A Primer of Ecosystem Modeling

Hank Stevens, BIO 672

2021-01-04

Contents

Chapter 1

Prerequisites

Chapter 2

Introduction to ecosystem modeling

In this course, we'll cover the very basics of ecosystem modeling. There are several goals I have for you, the reader. I hope the you,

- become more fluent in the discipline of ecosystem ecology; that you understand and can use basic terminology, and can identify quantitative pieces of the literature you read, and presentations you hear and see;
- understand and describe the quantitative and qualitative features of ecosystem dynamics and models of those dynamics;
- assess the relative merits of different modeling approaches and different mathematical formalisms of those apporaches;
- create models of ecosystem dynamics of your own;
- write R code to implement ecosystem models.

To do all this, the following text relies heavily on selected secondary sources including **?**] and **?**]. I also cite selected primary soruces werhe appropriate.

2.1 What's a model?

```
knitr::include_graphics("figs/LakeEcosystemCarpenteretal1992.png")
```

We model to aid understanding, because, at some level, the model is a quantitative and qualitative explanation for some phenomenon. We can use models to test hypotheses, guide experiments, and predict manage ecosystems and populations.

Statistical models (e.g., regression) are concerned with describing patterns and hypothesis testing. Process models (e.g., stock and flow models) are also descrip-
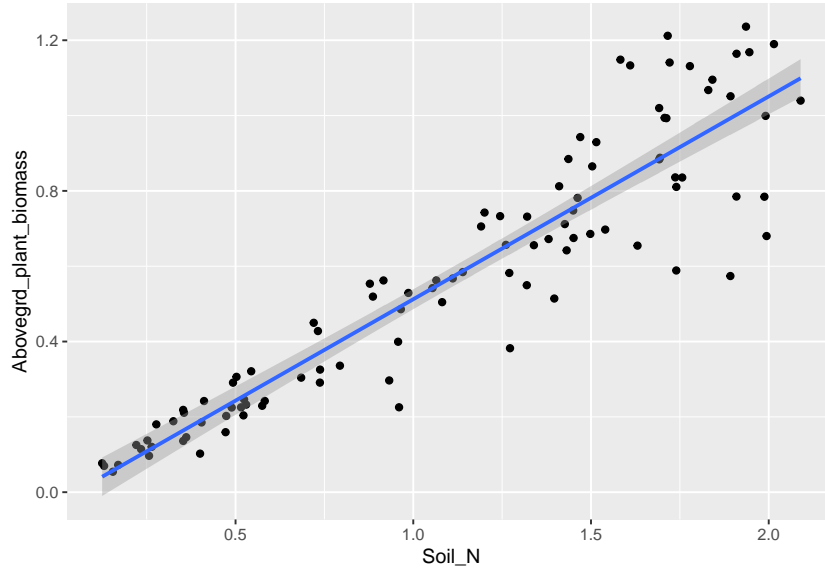
Figure 2.1: *A statistical model of aboveground plant biomass as a function of available soil nitrogen.*

tions of natural systems, but they include more mechanism and seek to describe mechanism and understand process. People sometimes call these *mechanistic* models.

A theory is a well supported hierarchical framework that contains clearly formulated postulates, based on a minimal set of assumptions, from which a set of predictions logically follows [**?**]. *Efficient* theory is based to as much as possible on first principles, and relies on as few assumptins as possible.

In contrast to theory, models are specific implementations of theory and specific descriptions of nature. Remember that in principle, *all models are wrong, but some are useful*[1].

2.2　Some useful terminology

- Domain and boundaries - the spatial, temporal and conceptual limits on a theory or model.
- Spatial and temporal scale
- State *factors* are forcing functions, constraints, and outside influences
- State *variables* are pools or stocks, have one currency (e.g., kg, individuals), vary through time; stuff whose variation we model.

---

[1]Box, G. E. P. (1979), "Robustness in the strategy of scientific model building", in Launer, R. L.; Wilkinson, G. N., Robustness in Statistics, Academic Press, pp. 201–236.
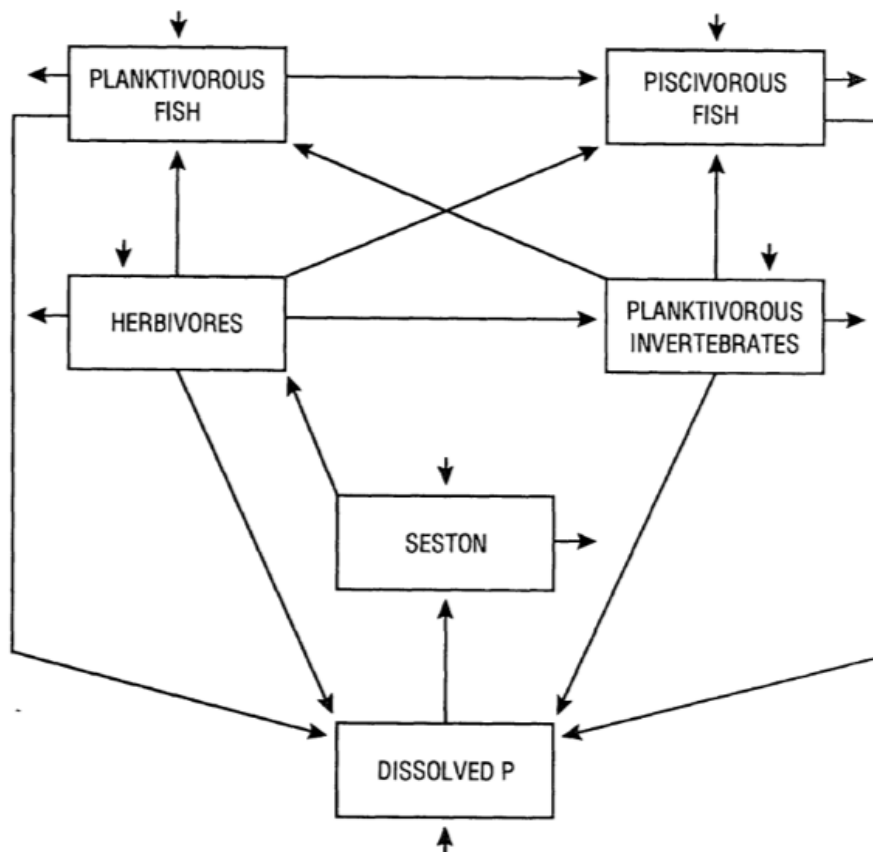
Figure 2.2: *Compartment models of other stuff*

- Flows = processes. They may be exchange between state variables (ecological, biogeochemical) such as ingestion, respiration, nutrient uptake; exchange with the external world we often refer to as transport, import, or export.
- Parameters are typically fixed rate constants governing process/flows

You might pause now, and ask yourself "is a lake a carbon sink or a source?" Draw an appropriate compartment model to address this question. After having done so, ask yourself what assumptions you've made about the temporal and spatial scales. What mechanisms have you included? Why?

2.3   Model making

Chapter 3

Describing a nitrogen budget

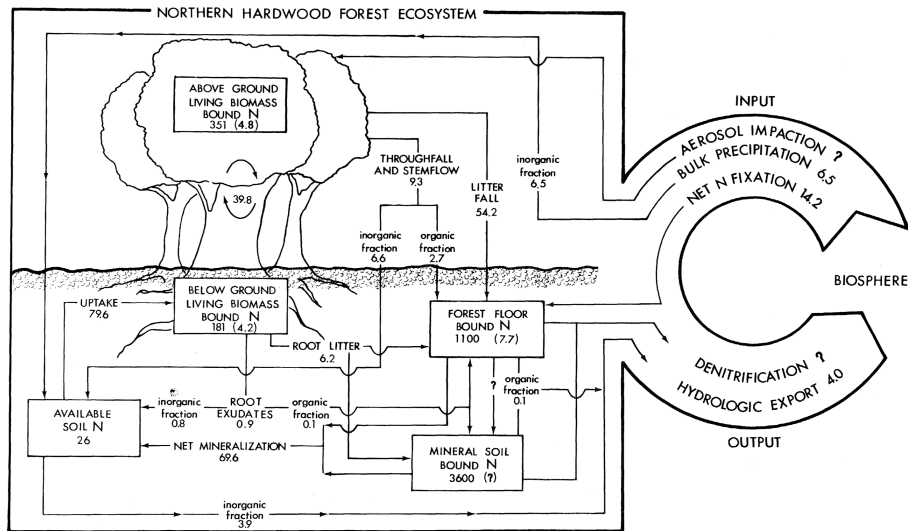Figure 3.1: *Nitrogen bugdget for a temperate northern hardwood forest (Hubbard Brook Watershed 6, Bormann et al. 1977).*

3.1 First mathematical form for plant uptake

A common starting point for dynamics depending on two pools is the law of *mass action*. This states that the reaction rate is proportional the product of the pools. In the case of plant uptake of N, which depends on the amounts of
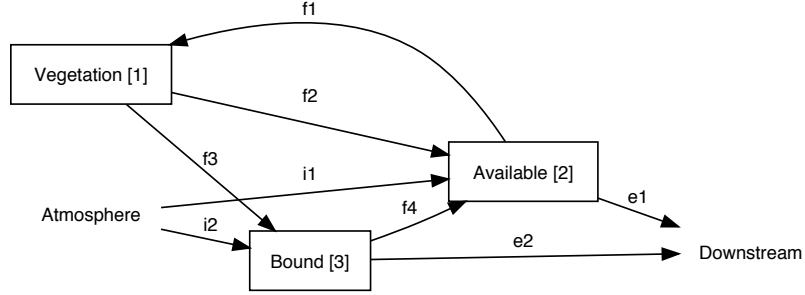
Figure 3.2: *A simpler compartment model for Hubbard Brook Watershed 6, based on Bormann et al. (1977).*

N in the available pool and the vegetation pool, this would be aVA, where a is a proportionality constant. In some circumstances, these pools might also have exponents different than one, or aVA^2. This occurs in chemistry when a reaction requires two molecules of "A" for each molecule of "V". It might occur in ecology if a rate depends differentially on A and B.

Use the law of mass action for plant uptake, we will describe the fluxes in the simple N budget above with the following expressions.

$$\frac{dV}{dt} = a_1AV - a_2V - a_3V \tag{3.1}$$

$$\frac{dA}{dt} = I_1 + a_2V + a_4B - a_5A - a_1AV \tag{3.2}$$

$$\frac{dB}{dt} = I_2 + a_3V - a_4B - a_6B \tag{3.3}$$

3.2   Paramaterization

Next, we find initial estimates for the parameters in our model. We use the literature for this purpose.

If we have the data (we do) and relatively simple mathematical forms (we do), it is fairly straightforward to estimate parameters. For instance, we decided that net mineralization would be directly proportional to the size of the organic pool, B, that is, $F_4 = a_4B$. To calculate a_4, we substitute data where we can,

Table 3.1: Parameters, variables, units and estimates for a simplified model of Bormann et al. (1977). All fluxes (dX/dy) are in units of $\mathrm{kg\,ha^{-1}\,y^{-1}}$. (Note that calculations should not be included in your final table, but are presented here for comparison to your own calculations.)

p, V	unit	Estimate
A, B, V state variables	$\mathrm{kg\,ha^{-1}}$	26, 4700, 532
a_1, uptake rate by V from A	$\mathrm{(kg\,ha^{-1})^{-1}\,y^{-1}}$	$79.6/(26 \cdot 532) = 0.0058$
a_2, loss rate from V to A	$\mathrm{y^{-1}}$	$(6.6 + 0.8)/532 = 0.014$
a_3, loss rate from V to B	$\mathrm{(kg\,ha^{-1})^{-1}\,y^{-1}}$	$(54.2 + 2.7 + 0.1 + 6.2)/532 = 0.00022$
a_4, mineralization	$\mathrm{y^{-1}}$	$69.6/4700 = 0.015$
a_5, export from A	$\mathrm{y^{-1}}$	$3.9/26 = 0.15$
a_6, export from B	$\mathrm{y^{-1}}$	$0.1/4700 = 0.000021$
I_1, bulk precip	$\mathrm{kg\,ha^{-1}\,y^{-1}}$	6.5
I_2, N fixation	$\mathrm{kg\,ha^{-1}\,y^{-1}}$	14.2

and solve what we need.

$$F_2 = a_4 B$$
$$69.6 = a_4 4700$$
$$a_4 = \frac{69.6}{4700}$$
$$a_4 \approx 0.015$$

We use the same approach for second order equations as well.

Enter parameters into R.

```r
# Atmospheric inputs
##  Precip
i1 <- 6.5
## fixation
i2 <- 14.2
# uptake
a1 <- 79.6 / (26 * 532)
# throughfall and exudates (inorganic)
a2 <- (6.6 + 0.8) / 532
# litter, throughfall, exudates (organic)
a3 <- (54.2 + 2.7 + 0.1 + 6.2 ) / 532
# net mineralization
a4 <- 69.6 / 4700
# export from available
a5 <- 3.9 /26
#export from bound
a6 <- 0.1/4700
```

```
# make a vector
params <- c( a1 = a1, a2 = a2, a3 = a3, a4 = a4,
             a5 = a5, a6 = a6, i1 = i1, i2 = i2)
#...and look at it.
params
```

```
##           a1           a2           a3           a4           a5           a6
## 5.754772e-03 1.390977e-02 1.187970e-01 1.480851e-02 1.500000e-01 2.127660e-05
##           i1           i2
## 6.500000e+00 1.420000e+01
```

3.3   Mathematical solution

The *mathematical solution* is the process of making the predictions using our model and our parameters. We *solve* the model. With simple models, we can sometimes find analytical solutions. For most ecosystem models, we have to solve the models numerically using *numerical integration*.

Here we write a function that includes our system of differential equations. This will allow R to integrate change through time.

```
bormann1 <- function(t, y, p) {
  # time, vector of state variables and parameters must be in this order
  # we can use as.list for both the state variables and parameters
  # a1 = uptake
  # a2 = loss from veg to avail
  # a3 = loss from veg to bound
  # a4 = net mineralization
  # a5 = export from avail
  # a6 = export from bound
  with( as.list( c(y, p) ), {
    dV.dt <- a1 * A * V - a2 * V - a3 * V
    dA.dt <- i1 + a2 * V + a4 * B - a1 * A * V - a5 * A
    dB.dt <- i2 + a3 * V - a4 * B - a6 * B
    # Here we return a list whose first element is the vector of
    # rates of change for the state variables. The first element must be these rates,
    # in the same order as the state variables in y
    # The second element is the total N in the system
    return(list( c(dV.dt, dA.dt, dB.dt),
                total = V + A + B
                ) )})
}
```

Now that we have the function, we tell R what to do with it. We will define the *initial state of the system*, and then tell R which time point we want it to return.

The initial state of the system is the set of starting values for the state variables. We could choose any values, but I select the values given in Bormann et al. (1977).

```
initial.state <- c( V = 532, A = 26, B = 4700)
```

3.4 Calibration, take 1

Calibration, in general, is the process of finding better estimates for our parameters. We can do that by looking into the literature, or by using independent data to estimates parameters directly.

If we take a look at the model output (above), we see both differences and similarities with Bormann et al. What are they?

What if we run it out longer, say, 100 years?

```
time <- 0:250
out <- ode(y = initial.state, times=time, func=bormann1, parms = params)
head(round(out,1))
```

```
##      time     V    A      B  total
## [1,]    0 532.0 26.0 4700.0 5258.0
## [2,]    1 540.8 25.8 4708.2 5274.7
## [3,]    2 548.8 25.5 4717.2 5291.5
## [4,]    3 556.0 25.2 4727.0 5308.3
## [5,]    4 562.6 25.0 4737.5 5325.1
## [6,]    5 568.6 24.8 4748.6 5342.0
```

```
#plot(out)
```

Use `gather()` and `ggplot()` to make a graph. We use `gather()` gather multiple columns into one with a new name (`value=kg.N`), keeping track of the names of the original columns in a new column (`key=State.var`). We can use `spread()` if we ever want to spread those columns back out.

```
outg <- gather(as.data.frame(out), key=State.var, value=kg.N,
               V, A, B, total, -time,
               factor_key=TRUE)
ggplot(outg, aes(x=time, y=kg.N)) + geom_line() + facet_wrap(~State.var, scale="free_y")
```

In some ways, we have been moderately successful in our first pass at converting a purely budgetary model into a dynamic process model. We mimicked total load, and see similar changes through time of all the state variables.

Questions to ponder

We replicated approximately the N budget of Bormann et al. (1977), but clearly vegetation cannot keep accumulating N indefinitely.
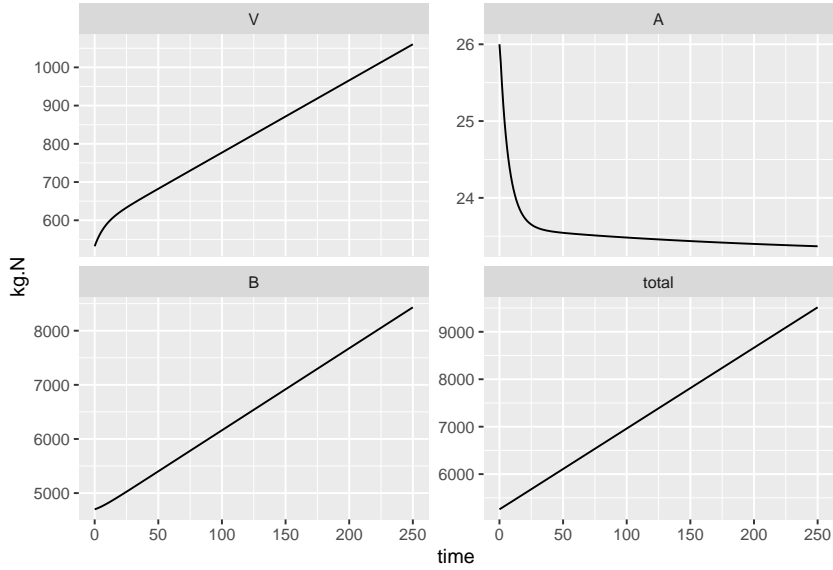
Figure 3.3: Dynamics of a simple N budget, based on Bormann et al. (1977).

What are our next steps? One logical step is to assume that as vegetation eventually gets limited by some factor or resource that is not in our model. If, at first approximation, the vegetation reaches a carrying capacity independent of high resource availability, we can use an approximation suggested by Soetaert and Hermann (2009) for *self-limitation*,

$$f(X)V\left(1 - \frac{V}{K}\right)$$

where $f(X)$ is everything else that regulates mass-specific growth rate.

Exercise

Include self-limitation in your model of vegetation, estimate K, and produce output.

3.5   Budget, take II

Remember that following our template, we have a maximum rate times resource and self limitation, and inhibition. Currently, we have

$$\frac{dV}{dt} = a_1 AV - a_2 V - a_3 V$$

and rearranging,

$$\frac{dV}{dt} = (a_1 A - a_2 - a_3) V$$

If we add self-limitation, we get

$$\frac{dV}{dt} = (a_1 A - a_2 - a_3) V \left(1 - \frac{V}{K}\right)$$

where K is the maximum amount of live vegetation that the ecosystem can sustain, in kg,N,ha^{-1}. We don't know exactly what that is yet, but we may be able to get estimates from the literature. For know we can pretend that it is just a bit more than was there in the mid-1970s, say, $K = 600$.

Now we rewrite the R function with self-limitation.

```r
bormann2 <- function(t, y, p) {
  # time, vector of state variables and parameters must be in this order
  # we can use as.list for both the state variables and parameters
  # a1 = uptake
  # a2 = loss from veg to avail
  # a3 = loss from veg to bound
  # a4 = net mineralization
  # a5 = export from avail
  # a6 = export from bound
  with( as.list( c(y, p) ), {
    dV.dt <- (a1 * A  - a2  - a3) * V * (1-V/K)
    dA.dt <- i1 + a2 * V + a4 * B - a1 * A * V - a5 * A
    dB.dt <- i2 + a3 * V - a4 * B - a6 * B
    # Here we return a list whose first element is the vector of
    # rates of change for the state variables. The first element must be these rates,
    # in the same order as the state variables in y
    # The second element is the total N in the system
    return(list( c(dV.dt, dA.dt, dB.dt),
                 total = V + A + B
                 ) )})
}
```

We can add a new parameter to our vector of parameters, and then solve our new function `bormann2` for the same time interval, and plot it.

```r
params["K"] <- 600
out <- ode(y = initial.state, times=time, func=bormann2, parms = params)
outg <- gather(as.data.frame(out), key=State.var, value=kg.N,
               V, A, B, total, -time,
               factor_key=TRUE)
ggplot(outg, aes(time, kg.N)) + geom_line() + facet_wrap(~State.var, scale="free_y")
```

Unlike our first model of this system, we see state variables on curved trajectories and perhaps reaching asymptotes. This makes greater intuitive sense - over the short term, it is the same as the simple N budget shown in **?**] and it also shows a reasonable longterm trajectory for the vegetation, and the predicted consequences for the available and bound pools.
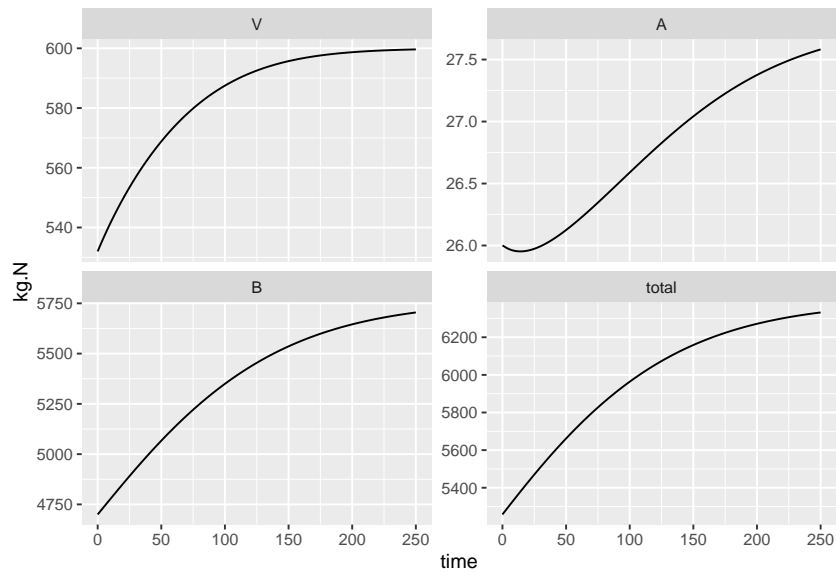
Figure 3.4: Dynamics of an N budget, assuming density-dependence in vegeta-tion with a fixed carrying capacity (Bormann et al. 1977).

Chapter 4

References

Bibliography