

# **Лабораторная работа №6**

**Компьютерный практикум по статистическому анализу данных**

Еюбоглу Тимур

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Выполнение лабораторной работы</b>	<b>6</b>
2.1 Решение обыкновенных дифференциальных уравнений . . . . .	6
2.2 Модель экспоненциального роста . . . . .	6
2.3 Система Лоренца . . . . .	8
2.4 Модель Лотки–Вольтерры . . . . .	10
2.5 Самостоятельное выполнение . . . . .	12
<b>3 Вывод</b>	<b>18</b>
<b>4 Список литературы. Библиография</b>	<b>19</b>

# Список иллюстраций

2.1 График модели экспоненциального роста . . . . .	7
2.2 График модели экспоненциального роста (задана точность решения) . . . . .	8
2.3 Аттрактор Лоренца . . . . .	9
2.4 Аттрактор Лоренца (интерполяция отключена) . . . . .	10
2.5 Модель Лотки–Вольтерры: динамика изменения численности популяций . . . . .	11
2.6 Модель Лотки–Вольтерры: фазовый портрет . . . . .	12
2.7 Решение задания №1 . . . . .	13
2.8 Решение задания №2 . . . . .	14
2.9 Решение задания №3 . . . . .	15
2.10 Решение задания №4 . . . . .	15
2.11 Решение задания №5 . . . . .	16
2.12 Решение задания №6 . . . . .	16
2.13 Решение задания №7 . . . . .	17
2.14 Решение задания №8 . . . . .	17

# **Список таблиц**

# **1 Цель работы**

Основной целью работы является освоение специализированных пакетов для решения задач в непрерывном и дискретном времени.

## **2 Выполнение лабораторной работы**

### **2.1 Решение обыкновенных дифференциальных уравнений**

Вспомним, что обыкновенное дифференциальное уравнение (ОДУ) описывает изменение некоторой переменной  $u$ .

Для решения обыкновенных дифференциальных уравнений (ОДУ) в Julia можно использовать пакет differentialEquations.jl.

### **2.2 Модель экспоненциального роста**

Рассмотрим пример использования этого пакета для решения уравнения модели экспоненциального роста, описываемую уравнением, где  $a$  – коэффициент роста.

Численное решение в Julia будет иметь следующий вид, а также график, соответствующий полученному решению (рис. 2.1):

## 1.1. Модель экспоненциального роста

```
using DifferentialEquations, Plots

# задаём описание модели с начальными условиями:
a = 0.88
f(u,p,t) = a*u
u0 = 1.0
# задаём интервал времени:
tspan = (0.0,1.0)
# решение:
prob = ODEProblem(f,u0,tspan)
sol = solve(prob)
# строим графики:
plot(sol, linewidth=5,title="Модель экспоненциального роста", xaxis="Время",yaxis="u(t)",label="u(t)")
plot!(sol.t, t->1.0*exp(a*t),lw=3,ls=:dash,label="Аналитическое решение")
```

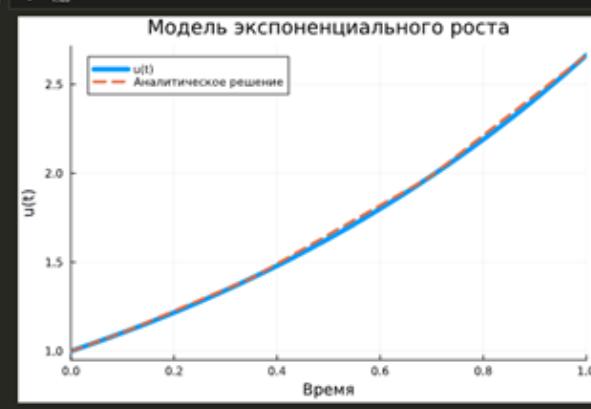


Рис. 2.1: График модели экспоненциального роста

При построении одного из графиков использовался вызов `sol.t`, чтобы захватить массив моментов времени. Массив решений можно получить, воспользовавшись `sol.u`.

Если требуется задать точность решения, то можно воспользоваться параметрами `abstol` (задаёт близость к нулю) и `reltol` (задаёт относительную точность). По умолчанию эти параметры имеют значение `abstol = 1e-6` и `reltol = 1e-3`.

Для модели экспоненциального роста (рис. 2.2):

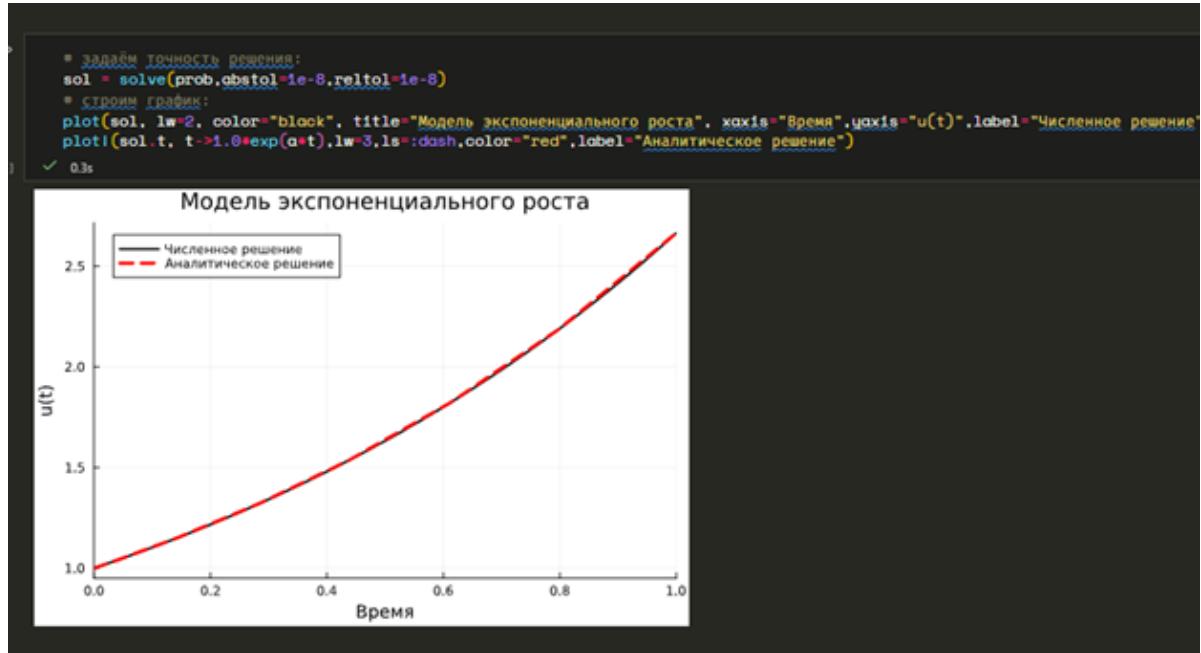


Рис. 2.2: График модели экспоненциального роста (задана точность решения)

## 2.3 Система Лоренца

Динамической системой Лоренца является нелинейная автономная система обыкновенных дифференциальных уравнений третьего порядка.

Система получена из системы уравнений Навье–Стокса и описывает движение воздушных потоков в плоском слое жидкости постоянной толщины при разложении скорости течения и температуры в двойные ряды Фурье с последующем усечением до первых-вторых гармоник.

Решение системы неустойчиво на аттракторе, что не позволяет применять классические численные методы на больших отрезках времени, требуется использовать высокоточные вычисления.

Численное решение в Julia будет иметь следующий вид (рис. 2.3):

## 1.2. Система Лоренца

```
> % Задаём описание модели
function lorenzi(du, u, p, t)
    d, p, β = p
    du[1] = d * (u[2] - u[1])
    du[2] = u[1] * (p - u[3]) - u[2]
    du[3] = u[1] * u[2] - β * u[3]
end
% Задаём начальное условие
u0 = [1.0, 0.0, 0.0]
% Задаём значения параметров
p = (10, 28, 8 / 3)
% Задаём интервал времени
tspan = (0.0, 100.0)
% Решение
prob = ODEProblem(lorenzi, u0, tspan, p)
sol = solve(prob, Tsit5())
% Строим график
plot(sol, idxs=(1, 2, 3), lw=1, title="Аттрактор Лоренца", xaxis="x", yaxis="y", zaxis="z", legend=false)
```



Рис. 2.3: Аттрактор Лоренца

Можно отключить интерполяцию (рис. 2.4):

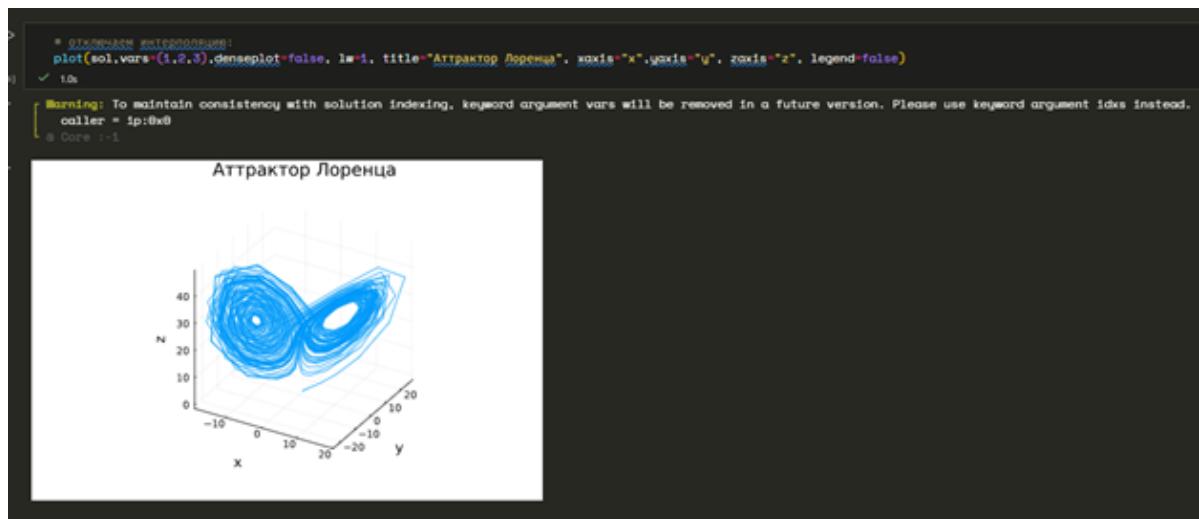


Рис. 2.4: Аттрактор Лоренца (интерполяция отключена)

## 2.4 Модель Лотки–Вольтерры

Модель Лотки–Вольтерры описывает взаимодействие двух видов типа «хищник – жертва».

Численное решение в Julia будет иметь следующий вид (рис. 2.5):

## 2. Модель Лотки–Вольтерры

```
# Определяем модель Лотки–Вольтерры
function lotka_volterra!(du, u, p, t)
    x, y = u
    a, b, c, d = p
    du[1] = a * x - b * x * y # Уравнение для жертв
    du[2] = -c * y + d * x * y # Уравнение для хищников
end
# Начальные условия
u0 = [1.0, 1.0] # начальные популяции жертв и хищников
# Параметры модели
p = [1.5, 1.0, 3.0, 1.0] # a, b, c, d
# Интервал времени
tspan = (0.0, 10.0)
# Создаём задачу
prob = ODEProblem(lotka_volterra!, u0, tspan, p)
# Решаем задачу
sol = solve(prob, Tsit5())
# Построение графика
plot(sol, label=["Жертвы" "Хищники"], color="black",
      linestyle = [:solid :dash], title="Модель Лотки–Вольтерры",
      xlabel="Время", ylabel="Размер популяции")
```

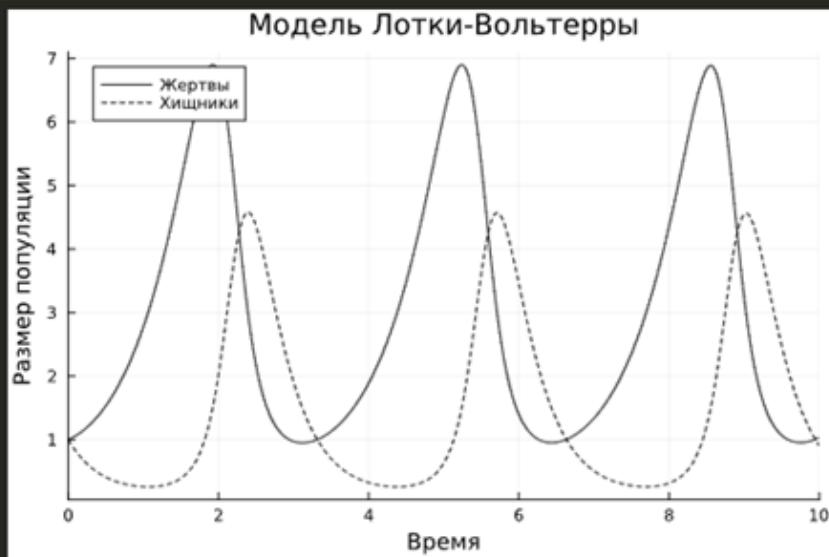


Рис. 2.5: Модель Лотки–Вольтерры: динамика изменения численности популяций

Фазовый портрет (рис. 2.6):

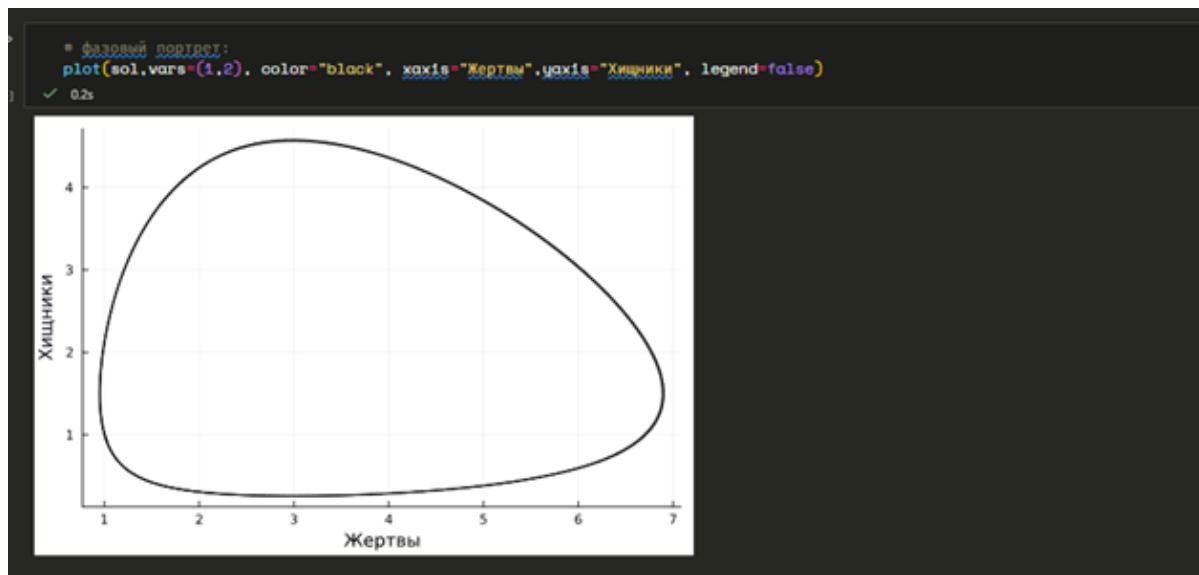


Рис. 2.6: Модель Лотки–Вольтерры: фазовый портрет

## 2.5 Самостоятельное выполнение

Выполнение задания №1 (рис. 2.7):

### Самостоятельное выполнение

1) Реализовать и проанализировать модель роста численности изолированной популяции (модель Мальтуса). Начальные данные и параметры задать самостоятельно и пояснить их выбор. Построить соответствующие графики (в том числе с анимацией):



Рис. 2.7: Решение задания №1

Выполнение задания №2 (рис. 2.8):

выбор. Построить соответствующие графики (в том числе с анимацией):

```
# Определение модели
function logisticI(du, u, p, t)
    r, k = p
    du[1] = r * u[1] * (1 - u[1] / k)
end
# Начальные условия
u0 = [1.0] # Начальная численность популяции
# Параметры
r = 0.5 # Коэффициент роста
k = 10.0 # Емкость экосистемы
p = (r, k)
# Интервал времени
tspan = (0.0, 20.0)
# Решение
prob = ODEProblem(logisticI, u0, tspan, p)
sol = solve(prob, Tsit5()) # Используем Tsit5 для не-жёсткой задачи
# Создание объекта анимации
anim = Animation()
# Построение анимации
for t in 0:0.5:20
    plot(sol, vars=(0, 1), linewidth=2, color=:blue, title="Логистическая модель роста",
        xlabel="Время", ylabel="Популяция", legend=false)
    scatter!([t], [sol(t)[1]], color=:red, label="", markersize=5) # Добавляем текущую точку
    frame(anim) # Добавляем кадр в анимацию
end
# Сохранение анимации
gif(anim, "logistic_growth.gif", fps=10) # Сохраняем анимацию в файл
3.4s
```

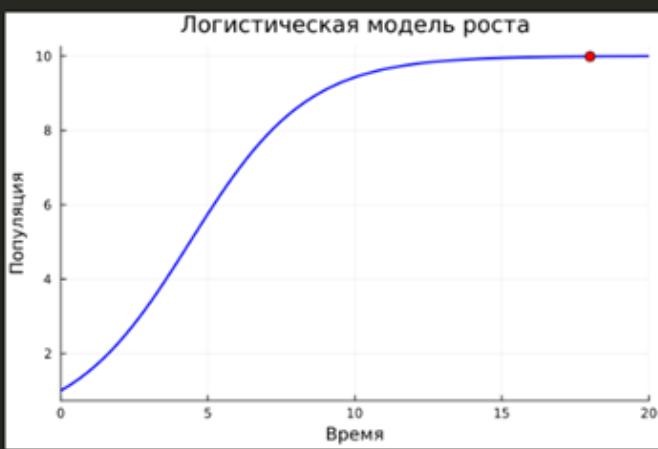


Рис. 2.8: Решение задания №2

Выполнение задания №3 (рис. 2.9):

3) Реализовать и проанализировать модель эпидемии Кермака–Маккендрика (SIR модель). Начальные данные и параметры задать самостоятельно и пояснить их выбор. Построить соответствующие графики (в том числе с анимацией):

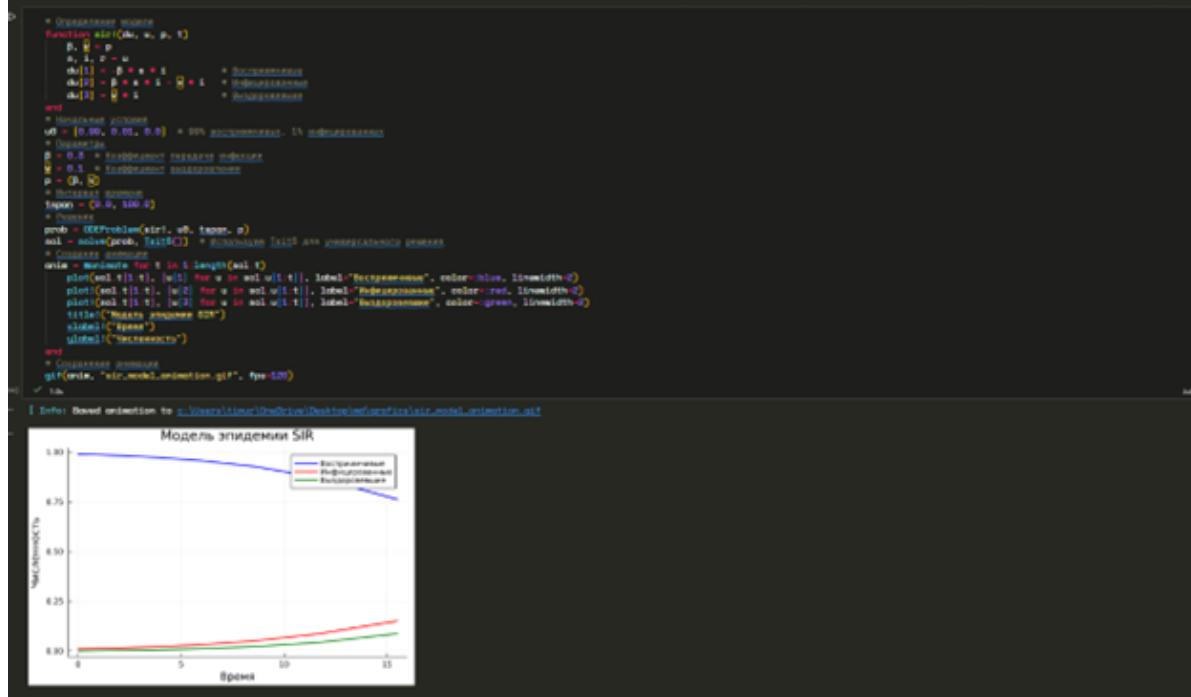


Рис. 2.9: Решение задания №3

Выполнение задания №4 (рис. 2.10):

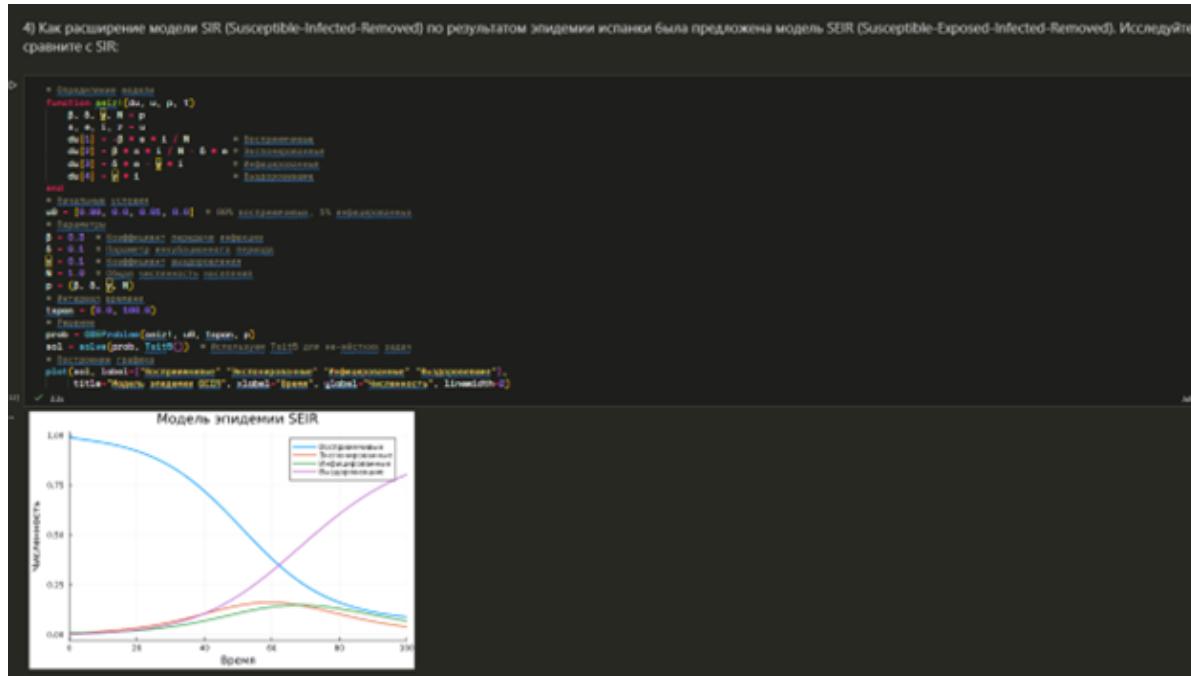


Рис. 2.10: Решение задания №4

### Выполнение задания №5 (рис. 2.11):

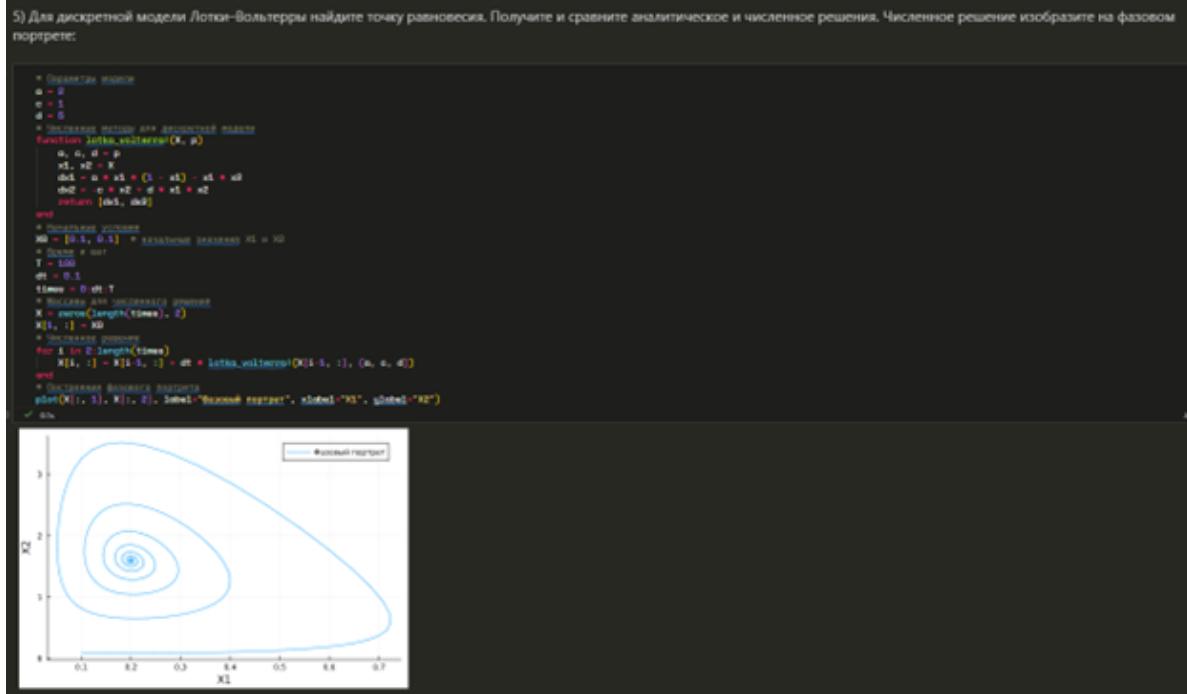


Рис. 2.11: Решение задания №5

### Выполнение задания №6 (рис. 2.12):

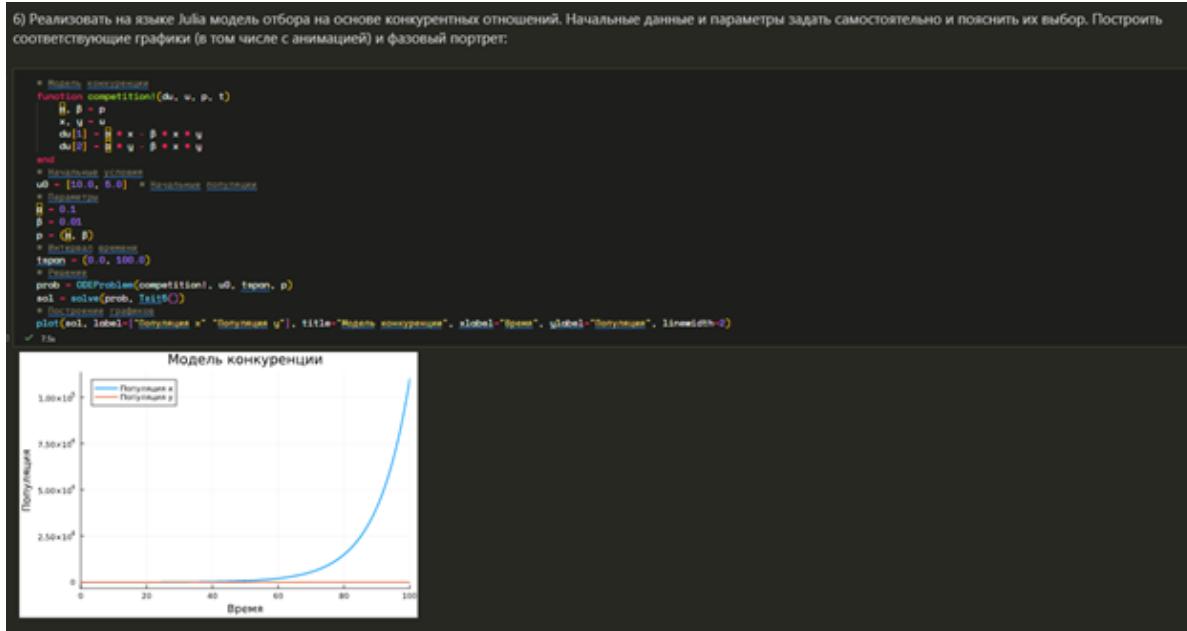


Рис. 2.12: Решение задания №6

### Выполнение задания №7 (рис. 2.13):

7) Реализовать на языке Julia модель консервативного гармонического осциллятора. Начальные параметры подобрать самостоятельно, выбор пояснить. Построить соответствующие графики (в том числе с анимацией) и фазовый портрет:

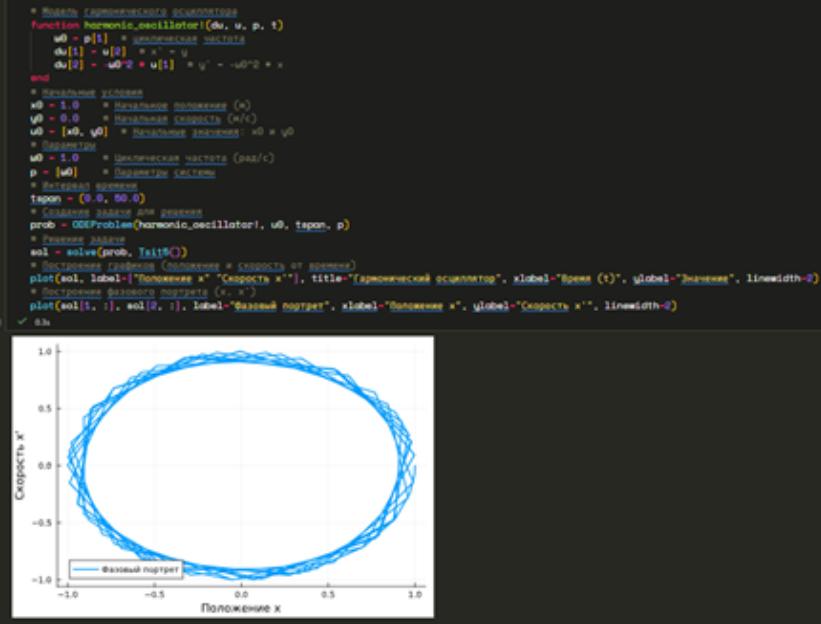


Рис. 2.13: Решение задания №7

### Выполнение задания №8 (рис. 2.14):

8) Реализовать на языке Julia модель свободных колебаний гармонического осциллятора. Начальные параметры подобрать самостоятельно, выбор пояснить. Построить соответствующие графики (в том числе с анимацией) и фазовый портрет:

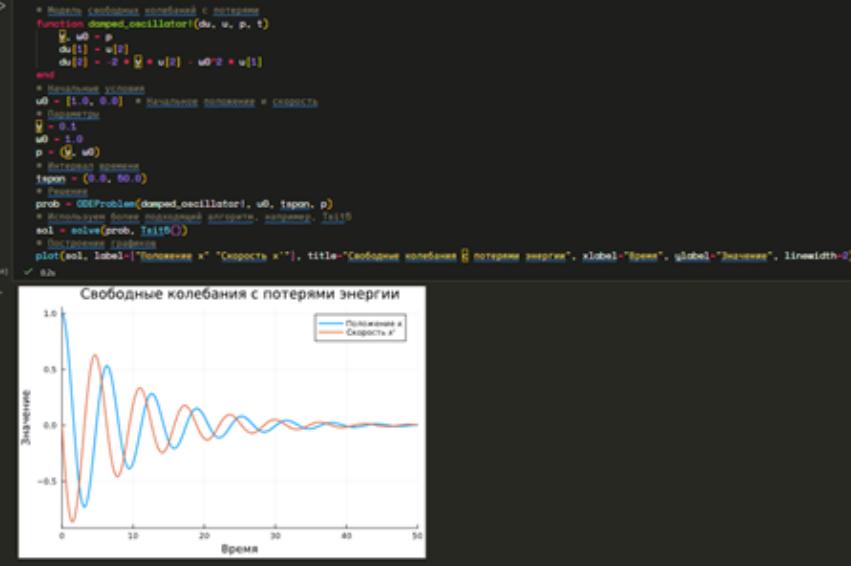


Рис. 2.14: Решение задания №8

## **3 Вывод**

В ходе выполнения лабораторной работы были освоены специализированные пакеты для решения задач в непрерывном и дискретном времени.

## **4 Список литературы. Библиография**

[1] Julia Documentation: <https://docs.julialang.org/en/v1/>