Module 4: Sustainable Fishing

Jessie Golding and Ellen Bledsoe
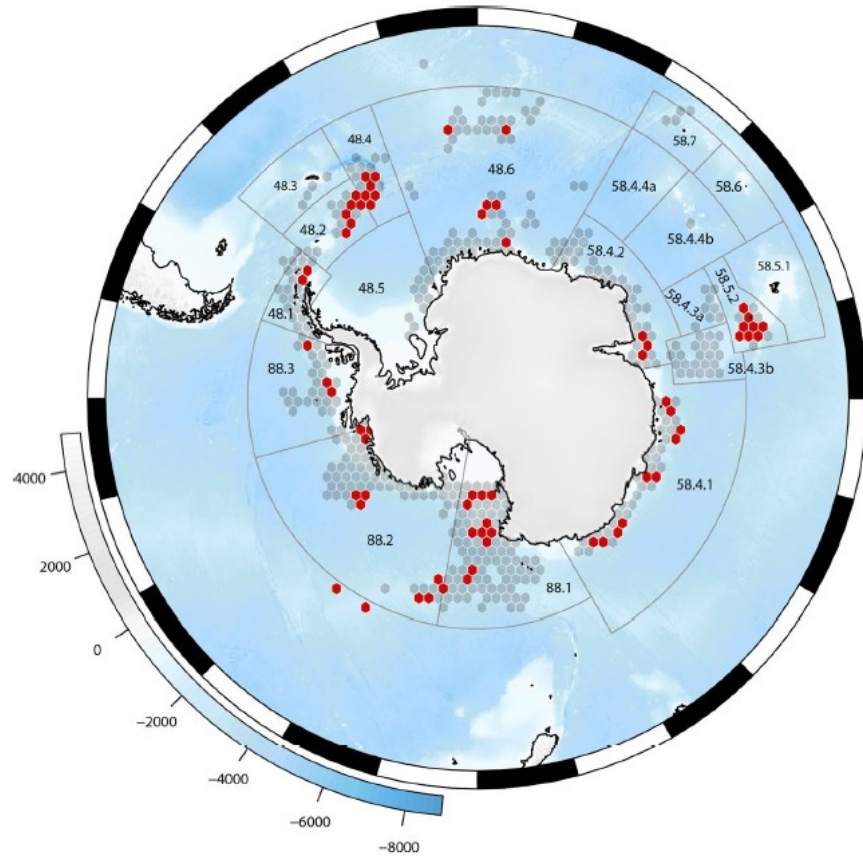
2024-12-03

Sustainable Fishing

Applying Principles of Population Growth

We have decided that the Antarctic toothfish (*Dissostichus mawsoni*) may be a good candidate for our fishing.

The Antarctic toothfish is fairly widely distributed around Antarctica. The hexagons below show where the Antarctic toothfish occurs. Gray represents known distribution and red represents locations where genetic samples were taken for a study on their population (Maschette et al. 2023).

Fig. 2 Locations of Antarctic toothfish (*Dissostichus mawsoni*) tissue samples collected (red) across areas where Antarctic toothfish have been caught (grey hexagons, amalgamated data from Robinson and Reid (2016) and Duhamel et al. (2014))

However, given concerns about maintaining the fishery (an important ecosystem service), we want to make sure we do not overfish the species but rather fish sustainably.

Let's keep exploring our fish populations to help us understand what we should be focusing on when making decisions about how many fish we can sustainably harvest from any given population.

Set-Up

As usual, we start with loading our packages and our data.

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----------------------- tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ----------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
fish <- read_csv("fish_pops.csv")
```

```
## Rows: 80 Columns: 6
## -- Column specification --------------------------------------------------------
```

```
## Delimiter: ","
## chr (1): growth_type
## dbl (5): N, year, r, carrying_capacity, population
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Let's take a look at the data and remind ourselves what the data look like.

```r
head(fish)
```

```
## # A tibble: 6 x 6
##       N  year growth_type     r carrying_capacity population
##   <dbl> <dbl> <chr>       <dbl>             <dbl>      <dbl>
## 1     5     1 exponential -0.45                NA          1
## 2     3     2 exponential -0.45                NA          1
## 3     2     3 exponential -0.45                NA          1
## 4     1     4 exponential -0.45                NA          1
## 5     1     5 exponential -0.45                NA          1
## 6     1     6 exponential -0.45                NA          1
```

Population Growth Models

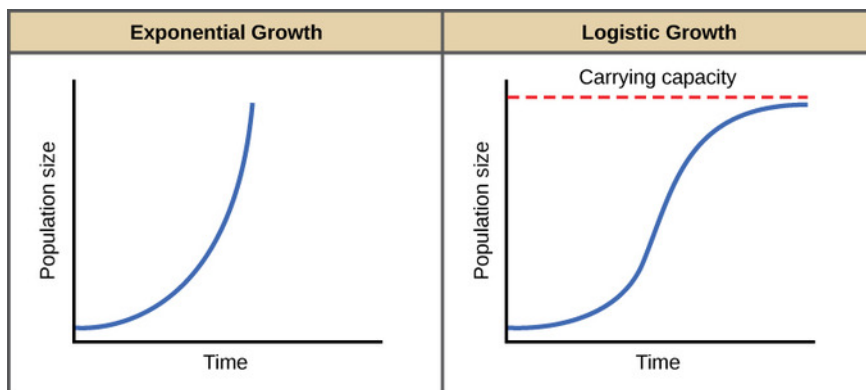Let's remind ourselves about exponential growth and logistic growth.

Expotential growth takes births and deaths (in the form of the intrinsic growth rate, `r`) into account:

$$N = rN_i$$

Logistic growth includes the intrinsic growth rate and another important factor—resources. The resources determine the maximum number of individuals a population can have, which we call the carrying capacity (`K`):

$$N = rN_i * (1 - \frac{N}{K})$$

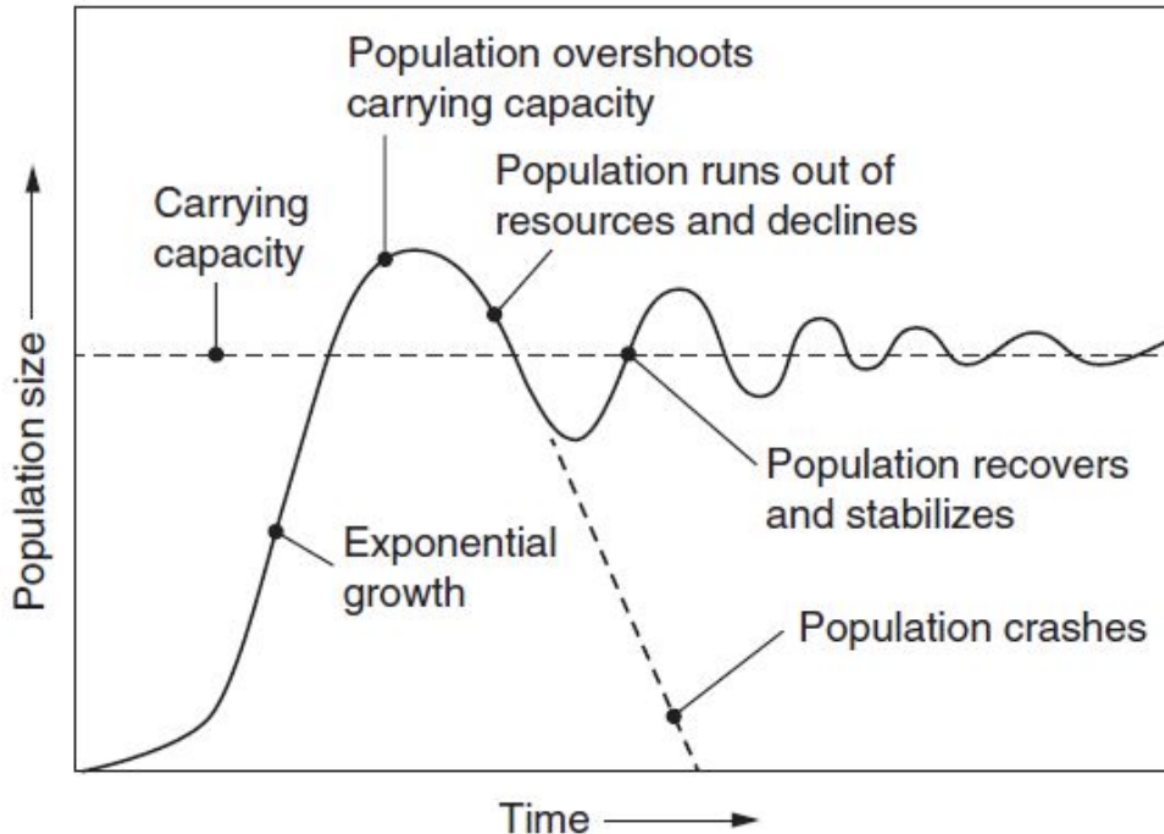This is what each type of population growth model looks like:

Which Population to Fish?

We've previously discussed what factors we want to focus on when we are deciding which population to fish. Ideally, we want to target both a large population and a stable population.

At first glance, a population that is growing exponentially seems ideal—and it is, in some ways.

But in reality, all populations will *eventually* hit some capacity on resources. Populations that are currently demonstrating exponential growth either haven't yet hit their carrying capacity or have dramatically overshot their carrying capacity, which can lead to a huge crash or a smaller crash with boom and bust cycles until it evens out around the carrying capacity.

So, in order for us to feel confident that our population will be large for a long time, we probably want to choose the safer option, which is choosing a population that is showing logistic growth.

Let's select the logistic growth population with the largest value, population 4, for our next analyses.

```
log_pop <- fish %>%
  filter(population == 4)
```

Estimating Carrying Capacity

In our journey to determine how many fish we can sustainably harvest (i.e., remove from the population without causing the population to go extinct), we first need to determine the carrying capacity.

The dataset contains carrying capacity estimates, but let's go ahead and get our own estimates. To do so, we are actually going to use a new package, `drc`!

This package will let us fit a logistic model and provide an estimate of the asymptote, which is `K`, our carrying capacity.

```
# load the drc package; the red text is fine (it doesn't say error)
library(drc)
```

```
## Loading required package: MASS
```

```
## 
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
## 
##     select

## 
## 'drc' has been loaded.

## Please cite R and 'drc' if used for a publication,

## for references type 'citation()' and 'citation('drc')'.

## 
## Attaching package: 'drc'

## The following objects are masked from 'package:stats':
## 
##     gaussian, getInitial
```

Now, we are going to use a function from the `drc` package called `drm` to fit our logistic growth model. It works very similarly to the inferential statistics models from Module 3.

What kind of growth do we think we have based on this plot? Logistic

```
# Fit the model to the data
# 'N ~ year' means we're predicting N (population) as a function of 'year'
# fct = LL.4() specifies to fit a logistic growth model
model <- drm(N ~ year, data = log_pop, fct = LL.4())
```

This time, unlike with the linear model, we are interested not in a p-value, put in the estimate of a specific value from the model–the carrying capacity.

To get all of the estimated values, we will use a function called `coef`, which is short for "coefficient," to give us the estimates from the `model` object.

```
estimates <- coef(model)
estimates
```

```
## b:(Intercept) c:(Intercept) d:(Intercept) e:(Intercept)
##     -3.671594      7.583561    104.457259      7.943917
```

In this case, we happen to be interested in the estimate they are calling `d:(Intercept)`; this is what we are calling K, the carrying capacity. Let's save that value.

The simplest way to choose one value from the vector `estimates` is to use square brackets for subsetting, something we haven't done since Module 1! We want the 3rd value.

```
K <- estimates[3]
K
```

```
## d:(Intercept)
##      104.4573
```

Our estimate for the carrying capacity is 104.5 fish, which is close to the carrying capacity value in the dataset (100).

We can remove the name (`d:(Intercept)`) if we want to by using the `unname()` function.
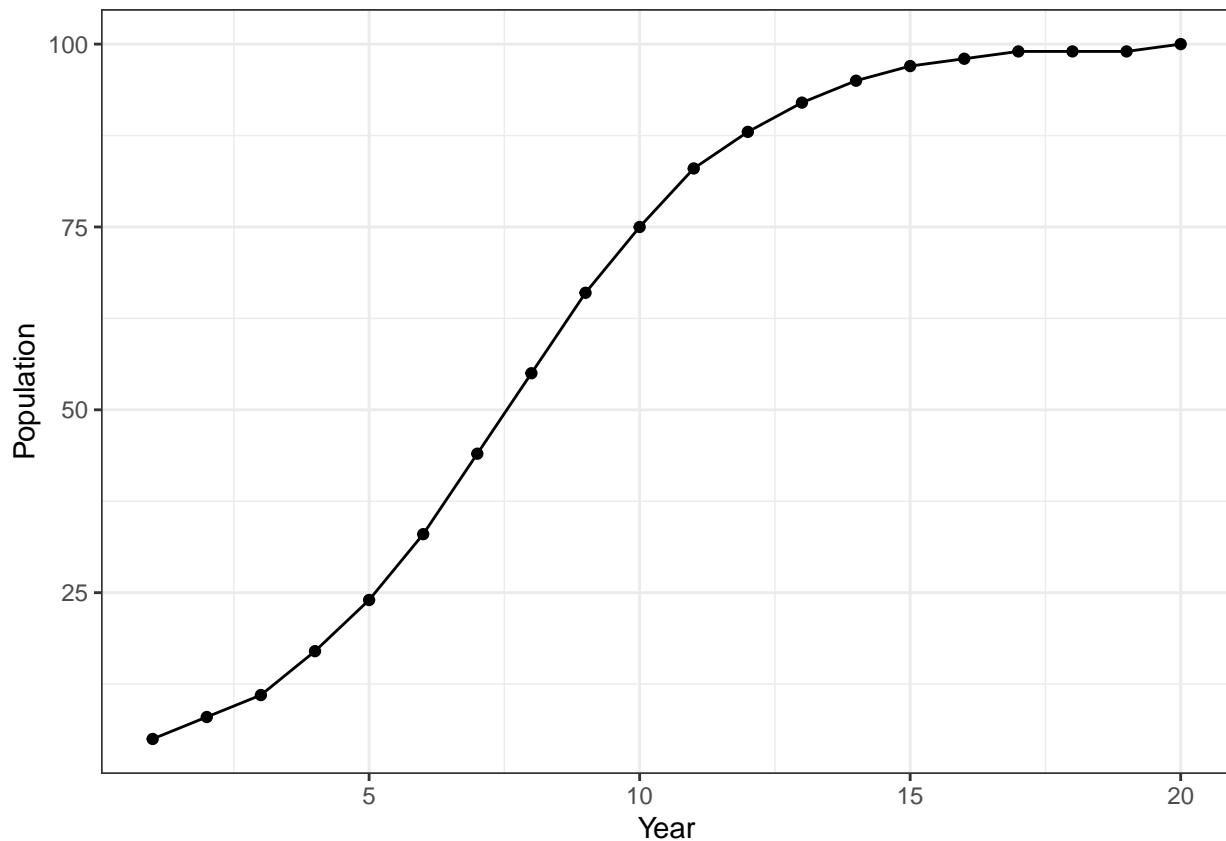
```
K <- unname(K)
K
```

```
## [1] 104.4573
```

Plotting the Carrying Capacity

Let's now add this estimated carrying capacity value to our plot of the data.
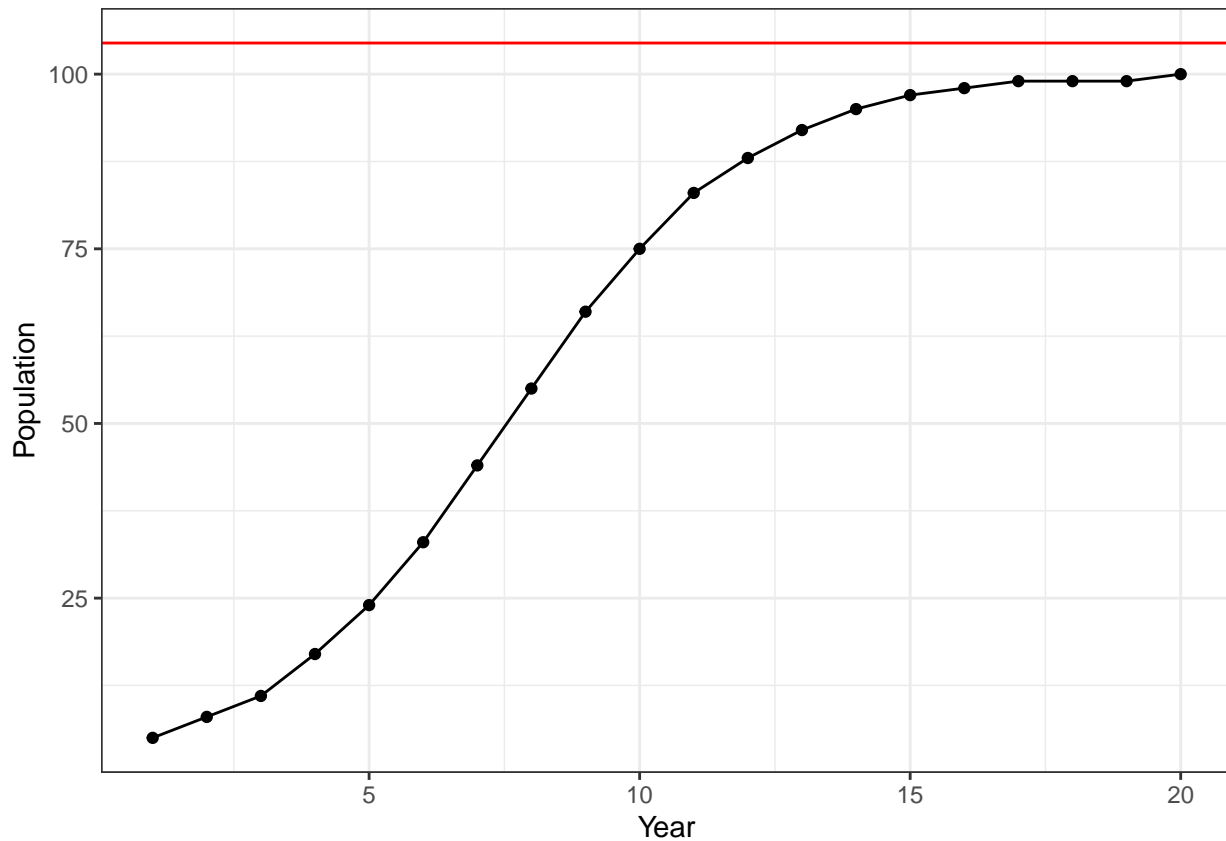
First, let's plot the population data alone.

```
ggplot(log_pop, aes(x = year, y = N)) + # Plot the original data points (year vs. population)
  geom_point() +
  geom_line() +
  labs(x = "Year", y = "Population") +
  theme_bw()
```


To add the a line representing the carrying capacity, we can use `geom_hline()` to add a horizontal line where the y-intercept is equal to the carrying capacity.

This is very similar to how we used `geom_vline()` to add a vertical line to histograms to represent the mean value.
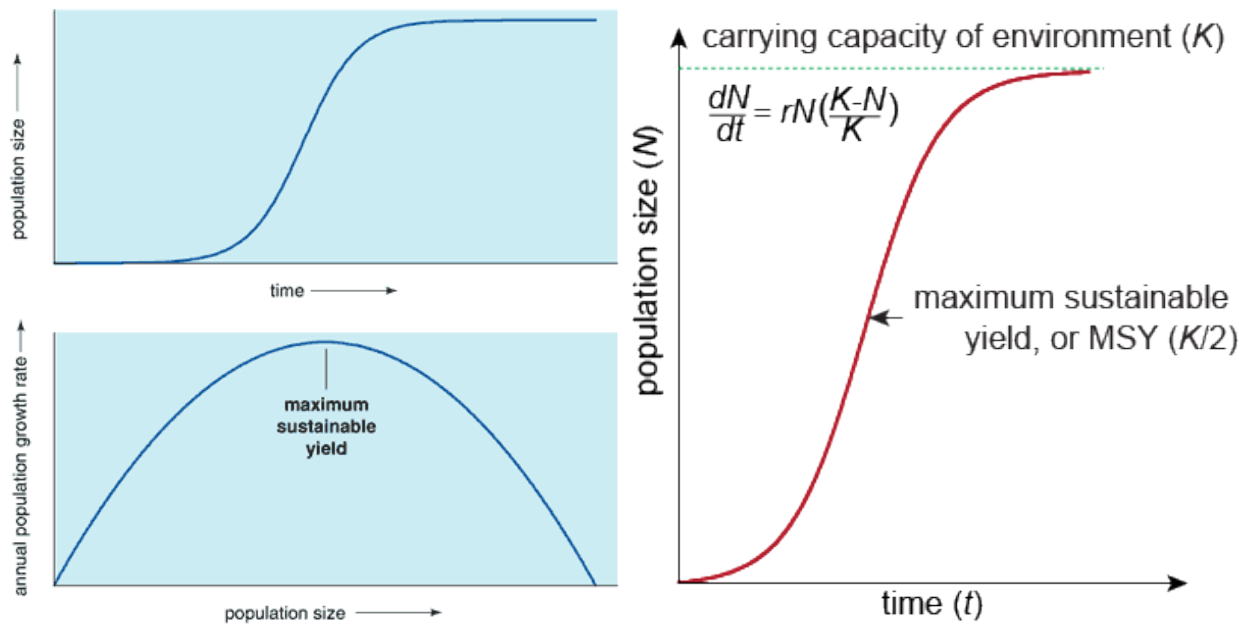
```
ggplot(log_pop, aes(x = year, y = N)) + # Plot the original data points (year vs. population)
  geom_point() +
  geom_line() +
  # use geom_hline to plot a horizontal line where the y-intercept is equal to K
  geom_hline(aes(yintercept = K), color = "red") + # outside of aes() sets color
  labs(x = "Year", y = "Population") +
  theme_bw()
```

Maximum Sustainable Yield

Once we have an estimate of the carrying capacity for a population, we can calculate something called the *maximum sustainable yield*, or MSY.

The MSY is the number of individuals that we can remove from the population and keep the population stable (or increasing). If we remove too many individuals (above MSY), the population will not be able to reproduce fast enough to replace the harvested individuals, causing a decline in the population—in this case, from over-fishing.

While the concept behind how we get to the equation for MSY may be a bit complex, thankfully, the equation itself is quite straightforward:

$$MSY = \frac{K}{2}$$

Let's calculate the maximum sustainable yield for population 4.

```
MSY <- K/2
MSY
```

```
## [1] 52.22863
```

This means that we can take about 52 fish from this population annually and keep the population stable!

If we take fewer than 52 fish, the population will increase (until the carrying capacity, that is). If we take more than 52 fish, however, we are over-fishing the population, meaning the population will not be able to reproduce enough to maintain the population and will eventually collapse.