset[i]-tset[i-1]
# d \leftarrow d.simu1[i-1]
#
       h \leftarrow h.simu1[i-1]
       c \leftarrow c.simu1[i-1]
# dc \leftarrow ((c+d+epsilon*h)*(1-c)-mu*c)*dt
# dd \leftarrow (rho_D*d*(k_D*c-d-h)/(k_D*c)-d*p)*dt
 \# \quad dh <- (rho\_H*h*(k\_H*c-h-alpha*d)/(k\_H*c)-phi*h*p)*dt 
       c.simu1[i] \leftarrow c+dc
\# d.simu1[i] \leftarrow d+dd
#
       h.simu1[i] \leftarrow h+dh
# }
# dstar[k,j] <- d.simu1[length(tset)];</pre>
#
       hstar[k, j] <- h.simu1[length(tset)];</pre>
       cstar[k,j] <- c.simu1[length(tset)];</pre>
# }}
#
# dstar_e <- dstar
# hstar_e <- hstar</pre>
# cstar_e <- cstar
# setwd('/Users/hollyum/Box Sync/StierMoeller_PredMedMutualisms/Code/RCodes')
\#\ save(dstar\_a, hstar\_a, cstar\_a, dstar\_p, hstar\_p, cstar\_p, dstar\_e, hstar\_e, cstar\_e, Pset, epsilonset, phiset, and the property of the p
Load results
setwd('/Users/hollyvm/Box Sync/StierMoeller_PredMedMutualisms/Code/RCodes')
#load('Heatmap1.Rdata') #low resolution: 20 x 25
load('Heatmap3.Rdata') #high resolution: 100 x 105
kot <- colorRampPalette(colors = c(kotare[3], 'white', kotare[2]))</pre>
rwb <- colorRampPalette(colors = c("red", "white", "blue"))</pre>
rwb <- kot
```

zlimraw <- c(0.1,.55)zlimrel <- c(.18,4.5)

par(mar=c(4,4,1,1),mfrow=c(3,2))

```
\#panelwidths = 2; panelheights = 2; layout(matrix(c(rep(7,1+2*panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(1,panelwidths),re
image.plot(x = Pset, y = alphaset, z = t(cstar_a),las=1,col=rwb(100),ylab='Alpha',xlab='Pred. Abun.',zl
relcstar_a <- cstar_a*NaN
for(i in 1:length(alphaset)){
     relcstar_a[i,1:length(Pset)] <- cstar_a[i,1:length(Pset)]/cstar_a[i,1]</pre>
image.plot(x = Pset, y = alphaset, z = t(relcstar_a),las=1,col=rwb(100),ylab='',xlab='Pred. Abun.',zlim
image.plot(x = Pset, y = phiset, z = t(cstar_p),las=1,col=rwb(100),ylab='Phi',xlab='Pred. Abun.',zlim=z
relcstar_p <- cstar_p*NaN</pre>
for(i in 1:length(phiset)){
     relcstar_p[i,1:length(Pset)] <- cstar_p[i,1:length(Pset)]/cstar_p[i,1]</pre>
image.plot(x = Pset, y = phiset, z = t(relcstar_p), las=1, col=rwb(100), ylab='', xlab='Pred. Abun.', zlim=z
image.plot(x = Pset, y = epsilonset, z = t(cstar_e),las=1,col=rwb(100),ylab='Epsilon',xlab='Pred. Abun.
relcstar_e <- cstar_e*NaN</pre>
for(i in 1:length(epsilonset)){
     relcstar_e[i,1:length(Pset)] <- cstar_e[i,1:length(Pset)]/cstar_e[i,1]</pre>
image.plot(x = Pset, y = epsilonset, z = t(relcstar_e),las=1,col=rwb(100),ylab='',xlab='Pred. Abun.',zl
```

```
0.8
                                                                                      0.8
                                                                   0.4
                                                                                                                                                 3
Alpha
        0.6
                                                                                      0.6
                                                                   0.3
                                                                                                                                                 2
        0.4
                                                                                      0.4
                                                                   0.2
                                                                                      0.2
         0.2
                                                                   0.1
         0.0
                                                                                      0.0 -
                0.00
                             0.04
                                            0.08
                                                                                             0.00
                                                                                                           0.04
                                                                                                                          0.08
                           Pred. Abun.
                                                                                                        Pred. Abun.
          10
                                                                                        10
                                                                   0.5
            8
                                                                                          8
                                                                   0.4
Phi
            6
                                                                                          6
                                                                   0.3
            4
                                                                                          4
                                                                   0.2
            2
                                                                                          2
                0.00
                             0.04
                                            0.08
                                                                                             0.00
                                                                                                           0.04
                                                                                                                         0.08
                          Pred. Abun.
                                                                                                        Pred. Abun.
         1.0
                                                                                       1.0
                                                                   0.5
         8.0
                                                                                      8.0
                                                                   0.4
        0.6
                                                                                      0.6
                                                                   0.3
        0.4
                                                                                      0.4
                                                                   0.2
         0.2
                                                                                      0.2
                                                                   0.1
                                                                                      0.0
         0.0
                0.00
                             0.04
                                            0.08
                                                                                             0.00
                                                                                                           0.04
                                                                                                                         0.08
                          Pred. Abun.
                                                                                                        Pred. Abun.
kot <- colorRampPalette(colors = c(kotare[3],'white',kotare[2]))</pre>
rwb <- colorRampPalette(colors = c("red", "white", "blue"))</pre>
rwb <- kot
par(mar=c(4,4,1,1),mfrow=c(3,2))
\#panelwidths = 2; panelheights = 2; layout(matrix(c(rep(7,1+2*panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,re
image.plot(x = Pset, y = alphaset, z = t(cstar_a),las=1,col=rwb(100),ylab='Alpha',xlab='Pred. Abun.',ma
relcstar_a <- cstar_a*NaN
for(i in 1:length(alphaset)){
      relcstar_a[i,1:(dim(relcstar_a)[2])] <- cstar_a[i,1:(dim(relcstar_a)[2])]/cstar_a[i,1]</pre>
image.plot(x = Pset, y = alphaset, z = t(relcstar_a),las=1,col=rwb(100),ylab='',xlab='Pred. Abun.',main
image.plot(x = Pset, y = phiset, z = t(cstar_p),las=1,col=rwb(100),ylab='Phi',xlab='Pred. Abun.');
relcstar_p <- cstar_p*NaN</pre>
for(i in 1:length(phiset)){
      \#relcstar\_p[i,1:length(Pset)] \leftarrow cstar\_p[i,1:length(Pset)]/cstar\_p[i,1]
     relcstar_p[i,1:(dim(relcstar_p)[2])] <- cstar_p[i,1:(dim(relcstar_p)[2])]/cstar_p[i,1]</pre>
```

Equilibrium Coral Biomass mass Relative to Predator-Fre

```
image.plot(x = Pset, y = phiset, z = t(relcstar_p),las=1,col=rwb(100),ylab='',xlab='Pred. Abun.');
image.plot(x = Pset, y = epsilonset, z = t(cstar_e),las=1,col=rwb(100),ylab='Epsilon',xlab='Pred. Abun.
relcstar_e <- cstar_e*NaN</pre>
for(i in 1:length(epsilonset)){
      \#relcstar_e[i,1:length(Pset)] \leftarrow cstar_e[i,1:length(Pset)]/cstar_e[i,1]
     relcstar_e[i,1:(dim(relcstar_e)[2])] <- cstar_e[i,1:(dim(relcstar_e)[2])]/cstar_e[i,1]</pre>
image.plot(x = Pset, y = epsilonset, z = t(relcstar_e),las=1,col=rwb(100),ylab='',xlab='Pred. Abun.')
   Equilibrium Coral Biomass mass Relative to Predator-Fre
         8.0
                                                                                        8.0
                                                                    0.52
                                                                                                                                                    1.2
                                                                    0.50
Alpha
         0.6
                                                                                        0.6
                                                                                                                                                   1.1
                                                                    0.48
        0.4
                                                                    0.46
                                                                                        0.4
                                                                                                                                                   1.0
                                                                    0.44
         0.2
                                                                                        0.2
                                                                                                                                                   0.9
                                                                    0.42
                                                                    0.40
         0.0
                                                                                        0.0
                0.00
                              0.04
                                            0.08
                                                                                               0.00
                                                                                                             0.04
                                                                                                                           0.08
                           Pred. Abun.
                                                                                                          Pred. Abun.
          10
                                                                                          10
                                                                                                                                                    1.3
                                                                     0.50
            8
                                                                                           8
                                                                                                                                                   1.2
                                                                     0.45
                                                                                                                                                    1.1
            6
                                                                                           6
٦.
                                                                    0.40
                                                                                                                                                    1.0
             4
                                                                                            4
                                                                                                                                                   0.9
                                                                     0.35
                                                                                                                                                   8.0
                                                                    0.30
                                                                                            2
            2
                0.00
                             0.04
                                         0.08
                                                                                               0.00
                                                                                                             0.04
                                                                                                                           0.08
                                                                                                          Pred. Abun.
                           Pred. Abun.
         1.0
                                                                                        1.0
                                                                     0.5
         8.0
                                                                                        8.0
                                                                    0.4
         0.6
                                                                                        0.6
                                                                    0.3
                                                                                        0.4
         0.4
                                                                    0.2
                                                                                        0.2
         0.2
                                                                     0.1
                                                                                        0.0
         0.0
                0.00
                            0.04
                                         0.08
                                                                                               0.00
                                                                                                             0.04
                                                                                                                           0.08
                           Pred. Abun.
                                                                                                          Pred. Abun.
kot <- colorRampPalette(colors = c(kotare[3],'white',kotare[2]))</pre>
rwb <- colorRampPalette(colors = c("red", "white", "blue"))</pre>
#rwb <- kot
par(mar=c(4,4,1,5),mfrow=c(1,3))
\#panelwidths = 2; panelheights = 2; layout(matrix(c(rep(7,1+2*panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(1,panelwidths),rep(c(8,rep(1,panelwidths),rep(1,panelwidths),rep(1,panelwidths)
```

```
\#image.plot(x = Pset, y = alphaset, z = t(cstar_a), las=1, col=rwb(100), ylab='Alpha', xlab='Pred. Abun.', more substitution and the substitution are substituted as a substitution of the substitution and the substitution are substitutionally as a substitution of the substitution and the substitution are substitutionally as a substitution of the substitution and the substitution are substitutionally as a substitution of the substitution and the substitution are substitutionally as a substitution are substitutionally as a substitution and the substitution are substitutionally as a substitution are substitutionally as a substitution and the substitution are substitutionally as a substitution are substitutionally as a substitution and the substitution are substitutionally as a substitution are substitutionally as a substitution and the substitution are substitutionally as a 
relcstar_a <- cstar_a*NaN
for(i in 1:length(alphaset)){
         relcstar_a[i,1:length(Pset)] <- cstar_a[i,1:length(Pset)]/cstar_a[i,1]</pre>
image.plot(x = Pset, y = alphaset, z = t(relcstar_a),las=1,col=rwb(100),ylab='Alpha',xlab='Pred. Abun.'
\#image.plot(x = Pset, y = phiset, z = t(cstar_p), las=1, col=rwb(100), ylab='Phi', xlab='Pred. Abun.');
relcstar_p <- cstar_p*NaN
for(i in 1:length(phiset)){
         relcstar_p[i,1:length(Pset)] <- cstar_p[i,1:length(Pset)]/cstar_p[i,1]</pre>
image.plot(x = Pset, y = phiset, z = t(relcstar_p),las=1,col=rwb(100),ylab='Phi',xlab='Pred. Abun.');
\#image.plot(x = Pset, y = epsilonset, z = t(cstar_e), las=1, col=rwb(100), ylab='Epsilon', xlab='Pred. Abun', xlab='Pred. Abu
relcstar_e <- cstar_e*NaN
for(i in 1:length(epsilonset)){
         relcstar_e[i,1:length(Pset)] <- cstar_e[i,1:length(Pset)]/cstar_e[i,1]</pre>
}
image.plot(x = Pset, y = epsilonset, z = t(relcstar_e),las=1,col=rwb(100),ylab='Epsilon',xlab='Pred. Ab
         1.0
        0.6
                                                                                                                                                                                                                                                            1.0
                                                                                                                           1.0
                                                                                                                                                                                                                                                            0.9
        0.2
                                                                                                                                                                                                                                                            0.8
                                                                                                                                                                                                                                                            0.7
        0.0
                                                                                                                                                                                                                                                                                                                                                                        0.10
                0.00
                                0.02
                                                0.04
                                                                   0.06
                                                                                    0.08
                                                                                                     0.10
                                                                                                                                                 0.00
                                                                                                                                                                  0.02
                                                                                                                                                                                    0.04
                                                                                                                                                                                                    0.06
                                                                                                                                                                                                                      0.08
                                                                                                                                                                                                                                       0.10
                                                                                                                                                                                                                                                                                   0.00
                                                                                                                                                                                                                                                                                                    0.02
                                                                                                                                                                                                                                                                                                                      0.04
                                                                                                                                                                                                                                                                                                                                       0.06
                                                                                                                                                                                                                                                                                                                                                        0.08
kot <- colorRampPalette(colors = c(kotare[3], 'white', kotare[2]))</pre>
rwb <- colorRampPalette(colors = c("red", "white", "blue"))</pre>
rwb <- kot
par(mar=c(4,4,1,5),mfrow=c(1,3))
 \#panelwidths = 2; panelheights = 2; layout(matrix(c(rep(7,1+2*panelwidths),rep(c(8,rep(1,panelwidths),rep(c)))
 \#image.plot(x = Pset, y = alphaset, z = t(cstar_a), las=1, col=rwb(100), ylab='Alpha', xlab='Pred. Abun.', market = final color for the state of t
relcstar_a <- cstar_a*NaN
for(i in 1:length(alphaset)){
         relcstar_a[i,1:length(Pset)] <- cstar_a[i,1:length(Pset)]/cstar_a[i,1]</pre>
}
image.plot(x = Pset, y = alphaset, z = t(relcstar_a),las=1,col=rwb(100),ylab='Alpha',xlab='Pred. Abun.'
\#image.plot(x = Pset, y = phiset, z = t(cstar_p), las=1, col=rwb(100), ylab='Phi', xlab='Pred. Abun.');
relcstar_p <- cstar_p*NaN
for(i in 1:length(phiset)){
```

```
relcstar_p[i,1:length(Pset)] <- cstar_p[i,1:length(Pset)]/cstar_p[i,1]</pre>
}
image.plot(x = Pset, y = phiset, z = t(relcstar_p),las=1,col=rwb(100),ylab='Phi',xlab='Pred. Abun.');
\#image.plot(x = Pset, y = epsilonset, z = t(cstar_e), las=1, col=rwb(100), ylab='Epsilon', xlab='Pred. Abun', xlab='Pred. Abu
relcstar_e <- cstar_e*NaN
for(i in 1:length(epsilonset)){
           relcstar_e[i,1:length(Pset)] <- cstar_e[i,1:length(Pset)]/cstar_e[i,1]</pre>
image.plot(x = Pset, y = epsilonset, z = t(relcstar_e),las=1,col=rwb(100),ylab='Epsilon',xlab='Pred. Ab
           0.8
          0.6
           0.4
                                                                                                                                                                                                                                                                                                                                   0.9
           0.2
                                                                                                                                                                                                                                                                                                                                                         0.2
                   0.00
                                         0.02
                                                                0.04
                                                                                      0.06
                                                                                                            0.08
                                                                                                                                                                                          0.00
                                                                                                                                                                                                                 0.02
                                                                                                                                                                                                                                       0.04
                                                                                                                                                                                                                                                             0.06
                                                                                                                                                                                                                                                                                    0.08
                                                                                                                                                                                                                                                                                                         0.10
                                                                                                                                                                                                                                                                                                                                                                 0.00
                                                                                                                                                                                                                                                                                                                                                                                        0.02
                                                                                                                                                                                                                                                                                                                                                                                                               0.04
                                                                                                                                                                                                                                                                                                                                                                                                                                    0.06
                                                                                                                                                                                                                                                                                                                                                                                                                                                           0.08
                                                                                                                                                                                                                                                                                                                                                                                                                 Pred. Abun.
                                                                   Pred. Abun
                                                                                                                                                                                                                                          Pred. Abun.
kot <- colorRampPalette(colors = c(kotare[3],kotare[3],'white',kotare[2],kotare[2]))</pre>
rwb <- colorRampPalette(colors = c("red", "white", "blue"))</pre>
rwb <- kot
par(mar=c(4,4,1,5), mfrow=c(1,3))
 \#panelwidths = 2; panelheights = 2; layout(matrix(c(rep(7,1+2*panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,re
\#image.plot(x = Pset, y = alphaset, z = t(cstar_a), las=1, col=rwb(100), ylab='Alpha', xlab='Pred. Abun.', model = rwb(100), ylab='Alpha', ylab='Pred. Abun.', model = rwb(100), ylab='P
relcstar a <- cstar a*NaN
for(i in 1:length(alphaset)){
           relcstar_a[i,1:length(Pset)] <- cstar_a[i,1:length(Pset)]/cstar_a[i,1]</pre>
image.plot(x = Pset, y = alphaset, z = t(log10(relcstar_a)),las=1,col=rwb(100),ylab='Alpha',xlab='Pred.
 \#image.plot(x = Pset, y = phiset, z = t(cstar_p), las=1, col=rwb(100), ylab='Phi', xlab='Pred. Abun.');
relcstar_p <- cstar_p*NaN</pre>
for(i in 1:length(phiset)){
           relcstar_p[i,1:length(Pset)] <- cstar_p[i,1:length(Pset)]/cstar_p[i,1]</pre>
image.plot(x = Pset, y = phiset, z = t(log10(relcstar_p)),las=1,col=rwb(100),ylab='Phi',xlab='Pred. Abu
\#image.plot(x = Pset, y = epsilonset, z = t(cstar_e), las=1, col=rwb(100), ylab='Epsilon', xlab='Pred. Abun', xlab='Pred. Abu
relcstar e <- cstar e*NaN
for(i in 1:length(epsilonset)){
           relcstar_e[i,1:length(Pset)] <- cstar_e[i,1:length(Pset)]/cstar_e[i,1]</pre>
}
```

```
image.plot(x = Pset, y = epsilonset, z = t(log10(relcstar_e)),las=1,col=rwb(100),ylab='Epsilon',xlab='P
                                                                                                                                                                                                                                                                                                                      0.15
                                                                                                                                                      0.10
          0.8
                                                                                                                                                                                                                                                                                                                                          0.8
                                                                                                                                                                                                                                                                                                                      0.05
                                                                                                                                                                                                                                                                                                                     0.00
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      0.0
                                                                                                                                                   - 0.00 등
         0.4
                                                                                                                                                                                                                                                                                                                                          0.4
                                                                                                                                                                                                                                                                                                                       -0.05
                                                                                                                                                      -0.05
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       -0.4
                                                                                                                                                                                                                                                                                                                        -0.10
          0.2
                                                                                                                                                                                                                                                                                                                                         0.2
                                                                                                                                                      -0.10
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       -0.6
                                                                                                                                                                                                                                                                                                                       -0.15
         0.0
                                                                                                                                                                                                                                                                                                                                          0.0
                                                                                                                                                                                                                                                                                                                                                                       0.02
                 0.00
                                       0.02
                                                            0.04
                                                                                 0.06
                                                                                                       0.08
                                                                                                                            0.10
                                                                                                                                                                                 0.00
                                                                                                                                                                                                      0.02
                                                                                                                                                                                                                            0.04
                                                                                                                                                                                                                                                 0.06
                                                                                                                                                                                                                                                                        0.08
                                                                                                                                                                                                                                                                                            0.10
                                                                                                                                                                                                                                                                                                                                                  0.00
                                                                                                                                                                                                                                                                                                                                                                                             0.04
                                                                                                                                                                                                                                                                                                                                                                                                                 0.06
                                                                                                                                                                                                                                                                                                                                                                                                                                        0.08
                                                               Pred. Abun.
                                                                                                                                                                                                                                                                                                                                                                                                Pred. Abun.
kot <- colorRampPalette(colors = c(kotare[3],'white',kotare[2]))</pre>
rwb <- colorRampPalette(colors = c("red", "white", "blue"))</pre>
rwb <- kot
par(mar=c(4,4,1,5), mfrow=c(1,3))
\#panelwidths = 2; panelheights = 2; layout(matrix(c(rep(7,1+2*panelwidths),rep(c(8,rep(1,panelwidths),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,rep(1,panelwidths)),rep(c(8,re
rwb <- colorRampPalette(colors = c(kotare[3],kotare[3],'white',kotare[2],kotare[2]))
\#image.plot(x = Pset, y = alphaset, z = t(cstar_a), las=1, col=rwb(100), ylab='Alpha', xlab='Pred. Abun.', more substitution for the substitution of the substitutio
relcstar_a <- cstar_a*NaN
for(i in 1:length(alphaset)){
          relcstar_a[i,1:length(Pset)] <- cstar_a[i,1:length(Pset)]/cstar_a[i,1]</pre>
image.plot(x = Pset, y = alphaset, z = t(log10(relcstar_a)),las=1,col=rwb(100),ylab='Alpha',xlab='Pred.
rwb <- colorRampPalette(colors = c(kotare[3],kotare[3], 'white',kotare[2],kotare[2]))</pre>
\#rwb \leftarrow colorRampPalette(colors = c(kotare[3], kotare[3], kotare[3], kotare[3], 'white', kotare[2], kotare[2], kotare[2], kotare[2], kotare[2], kotare[3], kotare[3]
\#image.plot(x = Pset, y = phiset, z = t(cstar_p), las=1, col=rwb(100), ylab='Phi', xlab='Pred. Abun.');
relcstar_p <- cstar_p*NaN
for(i in 1:length(phiset)){
          relcstar_p[i,1:length(Pset)] <- cstar_p[i,1:length(Pset)]/cstar_p[i,1]</pre>
image.plot(x = Pset, y = phiset, z = t(log10(relcstar_p)),las=1,col=rwb(100),ylab='Phi',xlab='Pred. Abu
rwb <- colorRampPalette(colors = c(kotare[3], 'white', kotare[2]))</pre>
\#image.plot(x = Pset, y = epsilonset, z = t(cstar_e), las=1, col=rwb(100), ylab='Epsilon', xlab='Pred. Abun', xlab='Pred. Abu
relcstar_e <- cstar_e*NaN
for(i in 1:length(epsilonset)){
          relcstar_e[i,1:length(Pset)] <- cstar_e[i,1:length(Pset)]/cstar_e[i,1]</pre>
image.plot(x = Pset, y = epsilonset, z = t(log10(relcstar_e)),las=1,col=rwb(100),ylab='Epsilon',xlab='P
```

