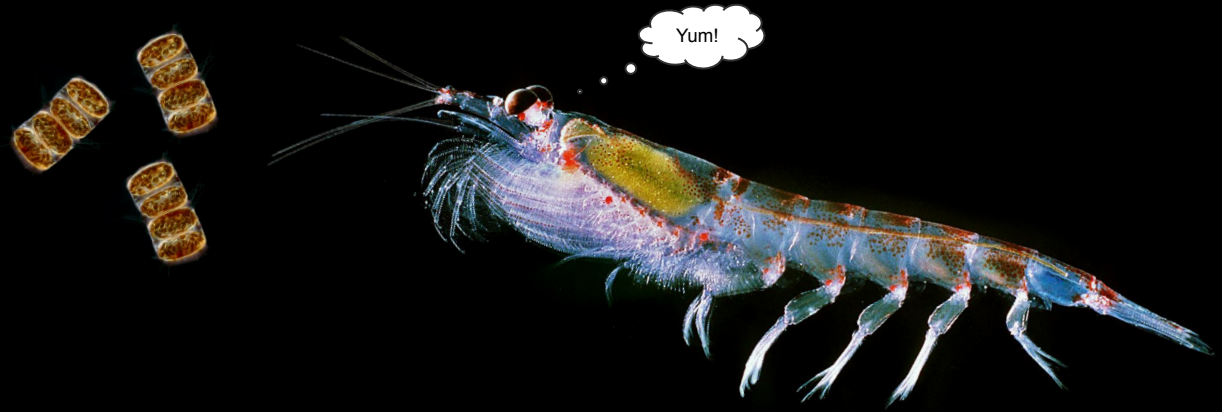


Trophic Control

Sea ice diatoms and Antarctic krill



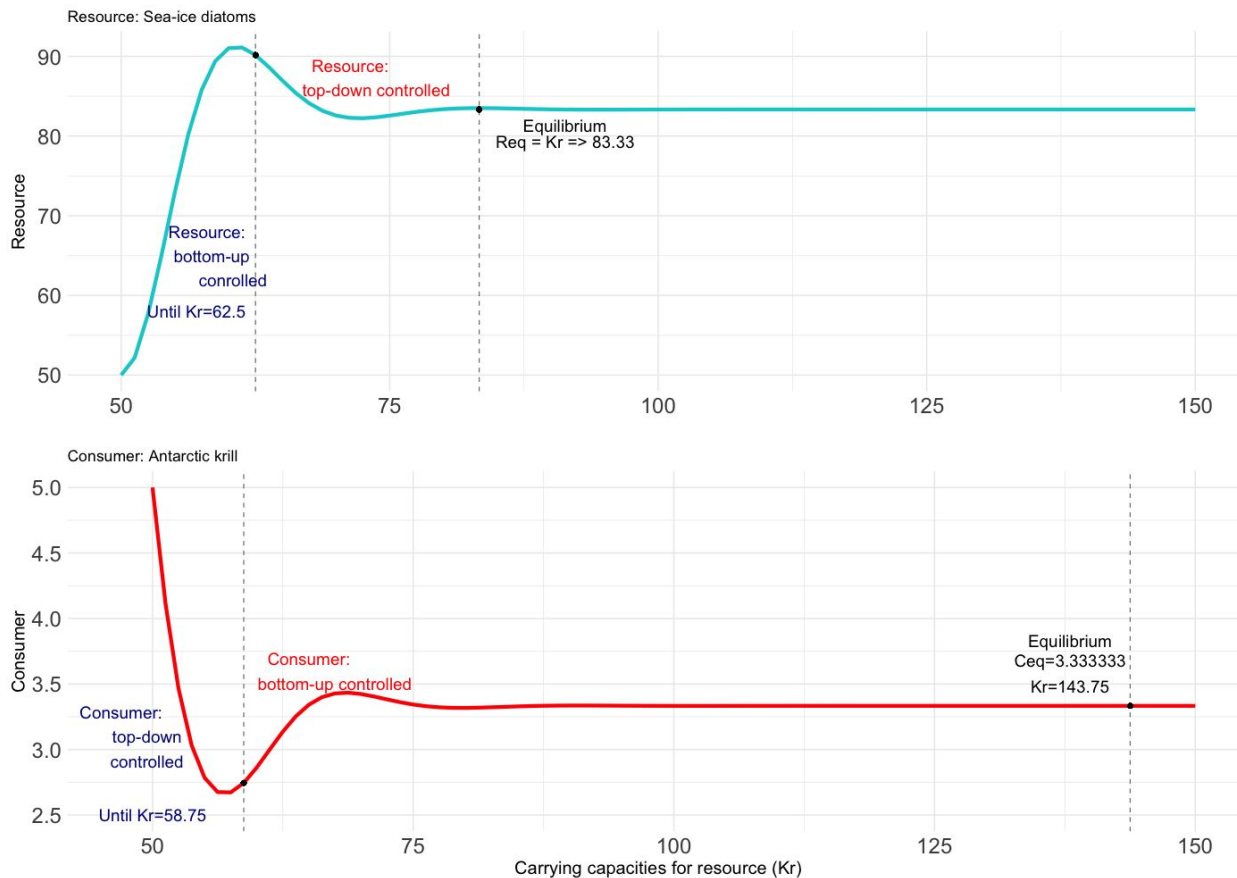
Swati Tak (s220868)

$$\frac{dR}{dt} = r R \left(1 - \frac{R}{K}\right) - b R C$$

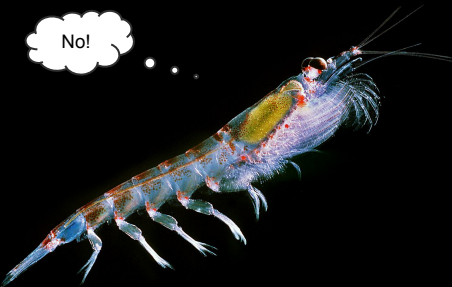
$$\frac{dC}{dt} = E b R C - d C$$

Initial number of diatoms	$R_0 \Rightarrow 50$
Initial number of krill	$C_0 \Rightarrow 5$
Growth rate of diatoms <i>per day</i>	$r \Rightarrow 0.75$
Clearance rate of diatoms by krill <i>diatoms eaten per day</i>	$b \Rightarrow 0.1$
Death rate of krill <i>Number of krill dying per day</i>	$d \Rightarrow 0.5$
Growth efficiency of krill per unit of diatoms consumed <i>Number of krill attaining adulthood per unit of diatoms consumed per day</i>	$E \Rightarrow 0.06$
Equilibrium without carrying capacity	Resource $\Rightarrow d/(E * b) \Rightarrow 83.33$ Consumer $\Rightarrow (r/b) \Rightarrow 7.5$
Actual equilibrium values obtained	$R_{eq} \Rightarrow 83.33$ $C_{eq} \Rightarrow 3.333333$

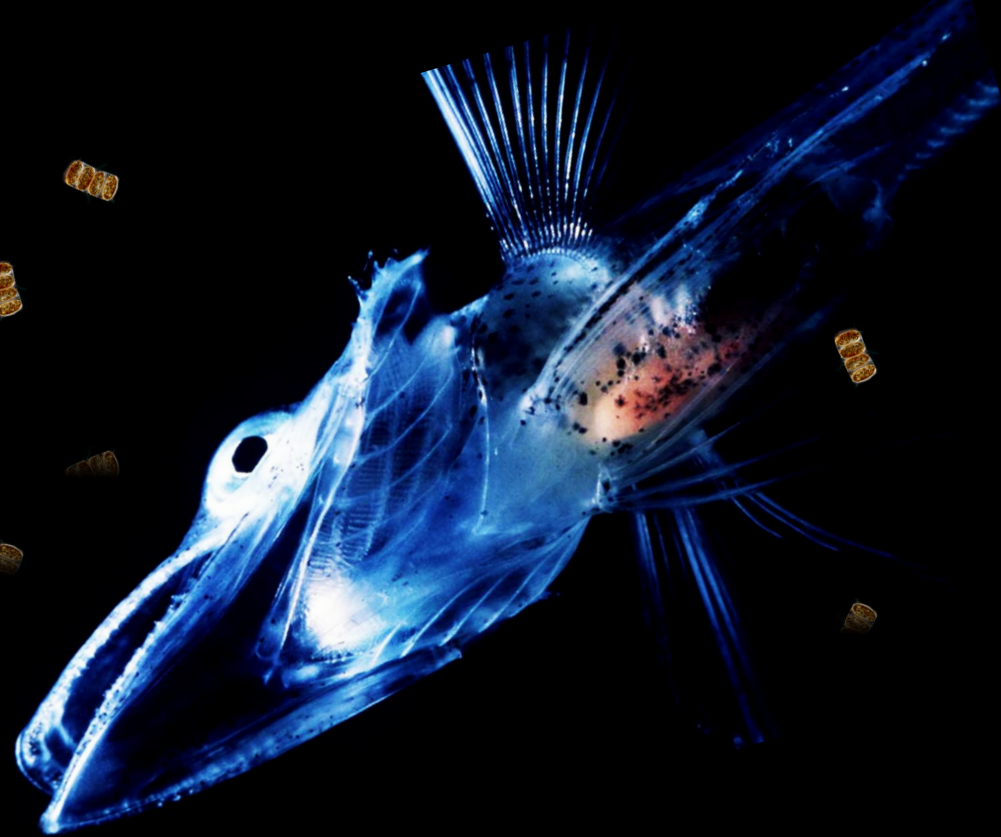
Bifurcation diagrams for trophic control with varying carrying capacity for resource



munch!
munch!



No!



Adding a predator: Antarctic Icefish

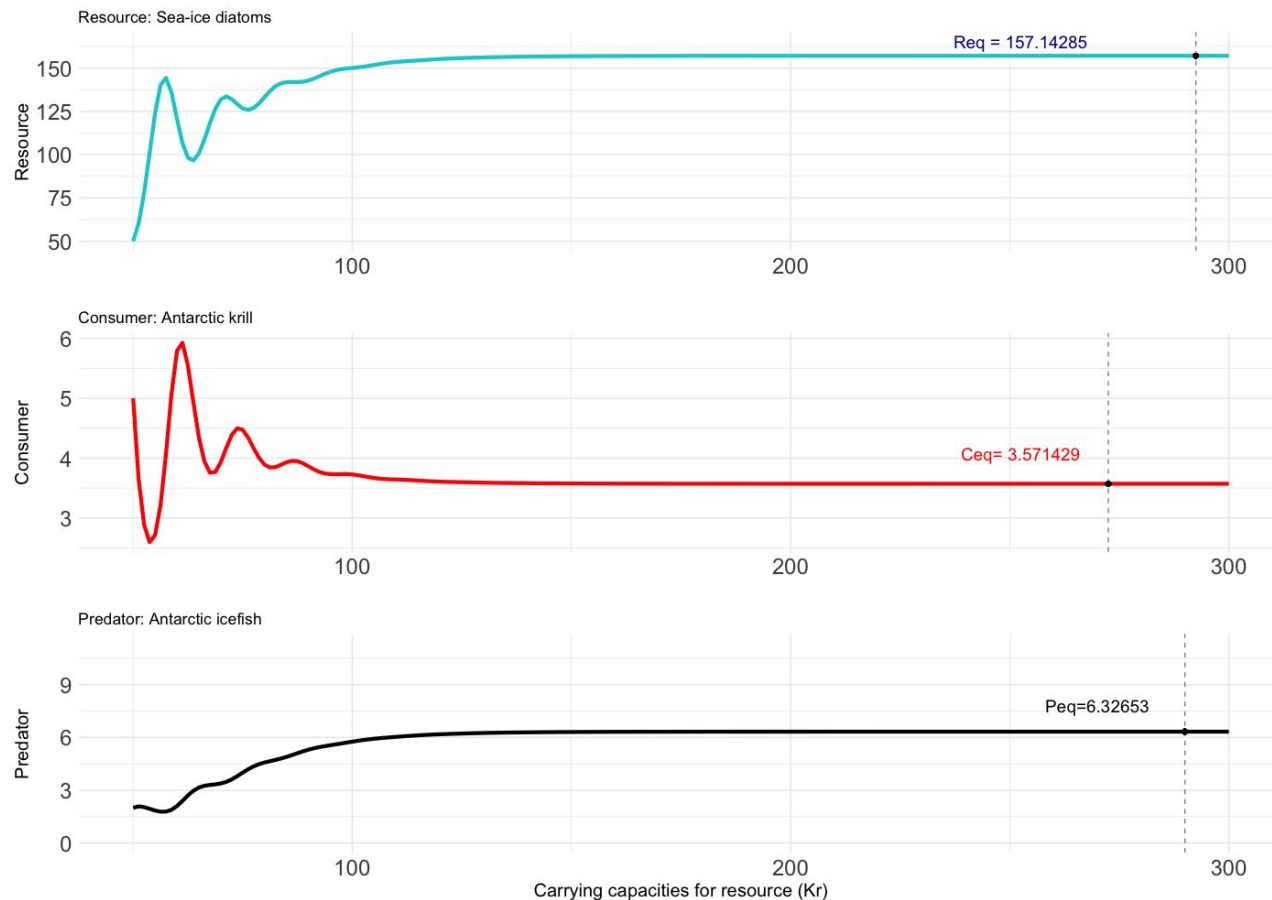
$$\frac{dR}{dt} = r R \left(1 - \frac{R}{K}\right) - b R C$$

$$\frac{dC}{dt} = E_c b R C - d_c C - a C P$$

$$\frac{dP}{dt} = E_p a C P - d_p P$$

Initial states	$R_0 \Rightarrow 50$ $C_0 \Rightarrow 5$ $P_0 \Rightarrow 2$ #predator
Growth rate of diatoms <small>per day</small>	$r \Rightarrow 0.75$
Clearance rates of diatoms by krill and icefish <small>consumption units per day</small>	$b \Rightarrow 0.1$ $a \Rightarrow 0.07$
Death rates for consumer and predator <small>Numbers dying per day</small>	$\text{death}_c \Rightarrow 0.5$ $\text{death}_p \Rightarrow 0.2$
Growth efficiencies for consumer and predator <small>Number of offsprings attaining adulthood per unit of consumption per day</small>	$E_c \Rightarrow 0.06$ $E_p \Rightarrow 0.8$
Equilibrium values obtained	$\text{Req} \Rightarrow 157.14285$ $\text{Ceq} \Rightarrow 3.571429$ $\text{Peq} \Rightarrow 6.326530$

**Adding a Predator leads to a more efficient and moderated trophic control
That is, higher numbers for Resource as well as Consumer, given a suitable K.**



Model extension



That bloodless
Channichthys !

What
happened!

Kicked the
bucket just
by seeing
Mr. Icey

Bifurcation with varying death rates of the consumer

Carrying capacity for resource

$K_r \Rightarrow 250$

Equilibrium values obtained

$R_{eq} \Rightarrow 130.9524$

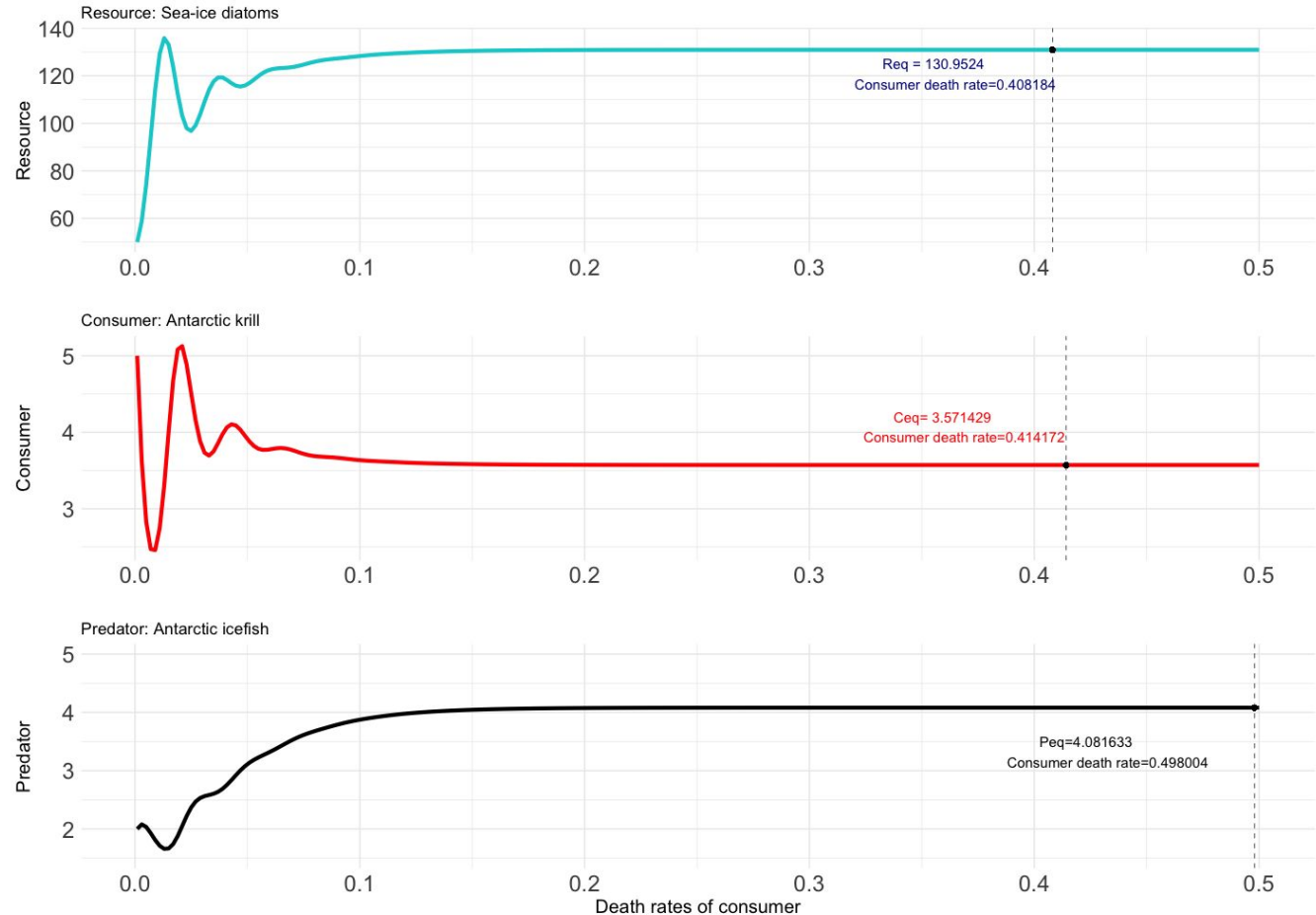
$C_{eq} \Rightarrow 3.571429$

$P_{eq} \Rightarrow 4.08163$

With varying death rates for consumer, the equilibrium for all three species takes slightly more time to reach.

Additionally, equilibrium population sizes for both resource and predator reduced, while that for the consumer remained unchanged.

Bifurcation diagrams for trophic control with varying death rates for consumer



```
> tail(results3)
  time    R      C      P death_C
246  245 130.9524 3.571429 4.081632 0.490020
247  246 130.9524 3.571429 4.081632 0.492016
248  247 130.9524 3.571429 4.081632 0.494012
249  248 130.9524 3.571429 4.081632 0.496008
250  249 130.9524 3.571429 4.081633 0.498004
251  250 130.9524 3.571429 4.081633 0.500000
```

```
> tail(results)
```

	time	R	C	K_values
76	75	83.33332	3.333333	143.75
77	76	83.33333	3.333333	145.00
78	77	83.33333	3.333333	146.25
79	78	83.33333	3.333333	147.50
80	79	83.33333	3.333333	148.75
81	80	83.33333	3.333333	150.00

Model 1: Resource +
Consumer + Varying K
for resource

Model 2: Resource +
Consumer + Predator +
Varying K for resource

```
> tail(results2)
```

	time	R	C	P	K_values
196	195	157.1428	3.571429	6.32653	293.75
197	196	157.1428	3.571429	6.32653	295.00
198	197	157.1428	3.571429	6.32653	296.25
199	198	157.1428	3.571429	6.32653	297.50
200	199	157.1428	3.571429	6.32653	298.75
201	200	157.1429	3.571429	6.32653	300.00

Model 3: Resource +
Consumer + Predator +
Varying death rates for
consumer

```
> tail(results3)
```

	time	R	C	P	death_C
246	245	130.9524	3.571429	4.081632	0.490020
247	246	130.9524	3.571429	4.081632	0.492016
248	247	130.9524	3.571429	4.081632	0.494012
249	248	130.9524	3.571429	4.081632	0.496008
250	249	130.9524	3.571429	4.081633	0.498004
251	250	130.9524	3.571429	4.081633	0.500000



AM ONE IN A KRILLION.

Mathematical Biology

Weekly exercise 9: Trophic Control

Swati Tak (s220868)

2023-11-06

Setting work environment

```
# Bifurcation Diagrams

## Resource: Phytoplankton
## Consumer: Zooplankton

setwd("/Users/swati/Desktop/Mathematical Biology/")
library(deSolve)
library(ggplot2)
library(patchwork)
```

Model 1: Resource, consumer and varying carrying capacities for resource

```
# Setting the parameters and initial state
r <- 0.75 # growth rate of the resource
b <- 0.1 # clearance rate of the resource by the consumer
d <- 0.5 # death rate of the consumer
E <- 0.06 # Growth efficiency of the consumer per unit of consumption

R0 <- 50 # Initial number of resource, plankton -- Thalassiosira antarctica
C0 <- 5 # Initial number of consumers -- Antarctic Krill

# Setting points of equilibrium without carrying capacity
Req <- d/(E * b)
Req
```

```
## [1] 83.33333
```

```
Ceq <- (r/b)
Ceq
```

```
## [1] 7.5
```



```

# Defining a function for the Lotka Volterra model using the equations
model <- function(t, state, parms) {
  with(as.list(c(state, parms)), {
    dR <- (r * (1 - (R / K_R)) * R) - (b * R * C)
    dC <- (E * b * R * C) - (d * C)
    return(list(c(dR, dC)))
  })}

# Time span
# times <- seq(0, 3000, by = 1)
times <- seq(0, 80, by = 1)

# Results from a simulation run for 3000 times gave the same equilibrium values
# Equilibrium for resource and consumer reached in a simulation of 300 steps

# Simulating multiple K values
# Adjusted to match times length
Kseq_R <- seq(R0, R0*C0-100, length.out = length(times))

# Variable to save different values of K
K_values_R <- c() # Resource K values

# Applying the model through a loop for all values of K
for (i in seq_along(Kseq_R)) {
  K_R <- Kseq_R[i]
  K_values_R <- append(K_values_R, Kseq_R[i])

  state <- c(R = R0, C = C0)
  parms <- c(E = E, b = b, r = r, d = d, K_R = K_R)
  out <- ode(y = state, times = times, parms = parms, func = model)
}

tail(out)

```

```

##      time      R      C
## [76,]  75 83.33332 3.333333
## [77,]  76 83.33333 3.333333
## [78,]  77 83.33333 3.333333
## [79,]  78 83.33333 3.333333
## [80,]  79 83.33333 3.333333
## [81,]  80 83.33333 3.333333

```

```

# Wrangling the results
results <- as.data.frame(out)
results$K_values <- K_values_R
head(results)

```

```

##   time      R      C K_values
## 1    0 50.00000 5.000000   50.00
## 2    1 52.17533 4.112117   51.25
## 3    2 57.64733 3.462916   52.50
## 4    3 65.04118 3.033002   53.75
## 5    4 72.99487 2.783531   55.00

```

```
## 6      5 80.22909 2.675206      56.25
```

```
tail(results)
```

```
##      time      R      C K_values
## 76    75 83.33332 3.333333   143.75
## 77    76 83.33333 3.333333   145.00
## 78    77 83.33333 3.333333   146.25
## 79    78 83.33333 3.333333   147.50
## 80    79 83.33333 3.333333   148.75
## 81    80 83.33333 3.333333   150.00
```

```
# Calculating equilibrium values at each step
```

```
Req <- tail(results$R, 1)
```

```
Req
```

```
## [1] 83.33333
```

```
# Ceq <- (r/b) * (1 - (Req/K_values_R))
```

```
Ceq <- tail(results$C, 1)
```

```
Ceq
```

```
## [1] 3.333333
```

```
# Setting values for aesthetics and annotation font sizes
```

```
annot_size <- 5
```

```
colors <- c("darkturquoise", "red", "black", "darkblue") # my colors
```

```
mytheme <- function() { # Creating the custom theme function
```

```
  theme_minimal() +
```

```
  theme(
```

```
    axis.text.x = element_text(size = 18),
```

```
    axis.text.y = element_text(size = 18),
```

```
    axis.title.x = element_text(size = 15),
```

```
    axis.title.y = element_text(size = 15)
```

```
  )
```

```
}
```

```
# Creating separate plots for each column
```

```
plot1 <- ggplot(data.frame(x = results[, 4], y = results[, 2]),
```

```
  aes(x = x, y = y)) +
```

```
  labs(x = " ", y = "Resource") +
```

```
  ggtitle("Resource: Sea-ice diatoms") +
```

```
  geom_line(color = colors[1], linewidth = 1.5) +
```

```
  xlim(50, tail(results$K_values, 1)) +
```

```
# Point where resource becomes top-down controlled
```

```
geom_point(aes(x = results$K_values[11], y = results$R[11]),
```

```
  color = colors[3], shape = 19) +
```

```
geom_vline(xintercept = results$K_values[11], linetype = "dashed",
```

```
  color = colors[3], linewidth = 0.25) +
```

```
annotate("text", x = Req - 12, y = Req + 4, label = "Resource:
```

```
  top-down controlled", size = annot_size, color = colors[2]) +
```

```

# Area where resource is bottom-up controlled
annotate("text", x = 58, y = 65, label = "Resource:
bottom-up
    controlled", size = annot_size, color = colors[4]) +
annotate("text", x = 57, y = 58, label = "Until Kr=62.5", size = annot_size,
    color = colors[4]) +

# Point where Resource attains equilibrium
geom_point(aes(x = Req, y = Req), color = colors[3], shape = 19) +
geom_vline(xintercept = Req, linetype = "dashed", color = colors[3],
    linewidth = 0.25) +

# Textual annotations
annotate("text", x = Req + 8, y = Req - 2, label = "Equilibrium",
    size = annot_size) +
annotate("text", x = Req + 8, y = Req - 4, label = "Req = Kr => 83.33",
    size = annot_size, color = colors[3]) +
mytheme()

plot2 <- ggplot(data.frame(x = results[, 4], y = results[, 3]),
    aes(x = x, y = y)) +
geom_line(color = colors[2], linewidth = 1.5) +
labs(x = "Carrying capacities for resource (Kr)", y = "Consumer") +
ggtitle("Consumer: Antarctic krill") +

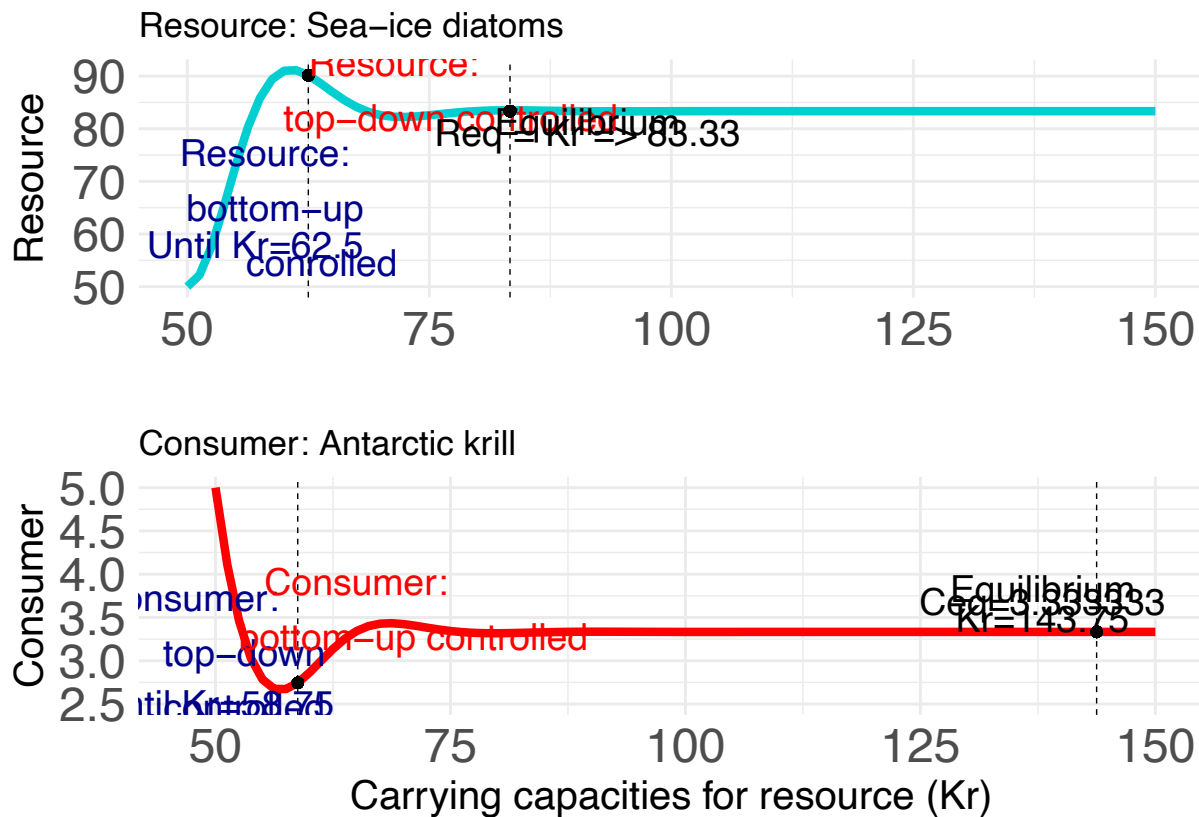
# Point where consumer start turning bottom-up controlled
geom_point(aes(x = results$K_values[8], y = results$C[8]), color = colors[3],
    shape = 19) +
geom_vline(xintercept = results$K_values[8], linetype = "dashed",
    color = colors[3], linewidth = 0.25) +
annotate("text", x = 47, y = 3.1, label = "Consumer:
top-down
    controlled", size = annot_size, color = colors[4]) +
annotate("text", x = 50, y = 2.5, label = "Until Kr=58.75", size = annot_size,
    color = colors[4]) +

# Equilibrium point for consumer
geom_point(aes(x = results$K_values[76], y = Ceq), color = colors[3],
    shape = 19) +
geom_vline(xintercept = results$K_values[76], linetype = "dashed",
    color = colors[3], linewidth = 0.25) +
annotate("text", x = 65, y = 3.6, label = "Consumer:
bottom-up controlled", size = annot_size, color = colors[2]) +
annotate("text", x = 138, y = Ceq+0.5, label = "Equilibrium",
    size = annot_size) +
annotate("text", x = 138, y = Ceq+0.35, label = "Ceq=3.333333",
    size = annot_size, color = colors[3]) +
annotate("text", x = 138, y = Ceq+0.15, label = "Kr=143.75",
    size = annot_size, color = colors[3]) +
mytheme()

# Combining plots using patchwork
combined_plot <- plot1 / plot2

```

```
# Displaying the combined plot
combined_plot
```



Model 2: Adding a predator that eats the consumer (krill) → Icefish (Channichthyidae)

```
# Setting the parameters and initial state
r <- 0.75 # growth rate of the resource
b <- 0.1 # clearance rate of the resource by the consumer
death_C <- 0.5 # death rate of the consumer
E_c <- 0.06 # Growth efficiency of the consumer per unit of consumption

# Predator parameters -- new elements
a <- 0.07 # clearance rate of the consumer by the predator
death_P <- 0.2 # death rate of the predator
E_p <- 0.8 # Growth efficiency of the predator per unit of predation

# Initial states
R0 <- 50 # Initial number of resource, plankton -- Thalassiosira antarctica
C0 <- 5 # Initial number of consumers -- Antarctic Krill
P0 <- 2 # Number of predators, Icefish (Channichthyidae)
```

```

# Redefining the model
model2 <- function(t, state, parms) {
  with(as.list(c(state, parms)), {
    dR <- (r * (1 - (R / K_R))*R) - (b * R * C)
    dC <- (E_c * b * R * C) - (death_C * C) - (a * C * P)
    dP <- (E_p * a * C * P) - (death_P * P)

    return(list(c(dR, dC, dP)))
  })}

# Time span
# times <- seq(0, 3000, by = 1)
times <- seq(0, 200, by = 1)

# Simulating multiple K values
# Adjusted to match times length
Kseq_R <- seq(R0, R0*C0*P0-200, length.out = length(times))

# Resetting the K values for the loop
K_values_R <- c()

for (i in seq_along(Kseq_R)) {
  K_R <- Kseq_R[i]
  K_values_R <- append(K_values_R, Kseq_R[i])

  state <- c(R = R0, C = C0, P = P0)
  parms <- c(E_c = E_c, E_p = E_p, a=a, b = b, r = r, death_C = death_C,
    death_P = death_P, K_R = K_R)
  out2 <- ode(y=state, times=times, parms = parms, func = model2)
}

# Wrangling the results
results2 <- as.data.frame(out2)
results2$K_values <- K_values_R
head(results2)

```

```

##   time      R      C      P K_values
## 1    0 50.00000 5.000000 2.000000   50.00
## 2    1 60.29458 3.642621 2.079279   51.25
## 3    2 77.93478 2.884726 2.038724   52.50
## 4    3 100.74949 2.596963 1.942240   53.75
## 5    4 123.85435 2.711057 1.841617   55.00
## 6    5 140.28449 3.217132 1.776789   56.25

```

```
tail(results2)
```

```

##   time      R      C      P K_values
## 196 195 157.1428 3.571429 6.32653   293.75
## 197 196 157.1428 3.571429 6.32653   295.00
## 198 197 157.1428 3.571429 6.32653   296.25
## 199 198 157.1428 3.571429 6.32653   297.50
## 200 199 157.1428 3.571429 6.32653   298.75
## 201 200 157.1429 3.571429 6.32653   300.00

```

```
Req2 <- tail(results2$R, 1)
Ceq2 <- tail(results2$C, 1)
Peq <- tail(results2$P, 1)
```

```
Req2
```

```
## [1] 157.1429
```

```
Ceq2
```

```
## [1] 3.571429
```

```
Peq
```

```
## [1] 6.32653
```

```
# Setting values for aesthetics and annotation font sizes
annot_size <- 5.5

# Create separate plots for each column
plot1 <- ggplot(data.frame(x = results2[, 5], y = results2[, 2]),
  aes(x = x, y = y)) +
  geom_line(color = colors[1], linewidth = 1.5) +

  # Equilibrium point for resource
  geom_point(aes(x = results2$K_values[195], y = results2$R[195]),
    color = colors[3], shape = 19) +
  geom_vline(xintercept = results2$K_values[195], linetype = "dashed",
    color = colors[3], linewidth = 0.25) +
  annotate("text", x = results2$K_values[195]-40, y = results2$R[195]+8,
    label = "Req = 157.14285", size = annot_size, colour=colors[4]) +

  labs(x = "", y = "Resource") +
  ggtitle("Resource: Sea-ice diatoms") +
  mytheme()

plot2 <- ggplot(data.frame(x = results2[, 5], y = results2[, 3]),
  aes(x = x, y = y)) +
  geom_line(color = colors[2], linewidth = 1.5) +

  # Equilibrium point for consumer
  geom_point(aes(x = results2$K_values[179], y = results2$C[179]),
    color = colors[3], shape = 19) +
  geom_vline(xintercept = results2$K_values[179], linetype = "dashed",
    color = colors[3], linewidth = 0.25) +
  annotate("text", x = results2$K_values[179]-20, y = results2$C[179]+0.5,
    label = "Ceq= 3.571429", size = annot_size, colour=colors[2]) +

  labs(x = "", y = "Consumer") +
  ggtitle("Consumer: Antarctic krill") +
  mytheme()
```

```

plot3 <- ggplot(data.frame(x = results2[, 5], y = results2[, 4]),
  aes(x = x, y = y)) +
  geom_line(color = colors[3], linewidth = 1.5) +

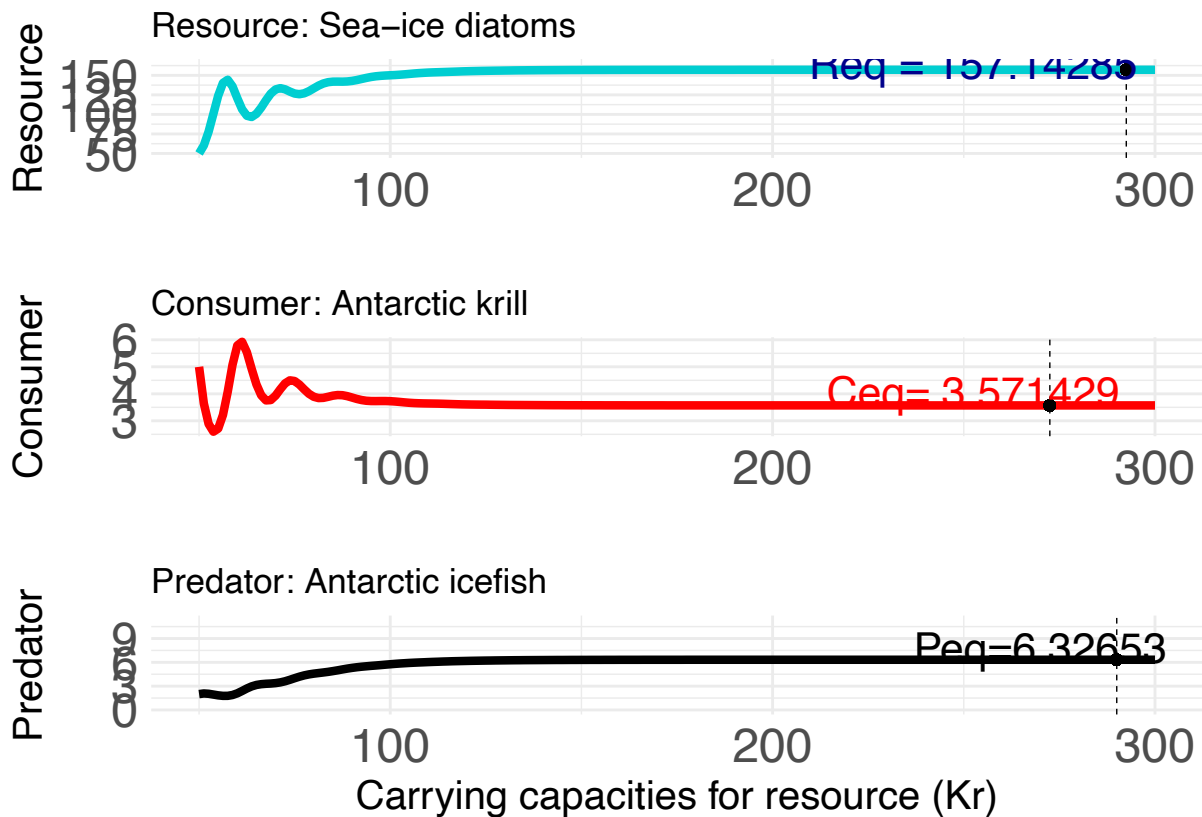
  # Equilibrium point for predator
  geom_point(aes(x = results2$K_values[193], y = results2$P[193]),
    color = colors[3], shape = 19) +
  geom_vline(xintercept = results2$K_values[193], linetype = "dashed",
    color = colors[3], linewidth = 0.25) +
  annotate("text", x = results2$K_values[193]-20, y = results2$P[193]+1.5,
    label = "Peq=6.32653", size = annot_size, colour=colors[3]) +
  ylim(0, Peq+5) +

  labs(x = "Carrying capacities for resource (Kr)", y = "Predator") +
  ggtitle("Predator: Antarctic icefish") +
  mytheme()

# Combine plots using patchwork
combined_plot2 <- plot1 / plot2 / plot3

# Display the combined plot
combined_plot2

```



Model 3 – Extension: Changing the values of consumer's death rates

```
# Setting the parameters and initial state
r <- 0.75 # growth rate of the resource
K_R <- 250 # Carrying capacity for resource
b <- 0.1 # clearance rate of the resource by the consumer
E_c <- 0.06 # Growth efficiency of the consumer per unit of consumption

# Setting time steps
# times <- seq(0, 3000, by = 1)
times <- seq(0, 250, by = 1)

# Simulating multiple values for consumer's death rate
# Adjusted to match times length
death_C_seq <- seq(0.001, 0.5, length.out = length(times))

# Resetting consumer's death rate values for the loop
death_C_values <- c()

# Predator parameters -- new elements
a <- 0.07 # clearance rate of the consumer by the predator
death_P <- 0.2 # death rate of the predator
E_p <- 0.8 # Growth efficiency of the predator per unit of predation

# Initial states
R0 <- 50 # Initial number of resource, plankton -- Thalassiosira antarctica
C0 <- 5 # Initial number of consumers -- Antarctic Krill
P0 <- 2 # Number of predators, Icefish (Channichthyidae)

# Redefining the model
model2 <- function(t, state, parms) {
  with(as.list(c(state, parms)), {
    dR <- (r * (1 - (R / K_R)) * R) - (b * R * C)
    dC <- (E_c * b * R * C) - (death_C * C) - (a * C * P)
    dP <- (E_p * a * C * P) - (death_P * P)

    return(list(c(dR, dC, dP)))
  })}

for (i in seq_along(death_C_seq)) {
  death_C <- death_C_seq[i]
  death_C_values <- append(death_C_values, death_C_seq[i])

  state <- c(R = R0, C = C0, P = P0)
  parms <- c(E_c = E_c, E_p = E_p, a=a, b = b, r = r, death_C = death_C,
            death_P = death_P, K_R = K_R)
  out3 <- ode(y=state, times=times, parms = parms, func = model2)
}

tail(out3)
```



```
##      time      R      C      P
## [246,] 245 130.9524 3.571429 4.081632
## [247,] 246 130.9524 3.571429 4.081632
## [248,] 247 130.9524 3.571429 4.081632
## [249,] 248 130.9524 3.571429 4.081632
## [250,] 249 130.9524 3.571429 4.081633
## [251,] 250 130.9524 3.571429 4.081633
```

```
head((out3))
```

```
##      time      R      C      P
## [1,]  0 50.00000 5.000000 2.000000
## [2,]  1 58.82661 3.627606 2.078667
## [3,]  2 74.26799 2.830600 2.034411
## [4,]  3 94.05524 2.472476 1.928830
## [5,]  4 114.10358 2.459617 1.810377
## [6,]  5 129.30968 2.747641 1.712580
```

```
# Wrangling the results
results3 <- as.data.frame(out3)
results3$death_C <- death_C_values
head(results3)
```

```
##   time      R      C      P death_C
## 1    0 50.00000 5.000000 2.000000 0.001000
## 2    1 58.82661 3.627606 2.078667 0.002996
## 3    2 74.26799 2.830600 2.034411 0.004992
## 4    3 94.05524 2.472476 1.928830 0.006988
## 5    4 114.10358 2.459617 1.810377 0.008984
## 6    5 129.30968 2.747641 1.712580 0.010980
```

```
tail(results3)
```

```
##      time      R      C      P death_C
## 246 245 130.9524 3.571429 4.081632 0.490020
## 247 246 130.9524 3.571429 4.081632 0.492016
## 248 247 130.9524 3.571429 4.081632 0.494012
## 249 248 130.9524 3.571429 4.081632 0.496008
## 250 249 130.9524 3.571429 4.081633 0.498004
## 251 250 130.9524 3.571429 4.081633 0.500000
```

```
Req3 <- tail(results3$R, 1)
Ceq3 <- tail(results3$C, 1)
Peq2 <- tail(results3$P, 1)
```

```
Req3
```

```
## [1] 130.9524
```

Ceq3

```
## [1] 3.571429
```

Peq2

```
## [1] 4.081633
```

Peq

```
## [1] 6.32653
```

```
# Setting values for aesthetics and annotation font sizes
annot_size <- 4
colors <- c("darkturquoise", "red", "black", "darkblue")

# Create separate plots for each column
plot1 <- ggplot(data.frame(x = results3[, 5], y = results3[, 2]),
  aes(x = x, y = y)) +
  geom_line(color = colors[1], linewidth = 1.5) +

  # Equilibrium point for resource
  geom_point(aes(x = results3$death_C[205], y = results3$R[205]),
    color = colors[3], shape = 19) +
  geom_vline(xintercept = results3$death_C[205], linetype = "dashed",
    color = colors[3], linewidth = 0.25) +
  annotate("text", x = results3$death_C[205]-0.053, y = results3$R[205]-10,
    label = "Req = 130.9524
    Consumer death rate=0.408184", size = annot_size, colour=colors[4]) +

  labs(x = " ", y = "Resource") +
  ggtitle("Resource: Sea-ice diatoms") +
  mytheme()

plot2 <- ggplot(data.frame(x = results3[, 5], y = results3[, 3]),
  aes(x = x, y = y)) +
  geom_line(color = colors[2], linewidth = 1.5) +

  # Equilibrium point for consumer
  geom_point(aes(x = results3$death_C[208], y = results3$C[208]),
    color = colors[3], shape = 19) +
  geom_vline(xintercept = results3$death_C[208], linetype = "dashed",
    color = colors[3], linewidth = 0.25) +
  annotate("text", x = results3$death_C[208]-0.055, y = results3$C[208]+0.5,
    label = "Ceq= 3.571429
    Consumer death rate=0.414172", size = annot_size, colour=colors[2]) +

  labs(x = " ", y = "Consumer") +
  ggtitle("Consumer: Antarctic krill") +
  mytheme()
```

```

plot3 <- ggplot(data.frame(x = results3[, 5], y = results3[, 4]),
               aes(x = x, y = y)) +
  geom_line(color = colors[3], linewidth = 1.5) +
  ylim(1.5,5) +

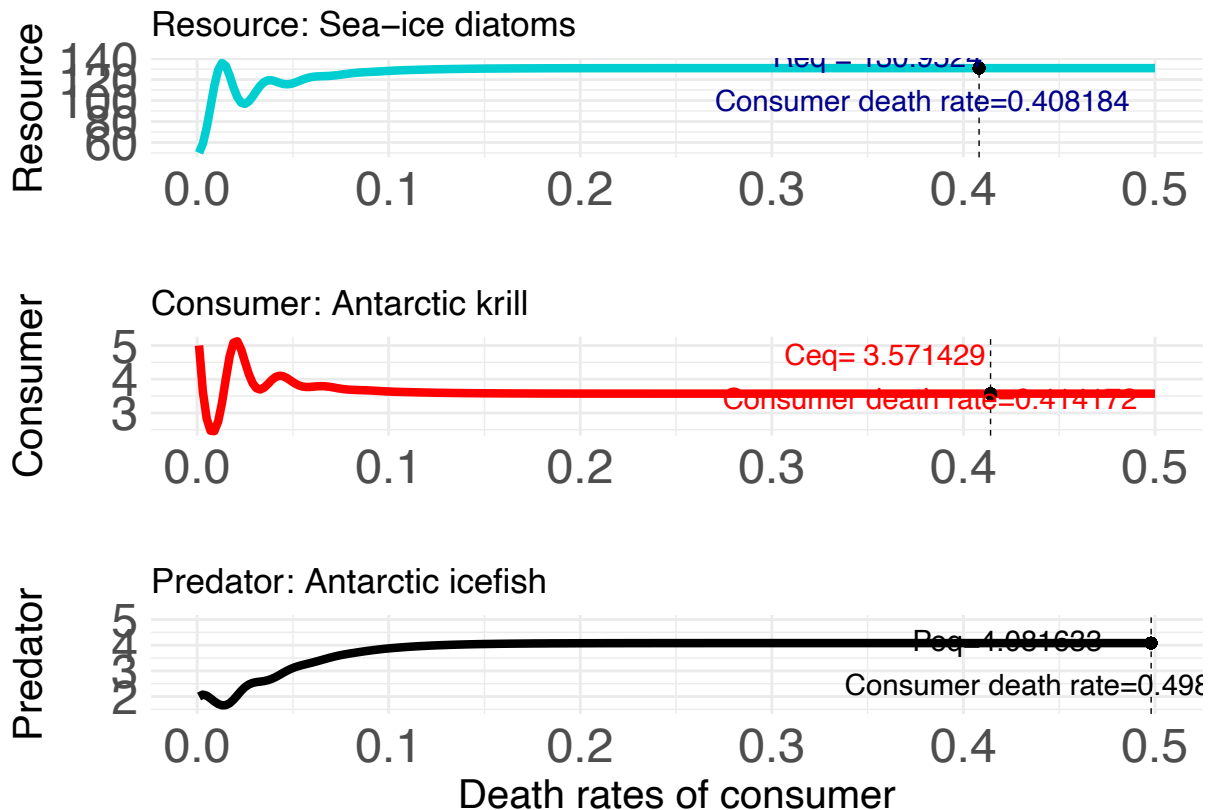
  # Equilibrium point for predator
  geom_point(aes(x = results3$death_C[250], y = results3$P[250]),
            color = colors[3], shape = 19) +
  geom_vline(xintercept = results3$death_C[250], linetype = "dashed",
            color = colors[3], linewidth = 0.25) +
  annotate("text", x = results3$death_C[250]-0.075, y = results3$P[250]-0.75,
         label = "Peq=4.081633
               Consumer death rate=0.498004", size = annot_size, colour=colors[3]) +

  labs(x = "Death rates of consumer", y = "Predator") +
  ggtitle("Predator: Antarctic icefish") +
  mytheme()

# Combine plots using patchwork
combined_plot3 <- plot1 / plot2 / plot3

# Display the combined plot
combined_plot3

```



Printing the results for three models

```
# Print all results  
tail(results)
```

```
##      time      R      C K_values  
## 76   75 83.33332 3.333333  143.75  
## 77   76 83.33333 3.333333  145.00  
## 78   77 83.33333 3.333333  146.25  
## 79   78 83.33333 3.333333  147.50  
## 80   79 83.33333 3.333333  148.75  
## 81   80 83.33333 3.333333  150.00
```

```
tail(results2)
```

```
##      time      R      C      P K_values  
## 196  195 157.1428 3.571429 6.32653  293.75  
## 197  196 157.1428 3.571429 6.32653  295.00  
## 198  197 157.1428 3.571429 6.32653  296.25  
## 199  198 157.1428 3.571429 6.32653  297.50  
## 200  199 157.1428 3.571429 6.32653  298.75  
## 201  200 157.1429 3.571429 6.32653  300.00
```

```
tail(results3)
```

```
##      time      R      C      P death_C  
## 246  245 130.9524 3.571429 4.081632 0.490020  
## 247  246 130.9524 3.571429 4.081632 0.492016  
## 248  247 130.9524 3.571429 4.081632 0.494012  
## 249  248 130.9524 3.571429 4.081632 0.496008  
## 250  249 130.9524 3.571429 4.081633 0.498004  
## 251  250 130.9524 3.571429 4.081633 0.500000
```