

**O SISTEMA LOTKA-VOLTERRA:  
MODELAGEM ECOLÓGICA DE PREDADORES E PRESAS E  
IMPLEMENTAÇÕES COM O SAGEMATH**

Pedro Victor S. Freitas

Universidade Federal Rural de Pernambuco - UFRPE

[pedro-victor@msn.com](mailto:pedro-victor@msn.com)

José Arthur de Lima Azevedo

Universidade Federal Rural de Pernambuco - UFRPE

[j.80.arthur@gmail.com](mailto:j.80.arthur@gmail.com)

Thiago Yukio Tanaka

Universidade Federal Rural de Pernambuco - UFRPE

[thiago.tanaka@ufrpe.br](mailto:thiago.tanaka@ufrpe.br)

Ricardo Nunes Machado Jr.

Universidade Federal Rural de Pernambuco - UFRPE

[ricardo.machadojunior@ufrpe.br](mailto:ricardo.machadojunior@ufrpe.br)

## Resumo

Investigaremos um caso particular do sistema Lotka-Volterra que modela uma variedade de problemas de nosso mundo, dentre eles a dinâmica ecológica envolvendo duas espécies, sendo uma de presas (com fonte abundante de alimento) e outra de predadores (que se alimenta das presas). O modelo matemático mais simples (modelo clássico) é descrito por meio de um sistema bidimensional de equações diferenciais ordinárias não lineares acopladas. Inicialmente, utilizaremos a abordagem por meio da linearização em torno dos pontos críticos de modo a entender como são localmente as trajetórias do sistema. Ainda nesta análise, a partir da solução para o caso linear, provaremos uma série de resultados para as soluções do sistema. Apesar do sistema clássico ser não linear, ele é equivalente a uma EDO separável, o que nos permite obter as soluções por meio de uma equação implícita. Para entender completamente o sistema, utilizamos o *software* livre *SageMath*, usando diversos métodos para obter soluções explícitas, numéricas e gerar gráficos associados como o campo de vetores, as trajetórias no plano de fase e como evoluem as soluções no tempo, e ainda comparamos as soluções obtidas numericamente com as soluções explícitas. Mais ainda, desenvolveremos noções qualitativas por meio da análise da estabilidade do sistema. Todas as implementações foram feitas pelos autores e são disponibilizadas ao fim deste trabalho e de maneira *online*, de forma

que os discentes e docentes interessados possam utilizar esses recursos já construídos para enriquecer seus estudos e pesquisas.

**Palavras-Chave:** Equações de Lotka-Volterra; Sistema Predador-Presa; Equações Diferenciais Ordinárias; Estabilidade; SageMath.

## Abstract

We will investigate a particular case of the Lotka-Volterra system that models a variety of problems in our world, including ecological dynamics involving two species, one of prey (with an abundant source of food) and the other of predators (which feeds on prey). The simplest mathematical model (classic model) is described using a two-dimensional system of coupled nonlinear ordinary differential equations. Initially, we will use the approach through linearization around critical points in order to understand how the system trajectories locally are. Still in this analysis, from the solution for the linear case, we will prove a series of results for the system solutions. Although the classical system is nonlinear, it is equivalent to a separable ODE, which allows us to obtain the solutions through an implicit equation. To fully understand the system, we use the open software *SageMath*, using several methods to obtain explicit, numerical solutions and generate associated graphics such as the vector field, phase plane trajectories, and how the solutions evolve in time, and we also compare the numerically obtained solutions with the explicit solutions. Furthermore, we will develop qualitative notions through the analysis of system stability. All implementations were made by the authors and are made available at the end of this work and online so that interested students and professors can use these resources already built to enrich their studies and research.

**Keywords:** Lotka-Volterra Equations; Predator-Prey System; Ordinary Differential Equation; Stability; SageMath.

## 1 Introdução

O interesse em estudar o comportamento de espécies na natureza é antigo e o uso de conceitos matemáticos têm sido uma importante ferramenta nesses estudos [4, p. 12].

Não é de hoje que grande parte de fenômenos variacionais são modelados por meio de equações diferenciais, pois a derivada é uma ferramenta que nos permite entender como determinadas grandezas variam em função de outras. O estudo sobre Equações Diferenciais Ordinárias (EDOs) permite modelagens de fenômenos diversos tais como: sociais

(dinâmica populacional); econômicos (capitalização de investimentos); epidemiológicos (dinâmica de vírus, bactérias, com ou sem vacinação, com ou sem reincidência da doença, contaminação em hospitais); ecológicos (extinção de espécies, competição entre espécies semelhantes, dinâmica entre predadores e presas); químicos (decaimento radioativo); físicos (na mecânica, elétrica, na física clássica, moderna) entre tantos outros contextos que podem ser abordados (ver [2, 6, 9, 21] e referências contidas nestes trabalhos).

Dentre os exemplos citados, nos concentraremos na dinâmica ecológica do predatismo, na qual envolve espécies diferentes, em que apenas uma delas é beneficiada. Tal relação é denominada interespecífica desarmônica. O difundido modelo ecológico do tipo predador-presa, determinado pelas equações Lotka-Volterra, podem ser encontradas em [5]. Este modelo como o próprio nome sugere, prevê a dinâmica entre as populações  $x$  e  $y$  respectivamente de presas e predadores confinadas em um determinado meio. Trata-se então de uma dinâmica na qual a espécie de predadores se alimenta das presas, esta última por sua vez se alimenta de outros tipos de comida que supostamente é abundante.

Ciente de que é possível criar e estudar modelos apenas com poucas variáveis e que modelos mais simples, em geral, podem não representar a totalidade do fenômeno real, o estudo do modelo predador-presa será feito com o modelo clássico, utilizando as equações de Lotka-Volterra, que mesmo sendo um modelo simples, permite-nos identificar estratégias para estudar casos de dinâmicas com estruturas mais complexas. As equações de Lotka-Volterra foram desenvolvidas por Alfred Lotka e Vito Volterra em momentos distintos e separadamente. As equações propostas por Volterra consistiam em um modelo que explicaria o aumento de certa população de peixes predadores, que se alimentariam de uma espécie de peixes presas no Mar Adriático durante a Primeira Guerra Mundial. Por outro lado, o modelo proposto por Lotka descrevia reações químicas em que as concentrações dos elementos químicos oscilavam, sendo semelhantes aos modelos de espécies em competição (ver [5]). Mais ainda, modelos desse tipo possuem aplicações diversas: nas Ciências Agrárias com o controle de pragas, na Economia com a descrição das oscilações de bolsas de valores e até mesmo em questões ambientais como na dinâmica envolvendo captura e emissão de carbono (ver [14] e as referências lá contidas).

Certamente, o caso clássico e bidimensional é extremamente simplificado, se compararmos com a inter-relação desses seres na natureza. Essa relação é bastante complexa, uma vez que diversos outros fatores podem vir a influenciar no modelo, por exemplo, aspectos topológicos associados ao ambiente, mudanças nas estações climáticas, diversidade de outras espécies. No entanto, ainda é possível compreender diversos princípios ecológicos estudando esse modelo, tais como: prever qual espécie irá predominar em um

determinado intervalo de tempo, se alguma das espécies entrará em extinção ou ainda, se a convivência entre eles promove a propagação de ambas as espécies.

O sistema predador-presa foi e continua sendo um objeto de estudo na área de Equações Diferenciais, pois há uma grande diversidade de problemas a serem considerados relacionados com esta modelagem que vão desde interferências extrínsecas como topologia relacionada ao ambiente ou perturbações provocadas por doenças infecciosas, quanto intrínsecas como dinâmicas mais complexas com mais espécies, modificações estruturais nos parâmetros. Em [11], estudando um sistema predador-presa do tipo Leslie-Gower, o autor caracteriza a estabilidade global do sistema por meio do método de Lyapunov. Essa modelagem considera outros tipos de modificações pertinentes, tais como ênfase dos limites superiores para as taxas de crescimento das populações de presas e predadores, fato que não existe no modelo clássico. Em [10] os autores investigam, usando equações do tipo SIS, os efeitos simultâneos de microparasitas e predadores na regulação da dinâmica entre predadores e presas. Nesse estudo, eles verificam a possibilidade de cenários em que a presença do parasita pode ou não interferir de maneira explícita nessa dinâmica, provocando, por exemplo, a extinção de uma das espécies ou mesmo, a dinâmica predatória eliminar completamente a doença. Essa análise é feita por meio do cálculo do  $R_0$ , número de reprodutibilidade básica, em cada situação. Citamos também o artigo [13] no qual os autores trazem uma análise numérica do sistema de Lotka-Volterra, utilizando a linguagem de programação *FORTRAN* e buscando soluções por meio do método de aproximação numérica Runge-Kutta de quarta ordem. Nesse trabalho, eles fazem algumas alterações no modelo clássico, considerando termos de saturação nas populações de presas e chegam à conclusão que nessa situação, dependendo da escolha de parâmetros, o novo modelo descreve situações não consideradas no modelo clássico, a saber com a extinção de uma das espécies. Por fim, convidamos os leitores para a leitura das dissertações de mestrado do PROFMAT [4, 15, 17].

Portanto, o objetivo deste trabalho é apresentar os estudos desenvolvidos durante o projeto de iniciação científica PIBIC-FACEPE 2020-2021 dos dois primeiros autores sob a orientação dos dois últimos autores. Mais precisamente, apresentaremos um estudo em detalhes do sistema predador-presa clássico, fazendo a análise da estabilidade de pontos críticos, os campos de vetores, trajetórias, e, em particular, apesar do sistema ser não linear, ele ainda pode ser resolvido, uma vez que ele é equivalente a uma EDO separável. Por meio da solução explícita, analisamos aspectos qualitativos dessas soluções. Como o sistema é não linear, seu estudo é feito por meio de implementações numéricas através do *software* computacional livre do SageMath. O SageMath é capaz de resolver simbolicamente e/ou numericamente algumas EDOs (ou sistemas de EDOs), permitindo uma melhor visualização por meio dos campos vetores associados

às soluções, às trajetórias dos sistemas associados e aos gráficos das soluções no decorrer do tempo. Por fim, ao decorrer do artigo, trazemos a construção de grande parte das implementações numéricas feitas por meio do SageMath. Mais ainda, disponibilizamos no *QR Code* abaixo, um site com as implementações pertinentes que funcionam *online*, com construções relacionadas ao conteúdo, a saber: calculadora de campo de vetores, calculadora das trajetórias do não linear e da linearização e calculadora da solução no tempo:



Esperamos, assim, que este trabalho sirva como fonte de pesquisa, mas também que forneça subsídios em termos de implementação computacional dos estudos sobre equações diferenciais ordinárias relacionadas com este trabalho.

Além disso, para um melhor entendimento do artigo, há a necessidade de conhecimentos relacionados com o conteúdo das EDOs e de Álgebra Linear. Nesse sentido, para o estudo de EDOs, recomendamos as referências [5, 16, 20] e as referências [1, 7] para estudos sobre Álgebra Linear.

Este trabalho está dividido da seguinte maneira. Na Seção 2, faremos a análise do sistema clássico utilizando três abordagens. Inicialmente analisaremos o comportamento do sistema linear associado nas proximidades de pontos críticos, fazendo um estudo da estabilidade do sistema, verificando as trajetórias e como se comportam localmente as soluções, o formato explícito e suas propriedades. Em seguida, faremos um estudo do sistema não linear por meio da análise da solução dada implicitamente, provaremos que determinadas propriedades e ilustraremos alguns casos com o auxílio do *software* livre SageMath. Por fim, comparamos as três soluções: a do sistema linear, a implícita e aquela obtida através do método numérico de Runge-Kunta de quarta ordem. A Seção 3 traz em detalhes todas as construções mais importantes relacionadas com implementações computacionais utilizando o *SageMath*, tanto dos aspectos da linearização, quanto da solução numérica, utilizando o método de Runge-Kunta de quarta ordem e os elementos estruturais gráficos envolvidos como o campo de vetores, as trajetórias no plano de fase e como variam as soluções no tempo. Por fim, há na Seção 4 a conclusão do trabalho seguida das referências utilizadas.

## 2 O Modelo Clássico Predador-Presa

A modelagem ecológica clássica de predadores e presas, usualmente utilizada em uma primeira abordagem do tema, pressupõe condições compatíveis com a realidade, fato que se traduz em sua construção, quando não há predadores,  $y \equiv 0$ , a população  $x$  de presas aumenta em uma taxa proporcional à população atual,  $\frac{dx}{dt} = ax$ ,  $a > 0$ . Então, na ausência de presas,  $x \equiv 0$ , os predadores se extinguem,  $\frac{dy}{dt} = -cy$ , com  $c > 0$ . Além disso, a quantidade de encontros entre predador e presa é proporcional ao produto das duas populações. Nesses encontros, a população de predadores tende a aumentar (uma vez que os predadores irão se alimentar) e a reduzir o crescimento da população de presas (pelo predatismo). Dessa maneira, a taxa de crescimento da população de predadores é aumentada por um fator da forma  $\gamma xy$ , enquanto a taxa de crescimento para a população de presas é diminuída por um termo da forma  $-\alpha xy$ , ambos  $\alpha$  e  $\gamma$  são constantes positivas. A condição do encontro ser proporcional ao produto das populações não é algo novo nas modelagens com equações diferenciais, por exemplo, nas dinâmicas epidemiológicas de modelos compartimentados isso também ocorre, uma diferença é que não há troca de indivíduos de um compartimento para outro, por isso as taxas  $\alpha$  e  $\gamma$  são supostamente distintas. Uma interpretação biológica desse fato é que a predação exclusiva de certa quantidade de presa pode não promover a sobrevivência da mesma quantidade de predadores. Ambos os modelos tomam por base o princípio da ação das massas na qual o encontro das variáveis (espécies ou compartimentos) e a interação entre elas é representada pelo produto das variáveis. Assumiremos também que, no caso da modelagem ecológica, a competição intraespecífica (com relação à mesma espécie) tem influência insignificante na população de presas e predadores.

Partindo dessas premissas, este modelo pode ser descrito pelo seguinte sistema acoplado bidimensional de equações diferenciais ordinárias de primeira ordem não lineares

$$\begin{cases} \frac{dx}{dt} = xa - \alpha xy = x(a - \alpha y) = F(x, y), \\ \frac{dy}{dt} = -cy + \gamma xy = y(-c + \gamma x) = G(x, y). \end{cases} \quad (2.1)$$

O sistema (2.1) é dito acoplado, pois a primeira equação na variável  $x$  depende da variável  $y$ , o mesmo acontece com a segunda equação.

A análise será feita dividida em duas partes. A análise do sistema linear que consiste em estudar o comportamento das soluções nas proximidades de pontos específicos, e do sistema não linear geral descobrindo assim um comportamento global associado. Em um segundo momento, estudaremos o sistema (2.1) sob a perspectiva computacional utilizando o *SageMath*.

## 2.1 Análise do Sistema Linear Associado

Investigaremos elementos pertinentes ao sistema (2.1). Perceba que (2.1) é um sistema não linear, portanto, requer uma análise mais específica. De fato, para entendermos de maneira satisfatória como são as trajetórias do sistema (2.1), analisaremos o comportamento das trajetórias nas proximidades dos pontos de equilíbrio fazendo uma aproximação por meio de um sistema linear. Inicialmente, procuraremos por pontos críticos ou soluções estáveis, que são soluções em que  $x(t)$  e  $y(t)$  permanecem constantes, sem variação, assim, nas proximidades desse ponto crítico as trajetórias do sistema não linear são, num certo sentido, semelhantes às trajetórias de um sistema linear associado. Antes disso, um importante questionamento é saber se o sistema admite solução, caso positivo se ela é única.

**Teorema 2.1.** *O sistema (2.1) munido com uma condição inicial  $(x(t_0), y(t_0)) = (x_0, y_0)$  admite única solução em um determinado intervalo contendo  $t_0$ .*

*Demonstração.* Como as funções  $F(x, y) = xa - \alpha xy$  e  $G(x, y) = -cy + \gamma xy$  são polinomiais, logo, são de classe  $C^\infty$ , ou seja, são infinitamente diferenciáveis com relação às variáveis  $x$  e  $y$  e, além disso, qualquer derivada, de qualquer ordem é contínua. Então, o resultado de Existência e Unicidade de solução para o sistema (2.1) é garantido pelo Teorema 7.1.1 de [5].  $\square$

Observando (2.1), percebemos que um ponto crítico  $(x_c, y_c)$  deve satisfazer  $F(x_c, y_c) = 0 = G(x_c, y_c)$ , ou de forma equivalente, esse ponto deve satisfazer o sistema

$$\begin{cases} x(a - \alpha y) = 0, \\ y(-c + \gamma x) = 0, \end{cases} \quad (2.2)$$

portanto, encontramos duas soluções  $P_1 = (0, 0)$  e  $P_2 = \left(\frac{c}{\gamma}, \frac{a}{\alpha}\right)$ . Como dito no começo desta seção, iremos linearizar o sistema em torno dos pontos críticos encontrados e usar resultados de sistemas lineares para entender localmente como são as soluções.

**Teorema 2.2.** *O sistema (2.1) é localmente linear nas proximidades dos pontos críticos  $P_1$  e  $P_2$ , e o sistema linear que aproxima o não linear é dado por*

$$\begin{cases} \frac{dx}{dt} = F_x(x_c, y_c)(x - x_c) + F_y(x_c, y_c)(y - y_c), \\ \frac{dy}{dt} = G_x(x_c, y_c)(x - x_c) + G_y(x_c, y_c)(y - y_c). \end{cases} \quad (2.3)$$

*Demonstração.* Como  $F(x, y)$  e  $G(x, y)$  possuem derivadas parciais contínuas até segunda ordem, podemos utilizar a Fórmula de Taylor Infinitesimal (ver [12]) em torno dos pontos críticos, de maneira que conseguimos escrever

$$\begin{cases} F(x, y) = F(x_c, y_c) + F_x(x_c, y_c)(x - x_c) + F_y(x_c, y_c)(y - y_c) + \rho_1(x, y), \\ G(x, y) = G(x_c, y_c) + G_x(x_c, y_c)(x - x_c) + G_y(x_c, y_c)(y - y_c) + \rho_2(x, y), \end{cases} \quad (2.4)$$

com os restos infinitesimais de ordem 2  $\rho_1$  e  $\rho_2$  satisfazendo

$$\lim_{(x,y) \rightarrow (x_c, y_c)} \frac{\rho_1(x, y)}{\|(x, y)\|^2} = 0 = \lim_{(x,y) \rightarrow (x_c, y_c)} \frac{\rho_2(x, y)}{\|(x, y)\|^2}. \quad (2.5)$$

Como  $F(x_c, y_c) = 0 = G(x_c, y_c)$ , pois  $(x_c, y_c)$  é um ponto crítico de (2.1), tomando o limite quando  $(x, y) \rightarrow (x_c, y_c)$  e como o resto tende a zero por (2.5), então nas proximidades do ponto crítico podemos reescrever (2.4) como (2.3).  $\square$

Em algumas circunstâncias, é preferível fazer a mudança de variável  $u = x - x_c$  e  $v = y - y_c$ , pois  $\frac{du}{dt} = \frac{dx}{dt}$  e  $\frac{dv}{dt} = \frac{dy}{dt}$ , assim, podemos reescrever o sistema (2.3) da seguinte forma

$$\begin{pmatrix} \frac{du}{dt} \\ \frac{dv}{dt} \end{pmatrix} = \begin{pmatrix} F_x(x_c, y_c) & F_y(x_c, y_c) \\ G_x(x_c, y_c) & G_y(x_c, y_c) \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}. \quad (2.6)$$

A matriz quadrada que aparece em (2.6) é dita a matriz Jacobiana, a qual denotaremos por  $J(x_c, y_c)$ . Uma vez que o sistema (2.6) é linear e homogêneo usaremos a teoria de estabilidade para classificar os pontos críticos, de forma que teremos uma noção de como são as trajetórias de (2.1) nas proximidades desses pontos.

Para determinar o comportamento das soluções nas proximidades de  $P_1$ , utilizamos a expressão em (2.6) e obtemos o seguinte sistema linear

$$\frac{d}{dt} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & 0 \\ 0 & -c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (2.7)$$

Neste ponto, para discutirmos sobre a solução dos sistemas lineares associados, vamos encontrar autovalores e autovetores associados à matriz  $J(P_1) = \begin{pmatrix} a & 0 \\ 0 & -c \end{pmatrix}$ .

Perceba que a equação

$$\det(J(P_1) - \lambda I) = 0,$$

é equivalente a  $(a - \lambda)(-c - \lambda) = 0$ , portanto os autovalores são  $\lambda_1 = a$  e  $\lambda_2 = -c$ . Substituindo  $\lambda_1$  e  $\lambda_2$  em  $J(P_1)v = \lambda v$ , encontramos os autovetores associados  $\xi_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$  associado a  $\lambda_1$  e  $\xi_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$  associado a  $\lambda_2$ . Usando os resultados da Seção 7.5 de [5], a solução geral é da forma

$$\begin{pmatrix} x \\ y \end{pmatrix} = c_1 \begin{pmatrix} 1 \\ 0 \end{pmatrix} e^{at} + c_2 \begin{pmatrix} 0 \\ 1 \end{pmatrix} e^{-ct}. \tag{2.8}$$

De fato, perceba que  $\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} e^{at}$  e  $\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} e^{-ct}$  são soluções de (2.7) substituindo na equação. Calculamos o wronskiano temos

$$W \left[ \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}, \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \right] = \det \begin{pmatrix} e^{at} & 0 \\ 0 & e^{-ct} \end{pmatrix} = e^{(a-c)t} \neq 0, \quad \forall t. \tag{2.9}$$

Dessa forma, pelo Teorema 7.4.3 de [5],  $\begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$  e  $\begin{pmatrix} x_2 \\ y_2 \end{pmatrix}$  formam um conjunto fundamental de soluções, de modo que qualquer outra solução é dada como uma combinação linear dessas, conforme (2.8).

Como as constantes  $a$  e  $c$  são positivas, então os autovalores são reais, distintos e com sinais opostos. Pelos resultados de estabilidade dados pelo Teorema 9.3.2 de [5],  $P_1$  é um ponto de sela instável. Perceba pela Figura 1 que pelo eixo  $y$ , em um determinado instante de tempo, as trajetórias se aproximam da origem e na direção do eixo  $x$ , elas se afastam, além disso, qualquer trajetória no primeiro quadrante que passe próximos aos eixos, também se aproximam e, em seguida, se afastam da origem.

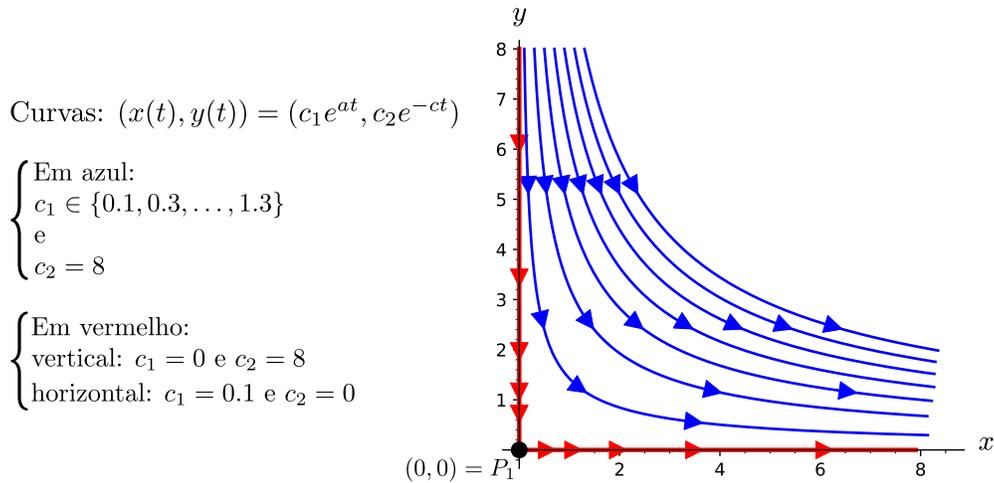


Figura 1: Ponto de sela  $P_1$  e curvas da Equação (2.8) (Fonte: Autores.)

Novamente, para determinar o comportamento das soluções nas proximidades de  $P_2$ , utilizamos a expressão em (2.6) e obtemos o seguinte sistema linear

$$\begin{cases} \frac{d}{dt} \left( x - \frac{c}{\gamma} \right) = \left( a - \alpha \frac{a}{\alpha} \right) \left( x - \frac{c}{\gamma} \right) + \alpha \left( -\frac{c}{\gamma} \right) \left( y - \frac{a}{\alpha} \right) = -\frac{\alpha c}{\gamma} \left( y - \frac{a}{\alpha} \right), \\ \frac{d}{dt} \left( y - \frac{a}{\alpha} \right) = \left( \gamma \frac{a}{\alpha} \right) \left( x - \frac{c}{\gamma} \right) + \left( -c + \gamma \frac{c}{\gamma} \right) \left( y - \frac{a}{\alpha} \right) = \frac{\gamma a}{\alpha} \left( x - \frac{c}{\gamma} \right). \end{cases} \quad (2.10)$$

Considere a mudança de variável  $u = x - \frac{c}{\gamma}$  e  $v = y - \frac{a}{\alpha}$ , a tradução matricial do sistema (2.10) assume o seguinte formato,

$$\frac{d}{dt} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 0 & -\frac{\alpha c}{\gamma} \\ \frac{\gamma a}{\alpha} & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}. \quad (2.11)$$

Análogo ao que fizemos antes, o cálculo dos autovalores da matriz em (2.11) nos fornece dois valores  $\lambda_3 = -i\sqrt{ac}$  e  $\lambda_4 = i\sqrt{ac}$ , ou seja, são números complexos conjugados com parte real nula, e novamente pelo Teorema 9.3.2 de [5], o ponto crítico  $P_2$  é um centro estável para o sistema linear. Do sistema acima, obtemos a seguinte relação diferencial

$$\gamma^2 au \, du + \alpha^2 cv \, dv = 0, \quad (2.12)$$

Portanto,

$$\gamma^2 au^2 + \alpha^2 cv^2 = k, \quad (2.13)$$

Aqui,  $k$  é uma constante de integração não negativa. Logo, as trajetórias do sistema linear dado por (2.11) são elipses. Mais ainda, as soluções podem ser encontradas explicitamente (ver [5]) e serão dadas por

$$u(t) = -\frac{\alpha c}{\sqrt{ac\gamma}} \operatorname{sen}(t\sqrt{ac}) v(0) + \cos(t\sqrt{ac}) u(0)$$

e

$$v(t) = \frac{a\gamma}{\sqrt{ac\alpha}} \operatorname{sen}(t\sqrt{ac}) u(0) + \cos(t\sqrt{ac}) v(0),$$

e que, tomando  $u(0) = \frac{c}{\gamma} K \cos(\phi)$  e  $v(0) = \frac{a}{\alpha} K \sqrt{\frac{c}{a}} \operatorname{sen}(\phi)$ , reescrevemos esta solução em um formato mais simples

$$u = \frac{c}{\gamma} K \cos(t\sqrt{ac} + \phi), \quad v = \frac{a}{\alpha} \sqrt{\frac{c}{a}} K \operatorname{sen}(t\sqrt{ac} + \phi). \quad (2.14)$$

Finalmente, desfazendo a mudança de variável, temos

$$x = \frac{c}{\gamma} + \frac{c}{\gamma}K \cos(t\sqrt{ac} + \phi) \quad \text{e} \quad y = \frac{a}{\alpha} + \frac{a}{\alpha}\sqrt{\frac{c}{a}}K \sin(t\sqrt{ac} + \phi). \quad (2.15)$$

A Figura 2 contém o retrato de fase com as trajetórias de (2.10) e deve ser interpretada da seguinte maneira: localmente, nas proximidades do ponto  $P_2$  as trajetórias do sistema não linear são semelhantes a elipses. Mais ainda, como a aproximação dada pelo método, supostamente, não funciona se o ponto não for um ponto crítico, à medida que nos afastamos do ponto destacado na Figura 2, devemos esperar que as trajetórias já não se comportem mais como elipses.

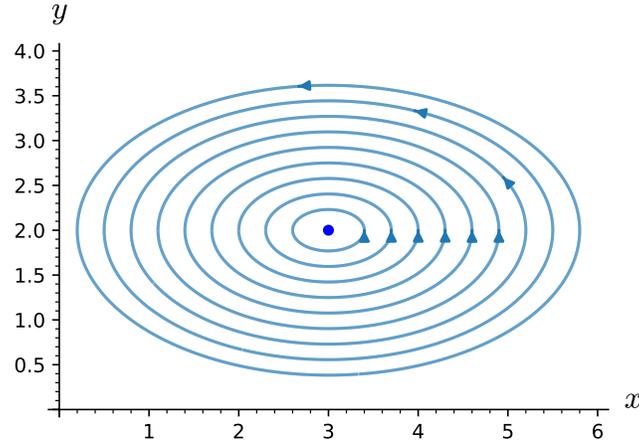


Figura 2: Gráfico de um retrato de fase pela aproximação linear, com  $a = 1$ ,  $\alpha = 0.5$ ,  $c = 0.75$  e  $\gamma = 0.25$ . (Fonte: Autores.)

Além disso, é esperado que as trajetórias do sistema não linear permaneçam no primeiro quadrante, uma vez que há trajetórias percorrendo os eixos  $x$  e  $y$  relacionadas ao sistema linear em torno do ponto  $P_1$ .

Uma vez que obtivemos soluções explícitas para as populações de predadores e presas, faremos agora algumas análises dessas soluções de modo a concluir, localmente, alguns de seus aspectos qualitativos.

**Proposição 2.3.** *As populações de predadores e presas variam de forma senoidal com período  $\frac{2\pi}{\sqrt{ac}}$ .*

*Demonstração.* Sejam  $s, t \in \mathbb{R}$  tais que  $s - t = \frac{2\pi}{\sqrt{ac}}$ , então

$$\begin{aligned}\cos(s\sqrt{ac} + \phi) &= \cos\left(\left[t + \frac{2\pi}{\sqrt{ac}}\right]\sqrt{ac} + \phi\right) = \cos([t\sqrt{ac} + \phi] + 2\pi) \\ &= \cos(t\sqrt{ac} + \phi)\cos(2\pi) - \sin(t\sqrt{ac} + \phi)\sin(2\pi) \\ &= \cos(t\sqrt{ac} + \phi).\end{aligned}$$

Isso nos garante que  $x(s) = x(t)$ . Analogamente, pode ser provado que, se  $s$  e  $t$  estão defasados de  $\frac{2\pi}{\sqrt{ac}}$ ,  $\sin(s\sqrt{ac} + \phi) = \sin(t\sqrt{ac} + \phi)$  donde  $y(s) = y(t)$ . Ainda observando as soluções dadas em (2.15), é fácil perceber que as amplitudes das populações de presas e predadores são dadas respectivamente por  $\frac{cK}{\gamma}$  e  $\frac{aK}{\alpha}\sqrt{\frac{c}{a}}$ , portanto, dependem apenas dos parâmetros do modelo e das condições iniciais.  $\square$

**Proposição 2.4.** *As populações médias de predadores e presas em um ciclo completo são numericamente iguais as populações de equilíbrio, ou seja,  $(x_{med}, y_{med}) = \left(\frac{c}{\gamma}, \frac{a}{\alpha}\right)$ .*

*Demonstração.* As populações médias, em um ciclo completo, são dadas por

$$y_{med} = \frac{1}{T} \int_t^{t+T} y(s) ds \quad \text{e} \quad x_{med} = \frac{1}{T} \int_t^{t+T} x(s) ds, \quad (2.16)$$

na quais  $t > 0$  representa um instante qualquer e  $T = \frac{2\pi}{\sqrt{ac}}$ . Utilizando a expressão em (2.15) para  $y$  em (2.16), obtemos

$$\begin{aligned}y_{med} &= \frac{1}{T} \int_t^{t+T} \left[ \frac{a}{\alpha} + \frac{a}{\alpha} \sqrt{\frac{c}{a}} K \sin(s\sqrt{ac} + \phi) \right] ds \\ &= \frac{1}{T} \int_t^{t+T} \left[ \frac{a}{\alpha} \right] ds + \frac{1}{T} \int_t^{t+T} \left[ \frac{a}{\alpha} \sqrt{\frac{c}{a}} K \sin(s\sqrt{ac} + \phi) \right] ds.\end{aligned} \quad (2.17)$$

Integrando o primeiro termo da soma, temos

$$\frac{1}{T} \int_t^{t+T} \left[ \frac{a}{\alpha} \right] ds = \frac{a}{\alpha}. \quad (2.18)$$

E ao integrar o segundo termo da soma, ficamos com

$$\frac{1}{T} \int_t^{t+T} \left[ \frac{a}{\alpha} \sqrt{\frac{c}{a}} K \sin(s\sqrt{ac} + \phi) \right] ds = \frac{1}{T} \frac{a}{\alpha} \sqrt{\frac{c}{a}} K \int_t^{t+T} \sin(s\sqrt{ac} + \phi) ds = (\star). \quad (2.19)$$

Fazendo a mudança de variável  $s\sqrt{ac} + \phi = S$  e como  $T\sqrt{ac} = 2\pi$ , obtemos

$$\begin{aligned} (\star) &= \frac{1}{T} \frac{a}{\alpha} \sqrt{\frac{c}{a}} K \int_{t\sqrt{ac}+\phi}^{t\sqrt{ac}+\phi+T\sqrt{ac}} \frac{\text{sen}(S)}{\sqrt{ac}} dS \\ &= \frac{1}{\sqrt{ac}T} \frac{a}{\alpha} \sqrt{\frac{c}{a}} K \int_{t\sqrt{ac}+\phi}^{t\sqrt{ac}+\phi+2\pi} \text{sen}(S) dS \\ &= 0, \end{aligned}$$

uma vez que a integral do seno em um ciclo completo é igual a zero. Consequentemente,  $y_{med} = \frac{a}{\alpha}$ . De forma análoga, substituindo o  $x(t)$  dado em (2.15) na Equação (2.16), temos

$$x_{med} = \frac{1}{T} \int_t^{t+T} x(s) ds = \frac{1}{T} \int_t^{t+T} \frac{c}{\gamma} ds + \frac{1}{T} \frac{c}{\gamma} K \int_t^{t+T} \cos(s\sqrt{ac} + \phi) ds = \frac{c}{\gamma}.$$

□

Vamos agora calcular as populações máximas de presas e predadores, usando a aproximação (2.15).

**Proposição 2.5.** *Sejam  $x$  e  $y$  dadas por (2.15). Denotando  $t_x$  e  $t_y$  os pontos críticos de  $x$  e  $y$  respectivamente, temos*

$$t_x = \frac{n\pi - \phi}{\sqrt{ac}} \quad e \quad t_y = \left( \frac{\pi}{2} + n\pi - \phi \right) \frac{1}{\sqrt{ac}}, \quad n \in \mathbb{N}.$$

Além disso,

- (a) Se  $n = 2m$ ,  $m \in \mathbb{N}$ ,  $x$  e  $y$  possuem um máximo local (e global) em  $t_x$  e  $t_y$ , respectivamente;
- (b) Por outro lado, se  $n = 2m + 1$ ,  $m \in \mathbb{N}$  então  $x$  e  $y$  assumem mínimo local (e global) em  $t_x$  e  $t_y$ , respectivamente;
- (c) A população de predador tem um atraso de um quarto de ciclo com relação ao ciclo da população de presas.

*Demonstração.* Primeiramente, encontraremos os pontos críticos  $t_x$  e  $t_y$  de  $x$  e  $y$ . Suponha  $K > 0$ , temos

$$\frac{dx}{dt} = \frac{d}{dt} \left( \frac{c}{\gamma} + \frac{c}{\gamma} K \cos(t\sqrt{ac} + \phi) \right) = -\frac{c}{\gamma} K \text{sen}(t\sqrt{ac} + \phi) \sqrt{ac}. \quad (2.20)$$

Para

$$-\frac{c}{\gamma}K\text{sen}(t_x\sqrt{ac} + \phi)\sqrt{ac} = 0 \quad \Rightarrow \quad \text{sen}(t_x\sqrt{ac} + \phi) = 0, \quad (2.21)$$

portanto,  $t_x\sqrt{ac} + \phi$  deve ser um múltiplo de  $\pi$ , donde

$$t_x = \frac{n\pi - \phi}{\sqrt{ac}}, \quad \forall n \in \mathbb{N}.$$

Analogamente,

$$t_y = \left(\frac{\pi}{2} + n\pi - \phi\right) \frac{1}{\sqrt{ac}}, \quad \forall n \in \mathbb{N}.$$

Isto prova a primeira parte da proposição.

Mostraremos agora (a), ou seja, analisaremos os pontos críticos acima para  $n = 2m$ ,  $m \in \mathbb{N}$ , porém utilizando um argumento de periodicidade, é suficiente considerar o caso  $n = 0$ . Nesse caso, os pontos críticos são  $t_x = \frac{-\phi}{\sqrt{ac}}$  e  $t_y = \left(\frac{\pi}{2} - \phi\right) \frac{1}{\sqrt{ac}}$ . Utilizaremos o Teste da Segunda Derivada, que nos informa se um ponto crítico é máximo ou mínimo local em termos do valor da segunda derivada calculada nos pontos críticos. Para mais informações ver [19]. Perceba que

$$\frac{d^2x}{dt^2} = \frac{d}{dt} \left( \frac{dx}{dt} \right) = \frac{d}{dt} \left( -\frac{c}{\gamma}K\text{sen}(t\sqrt{ac} + \phi)\sqrt{ac} \right) = -\frac{c}{\gamma}K \cos(t\sqrt{ac} + \phi)ac.$$

Assim, calculando a segunda derivada no ponto crítico  $t_x$ , obtemos

$$\frac{d^2x}{dt^2} \left( \frac{-\phi}{\sqrt{ac}} \right) = -\frac{c}{\gamma}K \cos \left( \sqrt{ac} \left( \frac{-\phi}{\sqrt{ac}} \right) + \phi \right) ac = -\frac{c}{\gamma}K \cos(0)ac = -\frac{ac^2K}{\gamma} < 0.$$

Então  $t_x$  é ponto de máximo local. Fazendo o mesmo para o ponto  $t_y$ , temos

$$\frac{d^2y}{dt^2} = -\frac{a}{\alpha} \sqrt{\frac{c}{a}} K \text{sen}(t\sqrt{ac} + \phi)ac, \quad (2.22)$$

e calculando o valor em  $t_y$ , obtemos

$$\frac{d^2y}{dt^2} \left( \left( \frac{\pi}{2} - \phi \right) \frac{1}{\sqrt{ac}} \right) = -\frac{a}{\alpha} \sqrt{\frac{c}{a}} K \text{sen} \left( \frac{\pi}{2} \right) ac = -\frac{a^2cK}{\alpha} \sqrt{\frac{c}{a}} < 0.$$

Logo,  $t_y$  é também ponto de máximo local.

Perceba que a diferença entre dois pontos críticos da forma  $n = 2m$  é sempre um múltiplo de  $\frac{2\pi}{\sqrt{ac}}$ , como  $x$  e  $y$  são funções periódicas de período  $\frac{2\pi}{\sqrt{ac}}$ , todos estes pontos críticos serão de máximo local.

Para demonstrar (b), é análogo ao argumento feito em (a), bastaria considerar o caso  $n = 1$ , provar que nessa situação  $t_x$  e  $t_y$  são pontos de mínimo local para  $x$  e  $y$  respectivamente. Em seguida, usar o fato que se  $n = 2m + 1$  com  $m \in \mathbb{N}$ , os pontos críticos também estão defasados em um múltiplo de  $\frac{2\pi}{\sqrt{ac}}$ , e o resultado novamente é obtido por periodicidade.

Provaremos agora (c). Mostraremos que uma vez que a população de presas é máxima (mínima), então após um quarto de ciclo, a população de predadores também será máxima (mínima):

$$t_y - t_x = \left(\frac{\pi}{2} + n\pi - \phi\right) \frac{1}{\sqrt{ac}} - \left(\frac{n\pi - \phi}{\sqrt{ac}}\right) = \frac{\pi}{2\sqrt{ac}} = \frac{2\pi}{4\sqrt{ac}} = \frac{T}{4}.$$

Portanto, apesar dos ciclos fechados de predador-presa terem períodos iguais a  $T$ , a população de predador tem um atraso de  $\frac{T}{4}$  com relação ao ciclo da população de presas.  $\square$

Por meio da análise do sinal da primeira derivada das funções  $x(t)$  e  $y(t)$ , concluiremos sobre as velocidades de crescimento e decrescimento dessas populações nos instantes críticos.

**Proposição 2.6.** *Analizando as variações das populações  $x(t)$  e  $y(t)$  no tempo, podemos concluir que*

- (a) *Quando a população das presas atinge seu número máximo, a velocidade de crescimento da população dos predadores é a maior possível;*
- (b) *Quando a população das presas atinge seu número mínimo, a velocidade de crescimento da população dos predadores é a menor possível;*
- (c) *Quando a população de predadores atinge seu número máximo, a velocidade de crescimento da população de presas é mínima;*
- (d) *Quando a população de predadores atinge seu número mínimo, a velocidade de crescimento da população de presas é máxima.*

*Demonstração.* Faremos a demonstração apenas do item (a), uma vez que os outros itens (b), (c) e (d) são complementemente análogos.

Analisaremos primeiro, a função que nos dá o crescimento da população dos predadores, de modo a encontrar o momento em que a velocidade de crescimento é máxima. Perceba que isso consiste em estudar a derivada da função que nos informa o crescimento, ou seja, estudar os pontos críticos da segunda derivada de  $y$ , ou de maneira

equivalente, o ponto de inflexão de  $y$ . Usando a Equação (2.22) e denotando o ponto de inflexão de  $y$  por  $\tilde{t}_y$ ,

$$\frac{d^2y}{dt^2} = 0 \quad \Rightarrow \quad \tilde{t}_y = \frac{n\pi - \phi}{\sqrt{ac}} = t_x, \quad (2.23)$$

isto é, o ponto no qual a população de presas atinge o seu máximo,  $t_x$  (ver Proposição 2.5), corresponde ao mesmo ponto de inflexão da população de predadores. Para  $n = 0$ , temos  $\tilde{t}_y = \frac{-\phi}{\sqrt{ac}}$ , que já vimos na Proposição 2.5 que consistia em um ponto de máximo local para a população de presas. Agora considere  $\theta > 0$  suficientemente pequeno e os pontos

$$\tilde{t}_{y-} = \frac{-\phi - \theta}{\sqrt{ac}} \quad \text{e} \quad \tilde{t}_{y+} = \frac{-\phi + \theta}{\sqrt{ac}},$$

consequentemente,  $\tilde{t}_{y-} < \tilde{t}_y < \tilde{t}_{y+}$ . Note que

$$\begin{aligned} \frac{d^2y}{dt^2}(\tilde{t}_{y-}) &= -\frac{a}{\gamma} \sqrt{\frac{a}{c}} \operatorname{sen} \left( \sqrt{ac} \cdot \frac{-\phi - \theta}{\sqrt{ac}} + \phi \right) ac = -\frac{a}{\gamma} \sqrt{\frac{a}{c}} \operatorname{sen}(-\theta) ac \\ &= \frac{a}{\gamma} \sqrt{\frac{a}{c}} \operatorname{sen}(\theta) ac > 0, \end{aligned}$$

na última igualdade usamos que o seno é uma função ímpar,  $\theta > 0$  é suficientemente pequeno e, além disso, todos os parâmetros  $a$ ,  $\gamma$  e  $c$  são positivos por hipótese. Perceba agora que

$$\frac{d^2y}{dt^2}(\tilde{t}_{y+}) = -\frac{a}{\gamma} \sqrt{\frac{a}{c}} \operatorname{sen} \left( \sqrt{ac} \cdot \frac{-\phi + \theta}{\sqrt{ac}} + \phi \right) ac = -\frac{a}{\gamma} \sqrt{\frac{a}{c}} \operatorname{sen}(\theta) ac < 0.$$

Ou seja, no intervalo  $(\tilde{t}_{y-}, \tilde{t}_y]$ , temos que  $\frac{d^2y}{dt^2} > 0$ , enquanto no intervalo  $[\tilde{t}_y, \tilde{t}_{y+})$ , temos que  $\frac{d^2y}{dt^2} < 0$ . Portanto, pelo Teste da Primeira Derivada (ver [19])  $\tilde{t}_y$  é um ponto de máximo local para  $\frac{dy}{dt}$ , isso significa que quando a população de presa alcança seu máximo, a velocidade de crescimento da população de predadores é a maior possível. Perceba que isso acontece também para todo  $n = 2m$  com  $m \in \mathbb{N}$ . Analogamente, tratando o caso  $n = 1$ , poderíamos repetir o processo anterior e provar que o ponto de inflexão seria um mínimo para  $\frac{d^2y}{dt^2}$ , ou seja, quando a população de presas atinge seu mínimo (ver Proposição 2.5), a velocidade de crescimento dos predadores é a menor possível. Também de forma completamente análoga, poderíamos repetir os argumentos para  $x(t)$  e concluir os resultados dados por (c) e (d). □

A Figura 3 apresenta o gráfico mostrando essas noções para um ciclo completo.

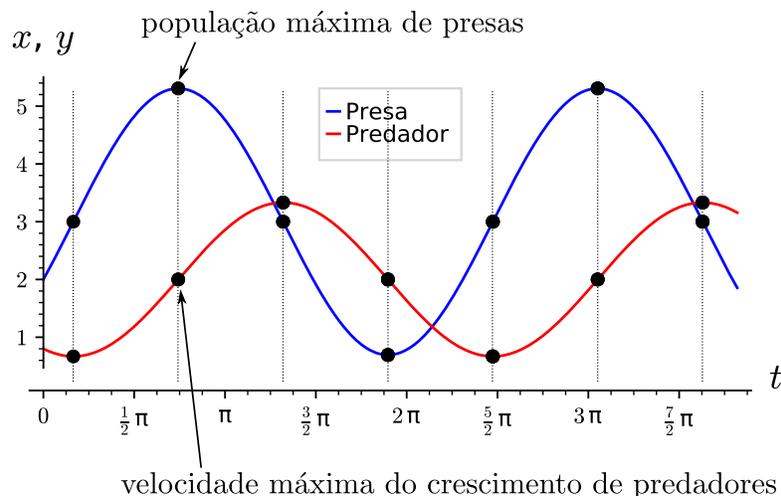


Figura 3: Comparações entre as populações de predadores e presas, com  $a = 1$ ,  $\alpha = 0.5$ ,  $c = 0.75$  e  $\gamma = 0.25$ . O período é  $T = \frac{4\sqrt{3}\pi}{3}$  e valores iniciais  $(x, y) = (2; 0, 8)$ . (Fonte: Autores.)

## 2.2 Análise do Sistema Não Linear

Para concluirmos o estudo do modelo clássico, faremos uma análise das soluções do sistema não linear (2.1). Durante esta e a próxima seção, descreveremos com explicações suficientes, os comandos utilizados por meio do *software* SageMath para as implementações, sejam na busca de soluções para os sistemas ou na composição das figuras. Salientamos que os comandos que disponibilizaremos aqui não retratam a totalidade do material que desenvolvemos para a composição deste artigo. Assim, os pesquisadores e discentes envolvidos com estudos de sistemas de EDOs, como apresentados neste trabalho, poderão usufruir de outras tecnologias que desenvolvemos como por exemplo as calculadoras automáticas de pontos críticos, de campos de vetores, de trajetórias dos sistemas e de gráficos das soluções no tempo, dentre tantos outros aspectos que possivelmente poderão auxiliar estudos nesta área. Relembramos que o endereço eletrônico para acesso ao material está disponibilizado na introdução. Para entender melhor o uso dos métodos usados, assim como a parte de programação, veja [18].

Apesar de sistemas não lineares não terem um método preciso para solucioná-los analiticamente, perceba que podemos reescrever (2.1), utilizando a Regra da Cadeia,

que pode ser vista em [19], da seguinte maneira:

$$\frac{dy/dt}{dx/dt} = \frac{dy}{dx} = \frac{y(-c + \gamma x)}{x(a - \alpha y)}, \quad (2.24)$$

e nessa escrita temos uma EDO separável. A solução geral é descrita implicitamente por meio da equação

$$a \ln(y) - \alpha y + c \ln(x) - \gamma x = C, \quad (2.25)$$

com  $C$  uma constante determinada pelas condições iniciais. Vimos que, no caso linear,  $P_1$  é um ponto de sela instável, uma vez que os autovalores associados são reais e com sinais opostos, e, portanto, pequenas perturbações ainda preservariam esses sinais, de forma que o mesmo ocorre com as trajetórias do não linear nas proximidades de  $P_1$ . Por outro lado,  $P_2$  é um centro estável para as trajetórias do sistema linear, uma vez que os autovalores são complexos puros. Assim, pequenas perturbações poderiam alterar a estabilidade desse ponto e as trajetórias do sistema não linear seriam espirais. Como veremos, nas proximidades de  $P_2$ , as soluções do não linear também se comportam como as trajetórias do linear, de modo que  $P_2$  também é um centro para o sistema não linear. A partir de (2.24) é possível verificar, computacionalmente, que o campo de vetores associados é dado por

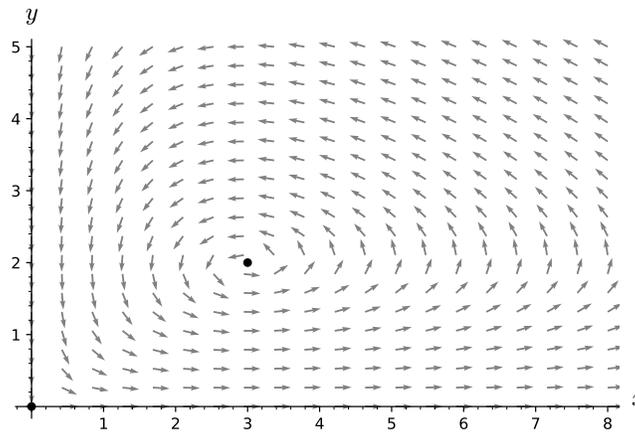


Figura 4: Campo de vetores para (2.1), com  $a = 1$ ,  $\alpha = 0,5$ ,  $c = 0,75$  e  $\gamma = 0,25$ . (Fonte: Autores.)

O campo de vetores acima nos dá uma noção de que as trajetórias de (2.25) são ciclos fechados em torno de  $P_2$ . A Figura 4 foi gerada no SageMath. Para isso, usamos

os métodos `vector` e `plot_vector_field`. Primeiro, precisamos definir a seguinte função vetorial:

$$F(x, y) = \frac{(x(1 - 0,5y), y(-0,75 + 0,25x))}{\|(x(1 - 0,5y), y(-0,75 + 0,25x))\|}$$

Os comandos a seguir definem a função  $F(x, y)$  no SageMath, e guarda na variável `F`.

```
1 y = var('y')
2 F = vector([(x*(1 - 0.5*y)), (y*(-0.75 + 0.25*x)) ])
3 F = F/F.norm()
```

Para gerar o gráfico do campo de vetores, usamos o comando:

```
1 plot_vector_field(F, (x, 0, 8), (y, 0, 5))
```

Até aqui, a figura ainda não exhibe os pontos críticos. Para isso, guardamos nas variáveis `pc1` e `pc2` a figura dos pontos críticos e na variável `cv` guardamos o campo de vetores com alguns parâmetros a mais. Feito isto, basta executar a soma `pc1 + pc2 + cv` para obter o gráfico, conforme o código abaixo:

```
1 pc1 = circle((0.75/0.25, 1/0.5), 0.06, color='black', fill=True)
2 pc2 = circle((0, 0), 0.06, color='black', fill=True)
3 cv = plot_vector_field(F, (x, 0, 8), (y, 0, 5), color='gray',
4     frame=False, axes=True, axes_labels=['$x$', '$y$'])
5 pc1 + pc2 + cv
```

Com o SageMath também podemos ver que as curvas são fechadas para diversos parâmetros e condições iniciais conforme as Figuras 5 e 6, cujos comandos serão mostrados na próxima seção.

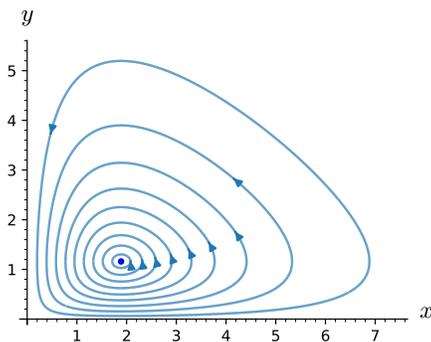


Figura 5: Retrato de fase com  $a = 0.58$ ,  $\alpha = 0.5$ ,  $c = 0.85$  e  $\gamma = 0.45$ .

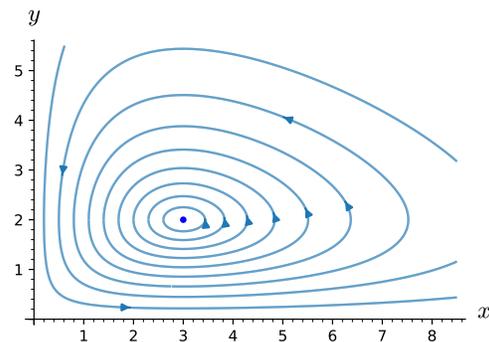


Figura 6: Retrato de fase com  $a = 1$ ,  $\alpha = 0.5$ ,  $c = 0.75$  e  $\gamma = 0.25$ .

É possível mostrar analiticamente (ver Corolário 4.1.9 de [8]) e geometricamente (ver p. 385-386 de [3]) que as curvas definidas por (2.25) são de fato curvas fechadas em torno do ponto crítico  $P_2$ .

Apesar de encontrarmos a solução de (2.24) em um formato implícito, esta solução não nos dá a variação de populações no tempo, isto é, ao ganharmos informações que as trajetórias são curvas fechadas sem auto interseção, não temos uma noção de como elas evoluem no tempo. Sendo assim, para obter respostas acerca da variação das populações em cada ciclo, utilizaremos métodos numéricos. Precisamente, usamos o método de Runge-Kutta de quarta ordem, que é um dos mais usados para obter soluções numéricas de problemas de valor inicial. O erro total acumulado deste método é proporcional a  $h^5$ . Para os detalhes do algoritmo veja [5], p. 373-374. Para detalhes sobre a precisão e estabilidade/instabilidade do algoritmo veja [5], p. 358-376 e [18], p. 376-381.

Com o método de Runge-Kutta de quarta ordem, disponível no SageMath através do método `desolve_system_rk4`, encontramos as variações das populações, como mostrado na Figura 7.

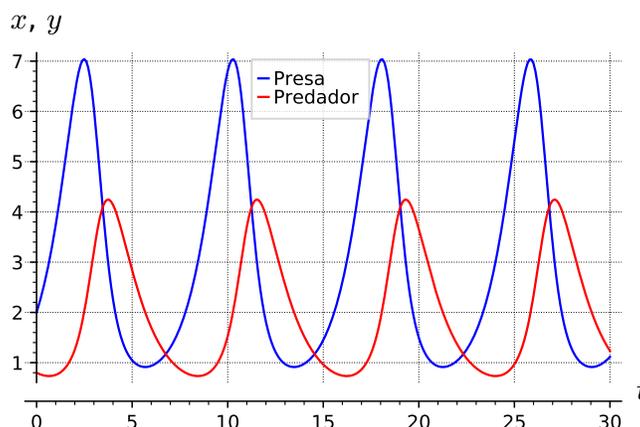


Figura 7: Variações nas populações de presas e de predadores em relação ao tempo para o Sistema (2.1), com  $a = 1, \alpha = 0,5, c = 0,75$  e  $\gamma = 0,25$ . Valores iniciais  $(x, y) = (2; 0, 8)$ . (Fonte: Autores.)

O método `desolve_system_rk4`, no SageMath, possui a seguinte sintaxe:

```
desolve_system_rk4([f,g], vars=[x,y], ics=[t0,x0,y0], ivar=t, end_points=tf)
```

na qual  $[f,g]$  são as funções do sistema,  $[x,y]$  são as variáveis,  $ics=[t0, x0, y0]$  são as condições iniciais,  $ivar=t$  indica a variável independente e  $end\_points=tf$  define o

tempo final. A saída desta função é uma lista com os valores numéricos de  $t$ ,  $x$  e  $y$ , com  $t$  variando de 0, 1 em 0, 1. Para definir um passo  $h$  de tamanho diferente, para o tempo, podemos acrescentar o parâmetro `step=h`. O código abaixo gera a Figura 7.

```

1 y, t = var('y, t')
2 f = x*(1 - 0.5*y)
3 g = y*(-0.75 + 0.25*x)
4 sn = desolve_system_rk4([f, g], vars=[x,y], ics=[0, 2, 0.8],
5 ivar=t, end_points=30)
6
7 l1 = []; l2 = [] #
8 for i in sn: #
9 l1.append([i[0], i[1]]) # guardando as info. temp e x
10 l2.append([i[0], i[2]]) # guardando as info. temp e y
11
12 p1_classic = line(l1, thickness=1.2, gridlines=True,
13 axes_labels=['t$', '$x$', '$y$'], legend_label='Presa')
14 p2_classic = line(l2, color='red', thickness=1.2, legend_label='Predador')
15 p1_classic + p2_classic

```

Uma vez que a solução é dada implicitamente, encontrar os pontos  $x$  e  $y$  de máximos e mínimos em um ciclo, não é um trabalho elementar de ser resolvido simbolicamente e explicitamente. De fato, como as trajetórias são centradas em  $P_2$ , bastaria fazer as interseções das trajetórias dadas por (2.25) com as retas  $x = \frac{c}{\gamma}$  e  $y = \frac{a}{\alpha}$ . Perceba, porém, que agora precisamos resolver explicitamente as seguintes equações

$$a \ln\left(\frac{a}{\alpha}\right) - a + c \ln(x) - \gamma x = C,$$

e

$$a \ln(y) - \alpha y + c \ln\left(\frac{c}{\gamma}\right) - c = C,$$

que não possuem solução em termos de funções elementares. Apesar disso, ainda é possível obter soluções para casos particulares usando métodos numéricos. Sendo assim, em comparação como o caso linear, encontrar os máximos e mínimos em um ciclo já é um problema muito mais difícil de ser resolvido.

Antes de obter as soluções numéricas, obtemos a solução para a condição inicial  $(x, y) = (2; 0, 8)$  com os comandos a seguir.

```

1 x, z = var('x, z')
2 y = function('y')(x)

```

```

3 sol = desolve(diff(y,x)==(y*(-0.75+0.25*x))/(x*(1-0.5*y)),y,[2,0.8])
4 solz = sol.subs(y==z)
5 show(solz)

```

$$-2z + 4 \log(z) = x + 3 \log(2) - 3 \log(x) - \frac{33892164}{7544041}$$

Observe que na linha 4 foi feita uma mudança de variável, pois  $y$  era uma função de  $x$  e vamos precisar de uma variável do tipo `var`. Usando a solução que está guardada em `solz`, podemos substituir  $x$  por 3 e usar o método numérico `find_root`, que procura pelos zeros de uma função  $f$  em um intervalo  $(a, b)$  através da sintaxe:

`find_root(f, a, b).`

Os códigos a seguir, exibem as soluções para  $y$  variando nos intervalos  $(0, 2)$  e  $(4, 5)$ .

```

1 s1=find_root(solz(x=3).lhs()-solz(x=3).rhs(), 0, 2)
2 s2=find_root(solz(x=3).lhs()-solz(x=3).rhs(), 4, 5)
3 (3, s1), (3, s2)

```

`((3, 0.7328463611981351), (3, 4.24683077371136))`

Ou seja, encontramos

$$(x, y) = (3; 0, 7328463611981351) \text{ e } (x, y) = (3; 4, 24683077371136).$$

De maneira análoga, fazendo a substituição no  $z$ , em vez de  $x$ , encontramos

$$(x, y) = (0, 9131473610947365; 2) \text{ e } (x, y) = (7, 040951217170739; 2).$$

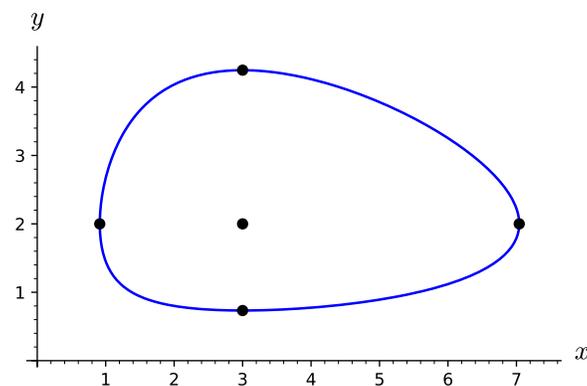


Figura 8: Máximos e Mínimos em uma trajetória com parâmetros  $a = 1$ ,  $\alpha = 0,5$ ,  $c = 0,75$  e  $\gamma = 0,25$  e valor inicial  $(2; 0, 8)$ . (Fonte: Autores.)

Este procedimento então nos fornece as populações máximas/mínimas de predadores e presas.

Prosseguindo conforme a análise do caso linear, vamos calcular agora a população média de presas e de predadores numericamente. Como estamos no caso discreto, usaremos a seguinte aproximação para o valor médio da função  $f$ ,

$$f_{med} \approx \frac{\Delta t(f(t_1) + \dots + f(t_n))}{T} = \frac{\Delta t(f(t_1) + \dots + f(t_n))}{\Delta t \cdot \text{qtd. de pontos}} = \frac{f(t_1) + \dots + f(t_n)}{\text{qtd. de pontos}}, \quad (2.26)$$

em que  $T$  é o período e  $t_1, \dots, t_n$  são igualmente espaçados com tamanho  $\Delta t$ , e esta aproximação melhora à medida que aumentamos a quantidade de pontos. Perceba que encontrar o período também não é um trabalho elementar, já que para cada ciclo, teremos um comportamento diferente para a trajetória. Comentaremos sobre isso mais à frente. Fazendo testes com o tempo final, percebemos que para  $T = 7,786617899$  obtemos praticamente um período para o caso da Figura 8. O ponto final encontrado foi  $(x, y) = (1,99999999811233; 0,8000000000314704)$ , cuja distância ao ponto inicial é  $1,91372322945032 \times 10^{-10}$ . Abaixo, temos o código que guarda na variável `sn` a solução numérica para este problema, porém com passos de  $0,00001$ .

```
1 x, y, t = var('x, y, t')
2 f = x*(1 - 0.5*y)
3 g = y*(-0.75 + 0.25*x)
4 sn = desolve_system_rk4([f, g], [x, y], ics=[0, 2, 0.8], ivar=t,
5                       end_points=7.786617899, step=0.00001)
```

Calculamos a distância entre o ponto final e o inicial no bloco de código a seguir.

```
1 v=vector((sn[-1][1], sn[-1][2]))-vector((2,0.8))
2 norm(v)
```

1.913723229450319e-10

O código seguinte calcula a média da população  $x$  utilizando (2.26).

```
1 listax=[]
2 for i in sn:
3     listax.append(i[1])
4 sum(listax)/len(listax)
```

2.9999984459449127

Analogamente, o código abaixo nos fornece a média da população  $y$  conforme (2.26).

```

1 listay=[]
2 for i in sn:
3     listay.append(i[2])
4 sum(listay)/len(listay)

```

1.99999813513556

Assim como a Proposição 2.4, como os valores entre o valor médio das populações e as populações de equilíbrio são próximas umas das outras, com erro de aproximação para a população de presas e predadores iguais a  $1,554055087310546 \times 10^{-6}$  e  $1,86486443998213 \times 10^{-6}$  respectivamente, esperamos que ainda no caso não linear as populações médias sejam iguais as populações de equilíbrio. De fato, além de simularmos o exemplo acima, fizemos diversas simulações com parâmetros diferentes e ainda encontramos resultados muito próximos de  $(3, 2)$ , até mesmo para trajetórias extremamente afastadas. A saber, testamos o valor inicial  $(0, 04; 2)$ , usando tempo final  $T = 11,05473819$  e obtemos a distância entre o ponto final e o inicial de aproximadamente  $9,713896356203176 \times 10^{-11}$ . Usando o mesmo passo do caso anterior obtemos  $x_{med} \approx 2,99999683775806$  e  $y_{med} \approx 2,000000000000958$ , apresentando um erro de  $9,57900425646585 \times 10^{-13}$  e  $3,16224193985803 \times 10^{-6}$  respectivamente.

### 2.3 Aproximação Linear vs Métodos Numéricos

Nesta seção, vamos comparar os resultados obtidos por meio da linearização e do método numérico que usa o algoritmo de Runge-Kutta de quarta ordem.

O gráfico das variações nas populações de presas e de predadores em relação ao tempo, presente nas Figuras 3 e 7, foram gerados usando a solução da aproximação linear e o algoritmo de Runge-Kutta, respectivamente. A seguir, vamos comparar os gráficos, mostrando os comandos do SageMath utilizados.

Para o gráfico da solução da EDO dada pela Equação (2.11), com as condições iniciais  $(x, y) = (2; 0, 8)$ , obtemos inicialmente a solução no SageMath com os comandos a seguir:

```

1 a = 1; alpha = 0.5; c = 0.75; gamma = 0.25
2 t= var('t')
3 u = function('u')(t); v = function('v')(t)
4 sedo=[diff(u, t)==-(alpha*c)/(gamma)*v, diff(v, t)==(a*gamma)/(alpha)*u]
5 sol = desolve_system(sedo, [u,v], [0, 2-0.75/0.25, 0.8-1/0.5])
6 show(sol)

```

$$\left[ u(t) = \frac{6}{5} \sqrt{3} \sin\left(\frac{1}{2} \sqrt{3}t\right) - \cos\left(\frac{1}{2} \sqrt{3}t\right), v(t) = -\frac{1}{3} \sqrt{3} \sin\left(\frac{1}{2} \sqrt{3}t\right) - \frac{6}{5} \cos\left(\frac{1}{2} \sqrt{3}t\right) \right]$$

Fazendo a substituição  $u = x - 3$  e  $v = y - 2$ , obtemos

$$\begin{cases} x(t) = 3 + \frac{6}{5} \sqrt{3} \sin\left(\frac{1}{2} \sqrt{3}t\right) - \cos\left(\frac{1}{2} \sqrt{3}t\right), \\ y(t) = 2 - \frac{1}{3} \sqrt{3} \sin\left(\frac{1}{2} \sqrt{3}t\right) - \frac{6}{5} \cos\left(\frac{1}{2} \sqrt{3}t\right). \end{cases} \quad (2.27)$$

Para gerar o gráfico com a solução guardada na variável `sol` do bloco de códigos anterior, com  $0 \leq t \leq 30$ , usamos os comandos a seguir

```

1 F = 0.75/0.25 + sol[0].rhs()
2 G = 1/0.5 + sol[1].rhs()
3 p1l = plot(F, (0, 30), color='black', axes_labels=['t$', '$x,y$'],
4         legend_label='Presas')
5 p2l = plot(G, (0, 30), color='purple', gridlines=True,
6         legend_label='Predador')
7 p1l+p2l

```

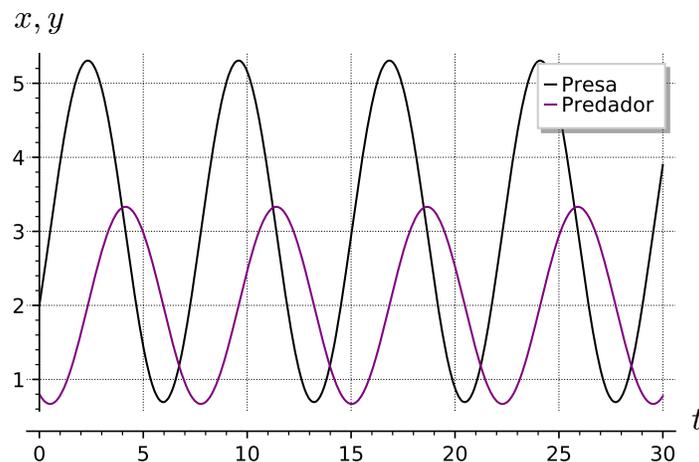


Figura 9: Gráfico da solução (2.27). (Fonte: Autores.)

Podemos comparar os gráficos das Figuras 7 e 9 com os comandos que estão nas linhas 1 e 2 do bloco de códigos a seguir, gerando as Figuras 10 e 11.

```

1 p1_classic+p1l
2 p2_classic+p2l

```

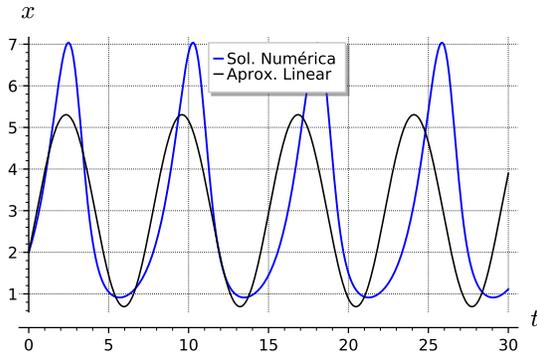


Figura 10: Solução numérica e solução linear ( $t \times x$ ) para a populações de presas. (Fonte: Autores.)

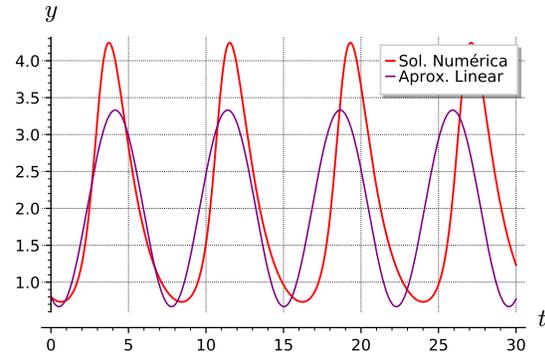


Figura 11: Solução numérica e solução linear ( $t \times y$ ) para a populações de predadores. (Fonte: Autores.)

Observe que `p1_classic` e `p2_classic` fazem parte do bloco de código que gera a Figura 7. A partir das Figuras 10 e 11, observamos que apesar das trajetórias do sistema não linear nas proximidades do ponto crítico serem semelhantes a elipses (trajetórias do linearizado), com o gráfico das variações no tempo conseguimos ver que as amplitudes entre as soluções do linear e não linear para parâmetros similares ainda são bem diferentes, e também as curvas não possuem o mesmo período, basta perceber que as trajetórias não se sobrepõem ao decorrer do tempo.

Abaixo, temos dois retratos de fase gerados com os dados da saída do algoritmo de Runge-Kutta. No gráfico da esquerda, é possível notar alguns segmentos de reta, enquanto no gráfico da direita está mais suave, pois foi exigido uma quantidade maior de passos.

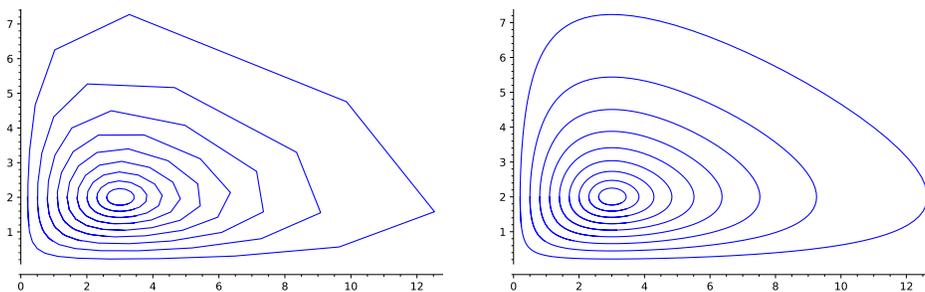


Figura 12: Gráficos gerados por duas soluções numéricas com números de passos distintos. (Fonte: Autores.)

Na figura acima, em ambos os casos o tempo variou de 0 até 10. Para cada curva fechada, no gráfico da esquerda foram calculados 21 passos, já para o gráfico da direita, foram 251 passos. Perceba que mesmo com uma quantidade pequena de passos, o algoritmo ainda nos fornece uma aproximação razoável para as trajetórias, mais ainda, nas proximidades do ponto crítico, note que as trajetórias mais internas das duas imagens virtualmente são iguais. É notável que, assim previsto pelo método (ver [5]), uma maior quantidade de passos nos fornece uma aproximação numérica mais eficiente, ao passo que o custo computacional para a geração dessas figuras também aumenta significativamente. A figura da esquerda foi gerada com um tempo 413 ms, já a da direita exigiu 760 ms. Por fim, apenas para simulação, fizemos uma implementação com 10001 passos e o tempo necessário foi de 16,2 s. Essas simulações foram executadas em um *PC* com processador *Core I5 9400* com *16Gb* de memória ram.

Na figura abaixo, colocamos o gráfico da Figura 2 sobre o gráfico da Figura 6.

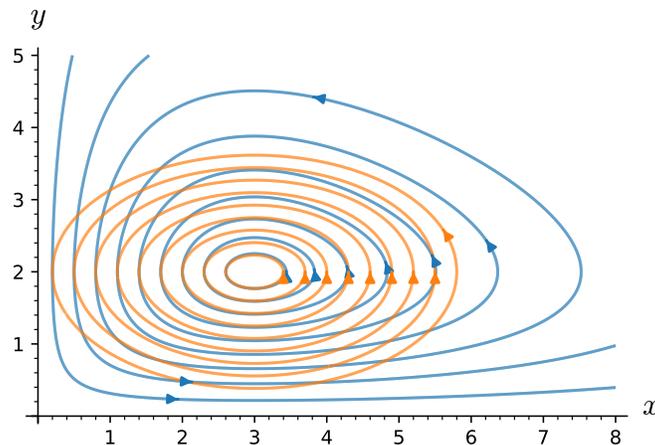


Figura 13: Gráfico com a união das Figuras 2 e 6. (Fonte: Autores.)

Com a Figura 13, comprovamos o que havíamos comentado na análise anterior. Podemos ver que as trajetórias mais internas tendem a se sobrepor, mas não são iguais, e à medida que tomamos trajetórias mais afastadas de  $P_2$  as curvas já são completamente diferentes. A Figura 13 pode ser gerada pelo bloco de códigos a seguir.

```

1 x,y = var('x,y')
2 vi = [[0.2, 2.0], [0.5, 2.0], [0.8, 2.0], [1.1, 2.0], [1.4, 2.0],
3       [1.7, 2.0], [2.0, 2.0], [2.3, 2.0], [2.6, 2.0]]
4
5 f1 = x*(1 - 0.5*y)          # não linear

```

```

6 g1 = y*(-0.75 + 0.25*x) # não linear
7 spl=streamline_plot((f1, g1), (x,0, 8.5), (y,0,5.5), start_points=vi,
8                       density=50, aspect_ratio=1)
9
10 f2 = -(0.5*0.75)/(0.25)*(y-2) # linear
11 g2 = +(1*0.25)/(0.5)*(x-3) # linear
12 sp2=streamline_plot((f2, g2), (x,0, 6), (y,0,4), start_points=vi,
13                       density=50, aspect_ratio=1, frame=False,
14                       axes_labels=['x$', 'y$'], figsize=5)
15
16 spl+sp2

```

De acordo com a Proposição 2.3, usando o método `desolve_system_rk4`, definimos um período de  $2\pi/\sqrt{ac}$  para  $a = 1$  e  $c = 0,75$ , ao simular a evolução das trajetórias de 9 curvas, cada uma partindo do ponto inicial  $(j, 2)$  com  $j$  variando de 0,2 até 2,6 de 0,3 em 0,3. Um fato interessante nesta análise é que observando as Tabelas 1 e 2 cujos tempos finais são  $T_1 = 7$  e  $T_2 = 2\pi/\sqrt{0,75} \approx 7,25519745693687$  respectivamente, o ângulo formado entre os pontos iniciais e finais, com relação à  $P_2$  é praticamente constante para todas as trajetórias, ciente que o método numérico nos dá um resultado aproximado.

	$p_0$	$p_f$	$\ p_f - p_0\ $	$\theta$ (em rad.)
$C_1$	(0.200, 2.0)	(0.26810, 2.3544)	0.36086	0.128997314034546
$C_2$	(0.500, 2.0)	(0.56081, 2.3164)	0.32220	0.128997314034544
$C_3$	(0.800, 2.0)	(0.85351, 2.2784)	0.28353	0.128997314034547
$C_4$	(1.10, 2.0)	(1.1462, 2.2405)	0.24487	0.128997314034544
$C_5$	(1.40, 2.0)	(1.4389, 2.2025)	0.20621	0.128997314034547
$C_6$	(1.70, 2.0)	(1.7316, 2.1645)	0.16754	0.128997314034544
$C_7$	(2.00, 2.0)	(2.0243, 2.1266)	0.12888	0.128997314034543
$C_8$	(2.30, 2.0)	(2.3170, 2.0886)	0.090215	0.128997314034540
$C_9$	(2.60, 2.0)	(2.6097, 2.0506)	0.051551	0.128997314034554

Tabela 1:  $p_0 = (x(0), y(0))$  e  $p_f = (x(T_1), y(T_1))$ , com  $T_1 = 7$ . (Fonte: Autores.)

A última observação toma forma visual quando construímos as trajetórias relacionadas com os dados da Tabela 1 acima. Esta construção está feita na figura abaixo.

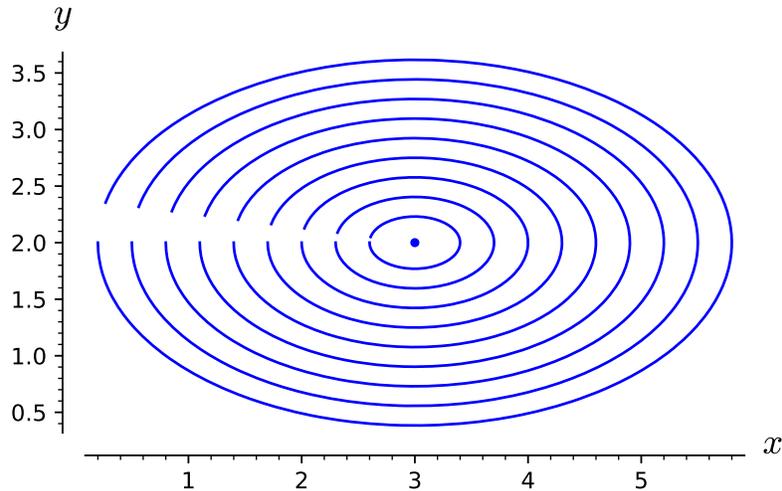


Figura 14: Trajetórias relacionadas com os parâmetros da Tabela 1 e tempo final  $T = 7$ .

Perceba que como o tempo final  $T_1$  é menor do que o tempo aproximado de um período, dado por  $T_2$ , é esperado que as trajetórias não completassem um ciclo inteiro, mas já era esperado que os pontos finais estivessem contidos em uma mesma reta, uma vez que para o sistema linear, o período é constante e completamente determinado apenas pelos parâmetros  $a$  e  $c$ .

Na Tabela 2, fizemos a construção análoga a anterior, mas considerando o tempo final  $T_2 = 2\pi/\sqrt{0,75}$ . Perceba agora que o ângulo formado entre os pontos iniciais e finais, com relação à  $P_2$  é praticamente igual a zero.

	$p_0$	$p_f$	$\ p_f - p_0\ $	$\theta$ (em rad.)
$C_1$	(0.200, 2.0)	(0.20000, 2.0000)	$4.7584 \times 10^{-10}$	$1.49011611938477 \times 10^{-8}$
$C_2$	(0.500, 2.0)	(0.50000, 2.0000)	$4.2485 \times 10^{-10}$	0.0000000000000000
$C_3$	(0.800, 2.0)	(0.80000, 2.0000)	$3.7387 \times 10^{-10}$	$1.49011611938477 \times 10^{-8}$
$C_4$	(1.10, 2.0)	(1.1000, 2.0000)	$3.2289 \times 10^{-10}$	0.0000000000000000
$C_5$	(1.40, 2.0)	(1.4000, 2.0000)	$2.7191 \times 10^{-10}$	0.0000000000000000
$C_6$	(1.70, 2.0)	(1.7000, 2.0000)	$2.2092 \times 10^{-10}$	0.0000000000000000
$C_7$	(2.00, 2.0)	(2.0000, 2.0000)	$1.6994 \times 10^{-10}$	0.0000000000000000
$C_8$	(2.30, 2.0)	(2.3000, 2.0000)	$1.1896 \times 10^{-10}$	0.0000000000000000
$C_9$	(2.60, 2.0)	(2.6000, 2.0000)	$6.7980 \times 10^{-11}$	$2.10734242552812 \times 10^{-8}$

Tabela 2:  $p_0 = (x(0), y(0))$  e  $p_f = (x(T_2), y(T_2))$ , com  $T_2 = 2\pi/\sqrt{0,75}$ . (Fonte: Autores.)

E as trajetórias relacionadas aos parâmetros da Tabela 2 estão dispostas na figura abaixo.

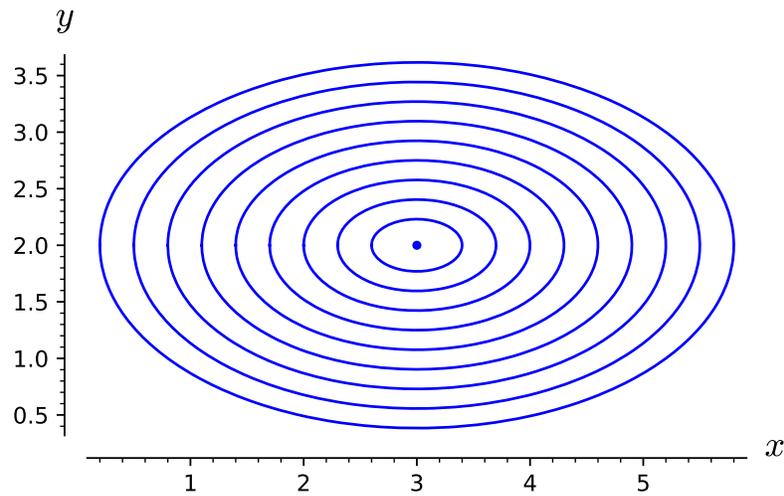


Figura 15: Trajetórias relacionadas com os parâmetros da Tabela 2 e tempo final  $T_2 = 2\pi/\sqrt{ac}$ .

Agora, com a mesma implementação usada para gerar as Tabelas 1 e 2 e Figuras 14 e 15, trocando apenas o sistema linear pelo não linear, mostramos que as trajetórias do sistema não linear não possuem período constante. De fato, usando o tempo final  $T = 2\pi/\sqrt{ac}$  (que é o mesmo tempo necessário para as trajetórias do sistema linear completarem um ciclo) para  $a = 1$  e  $c = 0,75$ . Verificamos que quanto mais próximo a curva está do ponto crítico, menor é o ângulo formado entre o ponto inicial e o ponto final, com relação à  $P_2$ . Isso pode ser visto na tabela abaixo.

	$p_0$	$p_f$	$\ p_f - p_0\ $	$\theta$ (em rad.)
$C_1$	(0.200, 2.0)	(1.7004, 6.9520)	5.1744	1.31415238646799
$C_2$	(0.500, 2.0)	(0.76248, 3.7691)	1.7884	0.669007037448034
$C_3$	(0.800, 2.0)	(0.90557, 2.8211)	0.82788	0.373633427453272
$C_4$	(1.10, 2.0)	(1.1445, 2.4110)	0.41343	0.217995860041385
$C_5$	(1.40, 2.0)	(1.4173, 2.2043)	0.20506	0.128390351727821
$C_6$	(1.70, 2.0)	(1.7058, 2.0951)	0.095281	0.0733527202760797
$C_7$	(2.00, 2.0)	(2.0015, 2.0386)	0.038636	0.0386459481190058
$C_8$	(2.30, 2.0)	(2.3002, 2.0120)	0.012038	0.0171983091732256
$C_9$	(2.60, 2.0)	(2.6000, 2.0021)	0.0020687	0.00517184020218028

Tabela 3:  $p_0 = (x(0), y(0))$  e  $p_f = (x(T), y(T))$ , com  $T = 2\pi/\sqrt{0,75}$ . (Fonte: Autores.)

A imagem de cada trajetória descrita a partir dos parâmetros da Tabela 3 é resumida na Figura 16 abaixo, e percebemos rapidamente que os ciclos aparentemente se fecham nas proximidades do ponto crítico, porém à medida que nos afastamos de  $P_2$ , as trajetórias percorrem apenas parte do ciclo. Isso se comprova com as informações da tabela acima quando o ângulo  $\theta$  é menor quanto mais próximo  $p_0$  está de  $P_2$ .

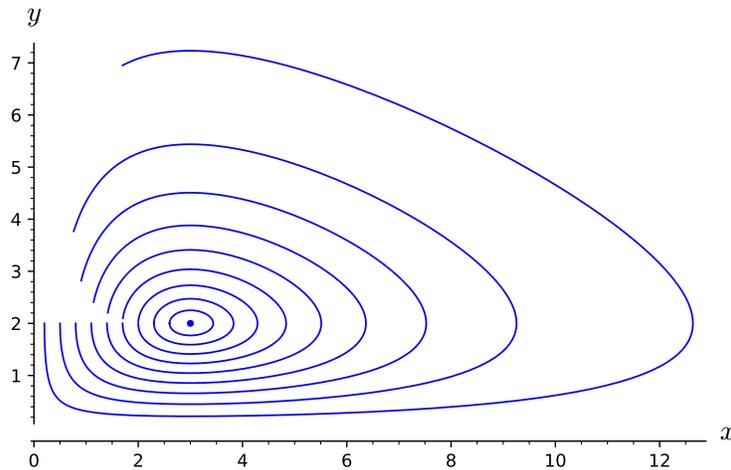


Figura 16: Trajetórias relacionadas com os parâmetros da Tabela 3 e tempo final  $T = 2\pi/\sqrt{ac}$ . (Fonte: Autores.)

Sendo assim, como o período não é mais constante no caso não linear, o atraso entre as populações de predadores não é mais necessariamente de um quarto de ciclo

com relação ao ciclo da população de presas, de modo que, geralmente, nenhuma das propriedades mostradas no caso linear se preservam para as trajetórias do sistema não linear, com exceção das trajetórias extremamente próximas de  $P_2$ , como mostramos em nossas implementações.

### 3 Conclusão

Neste trabalho, desenvolvemos parte da teoria envolvendo sistemas de EDOs não lineares de primeira ordem. Precisamente, fizemos um estudo detalhado sobre o modelo clássico do sistema predador-presa, caso particular das equações de Lotka-Volterra, abordando as soluções, como são os campos de vetores associados, como são as trajetórias associadas com as soluções do sistema e como estas soluções evoluem no tempo. Por se tratar de um sistema não linear, fizemos a abordagem inicial por meio da linearização em torno de pontos críticos. Como a aproximação nos dá um sistema linear, então além de utilizarmos a teoria de estabilidade para entender como são as trajetórias nas proximidades dos pontos críticos, descobrimos também o formato explícito das soluções, i.e., as equações que, localmente, governam as populações de predadores e presas. Também provamos uma série de resultados envolvendo estas populações como a periodicidade, a maximalidade e minimalidade das populações e de suas velocidades de crescimento. Uma particularidade do sistema clássico é que, apesar de ser um sistema não linear, ele pode ser reescrito como uma EDO separável, dessa maneira é possível encontrar a solução em um formato implícito utilizando a técnica de separação de variáveis. Usando o SageMath observamos em diversos casos que as trajetórias do sistema Lotka-Volterra são fechadas, além disso, indicamos duas referências contendo a prova deste fato, uma demonstra analiticamente e a outra utiliza um argumento geométrico. Ainda fizemos implementações acerca do sistema linear e não linear, apresentando as trajetórias do sistema, os campos de vetores associados e as soluções no tempo. O método numérico que utilizamos foi o de Runge-Kutta de quarta ordem, que está disponível no SageMath por meio da função `desolve_system_rk4`. Além disso, apresentamos ao decorrer da Seção 2, grande parte dos comandos utilizados para fazer os elementos computacionais. Dessa forma, esperamos que este trabalho facilite os estudos e pesquisas de docentes e discentes que se envolvam com estudos relacionados com os temas que aqui foram desenvolvidos ou que sejam relacionados. Destacamos, por fim, que estão disponíveis, *online*, materiais complementares desenvolvidos pelos autores no endereço eletrônico disponibilizado na introdução.

## Agradecimentos

Os autores agradecem o apoio financeiro da Fundação de Amparo a Ciência e Tecnologia de Pernambuco (FACEPE).

## Referências

- [1] H. Anton, R. C. Busby, *Álgebra Linear Contemporânea*. 1<sup>a</sup> edição. Porto Alegre: Bookman. 2006.
- [2] R. C. Bassanezi, *Ensino-Aprendizagem com Modelagem Matemática*. 4<sup>a</sup> edição, Editora Contexto. 2002.
- [3] R. C. Bassanezi, *Equações Diferenciais com Aplicações*. 1<sup>a</sup> Ed. São Paulo: Harbra Ltda, 1988.
- [4] G. R. Bessa, *Teoria de estabilidade de equações diferenciais ordinárias e aplicações: modelos presa-predador e competição entre espécies*. Dissertação, Universidade Estadual Paulista, Instituto de Geociências e Ciências Exatas, 2011.
- [5] W. E. Boyce, R. C. DiPrima, *Equações Diferenciais Elementares e Problemas de Valor de Contorno*, 9<sup>a</sup> edição, LTC Editora, Rio de Janeiro, 2010.
- [6] M. Braun, M. Golubitsky, *Differential equations and their applications*, second edition, Applied Mathematical Sciences, Vol. 15, Springer-Verlag, New York, 1978.
- [7] H. Bueno, *Álgebra Linear - um segundo curso*. 1<sup>a</sup> edição. Rio de Janeiro: SBM. 2006.
- [8] P. W. Draper, *Lotka-Volterra Predator-Prey Models Analytic and Numerical Methods*. Dissertation. Laurentian University of Sudbury, 2017.
- [9] L. Edelstein-Keshet, *Mathematical models in biology*, reprint of the 1988 original, Classics in Applied Mathematics, 46, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2005.
- [10] H. W. Hethcote et al. *A predator-prey model with infected prey*. Theoretical population biology, v. 66, n. 3, p. 259-268, 2004.
- [11] A. Korobeinikov, A Lyapunov function for Leslie-Gower predator-prey models, Appl. Math. Lett. **14** (2001), no. 6, 697–699.

- [12] E. L. Lima, *Curso de Análise Vol. 2*, Projeto Euclides, 13, Instituto de Matemática Pura e Aplicada, Rio de Janeiro, 1981.
- [13] T. V. R. Lima, O. E. Aiello, *Análise numérica das equações predador-presa*. IFSP, 2017.
- [14] P. L. Natti et al., *Modelagem matemática e estabilidade de sistemas predador-presa*.
- [15] R. Piva, *Modelos matemáticos e equações diferenciais ordinárias*. Dissertação (mestrado) - Universidade Estadual Paulista, Instituto de Geociências e Ciências Exatas, 2016.
- [16] B. Scárdua, *Equações Ordinárias e Aplicações*. 1ª edição. Rio de Janeiro: SBM, 2015.
- [17] R. Silva, *Modelagem de um Sistema Presa-predador em Ambientes Heterogêneos*. Dissertação (mestrado) - Univeridade Federal Rural de Pernambuco, 2014.
- [18] L. D. Silva, M. P. Santos, R. N. Machado Júnior, *Elementos de Computação Matemática com SageMath*, SBM, 2019.
- [19] J. Stewart, *Cálculo: volume 1*, 7ª edição, Thomson, 2013.
- [20] J. Stewart, *Cálculo: volume 2*, 7ª edição, Thomson, 2013.
- [21] D. G. Zill, *Equações diferenciais com aplicações em modelagem*. Cengage Learning Editores, 2016.