

## Praktikum – Lotka Volterra

Kode Python ([lotka\\_volterra.py](#)) yang dibuat oleh Sandy digunakan untuk mensimulasikan populasi mangsa (rusa) dan pemangsa (serigala) dengan model matematika Lotka-Volterra.

### Model Lotka-Volterra

#### Variabel yang digunakan:

- $x$  = jumlah mangsa (misal rusa)
- $y$  = jumlah pemangsa (misal serigala)

#### Model Lotka-Volterra sebagai berikut:

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

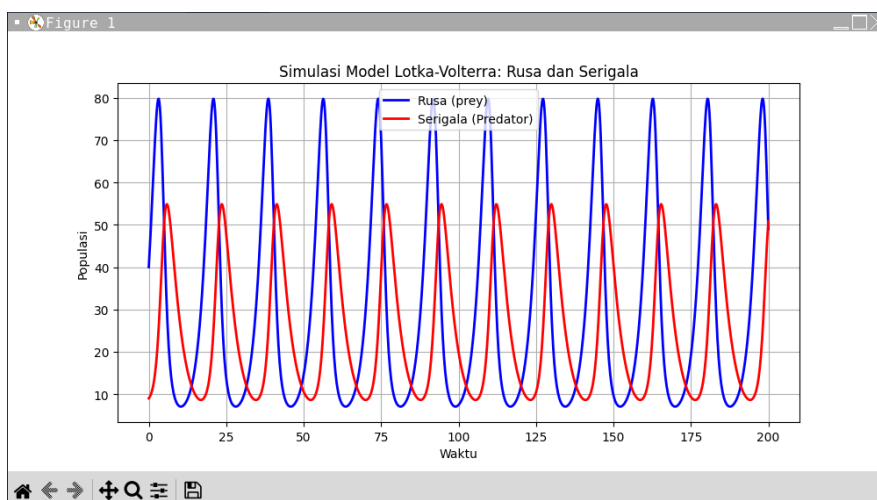
$$\frac{dy}{dt} = \delta xy - \gamma y$$

#### Parameter:

- Alpha ( $\alpha$ ): Tingkat pertumbuhan alami rusa (tanpa kehadiran serigala)
- Beta ( $\beta$ ): Tingkat perburuan rusa oleh serigala
- Delta ( $\delta$ ): Efisiensi konversi rusa yang dimakan menjadi pertumbuhan populasi serigala
- Gamma ( $\gamma$ ): Tingkat kematian alami serigala (karena kelaparan jika tidak ada rusa)

*Catatan Presentasi “Lotka-Volterra” FLEARN*

Output yang dihasilkan dalam menjalankan file `lotka_volterra.py` milik Sandy:



Grafik garis yang dihasilkan menunjukkan adanya naik turun dari masing-masing variabel terhadap waktu. Jumlah rusa yang bermula banyak akan menurun saat serigala memulai memangsa. Jumlah serigala yang bermula sedikit akan naik saat populasi rusa banyak. Siklus ini terjadi terus menerus ketika terdapat solusi setimbang atau *equilibrium point* selain saat 0.

### Penjelasan kode lotka\_volterra Arsandy:

#### 1. Mendefinisi Parameter yang digunakan dalam model:

- $\alpha = 0.5$  adalah Tingkat pertumbuhan alami rusa
- $\beta = 0.02$  adalah Tingkat perburuan rusa oleh serigala
- $\delta = 0.01$  adalah efisiensi konversi rusa yang dimakan menjadi pertumbuhan populasi serigala
- $\gamma = 0.3$  adalah Tingkat kematian alami serigala

#### 2. Membuat fungsi Lotka-Volterra:

```
def lotka_volterra(y,t,alpha,beta,delta,gamma):  
    x, y = y  
    dxdt = alpha*x - beta*x*y  
    dydt = delta*x*y - gamma*y  
    return [dxdt, dydt]
```

Fungsi tersebut digunakan untuk menyelesaikan persamaan diferensial biasa orde 2 dengan mengonversi ke orde 1 terlebih dahulu ( $\frac{dx}{dt}$  dan  $\frac{dy}{dt}$ ).

#### 3. Menetapkan kondisi awal:

- $y_0 = [40,9]$  adalah populasi awal rusa dan serigata secara berurutan

#### 4. Mendefinisi jangka waktu:

- $t = \text{np.linspace}(0, 200, 1000)$  membuat array/larik dengan jangka 0 hingga 200 sebanyak 1000 titik

#### 5. Menyelesaikan sistem persamaan Lotka-Volterra:

```
solution = odeint(lotka_volterra, y0, t, args=(alpha, beta, delta, gamma))
```

menggunakan fungsi [odeint](#) yang terdapat pada library SciPy untuk memudahkan menyelesaikan sistem persamaan diatas.

```
rusa, serigala = solution.T
```

baris tersebut memisahkan hasil penyelesaian diatas dengan metode transpose.

#### 6. Membuat grafik:

- Grafik garis perbandingan jumlah dengan waktu dengan dua variabel (rusa dan serigala).