Отчёт по лабораторной работе 6

дисциплина: Математическое моделирование

Никитаева А. С., НПИбд-02-18

Содержание

List of Tables

List of Figures

1 Цель работы

Построить простейшую модель эпидемии с помощью Python.

2 Задание

Вариант 18

На одном острове вспыхнула эпидемия. Известно, что из всех проживающих на острове ($N = 10400$) в момент начала эпидемии ($t = 0$) число заболевших людей (являющихся распространителями инфекции) $I(0) = 144$, а число здоровых людей с иммунитетом к болезни $R(0) = 28$. Таким образом, число людей восприимчивых к болезни, но пока здоровых, в начальный момент времени $S(0) = N - I(0) - R(0)$.

Постройте графики изменения числа особей в каждой из трех групп. Рассмотрите, как будет протекать эпидемия в случае:

1) если $I(0) \leq I^*$
2) если $I(0) > I^*$

3 Теоретическое введение

Предположим, что некая популяция, состоящая из N особей, (считаем, что популяция изолирована) подразделяется на три группы:

- $S(t)$ — восприимчивые к болезни, но пока здоровые особи;
- $I(t)$ — это число инфицированных особей, которые также при этом являются распространителями инфекции;
- $R(t)$ — это здоровые особи с иммунитетом к болезни.

До того, как число заболевших не превышает критического значения I^* считаем, что все больные изолированы и не заражают здоровых. Когда $I(t) > I^*$, тогда инфицирование способны заражать восприимчивых к болезни особей.

Таким образом, скорость изменения числа S(t) меняется по следующему закону:

$$\frac{\partial S}{\partial t} = \begin{cases} -\alpha S, I(t) > I^* \\ 0, I(t) \leq I^* \end{cases}$$

Поскольку каждая восприимчивая к болезни особь, которая в конце концов заболевает, сама становится инфекционной, то скорость изменения числа инфекционных особей представляет разность за единицу времени между заразившимися и теми, кто уже болеет и лечится, т.е.:

$$\frac{\partial I}{\partial t} = \begin{cases} -\alpha S - \beta I, I(t) > I^* \\ -\beta I, I(t) \leq I^* \end{cases}$$

А скорость изменения выздоравливающих особей (при этом приобретающие иммунитет к болезни):

$$\frac{\partial R}{\partial t} = \beta I$$

Постоянные пропорциональности:

- α — коэффициент заболеваемости
- β — коэффициент выздоровления

Для того, чтобы решения соответствующих уравнений определялись однозначно, необходимо задать начальные условия. Считаем, что на начало эпидемии в момент времени $t = 0$ нет особей с иммунитетом к болезни $R(0) = 0$, а число инфицированных и восприимчивых к болезни особей $I(0)$ и $S(0)$ соответственно. Для анализа картины протекания эпидемии необходимо рассмотреть два случая: $I(0) \leq I^*$ и $I(0) > I^*$.

4 Выполнение лабораторной работы

1. Изучила начальные условия. Популяция состоит из 10400 особей. В начальный момент времени: 144 особей инфицированы; 28 здоровых особей с иммунитетом; (10400 - 144 - 28) особей, восприимчивых к болезни. Задала коэффициент заболеваемости, равный 0,25, и коэффициент выздоровления, равный 0,3.

2. Оформила начальные условия в код на Python:

```
a = 0.25
b = 0.3

N = 10400
I0 = 144
R0 = 28
S0 = N - I0 - R0
x0 = [S0, I0, R0]
```

3. Задала условия для времени: $t_0 = 0$ – начальный момент времени, $t_{max} = 200$ – предельный момент времени, $dt = 0,01$ – шаг изменения времени.

4. Добавила в программу условия, описывающие время:

```
t0 = 0
tmax = 200
dt = 0.01
t = np.arange(t0, tmax, dt)
```

5. Запрограммировала систему уравнений, соответствующую 1-ому случаю ($I(0) \leq I^*$):

```
def S1(x, t):
    dx1_0 = 0
    dx1_1 = - b*x[1]
    dx1_2 = b*x[1]
    return dx1_0, dx1_1, dx1_2
```

6. Запрограммировала систему уравнений, соответствующую 2-ому случаю ($I(0) > I^*$):

```
def S2(x, t):
    dx2_0 = -a*x[0]
    dx2_1 = a*x[0] - b*x[1]
    dx2_2 = b*x[1]
    return dx2_0, dx2_1, dx2_2
```

7. Запрограммировала решение систем уравнений:

```
y1 = odeint(S1, x0, t)
y2 = odeint(S2, x0, t)
```

8. Описала построение графика для 1-ого случая ($I(0) \leq I^*$):

```
plt.plot(t, y1[:,0], label='S(t)')
plt.plot(t, y1[:,1], label='I(t)')
plt.plot(t, y1[:,2], label='R(t)')
plt.title('I(0) <= I*')
plt.legend()
```

9. Описала построение графика для 2-ого случая ($I(0) > I^*$):

```
plt.plot(t, y2[:,0], label='S(t)')
plt.plot(t, y2[:,1], label='I(t)')
plt.plot(t, y2[:,2], label='R(t)')
plt.title('I(0) > I*')
plt.legend()
```

10. Собрала код программы воедино и получила следующее:

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

a = 0.25
b = 0.3

N = 10400
I0 = 144
R0 = 28
S0 = N - I0 - R0
x0 = [S0, I0, R0]

t0 = 0
tmax = 200
dt = 0.01
t = np.arange(t0, tmax, dt)

def S1(x, t):
    dx1_0 = 0
    dx1_1 = - b*x[1]
    dx1_2 = b*x[1]
```

```
    return dx1_0, dx1_1, dx1_2

def S2(x, t):
    dx2_0 = -a*x[0]
    dx2_1 = a*x[0] - b*x[1]
    dx2_2 = b*x[1]
    return dx2_0, dx2_1, dx2_2

y1 = odeint(S1, x0, t)
y2 = odeint(S2, x0, t)

plt.plot(t, y1[:,0], label='S(t)')
plt.plot(t, y1[:,1], label='I(t)')
plt.plot(t, y1[:,2], label='R(t)')
plt.title('I(0) <= I*')
plt.legend()

plt.plot(t, y2[:,0], label='S(t)')
plt.plot(t, y2[:,1], label='I(t)')
plt.plot(t, y2[:,2], label='R(t)')
plt.title('I(0) > I*')
plt.legend()
```

11. Получила следующие динамики изменения числа людей из каждой группы (см. рис. 4.1 и 4.2):

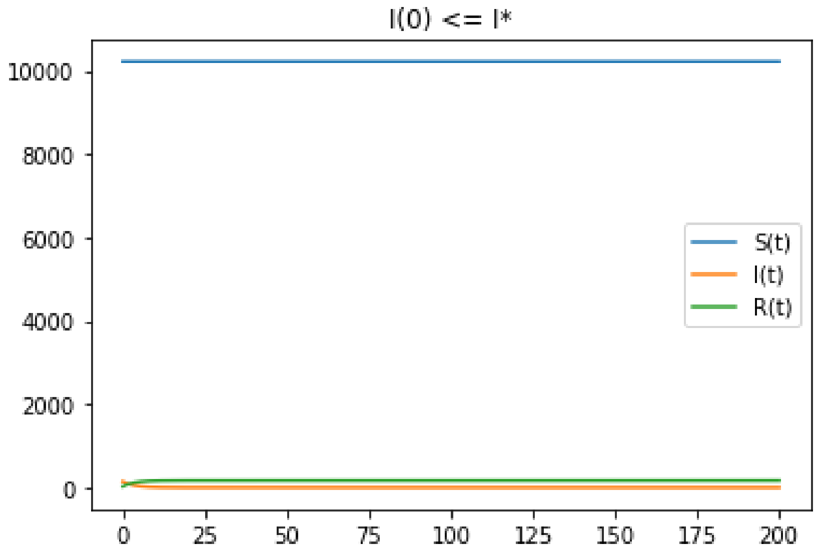

Figure 4.1: Динамика изменения числа людей в каждой из трех групп при $I(0) \leq I^*$

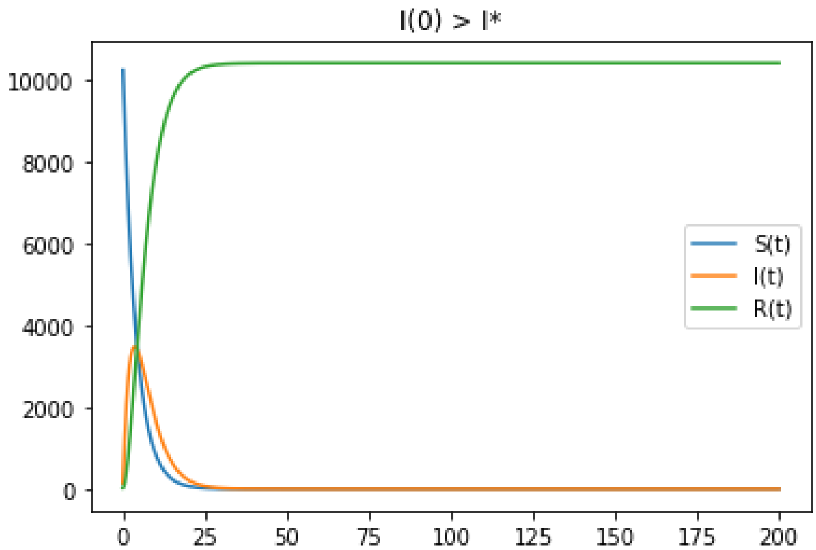

Figure 4.2: Динамика изменения числа людей в каждой из трех групп при $I(0) > I^*$

5 Выводы

Построила простейшую модель эпидемии с помощью Python.

В обоих случаях люди острова смогут победить болезнь.