

Coupled predator-prey models and its application in investing in stock market

Aurél György Próz, XGRP0J

2018 október 9

Introduction

Predator-prey models are important topics not only in biology, but in other fields like plasma physics[<http://iopscience.iop.org/article/10.1088/0741-3335/56/1/015002>], economics[<https://www.sciencedirect.com/science/article/pii/0895717794900124>], or even in criminology [<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5043299/>].

The first, simplest model originated from the study of fish populations of the Mediterranean after the first world war, by Lotka and Volterra. In their model there are two types of species: the prey, and the predator. They form a simple food-chain where the predator species hunts the prey species, while the prey grazes vegetation. Their behavior and the time evolution of the number of species are characterized by a simple system of two, nonlinear first order differential equations. By solving these equations we can have insights on how the number of predators and preys evolve in time.

In this document I present the classical Lotka-Volterra (LV) equations and its modified versions. I also reproduce a financial model described in [], which can be used by a stock market trader to eliminate some of its risks involving the trading of specific stocks by buying the shares of so called prey companies, and sell them to a predator company. Last, but not least, I present a simple extension to this model, which can be used to model more than two prey companies.

Project goals

Implement and interpret the:

- Classical LV model
- Classical LV model at equilibrium
- LV model with prey hill function and LV model with predator efficiency
- LV model with two preys
- LV model used in stock market
- An extension to the LV model used in stock market

My own contribution to the topic: blablabla

Materials and methods

R programming language

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, ...) and graphical techniques, and is highly extensible.

DeSolve library

R package deSolve is the successor of R package odesolve, which is a package to solve initial value problems (IVP) of ordinary differential equations (ODE), differential algebraic equations (DAE), partial differential equations (PDE), and delay differential equations (DeDE). I used this package to solve the systems of nonlinear differential equations present in predator-prey models.

Classical LV model

This model describes the interaction of a predatory and a prey species. The predators try to catch the preys, thus increasing their population, and decreasing the prey's. In this model the prey species has unlimited resources to feed, and in this way the population of the prey species grows exponentially. We can summarize the whole model in a system of nonlinear differential equations: Equation Source of errors

```
library(deSolve)

## Warning: package 'deSolve' was built under R version 3.4.4

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.4.4

library(reshape2)

## Warning: package 'reshape2' was built under R version 3.4.4

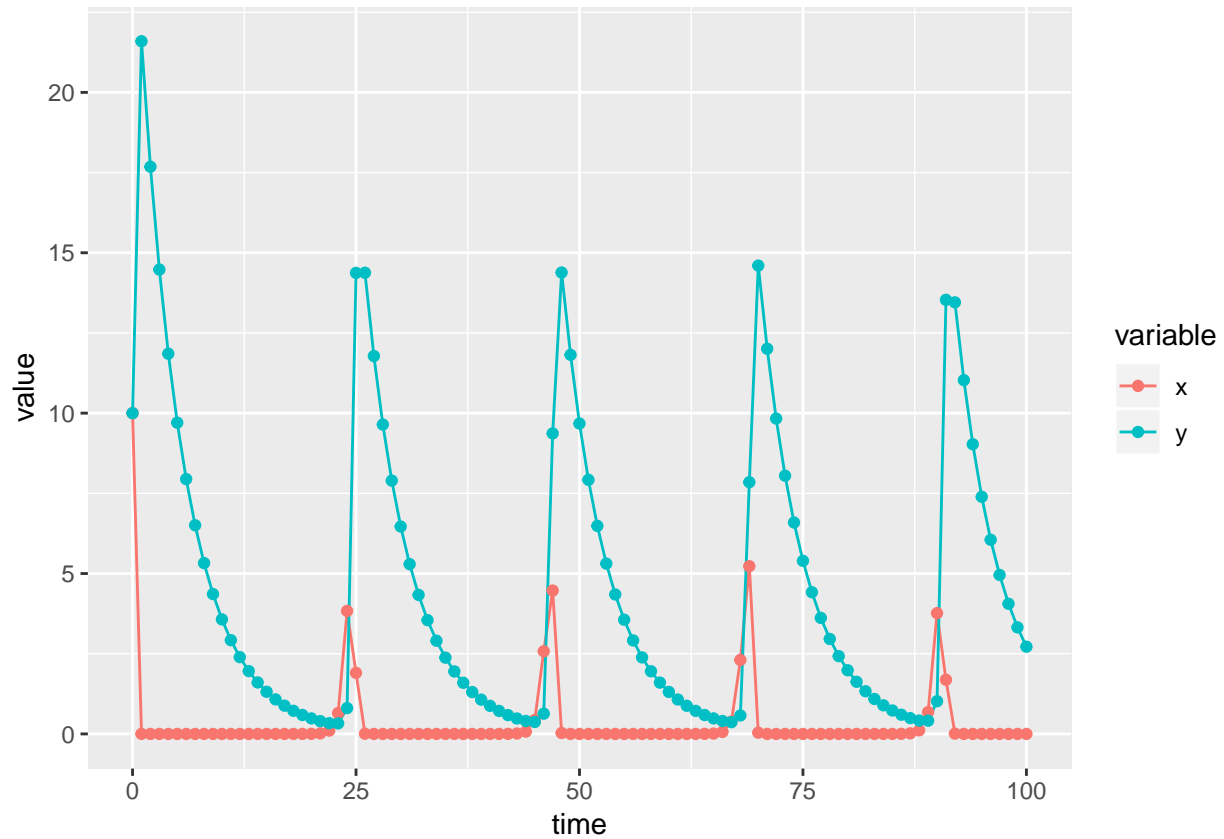
LotVmod <- function (Time, State, Pars) {
  with(as.list(c(State, Pars)), {
    dx = x*(alpha - beta*y)
    dy = -y*(gamma - delta*x)
    return(list(c(dx, dy)))
  })
}

Pars <- c(alpha = 2, beta = .5, gamma = .2, delta = .6)
State <- c(x = 10, y = 10)
Time <- seq(0, 100, by = 1)

out <- as.data.frame(ode(func = LotVmod, y = State, parms = Pars, times = Time))
out.r <- within(out, rm(time))
out.m <- melt(out.r)

## No id variables; using all as measure variables

out.m$time <- out$time
p<-ggplot(out.m, aes(x=time, y=value, group=variable)) +
  geom_line(aes(color=variable))+
  geom_point(aes(color=variable))
p
```

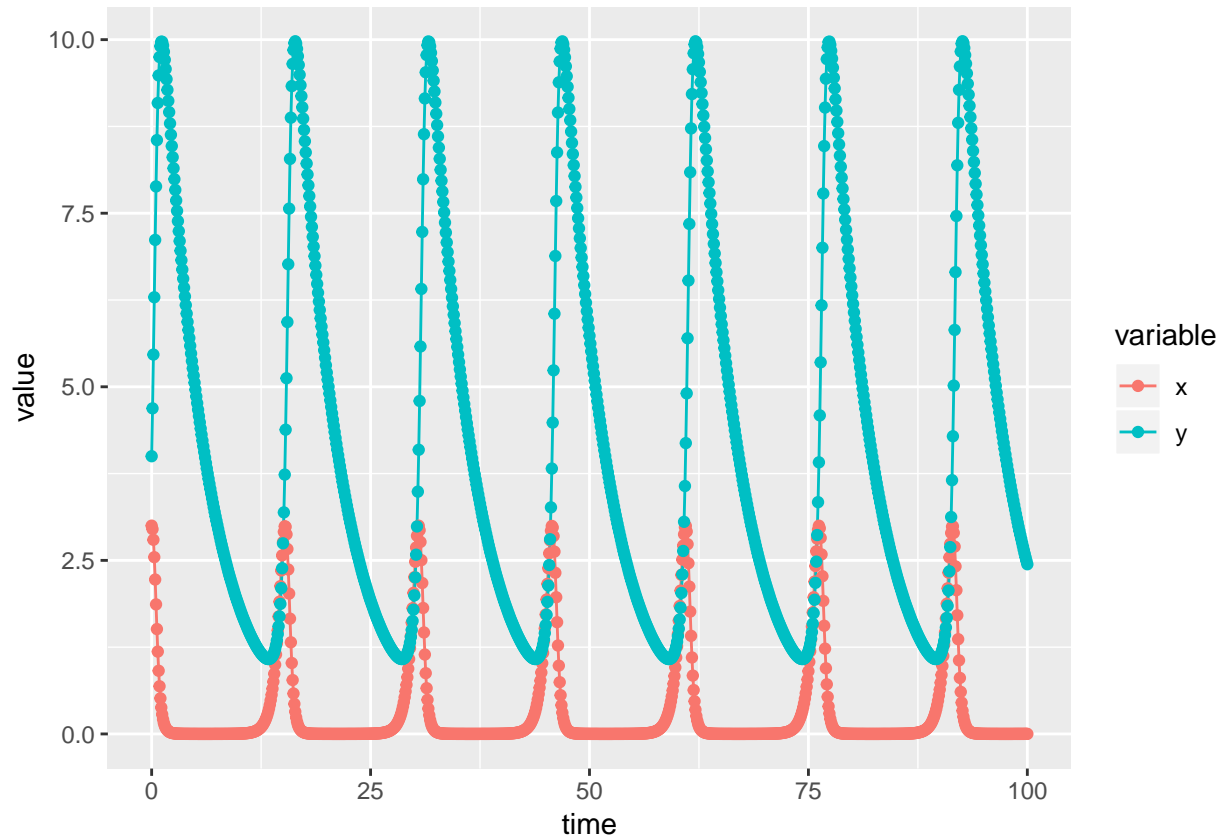


```
LotVmod <- function (Time, State, Pars) {
  with(as.list(c(State, Pars)), {
    dx = x*(alpha - beta*y)
    dy = -y*(gamma - delta*x)
    return(list(c(dx, dy)))
  })
}

Pars <- c(alpha = 2, beta = .5, gamma = .2, delta = .6)
State <- c(x = .6/.2, y = 2/.5)
Time <- seq(0, 100, by = 0.1)

out <- as.data.frame(ode(func = LotVmod, y = State, parms = Pars, times = Time))
out.r <- within(out, rm(time))
out.m <- melt(out.r)

## No id variables; using all as measure variables
out.m$time <- out$time
p<-ggplot(out.m, aes(x=time, y=value, group=variable)) +
  geom_line(aes(color=variable))+
  geom_point(aes(color=variable))
p
```



Pictures ## Derivation of steady state

LV model with prey limit and predator efficiency

This model extends the classical LV model in such a way, that it includes a so called K carrying capacity for the prey population, and a functional response α , as the probability a predator finds one prey. These two parameters makes the model more realistic, as in reality the resources available for the preys are not unlimited, and the hunt by the predators are not always successful, and takes time. Equation Source of errors Pictures # Two preys, one predator LV model

This modification of the original LV model includes the extensions mentioned above, and also includes one more prey species. In reality there can be preys which can be hunted down more easily by the predator, and could have different carrying capacity. These biological information are included in the parameters of this model. Equation Source of errors Pictures

Stock market application