Analytical Geometry and Linear Algebra II — Predator-Prey Model

Dmitriy Okoneshnikov
B22-DSAI-04
d.okoneshnikov@innopolis.university

April 30, 2023

1 Source code

The whole program also can be found here.

```
#include <iostream>
#include <iomanip>
#include <fstream>
#include <cmath>
#define USE_GNUPLOT 0
#define GNUPLOT_NAME "gnuplot -persist"

using namespace std;

int main(void)
{
    cout << fixed << setprecision(2);

    double V_0, K_0;
    cin >> V_0 >> K_0;

    double alpha_1, beta_1, alpha_2, beta_2;
    cin >> alpha_1 >> beta_1 >> alpha_2 >> beta_2;

    double T;
    cin >> T;

    int N;
    cin >> N;

    double v_0 = V_0 - alpha_2 / beta_2;
    double k_0 = K_0 - alpha_1 / beta_1;
    double a = sqrt(alpha_1 * alpha_2);

    // Create arrays
    double *T_arr = new double[N + 1];
    double *V_arr = new double[N + 1];
    double *K_arr = new double[N + 1];
```

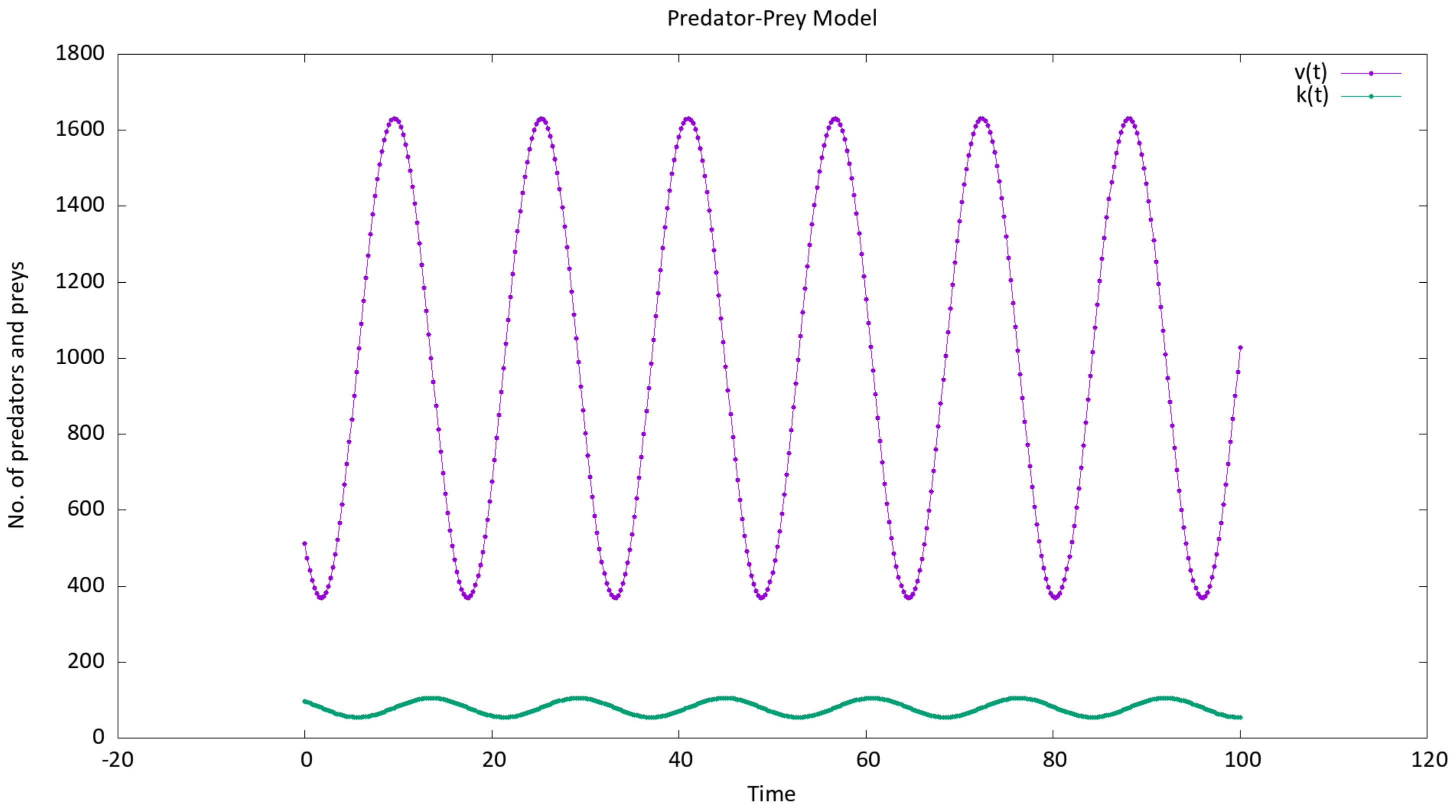
```
    // Calculate
    double t_i = 0;
    for (int i = 0; i <= N; i++)
    {
        T_arr[i] = t_i;

        // Eq. 23-24, p. 312, Matrix methods of approximating classical predator-prey problems, E. Y. Rodin et al.
        V_arr[i] = v_0 * cos(a * t_i) - k_0 * (sqrt(alpha_2) * beta_1 / (beta_2 * sqrt(alpha_1))) * sin(a * t_i) + alpha_2 / beta_2;
        K_arr[i] = k_0 * cos(a * t_i) + v_0 * (sqrt(alpha_1) * beta_2 / (beta_1 * sqrt(alpha_2))) * sin(a * t_i) + alpha_1 / beta_1;

        t_i += T / N;
    }

    // Print arrays
    cout << "t:" << '\n';
    for (int i = 0; i <= N; i++)
        cout << T_arr[i] << ' ';
    cout << '\n'
         << "v:" << '\n';
    for (int i = 0; i <= N; i++)
        cout << V_arr[i] << ' ';
    cout << '\n'
         << "k:" << '\n';
    for (int i = 0; i <= N; i++)
        cout << K_arr[i] << ' ';
    cout << '\n';

#if (defined(WIN32) || defined(_WIN32)) && USE_GNUPLOT
    FILE *pipe = _popen(GNUPLOT_NAME, "w");
#elif USE_GNUPLOT
    FILE *pipe = popen(GNUPLOT_NAME, "w");
#endif
#if USE_GNUPLOT
    ofstream out("points.txt");
    out << fixed << setprecision(4);
    for (int i = 0; i <= N; i++)
        out << T_arr[i] << '\t' << V_arr[i] << '\t' << K_arr[i] << '\n';
    out.close();
    // Draw v(t), k(t)
    fprintf(pipe, "%s\n", "set terminal png size 1920, 1080 font \"Calibri,24\"");
    fprintf(pipe, "%s\n", "set output 'v_t__k_t.png'");
    fprintf(pipe, "%s\n", "set title \"Predator-Prey Model\"");
    fprintf(pipe, "%s\n", "set key noautotitle");
    fprintf(pipe, "%s\n", "set autoscale xy");
    fprintf(pipe, "%s\n", "set offsets 0.05, 0.05, 0.05, 0.05");
```

```
    fprintf(pipe, "%s\n", "set xlabel \"Time\"");
    fprintf(pipe, "%s\n", "set ylabel \"No. of predators and preys\"")
       ↪ ;
    fprintf(pipe, "%s\n", "plot \"points.txt\" u 1:2 t 'v(t)' w
       ↪ linespoints pointtype 7, \"points.txt\" u 1:3 t 'k(t)' w
       ↪ linespoints pointtype 7");
    fflush(pipe);
    // Draw v(k)
    fprintf(pipe, "%s\n", "set output 'v_k.png'");
    fprintf(pipe, "%s\n", "set xlabel \"Number of predators\"");
    fprintf(pipe, "%s\n", "set ylabel \"Number of preys\"");
    fprintf(pipe, "%s\n", "plot \"points.txt\" u 3:2 t 'v(k)' w
       ↪ linespoints pointtype 7");
    fflush(pipe);
    // Draw k(v)
    fprintf(pipe, "%s\n", "set output 'k_v.png'");
    fprintf(pipe, "%s\n", "set xlabel \"Number of preys\"");
    fprintf(pipe, "%s\n", "set ylabel \"Number of predators\"");
    fprintf(pipe, "%s\n", "plot \"points.txt\" u 2:3 t 'k(v)' w
       ↪ linespoints pointtype 7");
    fflush(pipe);
#endif
#if (defined(WIN32) || defined(_WIN32)) && USE_GNUPLOT
    _pclose(pipe);
#elif USE_GNUPLOT
    pclose(pipe);
#endif

    // Delete arrays
    delete[] T_arr;
    delete[] V_arr;
    delete[] K_arr;

    return 0;
}
```

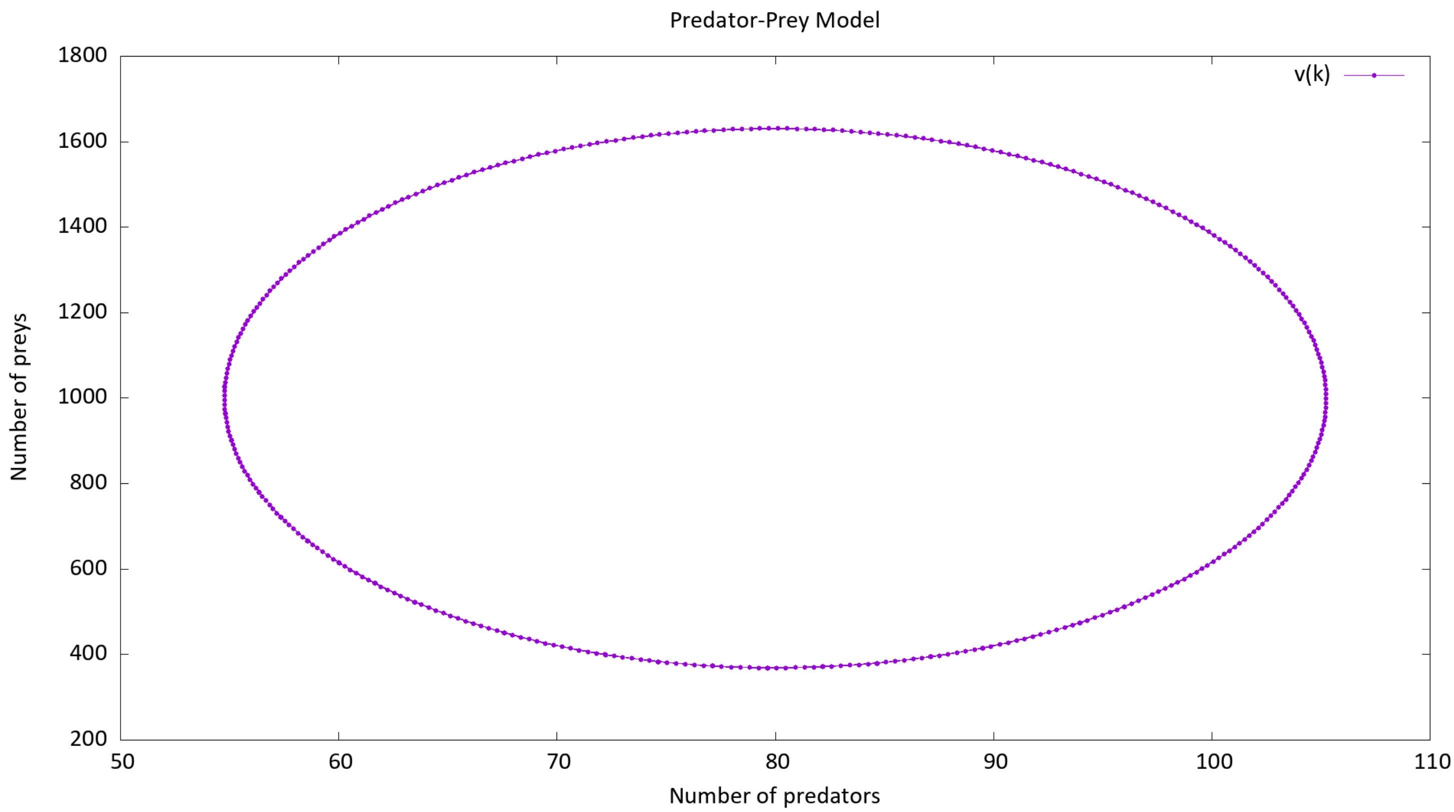
2 Input

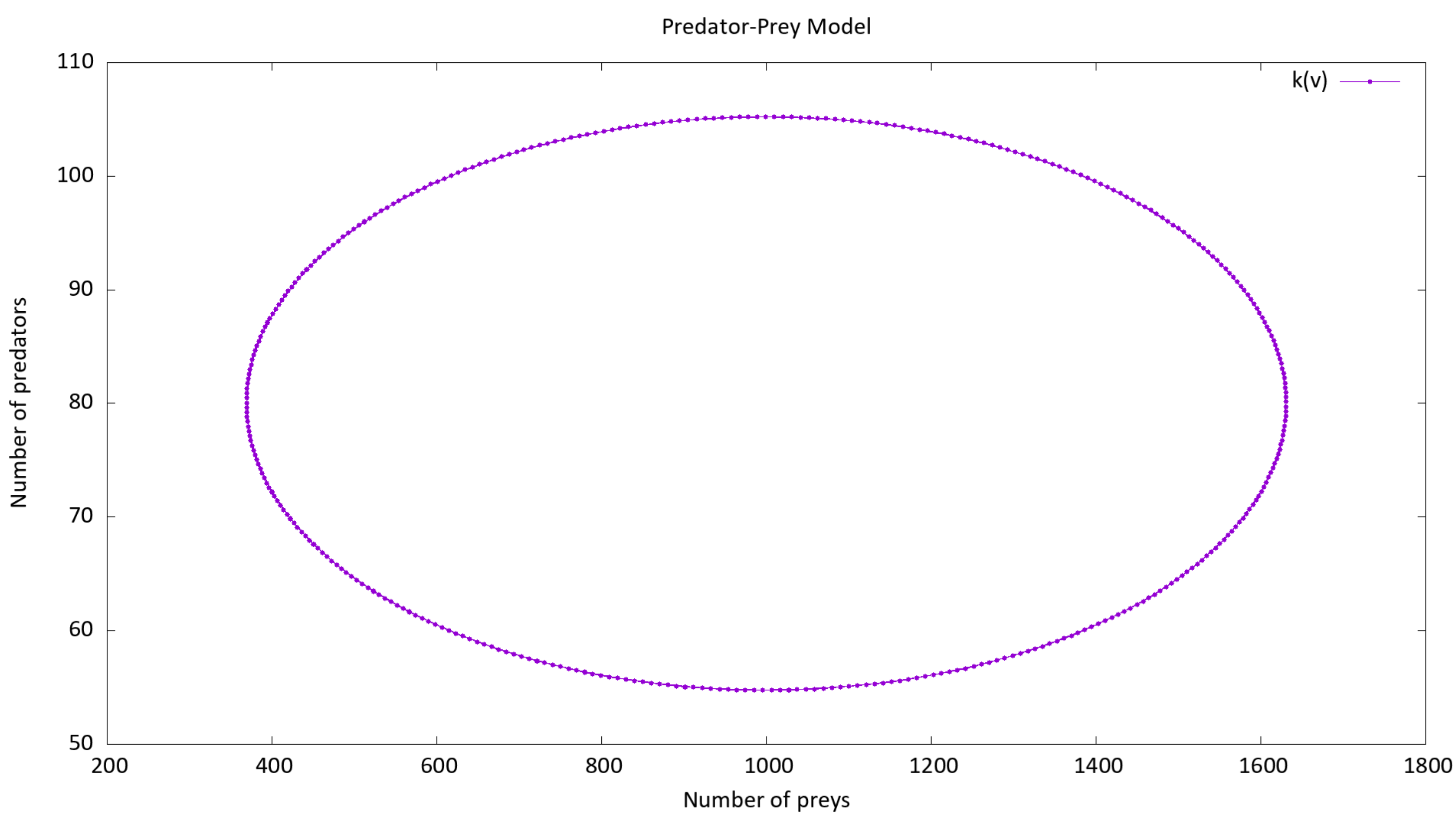
512
96
0.8
0.01
0.2
0.0002
100
400

3 Plot



Predator-Prey Model
v(t)
k(t)
No. of predators and preys
Time
1800
1600
1400
1200
1000
800
600
400
200
0
-20
0
20
40
60
80
100
120


Predator-Prey Model
v(k)
Number of preys
Number of predators
1800
1600
1400
1200
1000
800
600
400
200
50
60
70
80
90
100
110


Predator-Prey Model
k(v)
Number of predators
Number of preys
110
100
90
80
70
60
50
200
400
600
800
1000
1200
1400
1600
1800