Reverse Engineering Predator-Prey System

The predator-prey or Lotka-Volterra equations, as described below, consist of two differential equations modeling the interaction between two species in a closed system. Due to it's simplicity and applications, the Lotka-Volterra equations have been extensively studied and used to model several biological systems.

$$\frac{dx}{dt} = \alpha x - \beta xy$$

$$\frac{dy}{dt} = \delta xy - \gamma y$$

In biology, specially *Computational Biology*, one of the challenges is to understand the underlying dynamic or behavior of a system by only looking at limited data-point collected from experiments. In some cases, the prior knowledge about the system provide a very good starting point, e.g. knowing the underlying mathematical model. If the underlying model is known, then the question is to find the parameters of the model in such a way that describes the experimental data in the best way possible. This process is called *Reverse-engineering*.

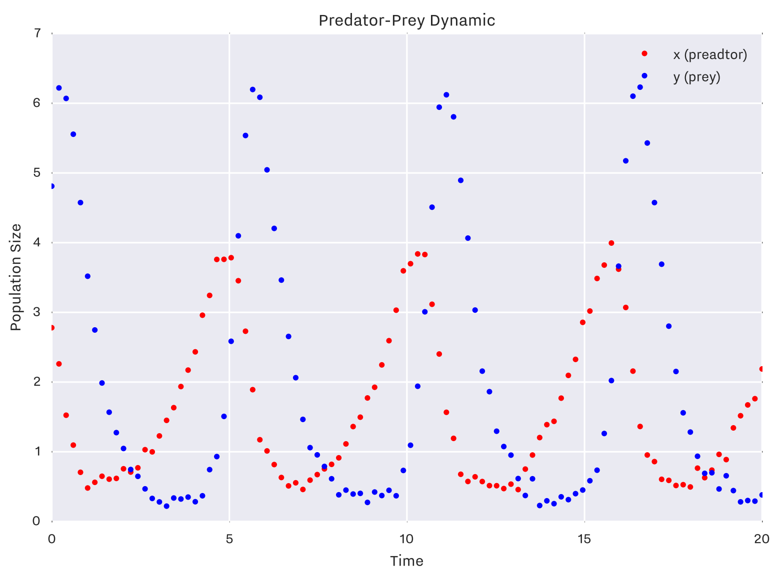

Figure 1: Predator-Prey Data

In this exercise, we are asking you to reverse-engineer a predator-prey model. The file `predator-prey-data.csv` consists of three columns: time, predator, prey. Given the fact that the data are generated from Lotka-Volterra equations (with added gaussian noise), try to find the parameters of the model using optimization algorithms.

- Give a brief introduction about Lotka-Volterra equations
  - Good understanding of the model helps you to design your optimization procedure more efficiently; e.g. *parameters range, time-scale, etc.*
- Construct at least two objective functions and explain their effects on optimization process
  - An objective function in curve fitting problems should measure the difference between the simulated results and experimental results.
- Use at least two optimization algorithms, a local optimization method, e.g. Hill Climbing; and a global optimization method, e.g. Simulated Annealing.
- Experiment with the number of data-points necessary for getting accurate result.
  - How many data-points from each time-series you could remove until you are not able to reverse-engineer the parameters any more?
    * First, try to find the importance of each time-series on reverse-engineering process by fixing one time-series and removing data-points from the other one. In this way, you will find the critical number of data-points for each time-series.
    * Then, if you combine two reduced time-series, could you still infer the parameters?
    * Does removing every data-point have the same effect on reverse-engineering process? In other words, what are the points that you could remove safely without affecting the inferance procedure? and what are those with critical effects on the process?