European Pine Marten: Martes martes

Population dynamics in Denmark

Case study: Modeling population dynamics of pine marten in Denmark, by including factors like maximum population growth rate (r_max) and carrying capacity of the environment (K)

Type of organism: A small omnivorous mammal found in Northern Europe

Significance:
They play an important role in the balance of woodland ecosystems.

They are quite rare and thus are a protected species in several regions in Europe, as well as in Ireland and Scotland.

© Richard McManus/Getty

The Model and the Parameters

Logistic growth model based on sightings during 1985-2005

Data source: *Atlas of Danish Mammals*

Assumption:

Number of sightings = Size of population in that year

To apply the logistical growth model, it was assumed that the number of individuals sighted and reported by volunteers across all regions in Denmark were the actual population sizes in those regions for their respective years. All the regional counts for a year were summed up to get the total counts for whole Denmark.

Estimation of Parameters:

Diff_growth = Difference in counts between years

r = yearly growth rate = Diff_growth / totalIndividualCount

r_max = max pop growth rate (per year)

K = carrying capacity of the environment (no unit)

	year	totalIndividualCount
1	1985	8
2	1986	1
3	1987	3
4	1989	2
5	1990	2
6	1991	1
7	1992	4
8	1993	2
9	1994	8
10	1995	8
11	1996	3
12	1997	6
13	1998	19
14	1999	7
15	2000	21
16	2001	70
17	2002	92
18	2003	29
19	2004	7
20	2005	7

The Model and the Parameters

Logistic growth model based on sightings during 1985-2005

Estimation of Parameters:

Diff_growth = Difference in counts between years

r = Diff_growth / totalIndividualCount

r_max = 0.75 or 75% per year

K = carrying capacity of the environment (no unit)

Estimating K as the mean of the total counts of individuals over 20 years:

K = mean(totalIndividualCount) = 15

	year	totalIndividualCount	Diff_growth	Rate
1	1985	8	NA	NA
2	1986	1	-7	-7.0000000
3	1987	3	2	0.6666667
4	1989	2	-1	-0.5000000
5	1990	2	0	0.0000000
6	1991	1	-1	-1.0000000
7	1992	4	3	0.7500000
8	1993	2	-2	-1.0000000
9	1994	8	6	0.7500000
10	1995	8	0	0.0000000
11	1996	3	-5	-1.6666667
12	1997	6	3	0.5000000
13	1998	19	13	0.6842105
14	1999	7	-12	-1.7142857
15	2000	21	14	0.6666667
16	2001	70	49	0.7000000
17	2002	92	22	0.2391304
18	2003	29	-63	-2.1724138
19	2004	7	-22	-3.1428571
20	2005	7	0	0.0000000

Modeling Results

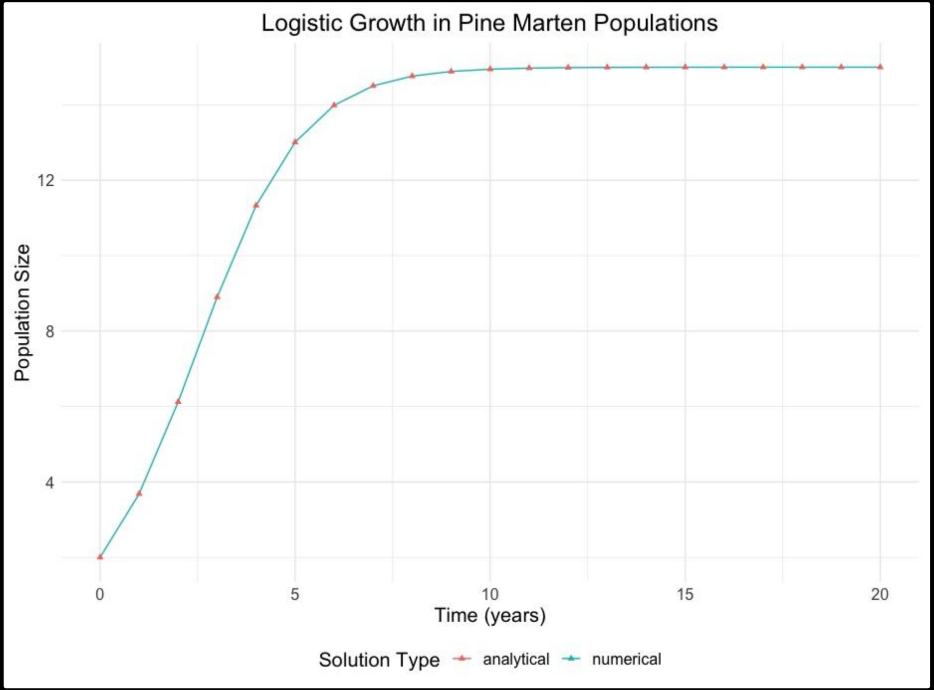
Logistic growth based on sightings during 1985-2005

Estimated from historical data:
Number of individuals (N) = 7 in 2005
r_max = 0.75 per year
K=15

Solving the logistic growth equation numerically:

$$\frac{dN}{dt} = r_{max} \times (1 - N/K) \times N$$
$$= 0.75 \times (1 - 7/15) \times 7$$
$$= 2.8 \text{ individuals per year}$$

The pine marten population in Denmark is estimated to be growing on average by 2.8 individuals per year since 2005.



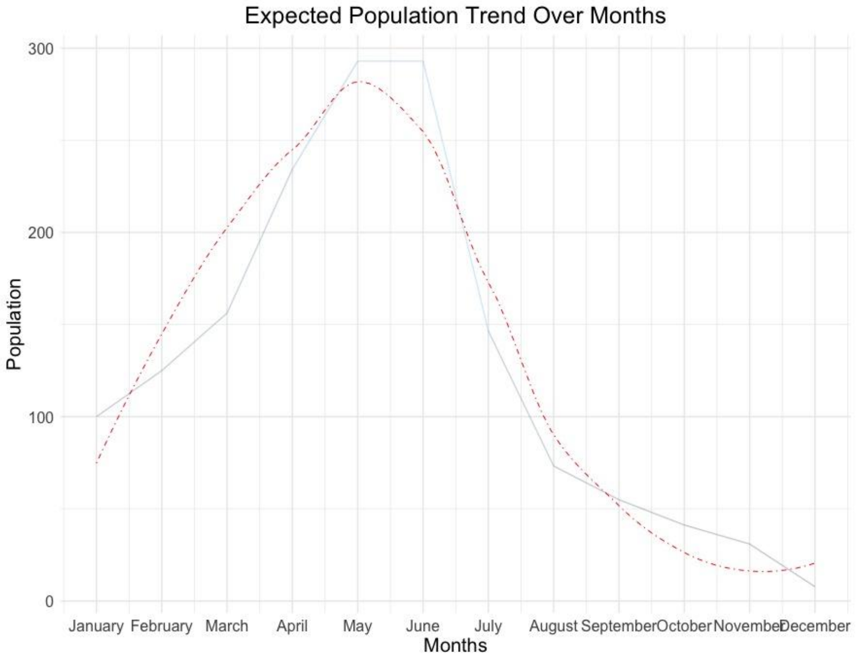
Modeling Results with Seasonal Variation

Variation due to births and deaths in a year

Mating months: July and August

Capable of delayed pregnancy: a remarkable adaptation that allows pine martens to give births during Spring instead of harsh winters, mainly from February to April, when they can protect and feed their kits well. Female kits mature in 2-3 years.

Deaths mainly happen due to hunting and human intervention, example, accidental killings. They have multiple predatory threats. Human activity and tourism during summer also adds to the rise in the number of deaths.



Modeling Results with Seasonal Variation

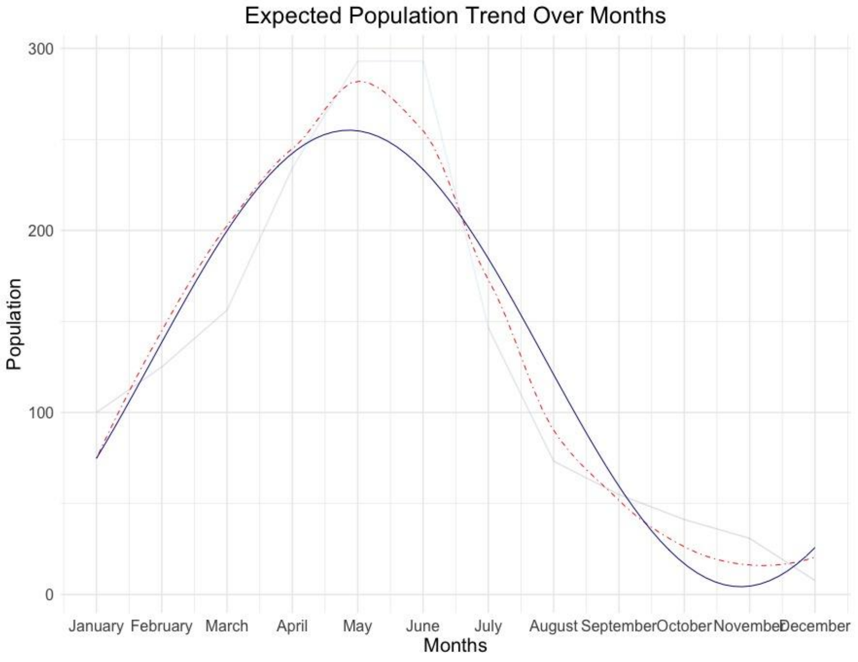
Variation due to births and deaths in a year

When variations are also visualized on the plot, it seems like the population dynamics for pine marten follow a sinusoidal trend.

With enough data, it could possibly completely follow a sinusoidal trend in terms of the rise and fall of actual population size.

So one can include a sinusoidal function in the logistic growth equation for pine martens by redefining the formula.

$$dN/dt = r_{max} \times (1 - N/K) \times N \times \sin(2 \times \pi \times (t - t_{peak})/12)$$



Modeling Results with Seasonal Variation

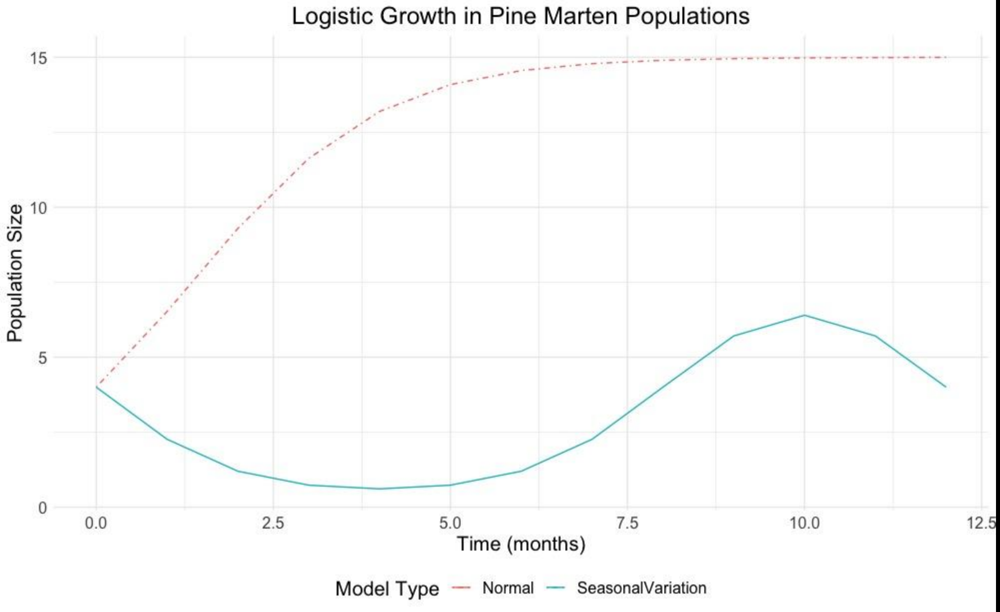
Variation due to births and deaths in a year

So one can include a sinusoidal function in the logistic growth equation for pine martens as follows.

dN/dt = r_max ✕ (1 – N/K) ✕ N ✕ sin(2✕pi✕(t – t_peak)/12)

where:

dN/dt = rate of change of the population size over time
r_max = maximum population growth rate
N = population size at a given time
K = carrying capacity = maximum population size the environment can sustain
t = time variable
t_peak = peak time of new births



Mathematical Biology

Weekly Exercise 1: Logistic Growth

Swati Tak

2023-09-10

```r
# Loading required libraries
library(knitr)
library(ggplot2)
library(readr)
library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(deSolve)
```

```r
# Calculation of maximum population growth rate r_max
occurrences <- read_tsv("occurrence.txt", col_names=TRUE)
```

```
## Rows: 57421 Columns: 79
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (18): id, license, bibliographicCitation, institutionCode, collectionCod...
## dbl  (9): datasetID, catalogNumber, individualCount, year, month, day, decim...
## lgl (52): references, institutionID, dynamicProperties, recordedBy, organism...
##
## 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.
```

```r
colnames(occurrences)
```

```
##  [1] "id"                                "license"
##  [3] "bibliographicCitation"             "references"
```

```
##  [5] "institutionID"                         "datasetID"
##  [7] "institutionCode"                       "collectionCode"
##  [9] "basisOfRecord"                         "dynamicProperties"
## [11] "occurrenceID"                          "catalogNumber"
## [13] "recordedBy"                            "individualCount"
## [15] "organismQuantity"                      "organismQuantityType"
## [17] "sex"                                   "lifeStage"
## [19] "establishmentMeans"                    "preparations"
## [21] "associatedMedia"                       "associatedReferences"
## [23] "associatedTaxa"                        "otherCatalogNumbers"
## [25] "occurrenceRemarks"                     "previousIdentifications"
## [27] "fieldNumber"                           "eventDate"
## [29] "startDayOfYear"                        "endDayOfYear"
## [31] "year"                                  "month"
## [33] "day"                                   "habitat"
## [35] "samplingProtocol"                      "fieldNotes"
## [37] "locationID"                            "continent"
## [39] "country"                               "stateProvince"
## [41] "county"                                "municipality"
## [43] "locality"                              "verbatimLocality"
## [45] "minimumElevationInMeters"              "maximumElevationInMeters"
## [47] "minimumDepthInMeters"                  "maximumDepthInMeters"
## [49] "minimumDistanceAboveSurfaceInMeters" "maximumDistanceAboveSurfaceInMeters"
## [51] "locationRemarks"                       "decimalLatitude"
## [53] "decimalLongitude"                      "geodeticDatum"
## [55] "coordinateUncertaintyInMeters"         "verbatimCoordinates"
## [57] "verbatimCoordinateSystem"              "georeferencedBy"
## [59] "georeferenceSources"                   "georeferenceRemarks"
## [61] "typeStatus"                            "identifiedBy"
## [63] "dateIdentified"                        "identificationRemarks"
## [65] "taxonID"                               "scientificNameID"
## [67] "scientificName"                        "namePublishedInYear"
## [69] "kingdom"                               "phylum"
## [71] "class"                                 "order"
## [73] "family"                                "genus"
## [75] "subgenus"                              "specificEpithet"
## [77] "infraspecificEpithet"                  "scientificNameAuthorship"
## [79] "vernacularName"
```

```r
# Filter the data for Lyra's Pantalaimon and group the number of sightings (counts) year wise
filteredData <- occurrences %>% select(catalogNumber, year, individualCount, scientificName) %>%
  filter(scientificName == "Martes martes") %>% group_by(year) %>%
  summarise(totalIndividualCount = sum(individualCount))

kable(filteredData)
```

year	totalIndividualCount
1985	8
1986	1
1987	3
1989	2
1990	2

year	totalIndividualCount
1991	1
1992	4
1993	2
1994	8
1995	8
1996	3
1997	6
1998	19
1999	7
2000	21
2001	70
2002	92
2003	29
2004	7
2005	7

```
# Estimating maximum growth rate using the data
growth_rate = filteredData %>%
  # first sort by year
  arrange(year) %>%
  mutate(Diff_growth = totalIndividualCount - lag(totalIndividualCount), # Difference in counts between
         Rate = Diff_growth/totalIndividualCount) # growth rate

kable(growth_rate)
```

year	totalIndividualCount	Diff_growth	Rate
1985	8	NA	NA
1986	1	-7	-7.0000000
1987	3	2	0.6666667
1989	2	-1	-0.5000000
1990	2	0	0.0000000
1991	1	-1	-1.0000000
1992	4	3	0.7500000
1993	2	-2	-1.0000000
1994	8	6	0.7500000
1995	8	0	0.0000000
1996	3	-5	-1.6666667
1997	6	3	0.5000000
1998	19	13	0.6842105
1999	7	-12	-1.7142857
2000	21	14	0.6666667
2001	70	49	0.7000000
2002	92	22	0.2391304
2003	29	-63	-2.1724138
2004	7	-22	-3.1428571
2005	7	0	0.0000000

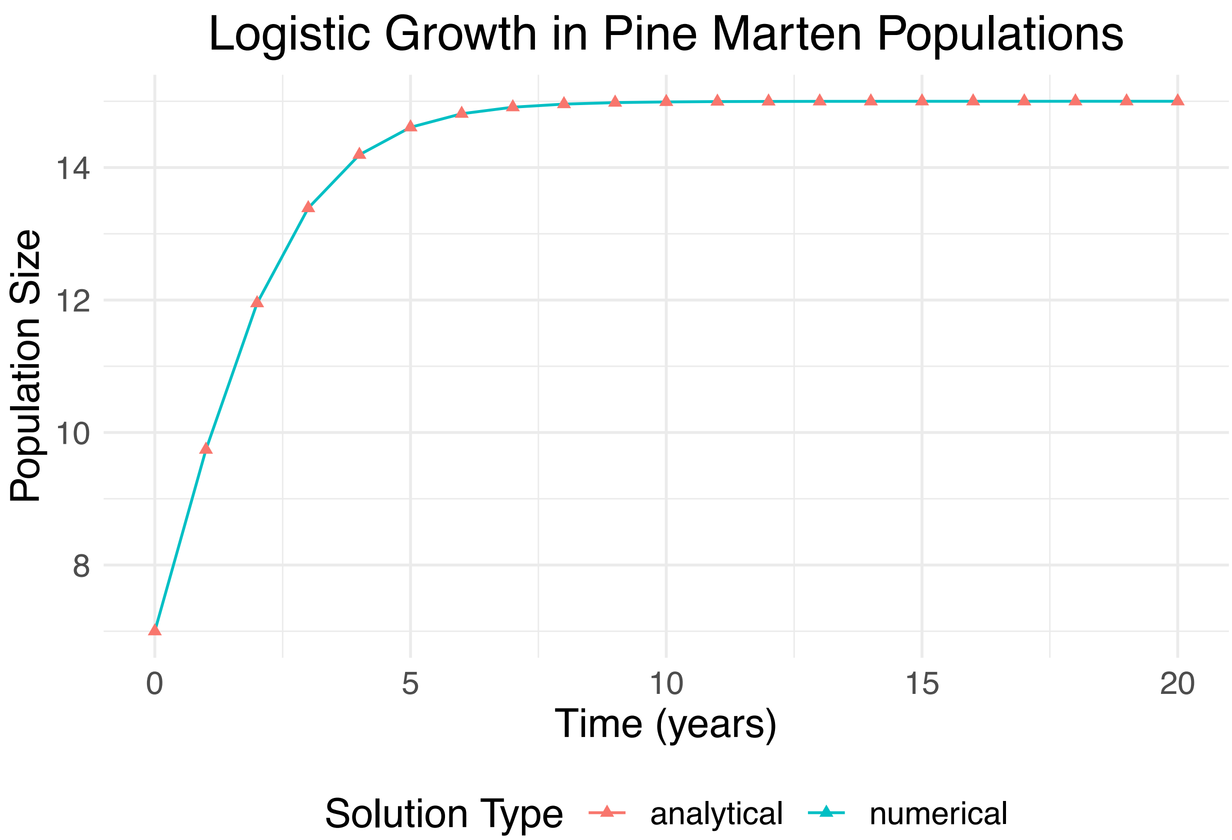
```
# Listing out all the calculated year-wise growth rates in an ascending order
growth_rate1 <- growth_rate %>%
```

```r
  arrange(Rate) %>%
  filter(Rate != "NA")

kable(growth_rate1)
```

year	totalIndividualCount	Diff_growth	Rate
1986	1	-7	-7.0000000
2004	7	-22	-3.1428571
2003	29	-63	-2.1724138
1999	7	-12	-1.7142857
1996	3	-5	-1.6666667
1991	1	-1	-1.0000000
1993	2	-2	-1.0000000
1989	2	-1	-0.5000000
1990	2	0	0.0000000
1995	8	0	0.0000000
2005	7	0	0.0000000
2002	92	22	0.2391304
1997	6	3	0.5000000
1987	3	2	0.6666667
2000	21	14	0.6666667
1998	19	13	0.6842105
2001	70	49	0.7000000
1992	4	3	0.7500000
1994	8	6	0.7500000

```r
# Estimated maximum growth rate is 0.75 (not actual)
r_max = 0.75 #As estimated for sightings in 1992 and 1994 in the Danish mammals atlas data

## Considering this is a non-linear phenomenon, K was decided to be as median value of counts
K = mean(growth_rate$totalIndividualCount)

# Final parameters
r_max = 0.75 # 0.75 per year
K = 15 # maximum number of individuals the environment can support, no unit

parameters <- c(r = r_max, K = K)
initial_conditions <- c(N = 7) # Random number of individuals at t=0
times <- seq(0, 20, 1)


# Writing the function for logistic growth
logistic_growth <- function(t, N, params) {
  with(as.list(c(N, params)), {
    dN <- r * N * (1 - N/K)
    list(c(dN))
  })
  }

out <- ode(y = initial_conditions, times = times, func = logistic_growth, parms = parameters)
```

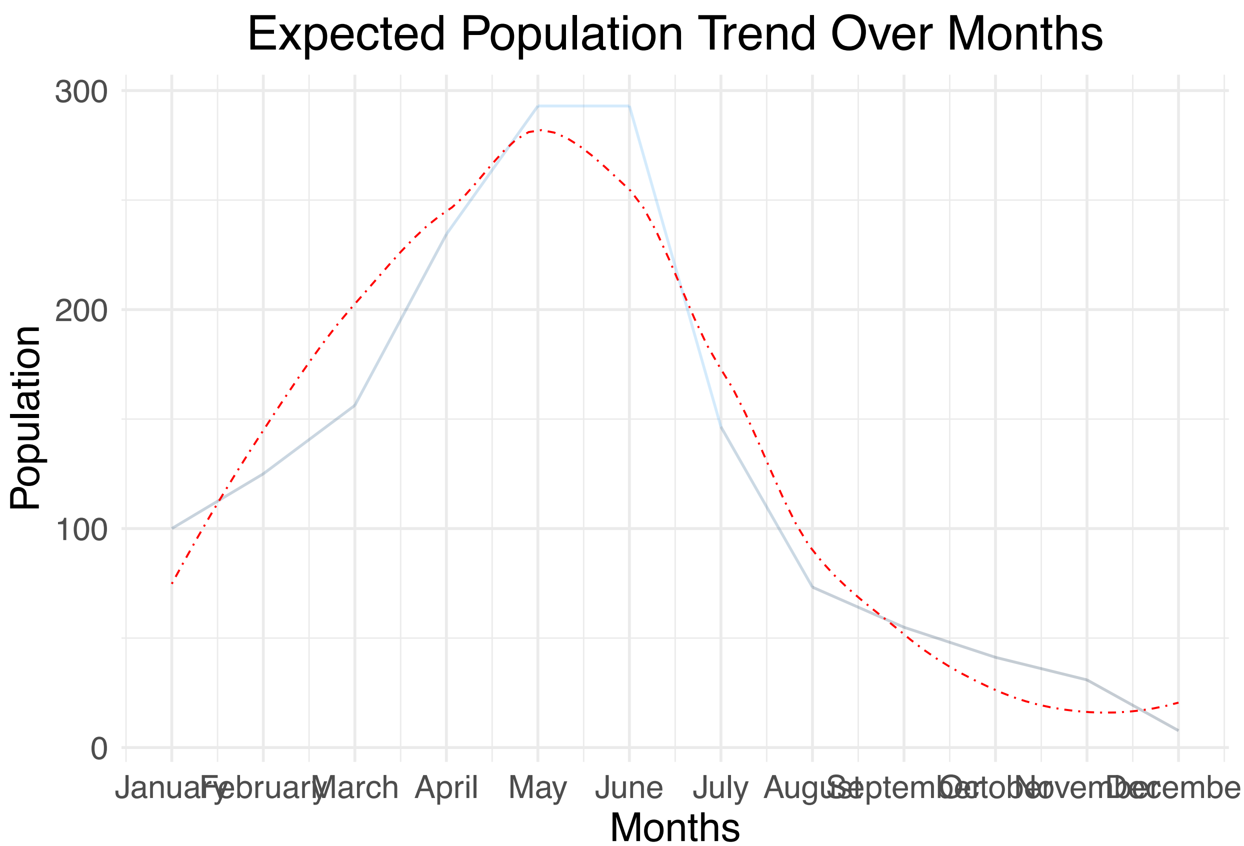
```r
# Analytical solution to the logistic growth equation
analytical_solution <- function(t, N, r, K) {
   N*K*exp(r*t)/(K + N*(exp(r*t)-1))
}

# Calculate the analytical solution
analytical_out <- analytical_solution(times, initial_conditions, r_max, K)

# Combine numerical and analytical solutions into a data frame
df <- data.frame(time = times, numerical = out[, 2], analytical = analytical_out)

# Transform data from wide to long format
df_long <- gather(df, key = "type", value = "population", -time)

# Create a ggplot
p <- ggplot(df_long, aes(x = time, y = population, color = type)) +
  geom_line(data = subset(df_long, type == "numerical"), linetype = "solid") +
  geom_point(data = subset(df_long, type == "analytical"), shape = 17) +
  labs(title = "Logistic Growth in Pine Marten Populations",
       x = "Time (years)",
       y = "Population Size",
       color = "Solution Type") +
  theme_minimal() +
  theme(text = element_text(size = 15),
        plot.title = element_text(hjust = 0.5),
        legend.position = "bottom")

print(p)
```




```r
# Explanation of seasonal variation in pine marten populations

# Create a vector of months
months <- c("January", "February", "March", "April", "May", "June", "July", "August", "September", "Octo

# Create a vector of reproduction rates corresponding to each month
reproduction_rates <- c(0.0, 0.5, 0.5, 0.75, 0.75, 0.5, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0)
death_rates <- c(0.25, 0.25, 0.25, 0.25, 0.5, 0.5, 0.5, 0.5, 0.25, 0.25, 0.25, 0.75)

# Simulate population growth over 1 year with seasonal variation
population <- rep(0, 12)
population[1] <- 100  # Initial population size

for (i in 2:12) {
  # Calculate the number of births based on the reproduction rate and the current population
  births <- reproduction_rates[i] * population[i-1]
  deaths <- death_rates[i] * population[i-1]

  # Update the population size
  population[i] <- population[i-1] + births - deaths
}

# Create a data frame
data <- data.frame(months, population)

# Create a ggplot object and specify the data and aesthetics
```

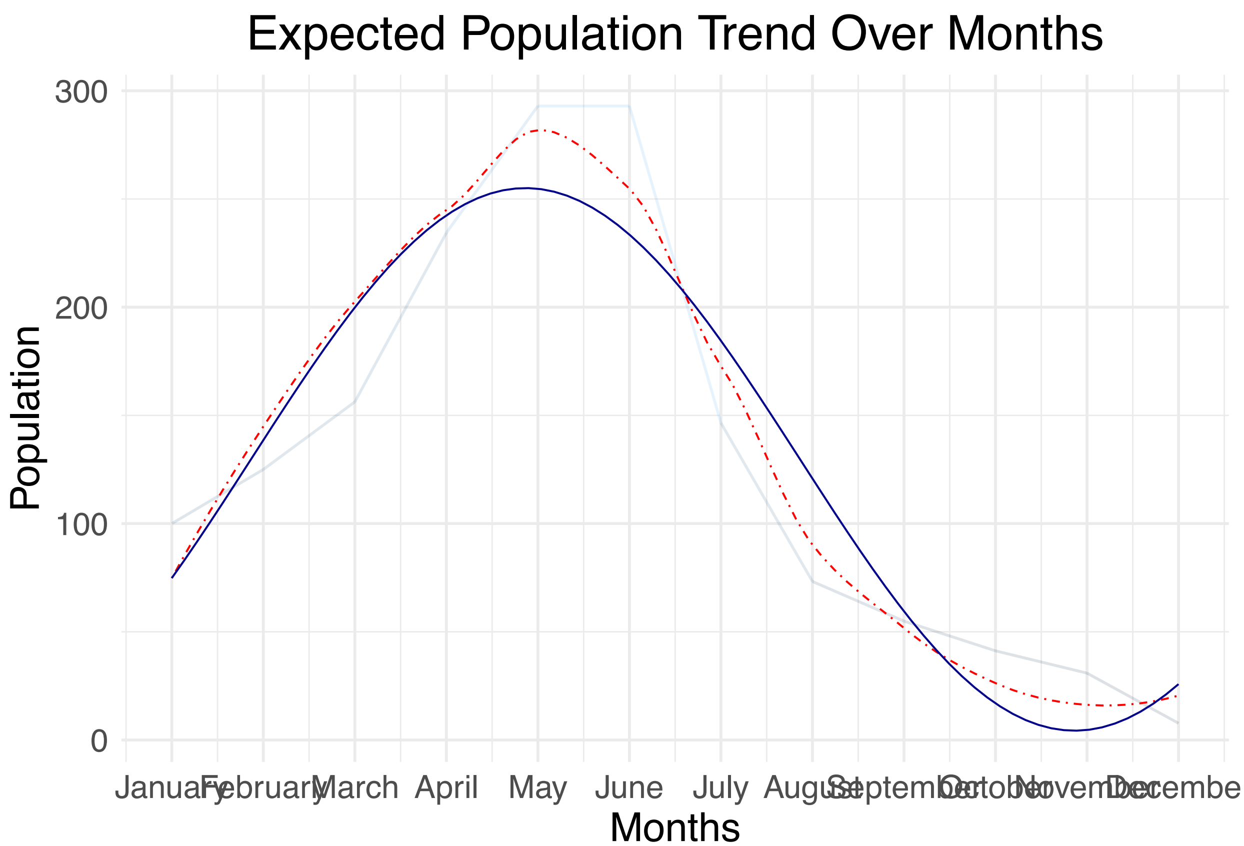
```r
# Convert months to a factor
data$months <- factor(data$months, levels = month.name)

# Create a data frame for the sinusoidal curve
sin_data <- data.frame(x = seq(min(as.numeric(data$months)), max(as.numeric(data$months)), length.out =
sin_data$y <- sin(sin_data$x)

# Create a vector of month names
month_labels <- c("January", "February", "March", "April", "May", "June", "July", "August", "September"

# Create a ggplot object and specify the data and aesthetics
ggplot(data, aes(x = as.numeric(months), y = population, color=population)) +
  # Add a line plot
  geom_line(alpha=0.25) +
  # Add x-axis and y-axis labels
  xlab("Months") +
  ylab("Population") +
  # Add a title
  ggtitle("Expected Population Trend Over Months") +
  # Specify the x-axis labels
  scale_x_continuous(breaks = 1:12, labels = month_labels) +
  # Add the sinusoidal curve as a separate layer
  geom_smooth(method="auto", formula= NULL, linetype=10, linewidth=0.35, se=FALSE, color="red") +
  theme_minimal() +
  theme(text = element_text(size = 15),
        plot.title = element_text(hjust = 0.5),
        legend.position = "none")
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



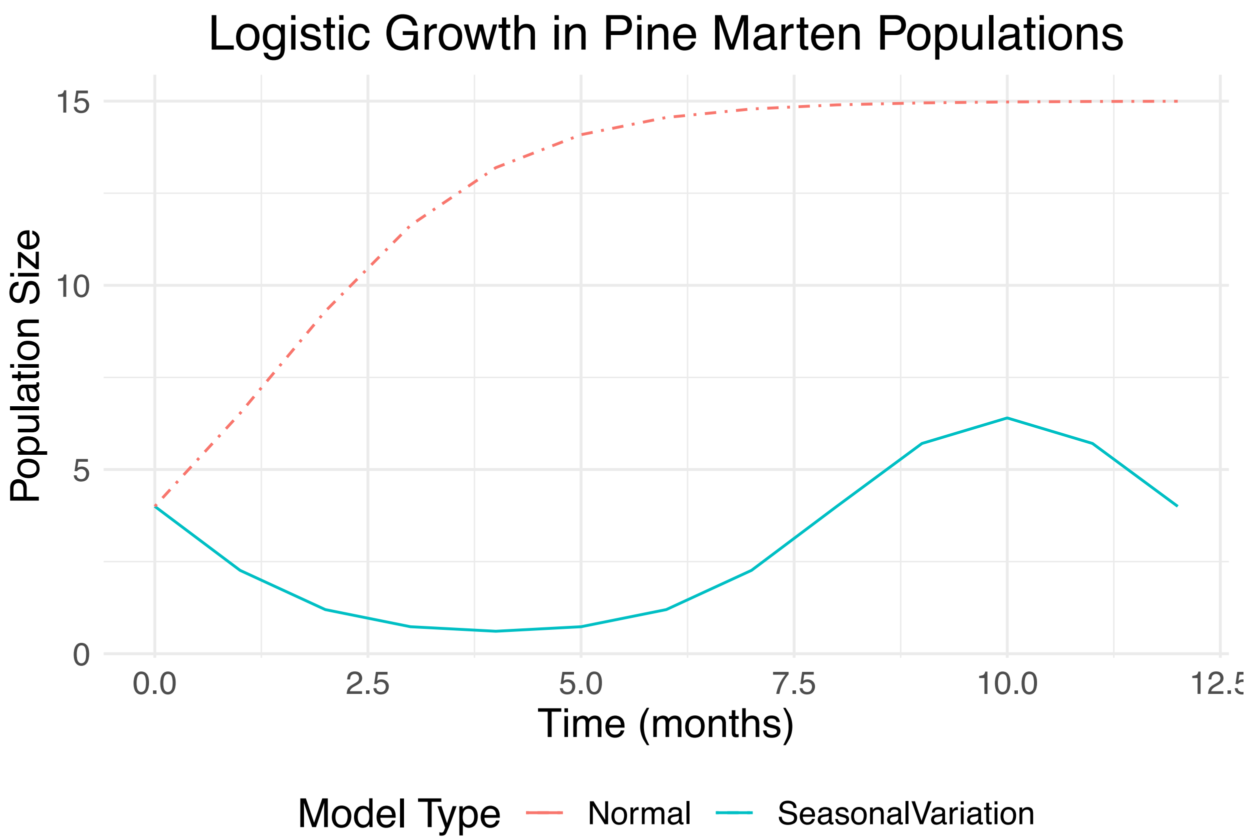

```r
# Create a ggplot object and specify the data and aesthetics
ggplot(data, aes(x = as.numeric(months), y = population, color=population)) +
  # Add a line plot
  geom_line(alpha=0.15) +
  # Add x-axis and y-axis labels
  xlab("Months") +
  ylab("Population") +
  # Add a title
  ggtitle("Expected Population Trend Over Months") +
  # Specify the x-axis labels
  scale_x_continuous(breaks = 1:12, labels = month_labels) +
  # Add the sinusoidal curve as a separate layer
  geom_smooth(method="auto", formula= NULL, linetype=10, linewidth=0.35, se=FALSE, color="red") +
  geom_smooth(method="lm", formula= y~cos(2*pi*x/12) + sin(2*pi*x/12), linetype=1, linewidth=0.35, se=F
  theme_minimal() +
  theme(text = element_text(size = 15),
        plot.title = element_text(hjust = 0.5),
        legend.position = "none")
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




```r
# Modeling the seasonal variation using sinusoidal function

# Writing the function for logistic growth with seasonal variation
logistic_growth <- function(t, N, params) {
  with(as.list(c(N, params)), {
    dN <- r_max * N * (1 - N/K) * sin(2*pi*(t-t_peak)/12)
    list(dN)
  })
}

# Writing the function for logistic growth without seasonal variation
logistic_growth2 <- function(t, N, params) {
  with(as.list(c(N, params)), {
    dN <- r * N * (1 - N/K)
    list(dN)
  })
}


# Define the parameters and initial conditions for population dynamics during an yearly cycle
r_max = 0.75
t_peak = 4  # peak growth rate in Spring around February-April
K = 15
times <- seq(0, 12, 1) # Assuming 12 months as the time range
```

```r
initial_conditions <- c(N = 4)
parameters <- c(r_max = r_max, t_peak = t_peak, K = K)
parameters2 <- c(r = r_max, K = K)


# Solve the logistic growth equation numerically with and without seasonal variations
out <- ode(y = initial_conditions, times = times, func = logistic_growth, parms = parameters)
out2 <- ode(y = initial_conditions, times = times, func = logistic_growth2, parms = parameters2)

# Combine numerical and analytical solutions into a data frame
df2 <- data.frame(time = times, Normal = out2[, 2], SeasonalVariation = out[, 2])

# Transform data from wide to long format
df_long2 <- pivot_longer(df2, cols = -time, names_to = "type", values_to = "population")

# Create a ggplot
p2 <- ggplot(df_long2, aes(x = time, y = population, color = type)) +
  geom_line(data = subset(df_long2, type == "SeasonalVariation"), linetype = "solid") +
  geom_line(data = subset(df_long2, type == "Normal"), linetype="dotdash") +
  labs(title = "Logistic Growth in Pine Marten Populations",
       x = "Time (months)",
       y = "Population Size",
       color = "Model Type") +
  theme_minimal() +
  theme(text = element_text(size = 15),
        plot.title = element_text(hjust = 0.5),
        legend.position = "bottom")

print(p2)
```



Logistic Growth in Pine Marten Populations
Population Size
15
10
5
0
0.0
2.5
5.0
7.5
10.0
12.5
Time (months)
Model Type
Normal
SeasonalVariation