

Отчёт по лабораторной работе 5

дисциплина: Математическое моделирование

Савченков Д.А., НПИбд-02-18

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	12

List of Tables

List of Figures

3.1	Колебания изменения числа популяции хищников и жертв	10
3.2	Зависимость изменения численности хищников от изменения численности жертв	11

1 Цель работы

Построить модель Лотки-Вольтерры типа “хищник – жертва” с помощью Python.

2 Задание

Вариант 38

Для модели «хищник-жертва»:

$$\begin{cases} \frac{\partial x}{\partial t} = -0.7x(t) + 0.06x(t)y(t) \\ \frac{\partial y}{\partial t} = 0.6y(t) - 0.07x(t)y(t) \end{cases}$$

Постройте график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при следующих начальных условиях: $x_0 = 8$, $y_0 = 15$. Найдите стационарное состояние системы.

3 Выполнение лабораторной работы

1. Полагаем для этой модели, что x – число жертв, а y – число хищников. Изучил начальные условия. Коэффициент 0,7 описывает скорость естественного прироста числа жертв в отсутствие хищников, 0,6 – естественное вымирание хищников, лишенных пищи в виде жертв. Вероятность взаимодействия жертвы и хищника считается пропорциональной как количеству жертв, так и числу самих хищников (xy). Каждый акт взаимодействия уменьшает популяцию жертв, но способствует увеличению популяции хищников (коэффициенты 0,06 и 0,07). Стационарное состояние будет в точке: $x_0 = 8, y_0 = 15$.

2. Оформил начальные условия в код на Python:

```
x0 = [8, 15]
```

```
a = 0.7
```

```
b = 0.06
```

```
c = 0.6
```

```
d = 0.07
```

3. Решение для колебаний изменения числа популяции хищников и жертв искал на интервале $t \in [0; 100]$ (шаг 0,1), значит, $t_0 = 0$ – начальный момент времени, $t_{max} = 37$ – предельный момент времени, $dt = 0,05$ – шаг изменения времени.
4. Добавил в программу условия, описывающие время:

```

t0 = 0
tmax = 100
dt = 0.1
t = np.arange(t0, tmax, dt)

```

5. Запрограммировал заданную систему уравнений:

```

def S(x, t):
    dx0 = -a*x[0] + b*x[0]*x[1]
    dx1 = c*x[1] - d*x[0]*x[1]
    return dx0, dx1

```

6. Запрограммировал решение системы уравнений:

```

y = odeint(S, x0, t)

```

7. Переписал отдельно x (жертв) в y_1 , а y (хищников) в y_2 :

```

y1 = y[:,0]
y2 = y[:,1]

```

8. Описал построение графика колебаний изменения числа популяции хищников и жертв:

```

plt.plot(t, y1, label='Хищники')
plt.plot(t, y2, label='Жертвы')
plt.legend()
plt.grid(axis = 'both')

```

9. Описал построение графика зависимости изменения численности хищников от изменения численности жертв:

```

plt.plot(y1, y2)
plt.grid(axis = 'both')

```


10. Добавил на второй график обозначение стационарного состояния:

```
plt.plot(x0[0], x0[1], 'ro')
```

11. Собрал код программы воедино и получил следующее:

```
import math
import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plt
```

```
x0 = [8, 15]
```

```
a = 0.7
```

```
b = 0.06
```

```
c = 0.6
```

```
d = 0.07
```

```
t0 = 0
```

```
tmax = 100
```

```
dt = 0.1
```

```
t = np.arange(t0, tmax, dt)
```

```
def S(x, t):
```

```
    dx0 = -a*x[0] + b*x[0]*x[1]
```

```
    dx1 = c*x[1] - d*x[0]*x[1]
```

```
    return dx0, dx1
```

```
y = odeint(S, x0, t)
```

```
y1 = y[:,0]
```

```
y2 = y[:,1]
```

```
plt.plot(t, y1, label='Хищники')
```

```
plt.plot(t, y2, label='Жертвы')
```

```
plt.legend()
```

```
plt.grid(axis = 'both')
```

```
plt.plot(y1, y2)
```

```
plt.plot(x0[0], x0[1], 'ro')
```

```
plt.grid(axis = 'both')
```

12. Получил графики колебаний изменения числа популяции хищников и жертв (см. рис. 3.1), а также график зависимости изменения численности хищников от изменения численности жертв (см. рис. 3.2):

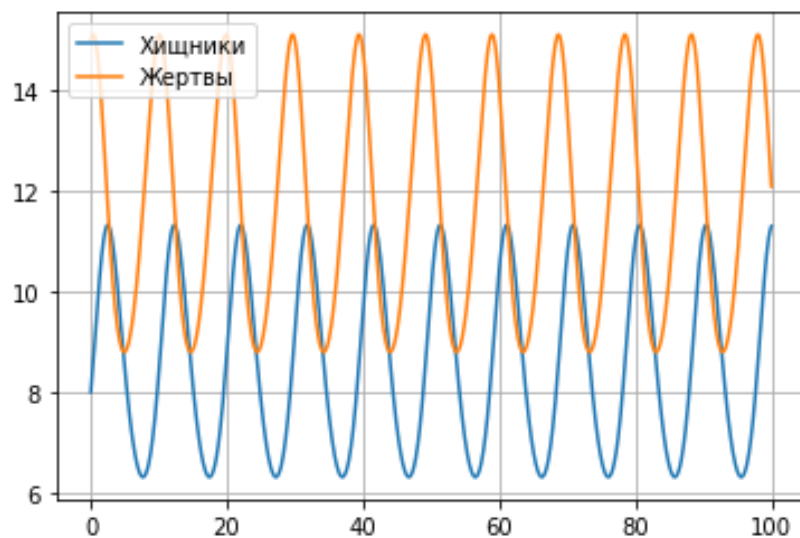


Figure 3.1: Колебания изменения числа популяции хищников и жертв

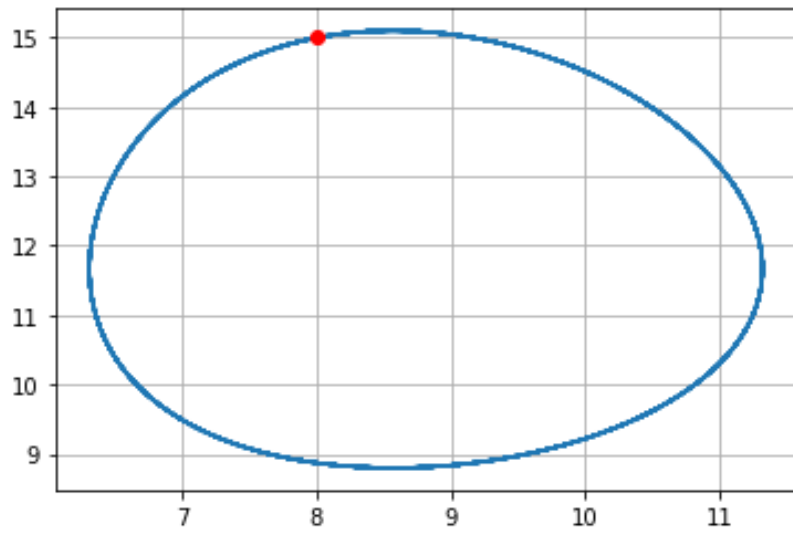


Figure 3.2: Зависимость изменения численности хищников от изменения численности жертв

4 Выводы

Построил модель Лотки-Вольтерры типа “хищник – жертва” с помощью Python.