Mathematical Biology

Simone Pezzuto, Cinzia Soresina

2024-10-04

Table of contents

Welcome

This is the webpage of **Mathematical Biology**! Here you will find all the material for the course. Stay tuned for regular updates!

Schedule

	Mon	Tue	Wed	Thu	Fri
13:30	Lecture A209				
14:30		Lecture A224			
15:30			Lab A202		
16:30					

Figure 1: Schedule of the course

Next lectures

- Mon 16th, 2024 (A209): Introduction to Mathematical Biology
- Tue 17th, 2024 (A224): Population dynamics
- Wed 18th, 2024 (A202): Introduction to numerical solution of ODEs
- ~~Mon 23rd, 2024 (A209)~~: no lecture! Use the slot to review basic concepts of ODEs
- ~~Tue 24th, 2024 (A224)~~: no lecture! As above
- Wed 25th, 2024 (A202): Correction to first assignment
- … TBA …

Office hours

Any time Mon-Wed, please make an appointment by email first Works best right after the lectures

Exams

The exam will consist of a written test (**50% of the final grade**), an oral exam (**20%**), and a project presentation (**30%**). Specifically:

- **Written Test**: Exercises similar to those covered in class and available on the course website.
- **Oral Exam**: One question at the blackboard on the theoretical part of the course.
- **Project**: Conducted in groups of up to 2 students. Typically involves reading an article, or implementing or studying (presentation of the model, qualitative analysis, simulations) a model not covered in class. The presentation will last about 25 minutes (total, not per student) and will be part of a mini-workshop open to other students in the course and the public. A selection of possible projects and articles will be available on the website.

Topics

The course "Mathematical Modeling" has a dual purpose: on one hand, to introduce students to some basic mathematical models in various areas of biology (demography, ecology, infectious diseases, enzyme reactions, physiology, molecular networks); on the other hand, to provide fundamental knowledge in the analysis and numerical simulation of ordinary and partial differential equations.

Specifically, the first part of the course is dedicated to modeling using ordinary differential equations and introduces various analytical techniques (linearization, equilibria and their stability, bifurcation, regular and singular perturbations).

- **Overview of ordinary differential equations (ODEs)**: Solution of linear equations; equilibria and linearized stability; phase plane, limit cycles; numerical schemes for solving ODEs.
- **One- or two-dimensional models** in demography, ecology, epidemiology, and immunology. Non-dimensionalization of variables and parameters.
- **Slow-fast systems**, enzyme reaction models and their simplification using perturbative methods.
- **Bifurcation of equilibria** and application to predator-prey systems and molecular networks. Simplified models of important biological phenomena, such as the cell cycle and glucose-insulin oscillations.
- **Excitable systems**: Hodgkin-Huxley equations (overview) and FitzHugh-Nagumo equations.
- **Parameter estimation** for differential models.

In the second part, partial differential equation models and some techniques for constructing or approximating solutions will be studied. Additionally, some of the most interesting phenomena

of reaction-diffusion equations (traveling wave solutions, Turing mechanism) will be presented in a biological context (morphogenesis).

- **Dynamical systems on networks**. Examples in epidemiology.
- **Introduction to partial differential equations (PDEs)**: Solutions by separation of variables. Fourier series. The heat equation and Brownian motion. Eigenfunctions of the Laplacian. Numerical approximation.
- **Skellam and Fisher equations**: Waveform solutions; stationary solutions of the boundary value problem.
- **Stability of stationary solutions** of reaction-diffusion systems and Turing's mechanism for morphogenesis. Conditions for its validity and examples. Chemotaxis: The Keller-Segel model.

Part I

Lectures

1 Bathtub model

1.1 The bathtub model

The models of Newtonian physics are made of differential equations built starting from the second law of the dynamics. The structure of the models discussed here is instead simpler; they are based on the "balance equation of the bathtub": if $Q(t)$ is the quantity of a substance in the bathtub we have

$$\frac{\mathrm{d}Q}{\mathrm{d}t} = Q'(t) = I(t) - O(t),$$

where

- $I(t)$ is the *input rate* (quantity that enters per unit time)
- $Q(t)$ is the *output rate* (quantity that leaves per unit time).

To be more precise, the assumption is that, if $I_{(t,t+\Delta t)}$ is the quantity that enters in the interval $(t, t+\Delta t)$, we have $I_{(t,t+\Delta t)} = I(t)\Delta t + o(\Delta t)$, where $o(\Delta t)$ is a higher order infinitesimal than Δt. Hence:

$$I(t) = \lim_{\Delta t \to 0} \frac{I_{(t,t+\Delta t)}}{\Delta t}.$$

The input rate $I(t)$ is like an instantaneous velocity: the quantity entered in a given time, when that time becomes very small. Hence $I(t)$ is measured in $[C][t^{-1}]$ units where $[C]$ represents the concentration of the quantity Q. Similarly for the exit rate $O(t)$.

Let us start from a very simple example. Assume $I(t) = \Lambda$ constant input flux; $O(t) = \gamma Q(t)$, i.e. exit flux is proportional to the quantity present at the moment; the proportionality constant γ is often called the *exit rate* and has the dimension $[t^{-1}]$, the inverse of time. From these assumptions we get:

$$Q'(t) = \Lambda - \gamma Q(t), \tag{1.1}$$

supplemented with some *initial condition*

$$Q(0) = Q_0.$$

The solution is:

$$Q(t) = e^{-\gamma t} Q_0 + \frac{\Lambda}{\gamma}\Big(1 - e^{-\gamma t}\Big).$$

Exercise

Solve Equation 1.1 with the method you prefer.

Exercise

Solve Equation 1.1 with the general formula for linear ODEs, by first defining the matrix exponential (here, just a scalar function).

Note that if $\Lambda = 0$ (no input), the solution is simply

$$Q(t) = Q_0 e^{-\gamma t}.$$

This means that the survival time of a molecule initially present follows the exponential distribution:

$$\mathbb{P}[\text{a molecule present at time 0 is present at time } t > 0] = \frac{Q(t)}{Q_0} = e^{-\gamma t}.$$

From the properties of the exponential distribution, we obtain that the mean survival time $\mathbb{E}[T] = 1/\gamma$; hence the exit rate γ can be interpreted as the inverse of the mean survival time.

To be more precise, let us define a continuous random variable T, which measures the lifetime of a particle present in the bathtub. Then, the cumulative distribution $F(t)$ of T is given by

$$\begin{aligned} F(t) &= \mathbb{P}[T \leq t] \\ &= 1 - \mathbb{P}[T > t] \\ &= 1 - \mathbb{P}[\text{a molecule present at time 0 is present at time } t > 0] \\ &= 1 - e^{-\gamma t}. \end{aligned}$$

So, we indeed have an exponential distribution. The probability density function is:

$$f(t) = F'(t) = \gamma e^{-\gamma t},$$

and the expectation is:

$$\mathbb{E}[T] = \int_0^\infty t f(t) \mathrm{d}t = \frac{1}{\gamma}.$$

Exercise

Compute the above integral explicitly.

1.2 Malthus equation

The metaphor of the bathtub can be used to model the dynamics of a population. Neglecting all differences among individuals (due to age, sex, genetic,...) we can represent a population through its size $N(t)$; this will increase through inputs due to births and outputs due to deaths (if immigration and emigration are not considered). Hence

$$N'(t) = B(t) - D(t),$$

where $B(t)$ = births and $D(t)$ = deaths.

Malthus model assumes

- within a (short) time period of length Δt, each individual gives, on average, birth to $\beta \Delta t$ new individuals; hence $B(t) = \beta N(t)$;
- within the same time period Δt, each individual has probability $\mu \Delta t$ of dying; hence $D(t) = \mu N(t)$.

We get the following equation

$$N'(t) = \beta N(t) - \mu N(t) = (\beta - \mu) N(t),$$

that represents the *Malthus model*. The parameter β is known as *fertility rate*, while μ is the *mortality rate*. Finally,

$$r = \beta - \mu$$

is the (instantaneous) *growth rate* and is also called *Malthus parameter* or *biological potential* of the population.

With the initial condition

$$N(0) = N_0,$$

the evolution of the population is completely determined. In fact, the solution is

$$N(t) = N_0 e^{rt},$$

and we see that the population will go to extinction or will grow without limits if $r < 0$ or $r > 0$, respectively. If instead $r = 0$, the population size is constant (births and deaths compensate.)

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

sns.set_theme("notebook", style="whitegrid")
t = np.linspace(0,1,100)

plt.plot(t,np.exp(1*t),label='r > 0')
plt.plot(t,np.exp(-1*t),label='r < 0')
plt.plot(t,np.exp(0*t),label='r = 0')
plt.grid()
plt.legend()
plt.xlabel('Time')
plt.ylabel('Population')
plt.show()
```

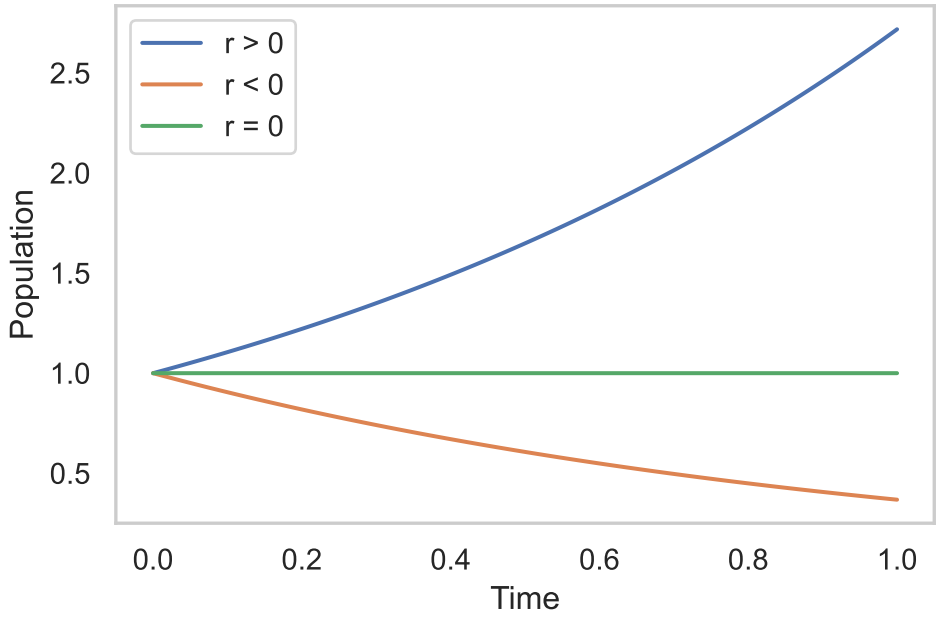


Figure 1.1: Example of solutions of Malthus equation

2 Population dynamics

This lecture is based on [@ip14, Section 1.1-1.4]

2.1 Population growth model

We have seen, using the bathtub analogy, that the *fundamental balance equation of population dynamics* take the form

$$N'(t) = B(t) - D(t) + I(t) - E(t),$$

where at time t we have that

- $N(t)$ is the population,
- $B(t)$ is the birth rate,
- $D(t)$ is the death rate,
- $I(t)$ is the immigration rate, and
- $E(t)$ is the emigration rate.

There is also an integral interpretation, in fact:

$$N(t) - N(t_0) = \underbrace{\int_{t_0}^{t} B(s)\,\mathrm{d}s}_{\text{births}} - \underbrace{\int_{t_0}^{t} D(s)\,\mathrm{d}s}_{\text{deaths}} + \underbrace{\int_{t_0}^{t} I(s)\,\mathrm{d}s}_{\text{immigration}} - \underbrace{\int_{t_0}^{t} E(s)\,\mathrm{d}s}_{\text{emigration}}.$$

A population growth model is an ODE of the above form with some specific form for each term above. Note that we could provide functions above explicitly, as function of time alone. However, it should be clear that some of them like $B(t)$ or $D(t)$ should depend on N.

2.1.1 Malthus model

Quoting Thomas R. Malthus (1766-1834):

> *Population, when unchecked, increases in a geometrical ratio. Subsistence increases only in an arithmetical ratio. A slight acquaintance with numbers will show the immensity of the first power in comparison of the second.*
>
> Essay on the Principle of Population, 1798

Malthus' model is mathematical formulation of the above statement. We already derived Malthus model, but let us recall the hypotheses.

1. The population is *homogeneous*, that is all individuals are identical. We have a single class to represent them, that is $N(t)$.
2. The population is *isolated*, so $E(t) = I(t) = 0$.
3. The habitat is *invariant*, so resources and life conditions are not affected by the environment nor the population itself.
4. The population is *very large*, so we can consider continuous functions.
5. On a short time scale Δt, each individual gives birth to $\beta \Delta t$ new individuals, $B(t) = \beta N(t)$.
6. On a short time scale Δt, each individual has probability $\mu \Delta t$ of dying, $D(t) = \mu N(t)$.

The non-negative parameters β and μ are the fertility and mortality rate, respectively. From hypothesis 4, we have that μ and β are constant. We introduce the *growth rate*

$$r = \beta - \mu,$$

also called *Malthus parameter* or *biological potential*.

The Malthus' model reads:

$$N' = rN, \quad \Rightarrow \quad N(t) = e^{rt} N(0),$$

thus, when $r > 0$, the growth of population is geometrical and unbounded, as predicted by Malthus.

2.1.2 Expected life

We have seen that, in absence of births, the population goes like

$$N(t) = e^{-\mu t} N_0,$$

thus, we could say that the probability of surviving up to time t is $e^{-\mu t}$. Specifically, the *life expectancy* L is a random variable such that

$$\mathbb{P}[L > t] = e^{-\mu t}.$$

Thus, the cumulative distribution of L is

$$F_L(t) = \mathbb{P}[L < t] = 1 - \mathbb{P}[L > t] = 1 - e^{-\mu t},$$

and the probability density function is

$$f_L(t) = F_L'(t) = \mu e^{-\mu t}.$$

We conclude that $L \sim \text{Exp}[\mu]$, the exponential distribution. The *average life expectancy* is:

$$\mathbb{E}[L] = \int_0^\infty s f_L(s)\,\mathrm{d}s = \int_0^\infty s\mu e^{-\mu s}\,\mathrm{d}s = \frac{1}{\mu}.$$

We have another interpretation of μ: it is the reciprocal of the expected life time.

2.1.3 Basic reproduction number

Let us rescale the equation and put it in non-dimensionalized form. This is a fundamental step in general, because

1. it reduces the number of parameters,
2. it removed scale effects (units are removed),
3. it highlights the determining factors of the model (maybe what matters is not this or that parameter, by their ratio or sum).

Here, we rescale as follows:

$$\tau = \mu t, \quad u = N,$$

so that time is now in units of "expected life time": $\tau = 1$ means $t = \mu^{-1} = \mathbb{E}[L]$. We have that

$$N' = \frac{\mathrm{d}N}{\mathrm{d}t} = \frac{\mathrm{d}u}{\mu^{-1}\mathrm{d}\tau} = \mu\frac{\mathrm{d}u}{\mathrm{d}\tau} = \mu\dot{u}.$$

We use the "dot" notation $\dot{u}$ for the derivative for the non-dimensional form, just to remember that now the time is τ and not t. We finally obtain:

$$\dot{u} = \mu^{-1}N' = \frac{\beta}{\mu}N - N = (R_0 - 1)u,$$

where we defined the *basic reproduction number*

$$R_0 = \frac{\beta}{\mu} = \beta \mathbb{E}[L].$$

We could interpret it as the average number of newborns produced by one individual during his whole life. Note that R_0 is non-dimensional.

> **Exercise**
>
> Why it does not make much sense to use the scaling $\tau = rt$?

2.1.4 Migration

In the presence of migration, say with a constant rate, we have the ODE:

$$N' = rN + m = f(N), \tag{2.1}$$

where $m = I - E$. If positive, there is a net immigration, otherwise emigration.

In order to study how the model will behave, we have 3 options:

1. Solve the problem analytically, that is finding $N(t) = ...$ explicitly.
2. Solve the problem numerically, which is always possible.
3. Study the problem qualitatively.

The last option has the advantage that we can be generic, there is no need for a specific value of the parameters or the initial condition. The qualitative study consists in the following steps:

1) Fixed points of the system. A fixed point or equilibrium (see Definition A.12) is a constant solution of the ODE. We can find it by setting the right hand side to zero:

$$N' = 0 \quad \Leftrightarrow \quad rN + m = 0.$$

The model has a single equilibrium for

$$N = N^* = -\frac{m}{r} = \frac{m\mathbb{E}[L]}{1 - R_0}$$

2) Biological feasibility. Equilibria must be biologically feasible. For this model, we need to check that N^* is non-negative, otherwise it doesn't make sense biologically speaking. Therefore

$$N^* \geq 0, \quad \Leftrightarrow \quad m \text{ and } r \text{ have opposite sign and } r \neq 0.$$

3) Local stability. Informally, an equilibrium is *locally stable* when, starting from a neighborhood of it, the solution stays close to it for $t \to \infty$. It is *asymptotically stable* when the solution converges toward the equilibrium for $t \to \infty$. It is *unstable* otherwise. See Definition A.13 for a more precise statement.

The local stability is determined by the sign of the derivative of the right hand side, that is $f'(N^*)$ for $f(N) = rN + m$. In general (see Section A.5.5),

- when $f'(N^*) < 0$, the equilibrium is asymptotically stable, and
- when $f'(N^*) > 0$, the equilibrium is unstable.

To see this in this specific case, let us define $w(t) = N(t) - N^*$. Then,

$$w' = N' = rN + m = r(N - N^*) = rw.$$

Note that $f'(N) = r$. We have that:

$$w(t) = w_0 e^{rt},$$

hence,

- if $r < 0$, then $w \to 0$ for all w_0, so $N(t) \to N^*$. The equilibrium is locally asymptotically stable.
- if $r > 0$, then $w \to \infty$ and the equilibrium is unstable.

4) Global stability. What if we start very far away from the equilibrium? In this particular case, with a linear ODE, the local stability argument applies also globally, thus the equilibrium is globally attractive. But for general, nonlinear ODEs this may not be the case, so we perform the analysis anyway. Note that if $N(0) = N_0 > N^*$, then

$$N'(0) = rN(0) + m = r(N_0 - N^*) < 0,$$

so the derivative of the solution is negative (assuming $r < 0$). Furthermore, for $N(t) > N^*$, the derivative is always negative. So, the solution must be monotonically decreasing. But the solution is bounded from below by the equilibrium $N(t) = N^*$, so we conclude that:

$$N(t) \to N^*$$

for all $N_0 \geq N^*$. Symmetrically, when $N_0 < N^*$, the derivative is positive and stays positive for all t, so the solution is monotonically increasing. Hence:

$$N(t) \to N^*$$

for all N^0. The equilibrium is therefore *globally* stable when $r < 0$.

5) Phase portrait. The phase portrait of a dynamical system is the collection of all possible orbits. Here, the phase space is $\Omega = [0, \infty)$. The only equilibrium we have, N^*, is a *barrier* to other orbits, because orbits cannot intersect (See Proposition A.2). Therefore:

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

sns.reset_defaults()
sns.set_context('notebook')
r,m = -0.5,1.7
Neq = -m/r

f = lambda N: r*N + m
N = np.linspace(1.5,5,10)

fig, ax = plt.subplots()
ax.spines[["bottom"]].set_position(("data", 0))
ax.spines[["top", "right", "left"]].set_visible(False)
ax.xaxis.set_ticks([])
ax.yaxis.set_ticks([])
ax.plot(1, 0, ">k", transform=ax.get_yaxis_transform(), clip_on=False)
ax.set_xlabel('N', loc='right', labelpad=10.0)

ax.plot(Neq,0,'r.',markersize=16, zorder=99)
ax.text(Neq, 3e-3, r'$N^*$', fontsize=12, ha='center', va='bottom')

ax.quiver(N,0*N, f(N), 0.0, color='blue', zorder=80)
ax.set_xlim((0,6))
ax.set_ylim((-1e-2,1e-2))
fig.subplots_adjust(left=0, right=1, top=0.1, bottom=0.05)
plt.show()
```

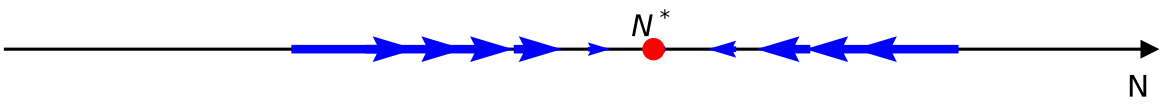


Figure 2.1: Phase space

For this problem we actually have the general solution

$$N(t) = (N_0 - N^*)e^{rt} + N^*.$$

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

r,m = -0.5,1.7
Neq = -m/r

sns.set_theme("notebook", style="whitegrid")
t = np.linspace(0,10,100)
N = lambda N0: (N0-Neq)*np.exp(r*t) + Neq

fig, ax = plt.subplots()
ax.plot(t,N(Neq - 1),'b',label='$N_0 < N^*$')
ax.plot(t,N(Neq + 1),'b',label='$N_0 > N^*$')
for delta in np.arange(0.1,0.9,0.1):
    ax.plot(t,N(Neq + delta),'b',lw=0.4)
    ax.plot(t,N(Neq - delta),'b',lw=0.4)
ax.plot(t,N(Neq),'r-',lw=2,label='$N^*$')
ax.grid()
ax.legend()
ax.set_xlabel('Time')
ax.set_ylabel('Population')
plt.show()
```

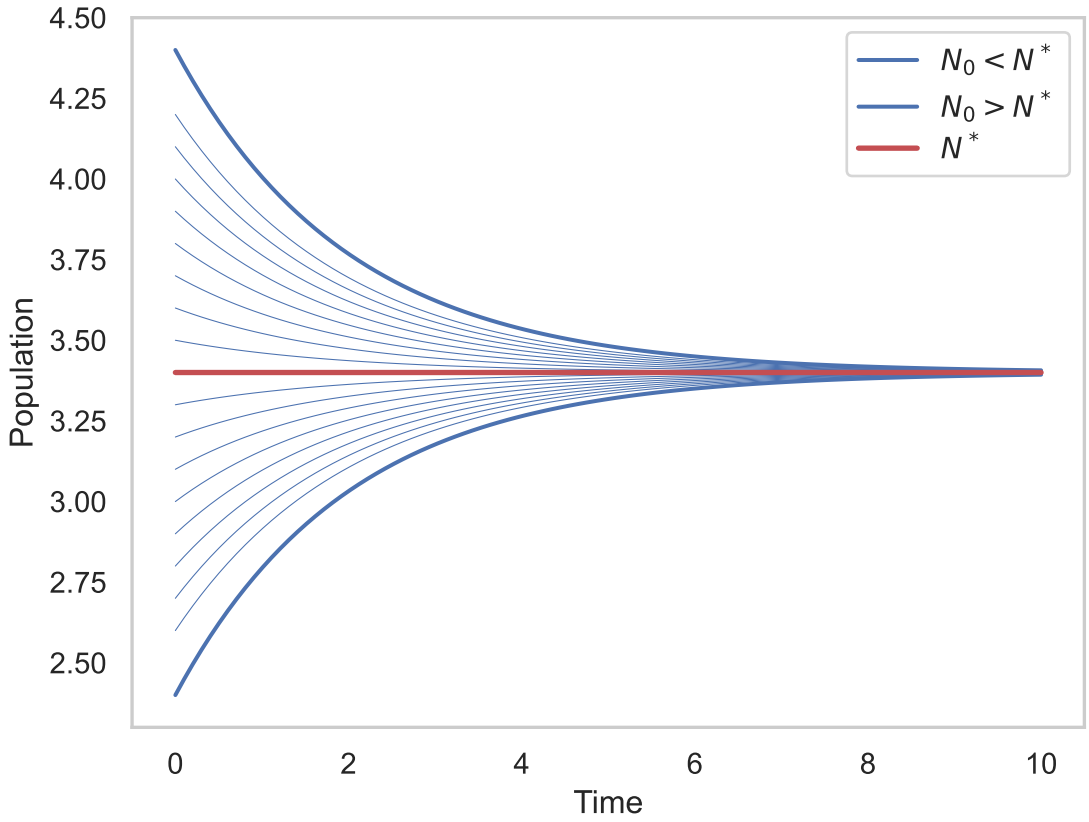


Figure 2.2: Example of trajectories

The case $m < 0$ and $r > 0$ is also interesting. Here, the equilibrium N^* is positive, but unstable. For $N_0 > N^*$ the population still grows exponentially, so emigration as no effect on the overall population. But for $N_0 < N^*$ the solution will become negative in finite time: the model is not correct. The reason is simple: the emigration m cannot be constant (like the immigration), it must depend on N as well.

> Exercise
>
> Solve Equation 2.1 with separation of variables.

2.1.5 Exogenous variability

By exogenous variability we mean variability in the parameter r that does not depend on the population. (Endogenous variability is when the parameters depend on internal variables like

N.) Thus, we consider the problem:

$$N' = r(t)N,$$

for $r(t)$ continuous function. An example could births and deaths that depend on climate or temperature. The solution is:

$$N(t) = N_0 e^{\int_{t_0}^{t} r(s)\,\mathrm{d}s}$$

We can also rewrite the solution as:

$$N(t) = N_0 e^{(t-t_0)\frac{1}{t-t_0}\int_{t_0}^{t} r(s)\,\mathrm{d}s},$$

showing that if the limit

$$r^* = \lim_{t\to\infty} \frac{1}{t-t_0}\int_{t_0}^{t} r(s)\,\mathrm{d}s$$

exists then the asymptotic behavior of the solution is:

$$N(t) \approx e^{r^*(t-t_0)}, \quad t \gg 1.$$

An interesting case is when $r(t)$ is *periodic*, that is there exists $T > 0$ such that $r(t+T) = r(t)$. In this case the above formula still applies, but we can "forget" the limit (check Assignment 4 for a proof) and take:

$$\bar{r} = \frac{1}{T}\int_0^T r(s)\,\mathrm{d}s,$$

and write the solution for any $t \geq t_0$ as:

$$N(t) = e^{\bar{r}(t-t_0)} N_\pi(t),$$

for some T-periodic function N_π. Note that

$$N(t_0 + kT) = e^{\bar{r}kT} N_\pi(t_0) = e^{k\bar{r}} N(t_0),$$

so after k periods the solution grows by a factor $e^{k\bar{r}}$. This factor is called *Floquet multiplier*.

3 Logistic model

This lecture is based on [@ip14, Section 1.5-1.6]

3.1 Logistic model

While populations can follow a phase of exponential growth for a limited amount of time, it seems impossible that this can go forever and that populations can grow to infinity. Indeed, we expect that there exists a negative effect of crowding, which can be stated in words as follows:

> An increase of the population size produces a fertility decrease and a mortality increase; since resources are limited, if the population size exceeds some threshold level, the habitat cannot support the growth.

This simple statement tries to summarize the complex phenomenology of *intraspecific competition* due to many factors such as resource availability, habitat pollution and waste, predation increase, energy consumption for social organization.

The simplest way to include this effect into a model, is to suppose that fertility decreases and mortality increases linearly with the number of individuals; namely

$$\begin{aligned}\beta(N) &= \beta_0 - \tilde{\beta}N,\\ \mu(N) &= \mu_0 + \tilde{\mu}N,\end{aligned}$$

where β_0, μ_0, $\tilde{\beta}$, and $\tilde{\mu}$ are non-negative constants. Hence:

$$\begin{aligned}B(N) &= \beta(N)N = \beta_0 N - \tilde{\beta}N^2,\\ D(N) &= \mu(N)N = \mu_0 N + \tilde{\mu}N^2.\end{aligned}$$

The resulting equation is generally written, after simple algebraic steps, as

$$\begin{cases} N' = r\Big(1 - \dfrac{N}{K}\Big)N,\\ N(0) = N_0,\end{cases}$$

where $r = \beta_0 - \mu_0$ and $K = r/(\tilde{\beta} + \tilde{\mu})$. These parameters are usually called *intrinsic growth rate* and *carrying capacity*.

A couple of comments are necessary: first of all, either $\tilde{\beta}$ or $\tilde{\mu}$ can be 0, but not both, otherwise there is no effect of crowding and K is not well-defined. Note also that, if $\tilde{\beta} > 0$, the birth rate $B(t)$ would become negative if $N(t)$ is too large, which does not make sense biologically. However, this does not cause mathematical problems and the biological nonsense would occur only at population levels not normally reached, so we neglect this problem.

We will generally assume that $r > 0$, so that also $K > 0$. In that case, the behaviour of solutions to that equation displays a first phase of exponential growth, followed by convergence to the limiting value K. The general solution is

$$N(t) = \frac{KN_0}{N_0 + (K - N_0)e^{-rt}}.$$

When $N_0 < K/2$, the resulting sigmoid curve have been called *logistic curve*, so that equation is also named *logistic equation*.

The logistic equation is extremely common in experimental biology. Below, data fitted to a logistic for micro-organisms.

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

sns.set_theme("notebook", style="whitegrid")

# from CRAN gauseR gause_1934_book_f04.rda
t = [1.0,2.0,3.0,4.0,5.0,5.0,6.0,6.0]
N = [ 22,129,334,374,376,356,397,367]
K = 375
r = 2.309
N0 = 2.0
tt = np.linspace(0,6,1000)
logistic = K*N0/(N0 + (K-N0)*(np.exp(-r*tt)))

plt.axhline(y=K,color='k',linestyle='--',linewidth=1.0)
plt.plot(tt, logistic,label='$\\frac{K N_0}{N_0+(K-N_0)e^{rt}}$')
plt.plot(t,N,'.',markersize=16,label='Data')
plt.xlabel('Days')
plt.ylabel('Number of individuals')
plt.title(f'Fit with K = {K}, $N_0$ = {N0}, r = {r}')
plt.annotate(f'K = {K}',(1,K),ha='center',va='bottom')
plt.legend()
```

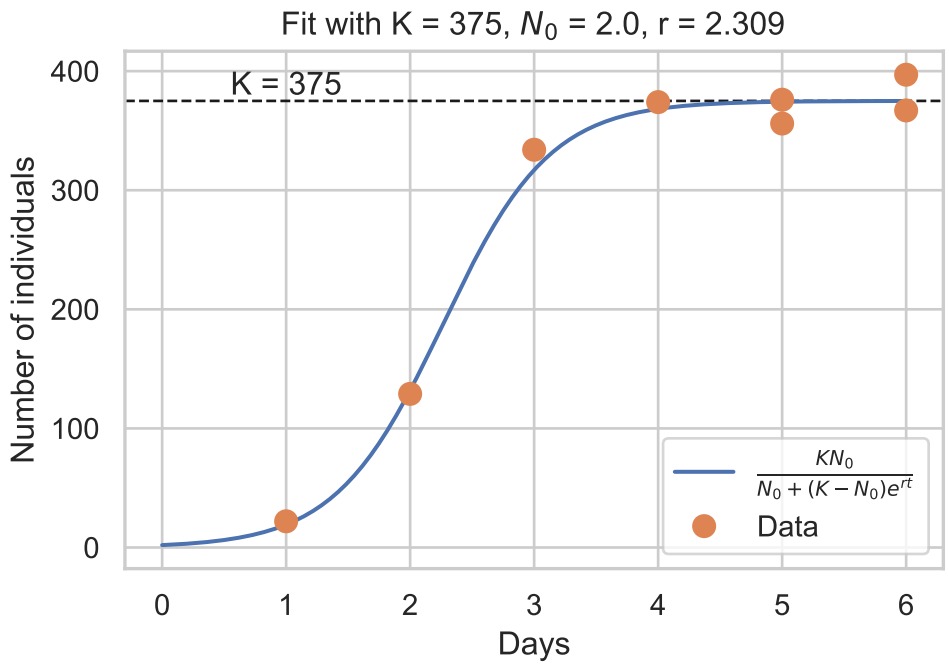


Figure 3.1: Fitting of *Paramecium caudatum* data using the logistic model. Data from (Gause 1934, fig. 4).

As before, we can non-dimensionalize the equation. Now we select:

$$\tau = rt, \quad u = \frac{N}{K},$$

so that $u = 1$ means that we are at carrying capacity. Substituting:

$$\dot{u} = u(1 - u).$$

The new equation has no parameters. Thus, the general solution of the logistic equation is just

$$N(t) = Ku(rt).$$

The absence of parameters in the non-dimensional equation means that its dynamic is always the same, up to a rescaling. We say that all parametric solutions (as we vary r and K) are topologically equivalent. Here, we cannot expect bifurcation, as we shall see.

3.2 Generalized logistic model

In general, we can take β and μ as generic functions of N, thus:

$$r(N) = \beta(N) - \mu(N)$$

is also a generic function of N. A generic growth model reads:

$$N' = r(N)N = f(N).$$

We can implicitly solve this equation, in fact:

$$N(t) = N(0)e^{\int_{t_0}^{t} r(N(s))\,\mathrm{d}s}.$$

Thus, if $N(0) > 0$, then $N(t) > 0$ for all time. This is important to verify, since a population cannot be negative. Here, we need no further restriction on $r(N)$ so to verify the condition.

How about equilibria? We have that $E_0 = 0$ is always an equilibrium. (Thus, since orbits cannot cross it, they positive stay positive foreover. This is another possible proof.) It is called *extinction equilibrium*. Its stability follows from:

$$f'(N)|_{N=E_0} = r'(0) \cdot 0 + r(0) = r(0).$$

The stability of E_0 is given by $r(0)$, which is called *intrinsic growth rate*. It is the growth rate we observe for very small population size. When $r(0) > 0$ the extinction equilibrium is *unstable*. Equivalently, it is unstable when

$$R_0 = \frac{\beta(0)}{\mu(0)} > 1.$$

How to capture a general logistic effect? By general we mean what we quoted above: a population increase should correspond to a decrease of fertility and an increase of mortality. Thus:

$$r'(N) < 0, \quad \text{and} \quad \lim_{N\to\infty} r(N) < 0.$$

The second hypothesis avoids the existence of positive horizontal asymptotes. Biologically, a sufficiently large population has always a negative growth rate.

We can now study more equilibria, those corresponding to $r(N) = 0$.

- If $r'(0) < 0$, then by monotonicity we conclude that $r(N) < 0$ for all N, so $E_0 = 0$ is the only equilibrium and the population is doomed.

- If $r'(0) > 0$, we have one additional (unique) equilibrium N^*, that we denote by K: that is, $r(K) = 0$. Since
$$f'(K) = r'(K)K + \underbrace{r(K)}_{=0} = r'(K)K < 0,$$

this equilibrium is (globally) asymptotically stable. This equilibrium is also called *carrying capacity*.

The classic logistic equation has $r(N) = r(1-N/K)$. The θ-logistic model (or Bernoulli model) has
$$r(N) = r(1 - (N/K)^\theta),$$

for $\theta > 0$. There are a multitude of models for $r(N)$, some we will explore in the assignments (see also figure below). Nonetheless, the above hypotheses always imply a sigmoid growth, when $N(0) \in (0, K)$.

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

sns.set_theme("notebook", style="whitegrid")

N = np.linspace(0,1.2,1000)

plt.plot(N, 1-N,label='$r=1-u$ (Verhulst)')
plt.plot(N, 1-N**2.0,
         label='$r=1-u^\\theta$ (Bernoulli, $\\theta=2$)')
plt.plot(N, (1-N)/(1+2*N),
         label='$r = \\frac{1-u}{1+\\alpha u}$ (Smith, $\\alpha=2$)')
plt.plot(N, (np.exp(3*(1-N))-1)/(np.exp(3)-1),
         label='$r = \\frac{e^{\\gamma u}-1}{e^\\gamma -1}$ (Ricker, $\\gamma = 3$)')
plt.plot(N[1:], -np.log(N[1:]),label='$r = -\\log u$ (Gompertz)')
plt.ylabel('Growth rate')
plt.xlabel('N / K')
plt.ylim([-0.5,1.5])
plt.legend()
```

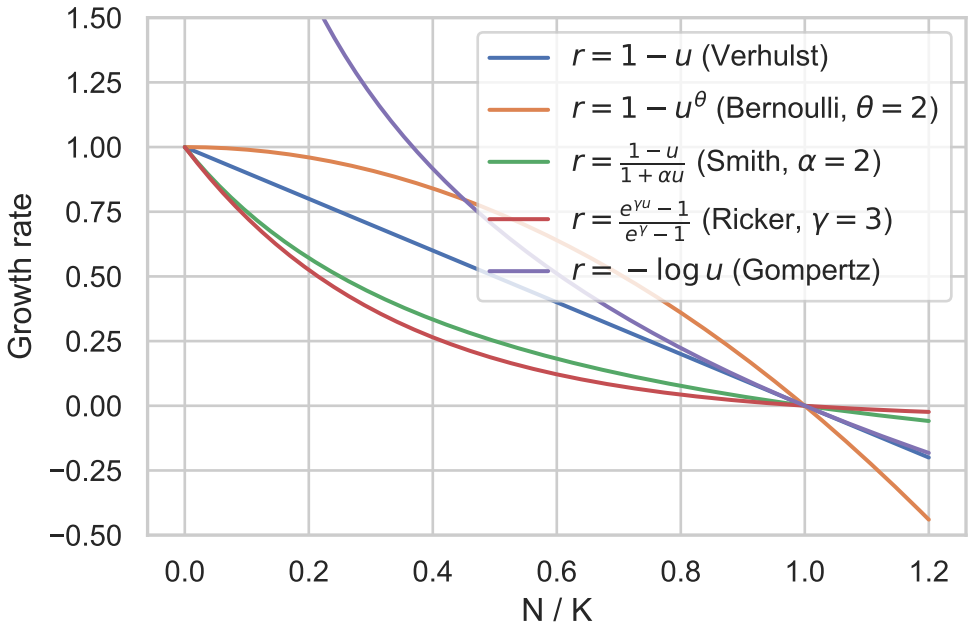


Figure 3.2: Different growth model giving a logistic effect. We set $u = N/K$, where K is the carrying capacity.

> **Exercise**
>
> Verify that the above models are "logistic", in the sense that they satisfy the hypotheses. Show that the Bernoulli model yields sigmoid solutions. (Hint: study N' and N''.)

> **Exercise**
>
> Integrate the Gompertz model, $\dot{u} = -u \log u$. This model is very common in the study of tumor cells proliferation. (Hint: set $u(t) = e^{w(t)}$.)

3.3 Allee effect

(Story time) The great auk was a bird that became extinct by the end of the 19th century. They were found at the northern Atlantic (Canada, Scotland, Iceland), usually on rocky islands. The overall population was composed by millions of individuals, before sailors and hunters started killing them for their meat, feathers, and oily fat.

Figure 3.3: The Great Auk (from Wikipedia)

As it could be expected, the auk population drastically reduced and the bird disappeared from many islands very quickly. Governments became aware of the situation in the 18th century, and some laws forbidden the hunt of the great auk, although with limited success.

Some species like the great auk are not able to fully recover, even in the absence of predation (or with a small one). In order for a species to go to excintion with no predation, say for

$$N' = r(N)N,$$

we would need $r(0) < 0$, otherwise the equilibrium $N = 0$ is not stable. But this would lead to no further equilibria under the hypotheses of general logistic growth. Hence, we can further generalize the hypotheses as follows: the function $r(N)$ is such that

1. there exists $N_m > 0$ such that $r'(N) > 0$ for $N < N_m$ and $r'(N) < 0$ otherwise;
2. $r(N_m) > 0$;
3. $\lim_{N\to\infty} r(N) < 0$.

Therefore, we certainly have the equilibrium $K > N_m$ (the carrying capacity). However, if $r(0) < 0$, we also have another equilibrium at $T \in (0, N_m)$. In this case, we say that we have a *strong Allee effect*. On the other hand, for $r(0) > 0$ there are no further equilibria, thus we have a *weak Allee effect*.

Let us study the stability for $r(0) < 0$. Remember that

$$f'(N) = r'(N)N + r(N).$$

We have 3 equilibria:

- $N = E_0 = 0$, which is asymptotically stable, since $f'(0) = r(0) < 0$.
- $N = T \in (0, N_m)$, which is unstable, since $f'(T) = r'(T)T > 0$.
- $N = K > N_m$, which is asymptotically stable, since $f'(K) = r(K)K < 0$.

Hence, we have the following result: if $N(0) < T$, then $N(t) \to 0$ (extinction), otherwise if $N(0) > T$, then $N(t) \to K$ (survival). We call T the threshold population for survival.

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from scipy.integrate import solve_ivp

sns.set_theme("notebook", style="whitegrid")

r = 0.5
T,K = 0.2,1.0
f = lambda t,N: r*(N/T-1)*(1-N/K)*N
```

```
fig,ax = plt.subplots()
u0 = [0.1,0.15,0.25,0.3,0.6,0.8]
sol = solve_ivp(f,[0,8],u0,max_step=0.05)
ax.plot(sol.t,sol.y.T,lw=2.0)
ax.axhline(y=T,color='r',linestyle='--')
ax.grid(True)
ax.legend([f'$u_0 = {u}$' for u in u0])
plt.show()
```

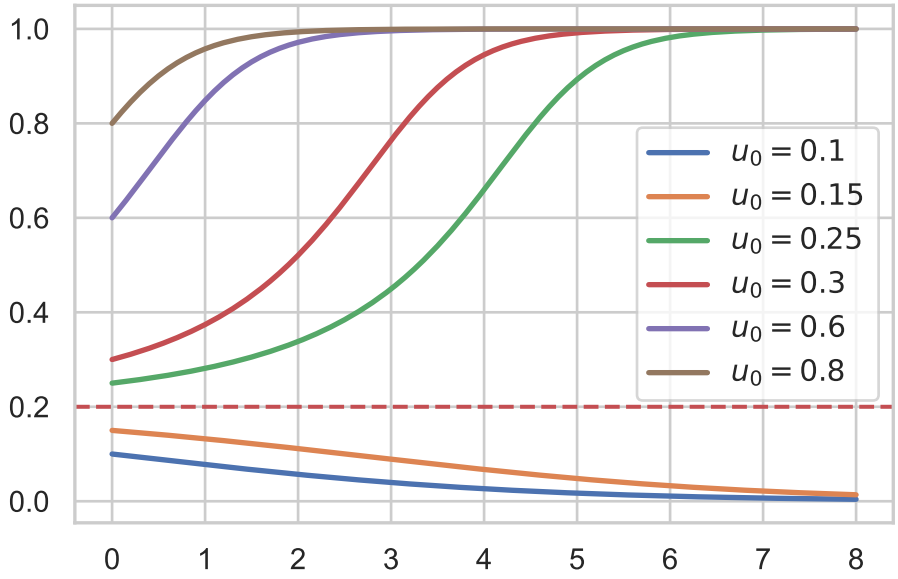


Figure 3.4: Solutions of the equation $N' = rN(N/T-1)(1-N/K)$. Note the threshold value for $N = T$.

The Allee effect may be found in other situations, for instance when the predation rate is a non-linear function of N.

4 Predation models

This lecture is based on [@ip14, Section 1.8-1.9]

4.1 Generalist predation

Predation is a fundamental topic in ecology and the main interaction between species. It must be understood in a broader sense: fishing, harvesting, hunting are all form of predation.

The simplest model of predation consists in formally increasing the mortality rate of N by an extra rate aP^*, that is

$$N' = r(N)N - aP^*N = f(N),$$

where $P^* > 0$ is the number of predators and $a > 0$ is the *attack rate* or *effective killing rate*. We note that the death rate aP^*N is *proportional* to both the number of predators P^* and the number of preys N, according to the *law of mass action*, as suggested by Volterra with the method of encounters. The idea is similar to collision theory for ideal gases.

This type of predation is also called *generalist*, in the sense that predators' survival does not depend on the survival of the prey population N. If the preys go to extinction, the predator will hunt something else. A *specialized* predator, on the other hand, will suffer from a low level of N, as we shall see next week.

The equilibria of the equation are: $N = E_0 = 0$, as usual, and the zeros of:

$$r(N) = aP^*.$$

- If $aP^* > r(0)$, then we have no additional equilibria, because $r'(N) < 0$.
- If $aP^* < r(0)$, then there exists a second equilibrium $N^* > 0$.

The stability of the second equilibrium follows from

$$f'(N^*) = r'(N^*)N^* + r(N^*) - aP^* = r'(N^*)N^* - aP^* < 0,$$

because $r'(N) < 0$. So N^* is asymptotically stable.

The stability of E_0 is similar:

$$f'(E_0) = r'(0) \cdot 0 + r(0) - aP^* = r(0) - aP^* > 0,$$

meaning that it is unstable. Concluding:

- If $aP^* > r(0)$, we have one stable equilibrium $E_0 = 0$, thus the population will go to extinction. In fact, the predation level is high.
- If $aP^* > r(0)$, $E_0 = 0$ is unstable but we have another equilibrium $0 < N^* < K$ asymptotically stable. Thus, the predation is sustainable.

The case $aP^* = r(0)$ is delicate. We have a single equilibrium, $E_0 = 0$, but $f'(E_0) = 0$, so we cannot deduce the stability from the linearization (why?). By inspecting the sign of E_0, we observe that $f(N) < 0$ for $N > 0$, so E_0 is attractive. However, for $N < 0$ we have that $f(N) < 0$, thus it is repulsive (biologically, we do not care because $N < 0$ is irrelevant.) This type of equilibrium is called *saddle-node*.

The point $aP^* = r(0)$ is a *bifurcation point*, specifically a *transcritical bifurcation* (See Section B.6). Roughly speaking, the dynamic before and after the bifurcation point is topologically different: in one case we have one equilibrium, in the other 2 equilibria. Actually, we still have 2 equilibria for $aP^* > r(0)$, one being negative. Thus, what really happens is that as we increase aP^* the two curves of equilibria ($N = 0$ and $N = E_0$) crosses and swap stability.

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

sns.set_theme("notebook", style="whitegrid")

with plt.xkcd(scale=0.5):
    fig, ax = plt.subplots()
    sns.despine()
    ax.xaxis.set_ticks([])
    ax.yaxis.set_ticks([])

    ax.plot([0,1],[0,0],'r-')
    ax.plot([0,1,1.2],[1,0,0],'b-')
    ax.plot([1,1.2],[0,-0.2],'r-',alpha=0.5)
    ax.plot([1],[0],'k.',markersize=16)
    ax.grid(False)
    ax.set_xlabel('$P^*$')
    ax.set_ylabel('$N^*$')
    ax.set_xlim([0,1.2])
```

```
[N,P] = np.mgrid[0.01:1.2:30j,0.01:1.2:30j]
D = N*(1-N) - P*N
ax.quiver(P,N,np.zeros_like(D),D,alpha=0.5)
plt.show()
```

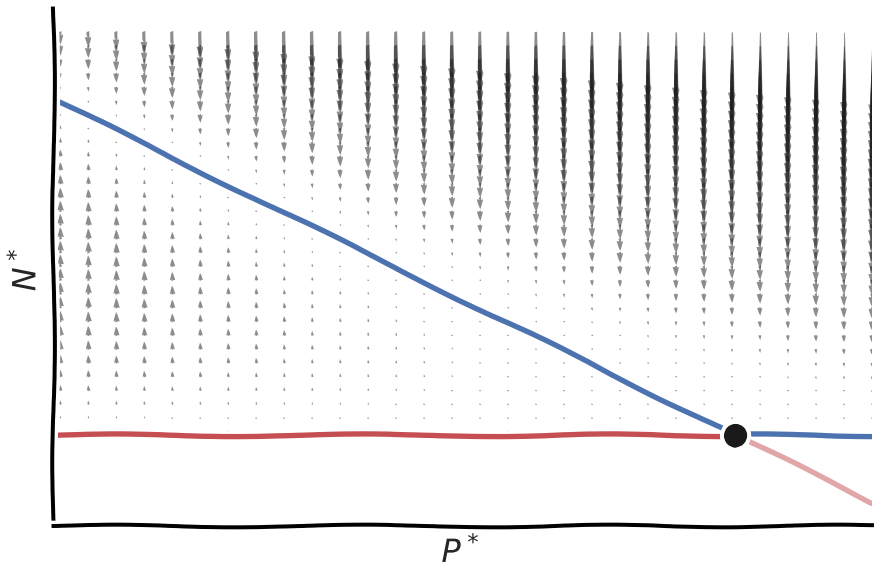


Figure 4.1: Bifurcation diagram for the model with generalist predation. In blue, the stable equilibrium, in red the unstable one. Beyond the bifurcation point (transcritical, in black) there is only one equilibrium.

4.2 Holling type predation models

More generally, we define the predation as:

$$N' = r(N)N - \pi(N)P^*,$$

where $\pi(N)$ is called *functional response*. The functional response can be interpreted as follows: $\pi(N)\Delta t$ is the number of preys killed in Δt units of time by a single predator.

For the simplest model, we have $\pi(N) = aN$. However, this is not realistic, because for very large N the predation rate cannot grow undefinitely, it must reach some limiting value. For instance, a predator needs some time to consume the prey, and this time cannot be reduced below a certain limit. Similarly, it is reasonable that at low density of preys, say when N is small, predation is harder.

In general, we can assume that:

1. $\pi(0) = 0$, no preys no predation, and
2. $\pi'(N) > 0$, the more preys, the higher the predation rate.

Holling (1965) proposed the following types of functional responses:

- **Holling type I**

$$\pi(N) = \begin{cases} aN, & 0 \leq N \leq N^*, \\ aN^*, & N > N^*. \end{cases}$$

  This model is exactly like the linear one, but it assumes that for $N > N^*$ we have constant predation rate aN^*. The parameter a is called *attack rate*, and it measures, after an encounter between prey and predator, the success rate of predation. Note that the function $\pi(N)$ is not $\mathcal{C}^1$ but it is Lipschitz.

- **Holling type II**

$$\pi(N) = \frac{aN}{1 + a\tau N},$$

  with $a, \tau > 0$. This is a smooth version of Holling type I. For small N, $\pi(N) \approx aN$, so the meaning of a is the same as above. For large N, we have that $\pi(N) \to 1/\tau =: \alpha$, which is called *maximum killing rate*. That is, α is the number of preys killed by one predator in a unit of time, when the number of preys is very large. Alternatively, we can interpret τ as the time required by the predator to consume the prey.

- **Holling type III**

$$\pi(N) = \frac{\alpha N^\theta}{\nu^\theta + N^\theta},$$

  with $\alpha, \nu > 0$ and $\theta > 1$. The last type also accounts for a lower predation rate as low density of preys. In fact, for small N and $\theta > 1$ we have $\pi'(0) = 0$. For $N = \nu$, $\pi(\nu) = \frac{\alpha}{2}$, and for $N \to \infty$ we have $\pi(N) \to \alpha$, the maximum killing rate. Thus, ν is the number of preys at which the killing rate is exactly half of the maximum one.

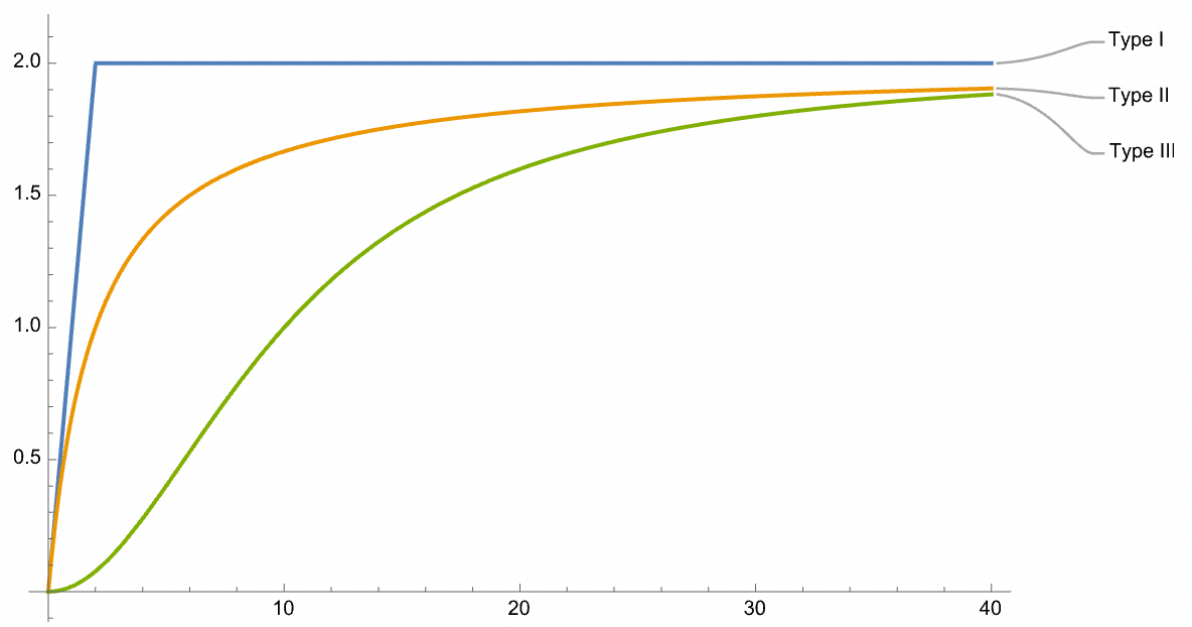


Figure 4.2: Different types of Holling predations

It is possible to justify Holling type II in a few ways. One, we will see in the completely different context of enzymatic reactions. A simpler one is as follows. In a time T, a single predator will spend $1/\pi(N)$ time in hunting. (Always keep in mind the bathtub example!) But the total time splits into T_s, the time spent seeking for a prey, and τ, the time needed to consume a prey. The time τ is fixed, no matter how large is N. The time T_s, however, is exactly $(aN)^{-1}$, because the more preys the easier is to catch them. Putting all together we have:

$$\frac{1}{\pi(N)} = T = \tau + T_s = \tau + \frac{1}{aN}, \quad \Rightarrow \quad \pi(N) = \frac{aN}{1 + a\tau N}.$$

4.3 Spruce budworm model

Figure 4.3: The effect of the spruce budworm on a forest.

The spruce budworm is an insect that feeds on needles of balsam fir trees (see this website). If needles are removed, the tree dies. Historical data in Canada evergreen forests, where the budworm is present, shows that in most years the budworm density is very low. However, in a outbreak year, the budworm population spikes and can kill up to 80% of mature trees in the forest. The period of the outbreak is roughly 30-70 years.

The spruce budworm model has been introduced in Ludwig et al. (1978). They proposed a system of 3 variables:

1. $N(t)$, the budworm density,
2. $S(t)$, the habitat space for larvae, and
3. $E(t)$, a measure of food energy reserves available to the budworm.

We focus here on the equation for $N(t)$. In fact, $S(t)$ and $E(t)$ will vary very slowly compared to $N(t)$, thus they can be assumed constant. (This argument can be made rigorous, as we shall see for enzymatic reactions.)

The equation for $N(t)$ reads as follows:

$$N' = rN\Big(1 - \frac{N}{K}\Big) - \frac{\alpha P^* N^2}{\nu^2 + N^2}.$$

The first term is the logistic growth. The second one is the predation of the budworms due to birds. This is Holling type III. Note that the parameters are fixed numbers (they do not depend on time), but they will depend on S and E, somehow. Thus, it will be interesting to see what happens to the system as we change them.

Since we have too many parameters, we proceed with non-dimensionalization. Here we select:

$$\tau = \frac{t}{T}, \quad u = \frac{N}{\nu},$$

for some $T > 0$ to be selected. We have:

$$\frac{\nu}{T}\dot{u} = r\nu u\Big(1 - \frac{\nu u}{K}\Big) - \frac{\alpha P^* u^2}{1 + u^2},$$

thus by selecting

$$T = \frac{\nu}{\alpha P^*}, \quad \rho = \frac{r\nu}{\alpha P^*}, \quad q = \frac{K}{\nu}$$

we arrive at

$$\dot{u} = \rho u\Big(1 - \frac{u}{q}\Big) - \frac{u^2}{1 + u^2},$$

where we are left with only 2 parameters:

- ρ is proportional to the intrinsic growth rate, while
- q is the carrying capacity normalized to the half-saturation population ν.

4.3.1 Equilibria and stability

As usual, we start by looking for equilibria of the system, that is solutions of

$$\rho u\Big(1 - \frac{u}{q}\Big) - \frac{u^2}{1 + u^2} = 0.$$

We have that $u = 0$ is an equilibrium. The others solve the equation

$$\rho\Big(1 - \frac{u}{q}\Big) - \frac{u}{1 + u^2} = 0,$$

which would lead to a 3rd-order polynomial equation, thus we can expect up to 3 real solutions. An analytical approach is not practical. However, equilibria are intersections of the two curves $f(u)$ and $g(u)$ where

$$f(u) = \rho\Big(1 - \frac{u}{q}\Big), \quad g(u) = \frac{u}{1+u^2}.$$

The function $f(u)$ represents the per capita growth rate of u, whereas $g(u)$ is the per capita death rate due to predation. Thus, solutions of the equation are equilibria of the system. Since the function $g(u)$ does not depend on any parameters, we can fix it and simply change $f(u)$, which is a segment.

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

sns.set_theme("notebook", style="whitegrid")

u  = np.linspace(0,12,1000)

fig,axs = plt.subplots(1,2,figsize=(9,3))
q = 4
uq = np.linspace(0,q,2)
axs[0].plot(u, u/(1+u**2))
for rho in np.linspace(0.2,0.8,10):
    pr = np.polynomial.Polynomial([q*rho,-q-rho,q*rho,-rho])
    rr = np.real(np.array([r for r in pr.roots() if np.isreal(r) and r >= 0]))
    axs[0].plot(uq,rho*(1-uq/q),'k',lw=0.5)
    axs[0].plot(rr,rho*(1-rr/q),'k.')

q = 10
uq = np.linspace(0,q,2)
axs[1].plot(u, u/(1+u**2))
for rho in np.linspace(0.1,0.8,6):
    pr = np.polynomial.Polynomial([q*rho,-q-rho,q*rho,-rho])
    rr = np.real(np.array([r for r in pr.roots() if np.isreal(r) and r >= 0]))
    axs[1].plot(uq,rho*(1-uq/q),'k',lw=0.5)
    axs[1].plot(rr,rho*(1-rr/q),'k.')

p = np.polynomial.Polynomial([q,0,-q,2])
rho = lambda u: q*u/((1+u**2)*(q-u))
bif_u   = np.real(np.array([r for r in p.roots() if np.isreal(r) and r >= 0]))
bif_rho = rho(bif_u)
for r,u in zip(bif_rho,bif_u):
```

```
    axs[1].plot(uq,r*(1-uq/q),'r',lw=1.0)
    axs[1].plot(u,r*(1-u/q),'r.',markersize=8)

for ax in axs:
    ax.grid(False)
    sns.despine()
    ax.xaxis.set_ticks([])
    ax.yaxis.set_ticks([])
    ax.set_xlabel('u',loc='right')
    ax.set_xlim(0,12)
    ax.set_ylim(0,1)

axs[1].yaxis.set_ticks(bif_rho)
axs[1].yaxis.set_ticklabels(['$\\rho_2$','$\\rho_1$'])

plt.show()
```

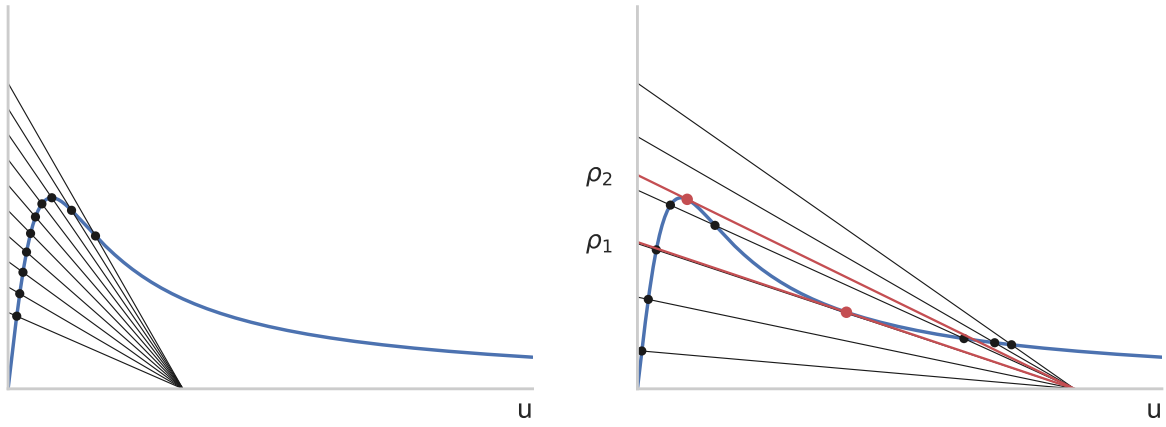


Figure 4.4: Equilibria for the spruce budworm model. On the left, the case of small q, on the right the case of large q.

We can see from the figure that when q is small, we only have a single equilibrium. When q is large, we can have 1, 2, or 3 additional equilibria, depending on ρ: given $\rho_1 < \rho_2$ we have:

1. If $\rho < \rho_1$, then there exists a single equilibrium $u_1^* \ll q$. This is the *refuge* equilibrium (low population). The equilibrium is globally stable, because $f(u) - g(u) > 0$ for $u < u_1^*$ and $f(u) - g(u) < 0$ for $u > u_1^*$.
2. If $\rho > \rho_2$, then there exists a single equilibrium u_3^* close to q. This is the *outbreak* equilibrium (large population). The equilibrium is globally stable, because $f(u) - g(u) > 0$ for $u < u_3^*$ and $f(u) - g(u) < 0$ for $u > u_3^*$.

3. If $\rho_1 < \rho < \rho_2$, then there exist 3 equilibria $u_1^* < u_2^* < u_3^*$. The stability of u_1^* and u_3^* is as above, but now locally. The equilibrium u_2^* is unstable.

The case 3. is the most interesting one. If $u(0) < u_2^*$, then $u \to u_1^*$, otherwise $u \to u_3^*$. This is again a threshold effect.

4.3.2 Bifurcation diagram

As we change ρ, we may have a different number of equilibria. Thus, there must be some bifurcation occurring. The equilibria are on the curve defined by

$$h(u, \rho) = \rho\Big(1 - \frac{u}{q}\Big) - \frac{u}{1+u^2} = 0.$$

The function $h(u, \rho)$ is smooth in ρ, since $\partial_\rho h \neq 0$ for $u \in [0, q)$. Thus we can write ρ as a function of u:

$$\rho(u) = \frac{qu}{(1+u^2)(q-u)}.$$

Since $h(u, \rho(u)) = 0$, the curve $(u, \rho(u))$ with $u \in [0, q)$ defines a curve of equilibria, shown below:

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

sns.set_theme("notebook", style="whitegrid")

q = 12 #3*np.sqrt(3)
u = np.linspace(0,0.9*q,1000)
rho = lambda u: q*u/((1+u**2)*(q-u))

p = np.polynomial.Polynomial([q,0,-q,2])
bif_u   = np.array([r for r in p.roots() if np.isreal(r) and r >= 0])
bif_rho = rho(bif_u)

rho_neg = p(u) < 0

with plt.xkcd(scale=0.5):
    fig, ax = plt.subplots()
    sns.despine()
    ax.xaxis.set_ticks([])
    ax.yaxis.set_ticks([])
```

```
    ax.axhline(y=q,color='k',linestyle='--',linewidth=1.0)
    ax.plot(rho(u),0*u,'r')
    ax.plot(rho(u), u, 'b')
    ax.plot(rho(u)[rho_neg], u[rho_neg], 'r')
    ax.plot([0],[0],'k.',markersize=16)
    ax.plot(bif_rho,bif_u,'k.',markersize=16)
    ax.grid(False)
    ax.set_xlabel('$\\rho$')
    ax.set_ylabel('$u^*$')
    ax.fill_between(rho(u), q, where=rho_neg, facecolor='gray', alpha=.2)

    if len(bif_u) > 0:
        ax.annotate('  $\\rho_2$',(bif_rho[0],bif_u[0]),ha='left',va='center')
        ax.annotate('$\\rho_1$  ',(bif_rho[1],bif_u[1]),ha='right',va='center')
        ax.annotate('$q$',(rho(u).max(),q),ha='left',va='top')

    plt.show()
```

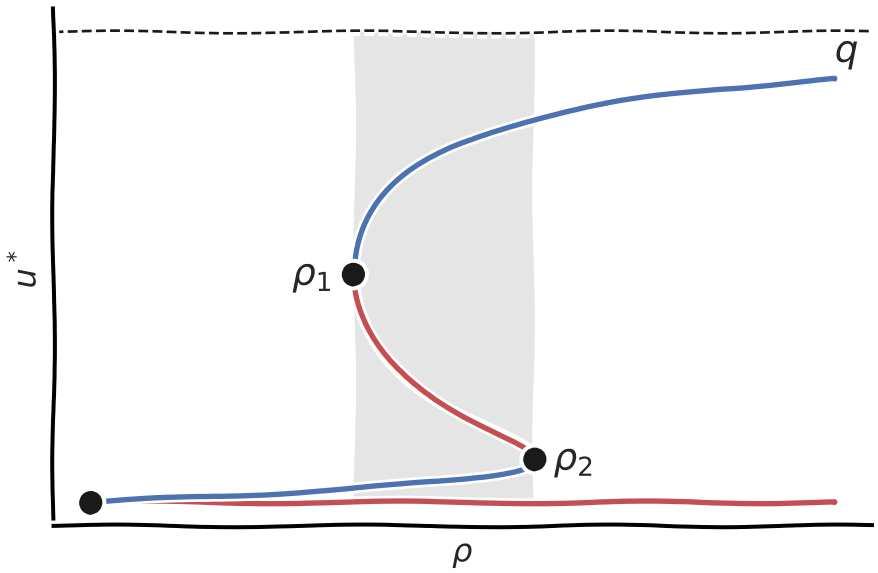


Figure 4.5: Bifurcation diagram with respect to ρ for the spruce budworm system, for $q = 12$. In blue, the stable equilibrium, in red the unstable one. The shaded region is bistable.

The plot above is a bifurcation diagram. We can interpret it as follows: given $\rho = \bar{\rho}$, the equilibria are those corresponding to the intersections between $\rho = \bar{\rho}$ and the curve $\rho = \rho(u)$.

So, for $\rho < \rho_1$ we have one equilibrium, for $\rho_1 < \rho < \rho_3$ we have 3, and for $\rho > \rho_3$ we have one.

For $\rho = \rho_1$ or $\rho = \rho_2$, we have a *tangent bifurcation* (See Section B.5). As we can see, the curve of equilibria is always smooth, with no branching or crossing on another curve of equilibria, like in the case of *transcritical bifurcation*. However, there is a change in stability: in fact, the branch between ρ_1 and ρ_2 corresponds to u_2^*, which is unstable.

When varying also q, the tangent bifurcations points ρ_1 and ρ_2 moves as well. In particular, as q is reduced, the two bifurcation points will get closer until they meet for $q = 3\sqrt{3}$. This point is another bifurcation, called *cusp bifurcation*. Beyond this point, the system is never bistable.

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

sns.set_theme("notebook", style="whitegrid")

rho = lambda u,q: q*u/((1+u**2)*(q-u))
Q = []
R = []
for q in np.arange(1,15,0.01):
    p = np.polynomial.Polynomial([q,0,-q,2])
    bif_u = np.array([r for r in p.roots() if np.isreal(r) and r >= 0])
    if not len(bif_u): continue
    Q.append(q)
    R.append([rho(u,q) for u in bif_u])

R = np.array(R)
Q = np.array(Q)

q_cusp = 3*np.sqrt(3)
u_cusp = np.sqrt(3)
r_cusp = rho(u_cusp,q_cusp)

fig, ax = plt.subplots()
ax.plot(Q,R.min(axis=1),'r-')
ax.plot(Q,R.max(axis=1),'r-')
ax.plot(q_cusp, r_cusp, 'k.',markersize=16)
ax.fill_between(Q,R.min(axis=1),R.max(axis=1),color='r',alpha=0.2)
ax.set_xlim([2,None])
ax.set_ylim([0.2,0.8])
ax.set_ylabel('$\\rho$')
```

```
ax.set_xlabel('$q$')

ax.annotate('  Cusp point',(q_cusp,r_cusp),ha='left')
ax.annotate('Bistable region',(12,0.45),ha='center')
ax.annotate('Refuge region',(10,0.25),ha='center')
ax.annotate('Outbreak region',(12,0.65),ha='center')
plt.show()
```

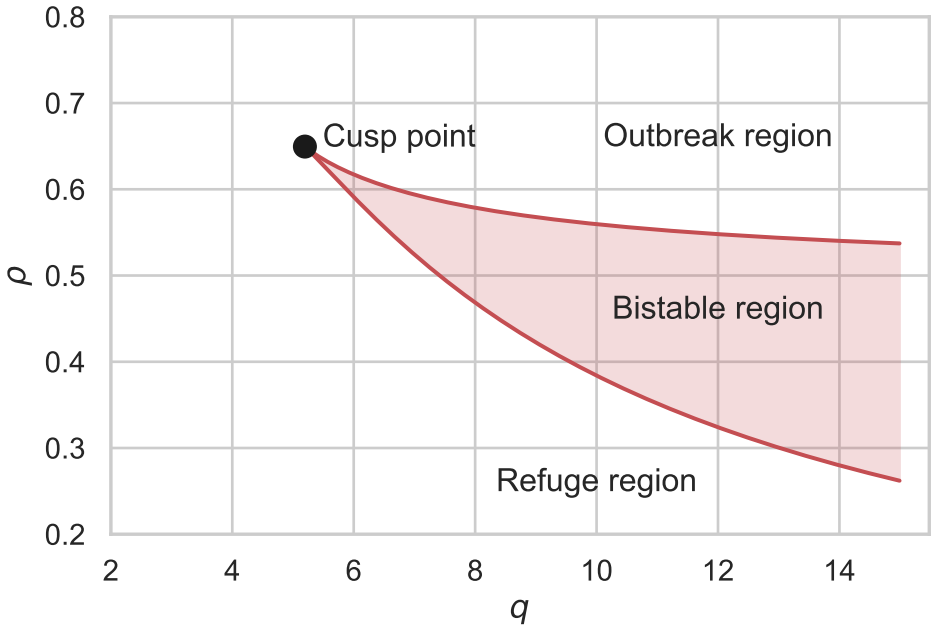


Figure 4.6: Bifurcation diagram with respect to (q, ρ) for the spruce budworm system. The shaded region is bistable. The cusp point is a bifurcation point of co-dimension 2. It occurs for $q = 3\sqrt{3}$.

If we plot the surface of equilibria in the space (q, ρ, u^*), as the solution of the equation $h(u, \rho, q) = 0$, we obtain the plot below. This plot clearly shows the bistable region and the threshold value (in red).

```
import numpy as np
import pyvista as pv
pv.set_jupyter_backend('static')

n = 50
```

```
bnd = np.array([[0.1,0.1,0],[20,1.0,20]])
grid = pv.ImageData(dimensions=(n,n,n),
                    spacing=(bnd[1,:]-bnd[0,:])/n,
                    origin=bnd[0,:])
Q,R,U = grid.points[:,0],grid.points[:,1],grid.points[:,2]
vals = R*(1-U/Q)-U/(1+U**2)

sols = U.copy()

grid.point_data['sols'] = sols
out = grid.contour(1,scalars=vals,rng=[0,0])
out.compute_normals(inplace=True,auto_orient_normals=True)
out.point_data['normals_u'] = out.point_data['Normals'][:,2] > 0
plotter = pv.Plotter()
plotter.add_mesh(out,scalars='normals_u',cmap=['red','blue'],
                 smooth_shading=True)
plotter.set_scale(xscale=1, yscale=20, zscale=0.5)
plotter.remove_scalar_bar()
plotter.add_axes(xlabel='q',ylabel='rho',zlabel='u')
plotter.camera_position = [
    (41, -11, 24),
    (11, 11, 1.7),
    (-0.33,0.4,0.84),
]
plotter.show()
```

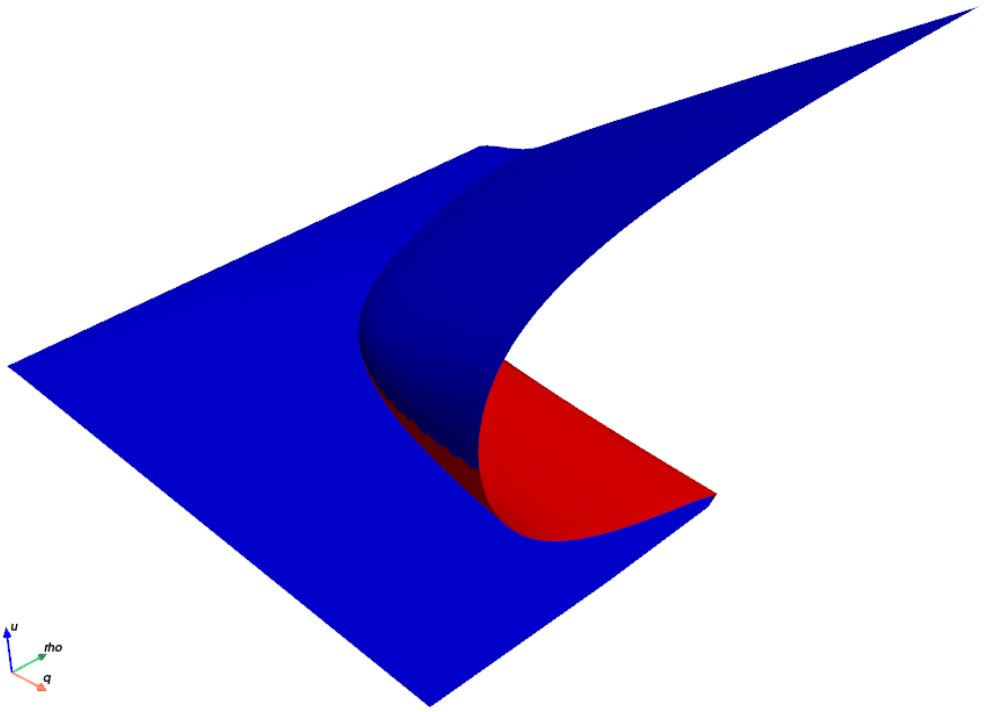


Figure 4.7: Bifurcation diagram with respect to (q, ρ) in 3D.

4.3.3 Hysteresis

Now that we have the bifurcation diagram, we can use it to find the periodic outbreaks. Remember that ρ and q are functions of S and E, but they vary very slowly compared to $N(t)$. On the other hand, when N is at equilibrium, it does not vary anymore, so even slow and small variations of E and S could matter.

Suppose to start with small ρ, say $\rho < \rho_1$. The spruce budworms is at refuge state u_1^*. Now we slowly increase ρ. The equilibrium $u_1^*(\rho)$ will only slightly increase. Once we react the point $\rho = \rho_2$, the equilibrium u_1^* disappears, so $u \to u_3^*$ (outbreak), the only other stable equilibrium. Note the the outbreak is fast, even for a small change of ρ. For this reason, the tangent bifurcation is a *catastrophic bifurcation*. As we keep increasing ρ, the outbreak equilibrium keeps increasing, but again slowly.

Since ρ is the intrinsic growth rate, we can assume that ρ will now start to decrease, because

there are too many spruce budworm consuming the resources. As we go back, decreasing ρ, we follow a specular path, jumping at $\rho = \rho_1$ from the outbreak to the refuge equilibrium. We go back to the original situation.

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

sns.set_theme("notebook", style="whitegrid")
q = 12
u = np.linspace(0,0.9*q,1000)
rho = lambda u: q*u/((1+u**2)*(q-u))

p = np.polynomial.Polynomial([q,0,-q,2])
bif_u   = np.array([r for r in p.roots() if np.isreal(r) and r >= 0])
bif_rho = rho(bif_u)

rho1 = bif_rho.min()
rho2 = bif_rho.max()

p1 = np.polynomial.Polynomial([q*rho1,-q-rho1,q*rho1,-rho1])
p2 = np.polynomial.Polynomial([q*rho2,-q-rho2,q*rho2,-rho2])

u0 = p1.roots().min()
u1 = bif_u.min()
u2 = p2.roots().max()
u3 = bif_u.max()

rho_neg = p(u) < 0

with plt.xkcd(scale=0.5):
    fig, ax = plt.subplots()
    sns.despine()
    ax.xaxis.set_ticks([])
    ax.yaxis.set_ticks([])

    ax.plot(rho(u), u, 'k', lw=1.0)
    for lb,ub in [[u0,u1],[u2,u3]]:
        uu = np.linspace(lb,ub,100)
        ax.plot(rho(uu),uu, 'r')
    for lb,ub in [[u1,u2],[u3,u0]]:
        ax.plot([rho(lb),rho(ub)],[lb,ub], 'r')
    ax.plot([rho(.7*u0+.3*u1)],[.7*u0+.3*u1],'r>')
```

```
    ax.plot([rho(0.5*(u2+u3))],[0.5*(u2+u3)],'r>')
    ax.plot([rho2],[0.5*(u1+u2)],'r^')
    ax.plot([rho1],[0.5*(u3+u0)],'rv')
    ax.grid(False)
    ax.set_xlabel('$\\rho$')
    ax.set_ylabel('$u^*$')

    ax.annotate('  $\\rho_2$',(bif_rho[0],bif_u[0]),ha='left',va='center')
    ax.annotate('$\\rho_1$  ',(bif_rho[1],bif_u[1]),ha='right',va='center')
    ax.annotate('$q$',(rho(u).max(),q),ha='left',va='top')

    plt.show()
```

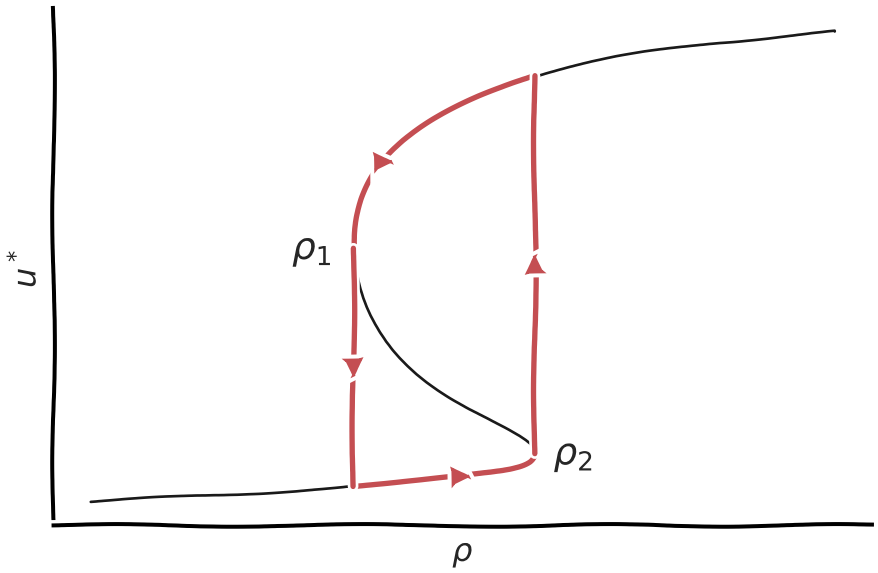


Figure 4.8: Bifurcation diagram with respect to ρ for the spruce budworm system. In blue, the stable equilibrium, in red the unstable one. The shaded region is bistable.

The result is a *hysteresis loop*, which could explain the periodic outbreaks.

We will analyze this model more in depth during the Lab session.

References

Gause, G. F. 1934. *The Struggle for Existence*. William & Wilkins Company.

Holling, Crawford Stanley. 1965. "The Functional Response of Predators to Prey Density and Its Role in Mimicry and Population Regulation." *The Memoirs of the Entomological Society of Canada* 97 (S45): 5–60. https://doi.org/10.4039/entm9745fv.

Ludwig, Donald, Dixon D Jones, Crawford S Holling, et al. 1978. "Qualitative Analysis of Insect Outbreak Systems: The Spruce Budworm and Forest." *Journal of Animal Ecology* 47 (1): 315–32. https://doi.org/10.2307/3939.

Pulley, Melissa. 2020. "The Marshmallow Lab: A Project-Based Approach to Understanding Functional Responses." Master's thesis, Utah State University. https://doi.org/https://doi.org/10.26076/neyp-cc41.

Part II

Assignments

Solving ODEs

The logistic model

Consider the following model for a population:

$$N' = \beta(N)N - \mu(N)N,$$

where $\beta(N)$ is the *fertility rate* and $\mu(N)$ is the *mortality rate*. We assume here both are function of the population size N as follows:

$$\beta(N) = \beta_0 - \tilde{\beta}N,$$
$$\mu(N) = \mu_0 + \tilde{\mu}N.$$

1. Show that the model can be written as

$$N' = r\Big(1 - \frac{N}{K}\Big)N. \tag{1}$$

2. Prove that the ODE, supplemented with an initial condition $N(0) = N_0$, has a unique local solution. *Hint: verify the hypotheses of the Cauchy-Lipschitz theorem.*

3. Consider $u(t) = 1/N(t)$. Show that $u(t)$ satisfies a linear ODE. Solve it, then use the solution to solve the original ODE in $N(t)$ with the initial condition $N(0) = N_0 > 0$.

4. Make the change of variables as follows: $\tau = rt$ and $y = N/K$. Find the corresponding equation in $y(\tau)$.

Blow-up of solutions

It may be considered reasonable that, for a sexual species, births are proportional to the number of encounters, hence, disregarding the mortality process, we have the equation

$$N'(t) = \beta N^2(t).$$

1. This equation can be solved by the method of separation of variables. Show that the solutions of this equation with $N(0) > 0$ tend to infinity in a finite time. (Blow-up of solution.)

2. Let us correct the equation, introducing deaths:

$$N'(t) = \beta N^2(t) - \mu N(t).$$

Discuss how the dynamics changes. In particular, does the equation still have the problem of solutions going to infinity in a finite time? *Hint: do not solve analytically the equation.*

3. Solve the above equation analytically and confirm the qualitative analysis of the previous point.

mRNA

Zeisel *et al.* considered the following system for the concentration of mRNA. The variables are $M(t)$, the concentration of mature mRNA, and $P(t)$, the concentration of precursor mRNA:

$$\begin{cases} P'(t) = b(t) - \alpha_1 P(t), \\ M'(t) = \alpha_1 P(t) - \alpha_2(t) M(t). \end{cases}$$

where $b(t)$ is the (gene-specific and time-dependent) production rate, α_1 is the conversion (splicing) rate of pre-mRNA to mRNA, $\alpha_2(t)$ the (time-dependent) degradation rate of mRNA.

1. Assume $b(t) \equiv b$ and $\alpha_2(t) \equiv \alpha_2$ are constant. Find the explicit solution given $P(0) = P_0$ and $M(0) = M_0$. *Hint: use the method of variation of constants.*

2. Assume now that $\alpha_2(t) \equiv \alpha_2$ and

$$b(t) = \begin{cases} \bar{b} & t < \bar{t}, \\ 0, & t > \bar{t}. \end{cases}$$

Show that $M(t)$ tends to 0 as $t \to \infty$. At which rate does it decay?

Thorium-Uranium dating

The thorium-uranium method for dating rocks is based on the fact that Uranium-234 decays into Thorium-230 which in turn decays into other elements. Set $t = 0$ the rock formation time and denoting $U(t)$ (resp. $T(t)$) the amount of Uranium-234 (resp. Thorium-230) in the rock at time t (measured in years), the following differential equation system is written:

$$\begin{cases} U'(t) = -aU(t), \\ T'(t) = aU(t) - bT(t), \\ U(0) = U_0, \\ T(0) = 0, \end{cases}$$

where $a \approx 5.9 \cdot 10^{-6}\,\text{years}^{-1}$, $b \approx 1.9 \cdot 10^{-5}\,\text{years}^{-1}$, U_0 represents the initial (generally unknown) amount of Uranium-234. Note that, based on geological principles, it is believed that there was no thorium at the time of rock formation.

1. Solve the equation for $U(t)$.

2. How can the quantities of a and b be interpreted? From the data provided can we infer the half-life of Uranium-234 and Thorium-230?

3. Calculate $T(t)$, solution of the second differential equation.

4. Compute
$$\lim_{t\to\infty} \frac{T(t)}{U(t)}.$$

5. Explain why it is possible to estimate the rock age from the knowledge of T/U at current time, but it is not possible from the knowledge of T alone. *Hint: study the function* $T(t)/U(t)$.

Population dynamics

Periodic solutions

Consider the time dependent Malthus model

$$\begin{cases} N(t) = r(t)N(t), \\ N(0) = N_0. \end{cases} \tag{4.1}$$

with periodic Malthus parameter: $r(t+T) = r(t)$ and denote by $\bar{r}$ the average over one period:

$$\bar{r} = \frac{1}{T}\int_0^T r(s)\,\mathrm{d}s. \tag{4.2}$$

1. After showing that the function

$$\pi(t) = \int_0^t r(s)\,\mathrm{d}s - \bar{r}t$$

is periodic with period T, prove that

$$\bar{r} = \lim_{t\to\infty} \frac{1}{t}\int_0^t r(s)\,\mathrm{d}s.$$

2. Show that the solution of the ODE is

$$N(t) = e^{\bar{r}t}N_\pi(t),$$

where $N_\pi(t)$ is a periodic function.

3. Modify the ODE by adding a time-dependent migration

$$N'(t) = r(t)N(t) + m(t), \tag{4.3}$$

assuming both $r(t)$ and $m(t)$ continuous and periodic with period T. Using Equation 4.2, show that, if $\bar{r} < 0$, then the function

$$N_\infty(t) = \int_{-\infty}^t e^{\int_s^t r(\sigma)\,\mathrm{d}\sigma} m(s)\,\mathrm{d}s$$

is well-defined, is a solution to the ODE, and is periodic with period T.

4. Show that if $\bar{r} < 0$, then the solution to Equation 4.3 with initial condition $N(0) = N_0$ is

$$N(t) = e^{\int_0^t r(\sigma)\mathrm{d}\sigma} N_0 + \int_0^t e^{\int_s^t r(\sigma)\mathrm{d}\sigma} m(s)\,\mathrm{d}s$$

and is such that $\lim_{t\to\infty}(N(t) - N_\infty(t)) = 0$.

Logistic model

Consider the logistic equation

$$N'(t) = r\Big(1 - \frac{N}{K}\Big)N.$$

1. Find the equilibria of the ODE and study their stability.
2. Study the global stability of the equilibria.
3. Would it make sense to assume $r < 0$ and $K > 0$?

Global well-posedness

Consider the following ODE model:

$$N'(t) = (\beta N^2 - \mu N)(1 - \gamma N).$$

Find all non-negative equilibria and discuss their stability. From the direction field conclude that all solutions are bounded, and hence solutions are globally defined (no blow-up in finite time).

Predation

Fishery model

(From Strogatz, 2015, exercise 3.7.4)

A fishery model with harvesting has the form

$$N' = r\Big(1 - \frac{N}{K}\Big)N - E\frac{N}{A+N},$$

where all parameters are positive. The parameter E is the harvest effort. Note that harvest rate increases with N because it is harder to catch fish when the population size is small.

1. Give a biological interpretation of A.
2. Show that the system can be rewritten in dimensionless form as

$$\dot{u} = u(1-u) - h\frac{u}{a+u}$$

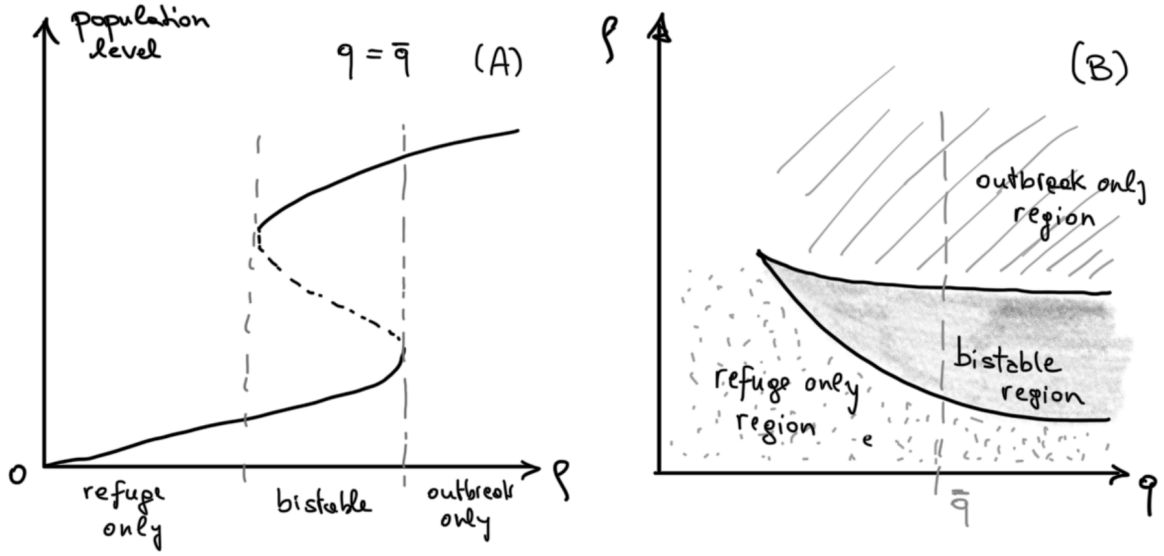
for a suitable choice of quantities.

3. Show that the system can have one, two, or three equilibria, depending on the values of a and h. Classify the stability in each case.
4. Plot the bifurcation diagram with respect to h, for a fixed value of a. Determine all bifurcation points and whether they are catastrophic or not. Is there a bistable region?
5. Suppose that we slowly increase the fishing effort from zero, assuming that the fish population is at the carrying capacity at the beginning. Show the dynamics.
6. Plot the stability diagram of the system in (a, h) parameter space. Can hysteresis occur in any of the stability regions?

Spruce-budworm model

The spruce-budworm model describes the population of budworms over time, for a given initial population. There are only 2 relevant parameters: ρ, the intrinsic growth rate of budworms; and q, the carrying capacity. An *outbreak* is when the stable population is very large, whereas a *refuge* is when it is very low.

Depending on ρ, we may have various stable population levels, according to the following diagrams:



In diagram (A) we see the effect of changing ρ with a fixed value of $q = \bar{q}$. In diagram (B) we visualize the region in the parameters space where 3 equilibria are present (bistable region).

We would like to analyze various strategies to efficiently go from an outbreak to a refuge.

1. Strategy (1): we reduce the population by reducing ρ (for instance, by preventing mating). Does it work? Use the diagrams to show how the population varies as we reduce ρ.

2. Strategy (2): we reduce the population by using an insecticide. Does it work? Use diagram (A). Observe that if we kill worms, we move the population away from the equilibrium, hence it will quickly rebound to the closest stable equilibrium.

3. Strategy (3): we reduce q, for instance by spraying a defoliant. Does it work? Use diagram (B).

4. Strategy (1+2+3): we reduce q and ρ simultaneously, and then use the insecticide. Does it work?

5. After the application of Strategy (1), ρ starts to increase again back to its original value (large). Explain how the population varies. Does it increase in the exactly same matter as it decreased?

Logistic with resources

Assume that the growth rate of a population depends on available resources $\rho(t)$, according to a general law $r(t) = G(\rho(t))$.

Assume for the moment that the total amount of resources is a fixed constant C, but they can be free (hence available, denoted by $\rho(t)$) or used by the population, denoted by $H(N(t))$, where dependence on the population is shown. In other words

$$\rho(t) = C - H(N(t)), \tag{4.4}$$

and the resulting model

$$N' = G(C - H(N))N, \tag{4.5}$$

will be specified when the functions $G(\cdot)$ and $H(\cdot)$ are given.

1. Explain why reasonable assumptions are that both G and H are increasing functions with $G(0) < 0$, $H(0) = 0$.

2. Choose a linear form for G and H and show that the growth rate has the form $r(N) = r - \alpha N$.

3. With the previous assumptions, does Equation 4.5 always have a positive equilibrium? If not, find the conditions under which it does and find the expressions for the intrinsic rate of growth r and the carrying capacity K.

4. Using generic functions G and H satisfying the assumptions in 1., find under which conditions the equation (2) has a unique positive equilibrium. When is it asymptotically stable?

5. What could be other reasonable assumptions for ρ instead of Equation 4.4?

6. Modify the equation to

$$\begin{cases} N' = G(R)N, \\ R' = \rho_0 - H(R, N), \end{cases}$$

where

$$G(R) = m\frac{R-a}{R+1} - \mu_0, \quad H(R,N) = (c+bN)\frac{R-a}{R+1}N - R.$$

Here $N(t)$ represents population density, and $R(t)$ available resources, and all constants are supposed to be positive.

With the help of Phase Plane in MATLAB, explore the solutions of the problem using parameter values such that:

$$\rho_0 > \frac{\mu_0 + ma}{m - \mu_0}.$$

Part III

Labs

Lab 01: Numerical integration

The purpose of this lab sesssion is to introduce some basic numerical schemes for the following initial value problem (IVP)

$$\begin{cases} \mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \\ \mathbf{y}(t_0) = \mathbf{y}_0. \end{cases}$$

The simplest method is *Forward Euler* method or *Explicit Euler*.

In the Forward Euler method, the numerical solution $\mathbf{u}$ is obtained from the following algorithm: given $h > 0$ (timestep), we set $\mathbf{u}_0 = y_0$ and we iterate for $n \geq 0$:

$$u_{n+1} = u_n + hf(t_n, u_n)$$

until $nh > T$, being T the final time.

Alternatively to Forward Euler, we can have the following scheme, called *Backward Euler*:

$$u_{n+1} = u_n + hf(t_{n+1}, u_{n+1}).$$

Please note that the unknown u_{n+1} is implicitly defined. If f is nonlinear, then it might not that trivial to obtain it! In our example, the ODE is linear and the solution is easily found:

$$u_{n+1} = u_n + h\lambda u_{n+1} \qquad \Rightarrow \qquad u_{n+1} = (1 - h\lambda)^{-1} u_n.$$

In the function below we directly use λ and not f.

Stability and convergence

Consider the IVP (in numerics, this is also called *test problem*):

$$\begin{cases} y' = \lambda y, \\ y(t_0) = y_0. \end{cases}$$

1. Write down the forward Euler scheme and solve for the equation for various values of h, with $\lambda = -1$. Is it always true that $y_n \to 0$ for $n \to \infty$? Use $t_0 = 0$ and $y_0 = 1$.

2. Write down the backward Euler scheme and repeat the previous step.

3. Solve the equation with the forward Euler scheme and compare the solution to the exact one. How does the absolute error decrease with respect to h? Plot the error in the logarithmic scale. Set $\lambda = 1$ and $y(0) = 1$.

4. Repeat the previous step with the Heun scheme:

$$y_{n+1} = y_n + \tfrac{h}{2}(f(y_n) + f(y_n + hf(y_n))).$$

Nonlinear equations

Consider the Nagumo or bistable model:

$$u' = au(1-u)(u-\alpha),$$

with $\alpha = 0.2$ and $a = 10$. Implement and solve the model in $t = 0, \dots, 2$ with forward Euler for $h = 10^{-3}$ and various values of $u(0)$.

Van Der Pol equation

Here a more complicate example of a system. We solve the Van Der Pol equation:

$$y'' - \mu(1-y^2)y' + y = 0.$$

1. Solve the van der Pol equation with forward Euler, using $\mu = 5$, $y(0) = 2$ and $y'(0) = 0$. How small h needs to be? Find the approximate stability limit by bisection.

2. Solve the van der Pol equation with backward Euler. Design a strategy to solve the non-linear equation. Is the scheme always stable?

Conservation of energy

Consider the 2nd-order ODE $y'' + \omega^2 y = 0$ (linear oscillator).

1. After recasting the equation to a first-order system, solve it with forward Euler using $\omega = 1$, $h = 0.01$, $y(0) = 1$, $y'(0) = 0$ and $t = 0, \dots, 100$. Does the solution match the expected behavior?

2. Try again with backward Euler. *Hint: the system is linear, you can use the backslash to solve it at each iteration.*

3. Compute the discrete energy of the system $E(t) = \frac{1}{2}((y'(t))^2 + \omega^2 y(t)^2)$ and inspect the problem.

4. Implement the following strategy: first, update the velocity, then the position with the newly compute velocity. (This is called the *symplectic Euler method*.) Does the method conserve the (discrete) energy?

Lab 02: Predation

Holling type predation

Holling type functional responses can be experimentally observed with a simple game. This is inspired to the MSc thesis from Pulley (2020), where the author proposes a similar game with marshmallows.

The experiment

We need the following tools:

- A desk or a clean area where to perform the experiment;
- Two stopwatches (there's an app in every phone);
- A bag of small objects to be collected or eaten. In the original proposal, they were marshmallow, which can be eaten. If you're concerned about your diet or teeth, you can use other objects (coins, beans, pens,…)
- A blindfold.
- A group of at least 4 students.

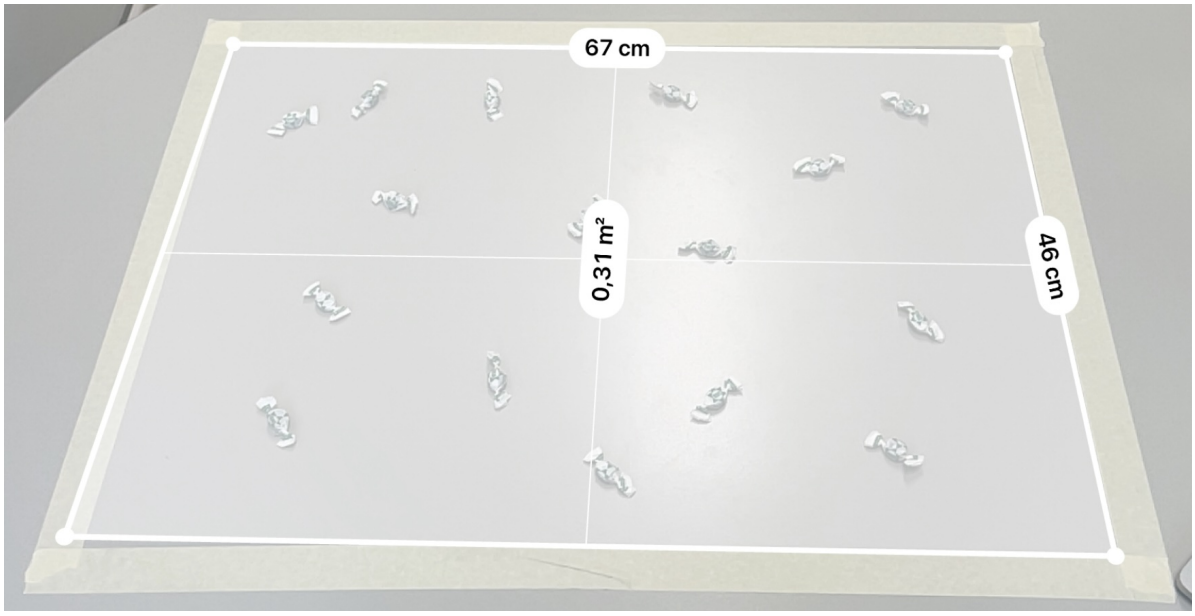


Figure 4.9: Desk with a $0.31\,\mathrm{m}^2$ delimited area with $N = 16$ candies. The density here is $16/0.31 \approx 52\,\mathrm{candies/m}^2$. You can measure the area with an app.

The game is as follows:

1. We *randomly* place on the desk a certain number of objects. We take note of this number N;
2. Student A is blindfolded;
3. Student B says "Go!" and starts the stopwatch. In the meanwhile, student A starts searching on the desk for an object.
4. When Student A finds one object, he will collect/eat it.
5. Student C, with another stopwatch, will measure the total time spent eating/collecting. Thus, it starts the stopwatch when student A picks up the object, and stops it when student A starts searching again.
6. Student D will put a new object at random on the desk so to keep N constant.

The experiment can last 90 seconds. At the end of the experiment, we need to mark

- the total number of preys N,
- the total number of preys captured C,
- the total handling time T^*.

To avoid statistical fluctuations, we repeat the experiment 3 times. Data can be collected in a table. You can use the following link to copy the table: CandyLab spreadsheet.

Number of candies	Number of captured prey				Total handling time [sec]				Handling time
	Trial 1	Trial 2	Trial 3	Avg	Trial 1	Trial 2	Trial 3	Avg	
2	1	1	1	1	2	2	2	2	2
4									
6									
15									
20									

Figure 4.10: Spreadsheet for data collection

Once done, we compute the handling time $T_h = \bar{T}^*/\bar{C}$.

The analysis

Suppose that T is the total experiment time (90 seconds for us). During this time, we spend T_s in searching for a prey, and T_h in handling a prey. Thus, we have

$$T = T_s + CT_h,$$

where the total handling time is multiplied by the number of prey captured. The assumption we make is that the total number of captured preys is proportional to the searching time and the total number of prey N. So:

$$C = aNT_s,$$

for some $a > 0$. Substituting T_s we obtain:

$$C = aNT_s = aN(T - CT_h),$$

that we can solve for C/T, that is the rate of predation (number of prey captured per unit time):

$$\frac{C}{T} = \frac{aN}{1 + aT_hN}$$

So, the functional response should be exactly Holling type II.

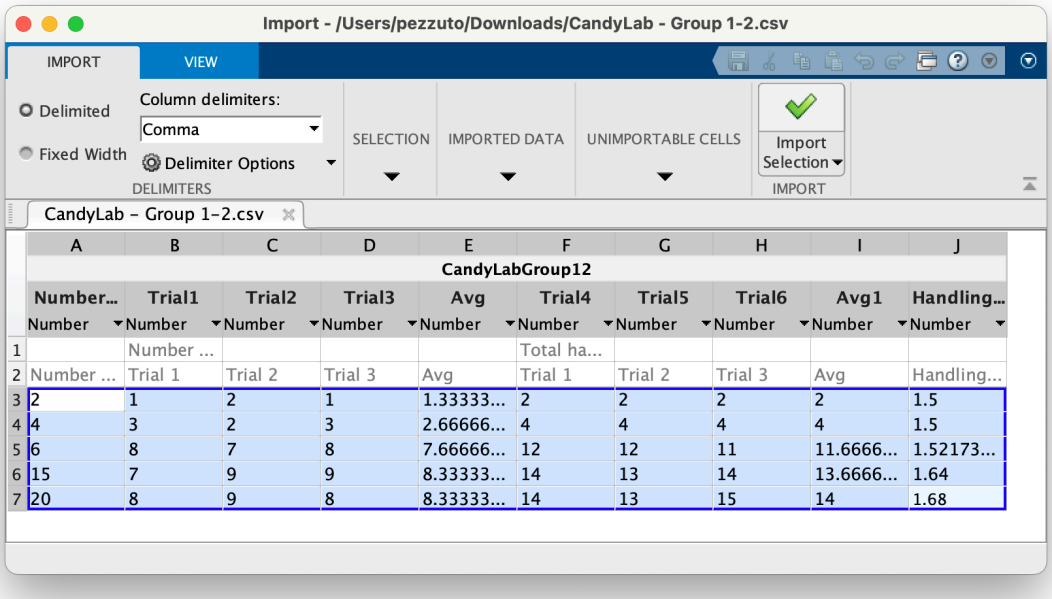
Import data

Before trying any fitting, we need to import the data into a software. For instance, this can be MATLAB or Python.

Save and download the spreadsheet as `csv` format.

4.3.1 MATLAB

Use the `Import Data` button, then select the columns of your spreadsheet. The variables will appear in your workplace.



You can access the variables with `CandyLabGroup.NumberOfCandies`, for instance. Just type `CandyLabGroup` to see all variables.

Plot the results.

4.3.2 Python

In Python there is no GUI for importing `csv` files, but you can use a module:

```
import csv

import csv
with open('CandyLab - Group 1.csv') as csvfile:
    reader = csv.reader(csvfile)
    lines = list(reader)

N  = [int(l[0]) for l in lines[3:]]
```

```
C1 = [int(l[1]) for l in lines[3:]]
C2 = [int(l[2]) for l in lines[3:]]
C3 = [int(l[3]) for l in lines[3:]]
```

Fitting

The objective is to fit the function:

$$\frac{C}{T} = \frac{aN}{1 + aT_h N}$$

Since T is fixed, we can fit C versus N for various values, and find a and T_h.

There are at least 2 options you can try:

Option 1: Linear fit via Lineweaver–Burk plot

The Lineweaver–Burk plot uses a simple transformation to make the above equation linear. In fact:

$$y = \frac{T}{C} = \frac{1}{a}\frac{1}{N} + \frac{1}{T_h} = \alpha x + \beta.$$

So, the Holling type II relationship is *linear* with respect to the reciprocal. We can perform a linear regression to find the coefficients (use `polyfit`).

Option 2: Nonlinear regression

We can approach the problem as follows: find (a, T_h) such that

$$g(a, T_h) := \sum_{i=1}^{3} \left(\frac{C}{T_i} - \frac{aN_i}{1 + aT_h N_i} \right)^2 \quad \to \quad \min .$$

This problem can be solved via optimization. In MATLAB, you can use `nlinfit` function, in Python `scipy.optimize.curve_fit`.

Otherwise, you can try to use a steepest descrent approach: given an initial guess $(a^{(0)}, T_h^{(0)})$, we update with:

$$(a^{(k+1)}, T_h^{(k+1)}) = (a^{(k)}, T_h^{(k)}) - \eta \nabla g((a^{(k)}, T_h^{(k)})),$$

where $\eta > 0$ is the step length or learning rate.

A Introduction to ODEs

A.1 Basic definitions

Let us start with some simple definitions about Ordinary Differential Equations (ODEs).

Definition A.1 (ODE)**.** An scalar ODE is an equation (generally nonlinear) involving a function $y(t)$ and its derivatives. Let $f\colon D \to \mathbb{R}$, $D \subseteq \mathbb{R}^{n+1}$ open set not empty. Then an ODE of order n is an equation of the type:

$$y^{(n)} = f(t, y, y', \dots, y^{(n-1)}).$$

Let's have a look to some of the equations we are going to deal with during this course. We will analyze the equations for a purely mathematical point of view, with no biological insights: this will come later.

i Example: Malthusian growth model

$$N'(t) = r\,N(t).$$

This is a simple model of population growth where $N(t)$ represents the population size at time t. We have that $f(t, N) = rN$. It is a *linear* ODE (f is linear with respect to N), with *constant coefficients* (r is a constant), *first-order*, and *autonomous* (f does not explictly depend on t).

i Example: Logistic Growth Model

$$N'(t) = rN\left(1 - \frac{N}{K}\right)$$

The logistic model modifies the Malthusian growth by introducing a carrying capacity. We have that $f(t, N) = rN(1 - N/K)$. It is a *nonlinear* ODE (f is non-linear with respect to N), *first-order*, and *autonomous*.

i Example: Damped Harmonic Oscillator

$$y''(t) + 2\beta y'(t) + \omega_0^2 y(t) = F(t).$$

This is a second-order ODE, linear, constant coefficients, non-autonomous (because of the presence of $F(t)$).

In general, an ODE does not tell anything about a specific solution; it is just the law of motion. We can have many solutions. As a rule of thumb, an n-order scalar ODE requires n constants for uniquely defining a solution. An **Initial Value Problem (IVP)** is an ODE supplemented wuth additional conditions. It is called "initial condition" because it sets the value of the solution at some initial time t_0. (Please note that an IVP is also called "Cauchy problem" on some books.)

Definition A.2 (Initial Value Problem)**.** An initial value problem (IVP) is a system of the form:

$$\begin{cases} y^{(n)} = f(t, y, y', \dots, y^{(n-1)}), \\ y(t_0) = y_0, \\ y'(t_0) = y_1, \\ \vdots \\ y^{(n-1)}(t_0) = y_{n-1}. \end{cases}$$

In all the above cases, the ODEs were scalar. A system of ODEs is the natural generalization to $\mathbb{R}^n$. We will only consider *first-order system of ODEs*, because their extension to higher-order ODEs is superfluous (we will see why in a moment.)

Consider the system of first-order ODEs:

$$\begin{cases} y_1' = f_1(t, y_1, y_2, \dots, y_n), \\ y_2' = f_2(t, y_1, y_2, \dots, y_n), \\ \quad \dots \\ y_n' = f_n(t, y_1, y_2, \dots, y_n). \end{cases}$$

We set $y(t) = (y_1, \dots, y_n)^\top$ and $f(t, y) = (f_1, \dots, f_n)^\top$ so the system rewrites to

$$y' = f(t, \mathbf{y}).$$

(Please note that the time derivative of the vector is the vector of the derivatives.) In compact form, the IVP reads as:

$$\begin{cases} y' = f(t, \mathbf{y}), \\ y(t_0) = y_0. \end{cases} \tag{A.1}$$

The *solution* $\phi \in \mathcal{C}^1(I, \mathbb{R}^n)$ of the system is a *curve* in $\mathbb{R}^n$ such that:

$$\phi'(t) = f(t, \phi(t)), \quad \text{for all } t \in I.$$

A.2 Separation of variables

Solving ODE analytically is not always possible. As for integration, sometimes we cannot find a solution in closed form. However, we have alternatives to rescue us:

- **Qualitative analysis**: we don't solve the ODE explicitly, we rather analyse the behaviour of the trajectories as we would do for a function study. Very often we proceed graphically (in 1-D and 2-D). It is very helpful for parametric studies.
- **Numerical integration**: we solve the IVP on a computer. We have no limits, but we always need to provide all parameters and initial values. Parametric studies are more difficult.

A combination of the above approaches in always wise. On the other hand, it is good to know how to integrate an ODE when it is possible.

A simple yet effective technique to solve IVPs is the method of *separation of variables*. It applies to IVPs of the form:

$$\begin{cases} y' = f(t)g(y), \\ y'(t_0) = y_0. \end{cases}$$

for some functions f and g. Using the Leibniz notation, we write the ODE as follows:

$$\frac{\mathrm{d}y}{\mathrm{d}t} = f(t)g(y),$$

which suggests to separate the variables t and y:

$$\frac{\mathrm{d}y}{g(y)} = f(t)\,\mathrm{d}t.$$

Here we consider y and t as independent variables. The advantage is that we can integrate both sides to get rid of differentials:

$$\int_{y_0}^{y} \frac{\mathrm{d}\tilde{y}}{g(\tilde{y})} = \int_{t_0}^{t} f(\tilde{t})\,\mathrm{d}\tilde{t}.$$

Please note that amongst all anti-derivatives, we select those satisfying the initial condition $y(t_0) = y_0$. Suppose that $F(t)$ and $G(y)$ are respectively anti-derivatives of $f(t)$ and $\frac{1}{g(y)}$,

i.e. $F'(t) = f(t)$ and $G'(y) = \frac{1}{g(y)}$. Then, according to the Fundamental Theorem of Calculus, we have:

$$G(y) - G(y_0) = F(t) - F(t_0),$$

which gives an implicit expression for $y(t)$:

$$G(y(t)) = F(t) + G(y_0) - F(t_0).$$

Please note that the expression on the right hand side only contains the variable t. When G is invertible, the expression of $y(t)$ can be made explicit.

i Example (Malthusian growth model)

We aim at solving the following IVP:

$$\begin{cases} N' = rN, \\ N(t_0) = N_0. \end{cases} \tag{A.2}$$

We apply the method of separation of variables. We have:

$$\int_{N_0}^{N} \frac{\mathrm{d}\tilde{N}}{\tilde{N}} = \int_{t_0}^{t} r \, \mathrm{d}\tilde{t}.$$

After integration (recall that $\ln'(y) = 1/y$):

$$\ln\left(\frac{N}{N_0}\right) = r(t - t_0),$$

or, by inverting the logarithm,

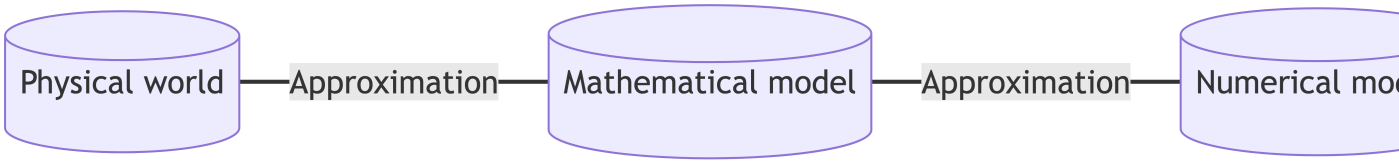
$$N(t) = N_0 e^{r(t-t_0)}.$$

! Exercise (Time-dependent Malthus equation)

Check that $N(t) = N_0 e^{r(t-t_0)}$ solves the problem (Equation A.2).

A.3 Well-posedness

Let us recall this (simplistic) view of mathematical modeling:



A good mathematical model shall approximate well the physical world. For instance, it is able to fit experimental data in a validation test. However, a validation test requires that we are able to *solve* the mathematical problem. For that, we require that the problem should satisfy **well-posedness properties**.

We are dealing with IVPs. Here, well-posedness translates into:

1. **Existence** of a solution $\phi \in \mathcal{C}^1([0,T],\mathbb{R}^n)$ for every choice of the initial data;
2. **Uniqueness** of the solution, in the sense that if $\phi_1(t)$ and $\phi_2(t)$ are both solutions of the IVP for the *same* initial data, then $\phi_1(t) = \phi_2(t)$ for all t.
3. **Stability** of the solution to perturbations, that is if $\tilde{\phi}(t)$ is the solution of the IVP with some perturbation, e.g., applied to the initial datum or to the right hand side, then $\|\tilde{\phi} - \phi\| \to 0$ as the perturbation goes to zero.

For IVPs, stability to perturbation is also called **zero stability**. The zero stability for initial data follows from the Cauchy-Lipschitz theorem, as shown below.

For the numerical realm, we have a similar definition of well-posedness. Additionally, we typically need to show that the numerical solution *converges* (in some sense) to the analytical solution. That is, the approximation error goes to zero as we refine the numerical problem. For an IVP, convergence means that that the error between the numerical solution and the true solution goes to zero as the time step goes to zero.

We start with a definition of a solution for the IVP.

Definition A.3 (Classic solution)**.** A classic solution of the IVP (Equation A.1) is a function $\phi \in \mathcal{C}^1(I;\mathbb{R}^n)$, where $I \subset \mathbb{R}$ is a closed interval, such that

1. $\phi'(t) = f(t,\phi(t))$ for all $t \in I$;
2. $\phi(t_0) = y_0$;
3. $(t,\phi(t)) \subset D$ for all $t \in I$, where $D \subset \mathbb{R}^{n+1}$ is the domain of f.

The first 2 conditions are simply the IVP (equation and initial condition). The third condition concern the domain of definition of the right hand side of the ODE.

i Example (lack of existence)

When the f is not continuous, we cannot expect existence in general. In fact, consider the IVP

$$\begin{cases} y' = \mathcal{H}(t) = \begin{cases} 0, & t \leq 0, \\ 1, & t > 0, \end{cases} \\ y(0) = y_0. \end{cases}$$

A possible and intuitive solution is

$$\phi(t) = y_0 + \max\{0, t\} = \begin{cases} y_0, & t \leq 0, \\ y_0 + t, & t > 0. \end{cases}$$

The function $\phi(t)$ is, however, not $\mathcal{C}^1(I)$ for any neighborhood I of the initial time $t_0 = 0$. Therefore, the function cannot be a solution, at least according to the above definition of solution. Note that with a different initial condition, say $y(1) = y_0$, we can find a $\mathcal{C}^1(I)$ solution, for I away from $t = 0$.

A.3.1 Local well-posedness

A general result for the local existence of a solution is due to Giuseppe Peano:

Theorem A.1 (Peano)**.** *Let* $\mathbf{f}\colon D \to \mathbb{R}^n$, $D \subseteq \mathbb{R}^{n+1}$ *open set, be continuous. Then, for all* $(t_0, \mathbf{y}_0) \in D$*, there exists a neighborhood of* t_0*, denoted by* $I_\delta = [t_0 - \delta, t_0 + \delta]$ *with* $\delta > 0$*, on which is possible to construct a solution* $\phi \in \mathcal{C}^1(I_\delta; \mathbb{R}^n)$ *to the IVP.*

Peano's theorem ensures the existence but not the uniqueness. He also provided a simple counterexample to uniqueness.

i Peano's counterexample on uniqueness

Consider the ODE

$$\begin{cases} y' = \sqrt[3]{y}, \\ y(1) = 0. \end{cases}$$

The function $f(t, y) = \sqrt[3]{y}$ is continuous with respect to t and y (note that it does not explicitly depends on time), thus Peano's Theorem ensures the existence of at least one solution.
We can find a solution by separation of variables. We have (please check!):

$$\phi_1(t) = \begin{cases} \left(\frac{2}{3}(t-1)\right)^{3/2}, & t \geq 1, \\ 0, & t < 1. \end{cases}$$

This solution is however not the only one. Also the following function is a solution (again, please check!)

$$\phi_2(t) = \begin{cases} -\left(\frac{2}{3}(t-1)\right)^{3/2}, & t \geq 1, \\ 0, & t < 1. \end{cases}$$

Yet another solution is $\phi_3(t) \equiv 0$. Actually, the IVP is so subtle that we can build an *infinite* number of solutions with the power of continuum! In fact, for any $\alpha \geq 1$, we have that

$$\phi_{\alpha^\pm}(t) = \begin{cases} \pm\left(\frac{2}{3}(t-\alpha)\right)^{3/2}, & t \geq \alpha, \\ 0, & t < \alpha, \end{cases}$$

are all solutions of the IVP. That's a lot to deal with.

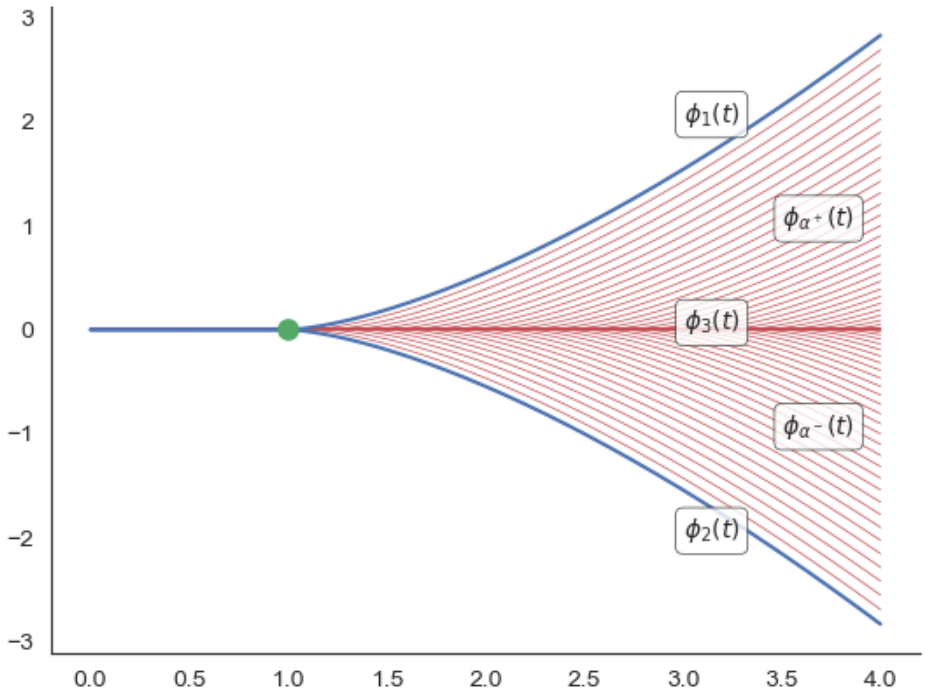


Figure A.1: Peano's brush

The example shows that the lack of uniqueness is somehow related to a lack of stability of the solution. Consider a particle located, at $t = 1$, in $y = 0$. The velocity at some generic time t is $y'(t) = \sqrt[3]{y(t)}$. Then, the trajectory of the particle is exactly the solution of Peano's example.

The example shows that we need some control on the growth rate of f. The correct concept is the Lipschitz continuity:

Definition A.4 (Lipschitz continuity)**.** Let $f\colon \Omega \to \mathbb{R}^n$, $\Omega \subseteq \mathbb{R}^n$, a continuous function. We say that f is *Lipschitz continuous* if there exists a constant $L > 0$ such that

$$\|f(y) - f(z)\| \leq L\|y - z\|,$$

for all $y, z \in \Omega$.

Definition A.5 (Local Lipschitz continuity)**.** Let $\mathbf{f}\colon \Omega \to \mathbb{R}^n$ a continuous function, with $\Omega \subseteq \mathbb{R}^n$ open domain. We say that f is *locally Lipschitz continuous* in Ω if for every $y_0 \in \Omega$ there exists a neighborhood of y_0 in which f is Lipschitz continuous, with a constant L possibly depending on the neighborhood.

> **!** Exercise
>
> Show that $\mathcal{C}^1$ functions are always locally Lipchitz. For simplicity consider functions on $\mathbb{R}$.

Theorem A.2 (Cauchy-Lipschitz)**.** *Let $f \in \mathcal{C}(D; \mathbb{R}^n)$, $D \subseteq \mathbb{R}^{n+1}$ open domain. If f is locally Lipschitz in D, with respect to y and uniformly in t, then for every $(t_0, y_0) \in D$ there exists an interval $I_\delta = [t_0 - \delta, t_0 + \delta]$, with $\delta > 0$, on which is possible to construct a solution $\phi \in \mathcal{C}^1(I_\delta; \mathbb{R}^n)$ to the IVP (Equation A.1). Such solution is also unique, in the sense that every other solution of the IVP coincides with $\phi(t)$ on I_δ.*

The Cauchy-Lipschitz theorem has very important consequences on the study of ODEs. We mention here the following

Corollary A.1 (Zero stability)**.** *Consider the following IVPs:*

$$\begin{cases} y' = f(t, \mathbf{y}), \\ y(t_0) = y_0, \end{cases} \quad \textit{and} \quad \begin{cases} \tilde{y}' = f(t, \tilde{\mathbf{y}}), \\ \tilde{y}(\tilde{t}_0) = \tilde{y}_0. \end{cases}$$

Suppose we are under the hypotheses of the Cauchy-Lipschitz theorem. Then, we have the following stability estimate:

$$\|\tilde{y}(t) - y(t)\| \leq M|\tilde{t}_0 - t_0| + e^{L|t-t_0|}\|\tilde{y}_0 - y_0\|,$$

where L is the Lipschitz constant of f.

In other words, if we perturb the initial value of the IVP, the perturbed solution remains close to the unperturbed solution as long as the perturbation is small and small time. For large time, however, the initial perturbation may grow exponentially, with a rate proportional to the Lipschitz constant.

A.3.2 Global well-posedness

The solution provided by the Cauchy-Lipschitz Theorem is only defined in a local interval, that is up to $t = t_0 + \delta$, with $\delta > 0$. Unfortunately, δ can be small, thus there is no guarantee that we can integrate the IVP for an arbitrarily long time.

We can try to circumvent the problem as follows. Let us call $\phi_1(t)$ the solution and $\delta_1 = \delta$. Now we set up a new problem of the form:

$$\begin{cases} y' = f(t, y), \\ y(t_1) = y_1, \end{cases}$$

where $t_1 = t_0 + \delta_1$ and $y_1 = \phi_1(t_1)$. Since also the above problem is well-posed, we get a new solution $\phi_2(t)$ up to $t_2 = t_1 + \delta_2$, for some $\delta_2 > 0$.

So, we can define a new solution on the interval $[t_0, t_0 + \delta_1 + \delta_2]$ by gluing the solutions around t_1:

$$\phi(t) = \begin{cases} \phi_1(t), & t \in [t_0, t_0 + \delta_1], \\ \phi_2(t), & t \in [t_0 + \delta_1, t_0 + \delta_1 + \delta_2]. \end{cases}$$

Please note that $\phi_1(t_1) = \phi_2(t_1)$ by construction, and the same applies to the derivative, so the full solution is still $\mathcal{C}^1([t_0, t_0 + \delta_1 + \delta_2])$.

By iterating the process, we obtain a sequence $\{\delta_n\}$ of extensions. The hope is that $\sum_n \delta_n \to \infty$, that is we can extend the solution indefinitely. However, it is certainly possible that $t_0 + \sum_n \delta_n \to T_{\max} < \infty$.

Definition A.6 (Maximal interval)**.** The *maximal right interval* $[t_0, T_{\max})$ is simply the maximum time $T_{\max}$ up to which we can extend the solution to the right. In general the interval is open. Similarly, the *left maximal interval* is $(T_{\min}, t_0]$. Finally, the *maximal interval* is just $(T_{\min}, T_{\max})$. We denote the maximal interval of the solution with J_ϕ.

i Example (blow-up of solutions)

Consider the problem

$$\begin{cases} y' = y^2, \\ y(0) = 1. \end{cases}$$

We solve it by separation of variables. So,

$$\begin{aligned}
&\int_1^y \frac{\mathrm{d}\tilde{y}}{\tilde{y}^2} = \int_0^t \mathrm{d}\tilde{t} \\
\Rightarrow\ &\left[-\frac{1}{\tilde{y}}\right]_1^y = t \\
\Rightarrow\ &-\frac{1}{y} + 1 = t \\
\Rightarrow\ &y(t) = \frac{1}{1-t}.
\end{aligned}$$

We know that the problem is locally well-posed, because $f(t, y) = y^2$ is continuous and locally Lipschitz. Thus, the above solution is locally unique. However, the interval of integration cannot be arbitrarily large: starting at $t = 0$, we can go forward in time only up to $t = 1 - \varepsilon$, with $\varepsilon > 0$. That is, we cannot reach $t = 1$. The reason is clear: the solution exhibit a vertical asymptote at $t = 1$, thus the solution *blows up*. In mathematics, this event is indeed called *blow-up in finite time* of the solution.
Conversely, backward integration exhibit no problem. Therefore, the maximal interval of well-posedness is $J_\phi = (-\infty, 1)$.

Computing the maximal interval for a solution by hand is clearly not practical. We would like to estimate the interval without having to solve the problem analytically.

Theorem A.3. *Let $\|\phi(t)\| \leq M$ for all $t \in (a, b)$. Then $J_\phi = (a, b)$.*

In the theorem, we can also take $a = -\infty$ and $b = +\infty$. In this case the maximal interval would be $\mathbb{R}$.

If the right hand side of the IVP grows almost linearly, then we have global existence as well. Indeed, the following theorem holds.

Theorem A.4. *Let $f\colon (a, b) \times \mathbb{R}^n \to \mathbb{R}^n$. Suppose we are under the assumptions of the Cauchy-Lipschitz theorem. If there exist 2 non-negative constants k_1, k_2 such that*

$$\|f(t, y)\| \leq k_1 + k_2\|y\|$$

for every $(t, y) \in [a, b] \times \mathbb{R}^n$, then for all (t_0, y_0) the unique solution $\phi(t)$ is defined in $[a, b]$.

In general, we use *a priori* information on the solution to apply the above Theorems. For that, we may use energy estimates (common for problems in physics), or we can study the qualitative behavior of the solutions for various initial conditions, e.g., in the phase space. Let us try that with a simple example.

Example (logistic equation)

Consider the ODE

$$y' = y(1-y),$$

where $y(t)$ may represent, for instance, the density of a population. We would like to show that $J = \mathbb{R}$ for every choice of the initial data $y(0) = y_0$. Note that the right hand side $f(t, y) = y(1-y)$ is quadratic: we cannot exclude a blow up as in the above example! We have the following cases:

- if $y_0 = 0$ or $y_0 = 1$, the solution is constant in time, $y(t) = y_0$. In fact, $f(t, y_0) = 0$, thus $y'(t) = 0$ and we have a constant solution. We are going to call this type of solutions *fixed points* or *equilibria* (see the next section.)
- if $y_0 \in (0, 1)$, then $y(t) \to 1^-$ as $t \to +\infty$ and $y(t) \to 0^+$ as $t \to -\infty$. Thus, $y(t) \in (0, 1)$ for all t. The limit follows from an analysis of the sign of $f(t, y)$. For $y \in (0, 1)$, $f(t, y) > 0$, thus $y' > 0$ for all t and $y(t)$ is monotonically increasing. But $y = 1$ is an equilibrium that cannot be crossed, and there are no other equilibria in $(0, 1)$. Thus, it must be that $y \to 1$ from below as $t \to \infty$.
- if $y_0 > 1$, then $y(t) \to 1^+$ as $t \to +\infty$. Again, the statement follows from $f(t, y) < 0$ for $y > 1$, thus $y' < 0$ and $y(t)$ is monotonically decreasing. Since it is bounded from below by $y(t) = 1$, we have $y(t) \to 1$. However, for $t \to -\infty$ we have a blow up in finite time, specifically for $t = \bar{t} < 0$ where (show this by explicit integration of the IVP!)

$$\bar{t} = \ln(1 - y_0) - \ln(y_0).$$

We conclude that the solution $y(t)$ is *bounded* in $[y_0, 0]$ for any choice of the initial condition $y_0 \geq 0$ and $t \geq 0$. (Why we exclude $y_0 < 0$?) Thus, we can apply the above theorem with $a = 0$ and $b = \infty$, concluding that $J = (0, \infty)$. For $t < 0$, the solution is bounded for $y_0 \in [0, 1]$, thus $J = \mathbb{R}$. However, for $y_0 > 1$, we have $J = (\bar{t}, \infty)$.

A.4 Linear ODEs

A general linear ODE is an equation of the form:

$$\mathbf{y}' = \mathbf{A}(t)\mathbf{y} + \mathbf{b}(t)$$

with

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \quad \mathbf{A}(t) = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ a_n(t) & a_{n-1}(t) & \cdots & a_2(t) & a_1(t) \end{pmatrix}, \quad \mathbf{b}(t) = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ b(t) \end{pmatrix}.$$

Note that $\mathbf{f}(t, \mathbf{y}) = \mathbf{A}(t)\mathbf{y} + \mathbf{b}(t)$ is only linear in $\mathbf{y}$, that is

$$\mathbf{f}(t, \alpha\mathbf{y} + \mathbf{z}) = \alpha\mathbf{f}(t, \mathbf{y}) + \mathbf{f}(t, \mathbf{z}),$$

for all choices of $\alpha \in \mathbb{R}$ and $\mathbf{y}, \mathbf{z} \in \mathbb{R}^n$.

A.4.1 Well-posedness

Theorem A.5. *Assuming that $\mathbf{A} \in \mathcal{C}^0([\alpha, \beta]; \mathbb{R}^{n\times n})$ and $\mathbf{b} \in \mathcal{C}^0([\alpha, \beta]; \mathbb{R}^n)$, that is all coefficients are continuous function of $t \in [\alpha, \beta]$, then we have a unique global solution $\phi \in \mathcal{C}^1((\alpha, \beta); \mathbb{R}^n)$ for each choice of the initial data.*

The proof of the local well-posedness follows for the Cauchy-Lipschitz theorem, because $\mathbf{f}(t, \mathbf{y})$ is locally Lipschitz uniformly in t. In fact, since $\mathbf{A}(t)$ is continous on a closed interval, it is also bounded. Thus:

$$\|\mathbf{f}(t, \mathbf{y}) - \mathbf{f}(t, \mathbf{z})\| = \|\mathbf{A}(t)(\mathbf{y} - \mathbf{z})\| \leq \|\mathbf{A}(t)\|\|\mathbf{y} - \mathbf{z}\| \leq \underbrace{\max_{t\in[\alpha,\beta]} \|\mathbf{A}(t)\|}_{L} \|\mathbf{y} - \mathbf{z}\|.$$

The global existence follows from Theorem A.4.

$$\|\mathbf{f}(t, \mathbf{y})\| = \|\mathbf{A}(t)\mathbf{y} + \mathbf{b}(t)\| \leq \|\mathbf{A}(t)\|\|\mathbf{y}\| + \|\mathbf{b}(t)\| \leq \underbrace{\max_{t\in[\alpha,\beta]} \|\mathbf{A}(t)\|}_{k_1} \|\mathbf{y}\| + \underbrace{\max_{t\in[\alpha,\beta]} \|\mathbf{b}(t)\|}_{k_2}.$$

A.4.2 Solution of the homogeneous problem

An interesting aspect of linear ODEs is that we can represent explicitly all the solutions of the ODE. Obviously, a linear IVP has a unique solution. The ODE, on the other hand, can have many solutions. We start with the *homogeneous problem*:

$$\mathbf{y}' = \mathbf{A}(t)\mathbf{y}.$$

Theorem A.6. *Let* $\mathcal{U} = \{\phi \in \mathcal{C}^1((\alpha,\beta);\mathbb{R}^n) : \phi'(t) = \mathbf{A}(t)\phi(t)\ \forall t \in (\alpha,\beta)\}$ *be the set of solutions of the linear ODE. Then* $\dim \mathcal{U} = n$.

Therefore, there exists a basis $\{\phi_1(t), \dots, \phi_n(t)\}$ of $\mathcal{U}$ such that each solution $\phi(t) \in \mathcal{U}$ reads as follows:

$$\phi(t) = \sum_{i=1}^{n} c_i \phi_i(t),$$

for some choice of $\mathbf{c} = [c_1, \dots, c_n]^T \in \mathbb{R}^n$.

Definition A.7 (Wronskian matrix)**.** We define the Wronskian matrix $\mathbf{W}(t)$ as the column-matrix of the basis of $\mathcal{U}$:

$$\mathbf{W}(t) = \left[\quad \phi_1(t) \quad | \quad \phi_2(t) \quad | \quad \cdots \quad | \quad \phi_n(t) \quad \right].$$

Note that from its definition it follows that:

$$\mathbf{W}(t)' = \mathbf{A}(t)\mathbf{W}(t).$$

The *general solution* of the linear ODE is:

$$\phi(t) = \mathbf{W}(t)\mathbf{c},$$

for some $\mathbf{c} \in \mathbb{R}^n$. For an IVP with initial condition $\mathbf{y}(t_0) = \mathbf{y}_0$, we have that:

$$\phi(t_0) = \mathbf{W}(t_0)\mathbf{c} = \mathbf{y}_0, \quad \Rightarrow \quad \mathbf{c} = \mathbf{W}^{-1}(t_0)\mathbf{y}_0,$$

thus the unique solution is:

$$\phi(t) = \mathbf{W}(t)\mathbf{c} = \mathbf{W}(t)\mathbf{W}^{-1}(t_0)\mathbf{y}_0 = \mathbf{W}(t,t_0)\mathbf{y}_0,$$

where $\mathbf{W}(t,t_0)$ is called *transition matrix*. The transition matrix bears this name because it *transfers* the initial condition $\mathbf{y}_0$ at time t_0 to the solution at time t. It is a particular case of *flow* of an ODE. We have the following very useful properties, all straightforward to prove using the definition above:

1. $\mathbf{W}(t_0,t_0) = \mathbf{I}$,
2. $\mathbf{W}(t,s)\mathbf{W}(s,t_0) = \mathbf{W}(t,t_0)$,
3. $(\mathbf{W}(t,t_0))^{-1} = \mathbf{W}(t_0,t)$.

Note that we haven't proved the invertibility of $\mathbf{W}(t_0)$. We are going to do it later for the case of $\mathbf{A}$ with constant coefficients.

> **!** Exercise
>
> Prove the properties of the transfer matrix.

A.4.3 Solution of the general problem

The general solution is useful for building a *particular solution* for the non-homogeneous problem

$$\mathbf{y}' = \mathbf{A}(t)\mathbf{y} + \mathbf{b}(t). \tag{A.3}$$

Starting from the solution $\phi(t) = \mathbf{W}(t)\mathbf{c}$, we can apply the method of variation of constants to find a solution of the form $\phi(t) = \mathbf{W}(t)\mathbf{c}(t)$, for some choice of $\mathbf{c}(t)$. It turns out that the *particular solution* is:

$$\phi(t) = \mathbf{W}(t) \int^t \mathbf{W}^{-1}(s)\mathbf{b}(s)\mathrm{d}s = \int^t \mathbf{W}(t,s)\mathbf{b}(s)\mathrm{d}s.$$

Supplemented with an initial condition $\mathbf{y}(t_0) = \mathbf{y}_0$, the solution of the IVP is

$$\phi(t) = \mathbf{W}(t,t_0)\mathbf{y}_0 + \int_{t_0}^t \mathbf{W}(t,s)\mathbf{b}(s)\mathrm{d}s. \tag{A.4}$$

> **!** Exercise
>
> Show that (Equation A.4) is the solution of the IVP (Equation A.3).

A.4.4 Matrix exponential

We focus now on the following case:

$$\begin{cases} \mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{b}(t), \\ \mathbf{y}(t_0) = \mathbf{y}_0. \end{cases}$$

We need the Wronskian matrix. For computing it, we use the fact that $\mathbf{W}' = \mathbf{A}\mathbf{W}$ and $\mathbf{W}(0) = \mathbf{I}$ (identity matrix). Formally, notice that

$$\begin{aligned} \mathbf{W}(0) &= \mathbf{I}, \\ \mathbf{W}'(0) &= \mathbf{A}\mathbf{W} = \mathbf{A}, \\ \mathbf{W}''(0) &= \mathbf{A}\mathbf{W}' = \mathbf{A}^2, \\ &\dots \\ \mathbf{W}^{(k)}(0) &= \mathbf{A}^n. \end{aligned}$$

Thus we can construct the solution as a Taylor expansion:

$$\mathbf{W}(t) = \sum_{n=0}^{\infty} \frac{\mathbf{W}^{(n)}(0)}{n!} t^k = \sum_{n=0}^{\infty} \frac{\mathbf{A}^n}{n!} t^k = \mathbf{I} + t\mathbf{A} + \frac{t^2}{2}\mathbf{A}^2 + ...$$

Definition A.8 (Matrix exponential)**.** We define the *matrix exponential* as:

$$e^{\mathbf{A}} := \sum_{n=0}^{\infty} \frac{\mathbf{A}^n(t)}{n!}.$$

Thus, the Wronskian is:

$$\mathbf{W}(t) = e^{t\mathbf{A}}.$$

The series converges for each $\mathbf{A} \in \mathbb{R}^{n\times n}$. Below some useful properties:

1. If $\mathbf{A} = \operatorname{diag}\{a_1, ..., a_n\}$, then $e^{\mathbf{A}} = \operatorname{diag}\{e^{a_1}, ..., e^{a_n}\}$. The matrix exponential of diagonal matrices is then straightforward to compute. The proof is simple: just plug the diagonal matrix in the definition and note that $\mathbf{A}^k = \operatorname{diag}\{a_1^k, ..., a_n^k\}$. Hence:

$$\sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!} = \operatorname{diag}\left\{\sum_{k=0}^{\infty} \frac{a_1^k}{k!}, ..., \sum_{k=0}^{\infty} \frac{a_n^k}{k!}\right\} = \operatorname{diag}\{e^{a_1}, ..., e^{a_n}\}.$$

2. $e^{\mathbf{0}} = \mathbf{I}$. If $\mathbf{A}$ is the matrix with all zero entries $\mathbf{0}$, then the matrix exponential is the identity matrix $\mathbf{I}$. This is trivial because all the entries of the sum are zero except for $k = 0$.

3. $\det(e^{\mathbf{A}}) = e^{\operatorname{tr}\mathbf{A}}$. This is reminescent of the Wronskian (determinant of Wronskian matrix), and the proof is similar, based on the Taylor expansion $|\mathbf{I}+\varepsilon\mathbf{A}| = 1+\varepsilon \operatorname{tr}\mathbf{A}+\mathcal{O}(\varepsilon^2)$. We use the alternative definition of matrix exponential, and the fact that the determinant is a continuous function with respect to the coefficients of the matrix:

$$\begin{aligned}
|e^{\mathbf{A}}| &= \left|\lim_{n\to\infty}\left(\mathbf{I} + \tfrac{1}{n}\mathbf{A}\right)^n\right| \\
\text{(continuity)} &= \lim_{n\to\infty}\left|\left(\mathbf{I} + \tfrac{1}{n}\mathbf{A}\right)^n\right| \\
\text{(det of product)} &= \lim_{n\to\infty}\left|\mathbf{I} + \tfrac{1}{n}\mathbf{A}\right|^n \\
\text{(det expansion)} &= \lim_{n\to\infty}\left(1 + \tfrac{1}{n}\operatorname{tr}\mathbf{A} + \mathcal{O}(\tfrac{1}{n^2})\right)^n = e^{\operatorname{tr}\mathbf{A}}.
\end{aligned}$$

4. $e^{\mathbf{A}}$ is invertible. This is a consequence of the previous property, since the determinant is always strictly positive.

5. If $\mathbf{A}$ and $\mathbf{B}$ commute, that is $\mathbf{AB} = \mathbf{BA}$, then $e^{\mathbf{A}+\mathbf{B}} = e^{\mathbf{A}}e^{\mathbf{B}}$. We are not proving this fact.

6. $e^{0\mathbf{A}} = \mathbf{I}$. This follows immediately from property 2. of the matrix exponential.

7. $e^{t\mathbf{A}}e^{s\mathbf{A}} = e^{(t+s)\mathbf{A}}$. In this case we could use property 5., but instead we use the formula for the product of two series and Newton's binomial formula:

$$\begin{aligned} e^{t\mathbf{A}}e^{s\mathbf{A}} &= \left(\sum_{k=0}^{\infty} \frac{t^k\mathbf{A}^k}{k!}\right) \cdot \left(\sum_{j=0}^{\infty} \frac{s^j\mathbf{A}^j}{j!}\right) \\ &= \sum_{k=0}^{\infty}\sum_{l=0}^{k} \frac{t^l\mathbf{A}^l}{l!}\frac{s^{k-l}\mathbf{A}^{k-l}}{(k-l)!} \\ &= \sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!}\sum_{l=0}^{k} \frac{k!}{l!(k-l)!}t^l s^{k-l} \\ &= \sum_{k=0}^{\infty} \frac{(t+s)^k\mathbf{A}^k}{k!} = e^{(t+s)\mathbf{A}}. \end{aligned}$$

8. $\frac{\mathrm{d}}{\mathrm{d}t}e^{t\mathbf{A}} = \mathbf{A}e^{t\mathbf{A}}$. It follows from the definition:

$$\frac{\mathrm{d}}{\mathrm{d}t}e^{t\mathbf{A}} = \sum_{k=0}^{\infty} \frac{\mathrm{d}}{\mathrm{d}t}\frac{t^k\mathbf{A}^k}{k!} = \sum_{k=1}^{\infty} \frac{kt^{k-1}\mathbf{A}^k}{k!} = \mathbf{A}\sum_{k=1}^{\infty} \frac{t^{k-1}\mathbf{A}^{k-1}}{(k-1)!} = \mathbf{A}e^{t\mathbf{A}}.$$

This last property in particular shows rigorously that the matrix exponential is the Wronskian matrix of the ODE $\mathbf{y}' = \mathbf{A}\mathbf{y}$, while with the other properties we define the transition matrix.

A.4.5 Solution of the case with constant coefficients

The solution of

$$\begin{cases} \mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{b}(t), \\ \mathbf{y}(t_0) = \mathbf{y}_0, \end{cases}$$

follows now trivially from the matrix exponential. We have:

$$\phi(t) = e^{(t-t_0)\mathbf{A}}\mathbf{y}_0 + \int_{t_0}^{t} e^{(t-s)\mathbf{A}}\mathbf{b}(s)\mathrm{d}s.$$

i Example (scalar case)

When $n = 1$, with $\mathbf{A} = a \in \mathbb{R}$, the matrix exponential is the usual exponential function. The solution is

$$\phi(t) = e^{a(t-t_0)}y_0 + \int_{t_0}^{t} e^{a(t-s)}b(s)\mathrm{d}s.$$

For instance, with $b = e^{\omega t}$, we have:

$$\int_{t_0}^{t} e^{a(t-s)}b(s)\mathrm{d}s = \int_{t_0}^{t} e^{a(t-s)}e^{\omega s}\mathrm{d}s = \frac{e^{at} - e^{\omega t}}{a - \omega}.$$

So the solution to the problem:

$$\begin{cases} y' = ay + e^{\omega t}, \\ y(t_0) = y_0, \end{cases}$$

is as follows:

$$\phi(t) = e^{a(t-s)} + \frac{e^{at} - e^{\omega t}}{a - \omega}.$$

A.5 Dynamical systems

Intuitively, a dynamical system is a state or a set of variables evolving over time according to some rule. The concept is very general. Markov chains, cellular automata and large language models are examples of dynamical systems. Roughly speaking, a dynamical system is composed by

1. A *state space*, the set of all admissible states of the system, and
2. A *rule* for moving from the current state to the next one.

We can associate to a system of ODEs a dynamical system. The state space (called *phase space*) is the subset of $\mathbb{R}^n$ of all possible states. The update rule is the solution of the ODE (called *flow*). For ODEs, one is not interested in studying a specific instance (or trajectory) of the system, but rather the system as a whole, for all possible initial states.

Definition A.9. A dynamical system is well-posed autonomous ODE (*autonomous* means that $\mathbf{f}$ is not an explicit function of t). Specifically, given $\mathbf{f} \in \mathcal{C}^1(\Omega; \mathbb{R}^n)$, with $\Omega \neq \emptyset$ an open set of $\mathbb{R}^n$, we suppose that for every initial condition $\mathbf{y}_0 \in \mathbb{R}^n$, the IVP

$$\begin{cases} \mathbf{y}' = \mathbf{f}(\mathbf{y}), \\ \mathbf{y}(0) = \mathbf{y}_0, \end{cases}$$

has a unique solution $\phi(t)$ defined for $t \in [0, \infty)$, and such that $\phi(t) \in \Omega$ for all $t \geq 0$. We call $\mathbf{y}' = \mathbf{f}(\mathbf{y})$ *dynamical system* on the *phase space* Ω. Succintly, the dynamical system is fully qualified by the pair $(\mathbf{f}, \Omega)$.

Definition A.10. The *flow* of an ODE is a function $\Phi \colon \mathbb{R} \times \Omega \to \Omega$ such that

$$\Phi(t, \mathbf{y}_0) = \phi(t),$$

where $\phi(t)$ is the solution of the ODE with initial condition $\mathbf{y}_0$.

For example, for a linear dynamical system $\mathbf{y}' = \mathbf{A}\mathbf{y}$ the flow is given by the transition matrix:

$$\Phi(t, \mathbf{y}_0) = \mathbf{W}(t, 0)\mathbf{y}_0 = e^{t\mathbf{A}}\mathbf{y}_0.$$

Note. It is worth to mention that also the general, non-autonomous ODEs $\mathbf{y}' = \mathbf{f}(t, \mathbf{y})$ in $\mathbb{R}^n$ can be recast to the autonomous form $\mathbf{z}' = \mathbf{g}(\mathbf{z})$ in $\mathbb{R}^{n+1}$ by setting

$$\mathbf{z}(t) = \begin{bmatrix} \tau(t) \\ \mathbf{y}(t) \end{bmatrix}, \quad \mathbf{g}(\mathbf{z}) = \begin{bmatrix} 1 \\ \mathbf{f}(\tau, \mathbf{y}) \end{bmatrix},$$

in fact, $\tau(t) = t$.

A.5.1 Orbits and trajectories

Definition A.11. For a given solution (or trajectory) $\phi(t)$, we consider the associated *orbit* $\mathcal{O}(\phi)$ defined as follows

$$\mathcal{O}(\phi) = \bigcup_{t \geq 0} \phi(t),$$

that is, the orbit is the "shadow" of a trajectory onto the phase space Ω.

In practice, the study of a dynamical system is not limited to a particular solution (defined by some initial condition), but rather it covers the entire domain. In some sense, there is a strong geometrical interpretation: the r.h.s. of the ODE determines a vectorial field in the phase space Ω, so that a given orbit is tangent at every point to such field.

Proposition A.1. *If $\phi(t)$ is a trajectory of the dynamical system for all $t \geq 0$, also $\psi(t) = \phi(t + c)$ is a solution for all $t \geq -c$, with $c \in \mathbb{R}$.*

The proposition is a simple consequence of the fact that a dynamical system is an autonomous ODE, that is it $\mathbf{f}$ is not a function of time.

Proposition A.2. *Two distinct orbits of a dynamical system either coincide or they never intersect.*

This proposition is extremely important when analyzing the phase portrait of a system, because some orbits may act as "barriers" for other orbits. Please note that the statement refers to *orbits*, not simply to the trajectories.

To prove the proposition, suppose there are two distinct orbits associated to the trajectories ϕ and ψ, and such that $\phi(t_1) = \psi(t_2)$ for some $t_1, t_2 \geq 0$. That is, the orbits intersect. Now consider the function $\tilde{\psi}(t) = \psi(t + t_2 - t_1)$. This function is still a solution, and $\tilde{\psi}(t_1) = \psi(t_2) = \phi(t_1)$. Thus, $\tilde{\psi}$ and ϕ have the same initial condition, and they must coincide for all t, that is $\tilde{\psi}(t) = \phi(t)$. In other words, $\tilde{\psi}$ and ϕ have the same orbit. But also $\tilde{\psi}$ and ψ have the same orbit, because the system is autonomous. In conclusion, if two orbits intersect at a least one point, they must concide at all points.

> **i** Example (Van der Pol)
>
> Below an example of the phase space for the equation $y'' - \mu(1 - y^2)y' + y = 0$, which models a stiff oscillating system.

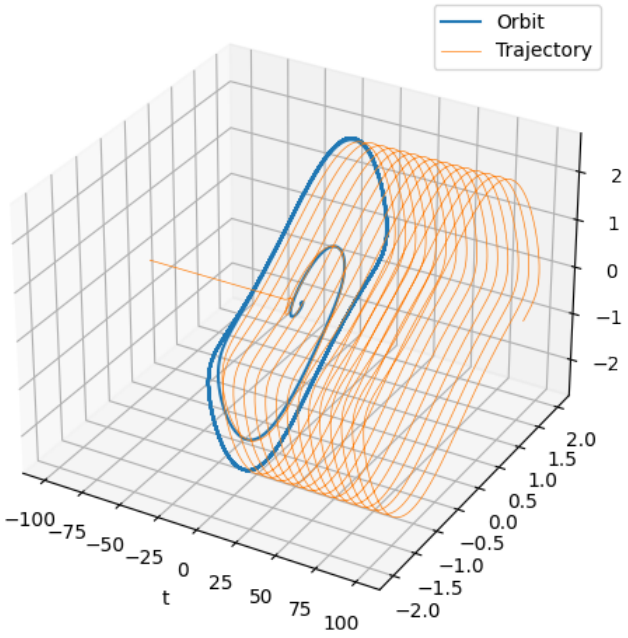


Figure A.2: Trajectory

In the figure, in blue we have the orbit associated to the trajectory in orange.

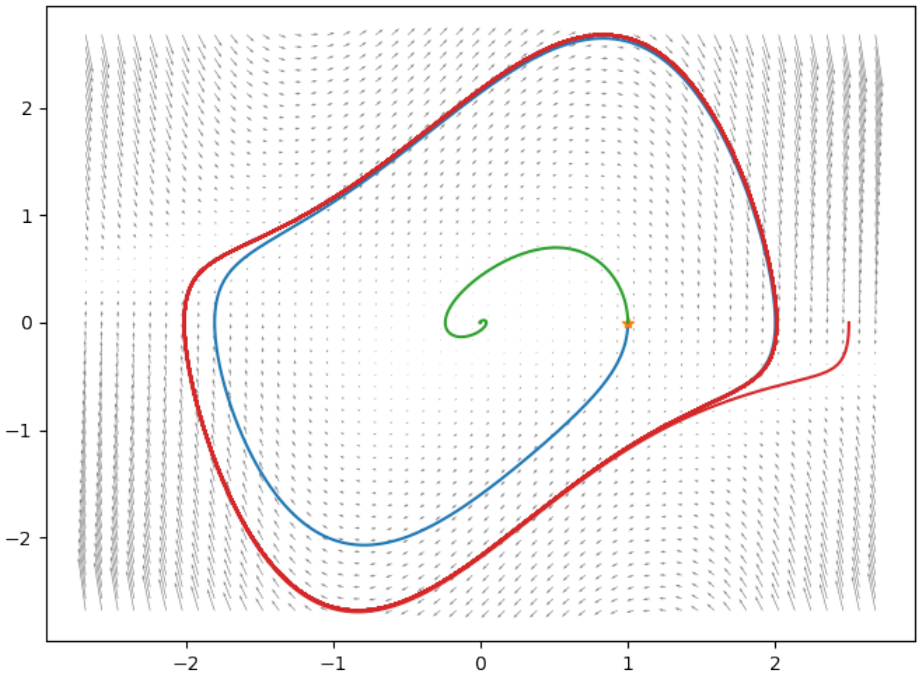


Figure A.3: Phase space

This second figure represents the phase portrait of the system, with the orbit and the limit cycle.

Example (positive systems)

An example are dynamical systems of the form:

$$\begin{cases} y_1' = y_1 \cdot f(y_1, y_2), \\ y_2' = y_2 \cdot g(y_1, y_2). \end{cases}$$

In fact, the line $(y_1, y_2) = (0, y_2)$ (the ordinate) is an orbit of the system, because if we start from $(0, y_2)$ we never leave the line, since $y_1' = 0$. Similarly, the line $(y_1, y_2) = (y_1, 0)$ (the abscissa) is also an orbit. Therefore, any other orbit starting in the positive quadrant will stay in the positive quadrant for all times. That is, the two orbits are barriers that ensures the positivity of the system for all times.

A.5.2 Types of orbits

Some orbits of a dynamical system are special. Here, we only mention the two most important ones: equilibria and periodic orbits.

Definition A.12. We say that $\mathbf{y}_0 \in \Omega$ is an *equilibrium* of the system if $\mathbf{f}(\mathbf{y}_0) = 0$. In particular, the trajectory $\phi(t) = \mathbf{y}_0$ for all $t \in \mathbb{R}$ is called solution of the equilibrium. The orbit is simply the point $\{y_0\}$.

What if an orbit self-intersect at some finite time? We have a *periodic orbit*.

Proposition A.3. *If there exist $\tau_1, \tau_2 \geq 0$ such that $\phi(\tau_1) = \phi(\tau_2)$, then $\phi(t)$ is a periodic solution.*

In fact, as above, consider the trajectory $\psi(t) = \phi(t - \tau_1 + \tau_2)$. Then $\psi(\tau_1) = \phi(\tau_2) = \phi(\tau_1)$. So we conclude that $\psi(t)$ and $\phi(t)$ coincide for all time. In particular, $\phi(t) = \phi(t + \tau_2 - \tau_1)$, that is, $\phi(t)$ is periodic with period $\tau_2 - \tau_1$.

The kind of orbits that we may find in a dynamical system is actually limited to

1. Orbits consisting of a single point (equilibria);
2. Orbits corresponding to closed regular curves (periodic solutions);
3. Orbits corresponding to open regular curves, with no self-intersections.

The dimension of the phase space plays a key role in the type of attractor. For $\Omega \subset \mathbb{R}$, we can only have equilibria. In fact:

Proposition A.4. *For $n = 1$ a dynamical system cannot have periodic solutions.*

For the proof, suppose that $\phi(t)$ is periodic with period $T > 0$. Then, we multiply by ϕ' both sides of the ODE and integrate in $[t, t + T]$:

$$\int_t^{t+T} \left(\phi'(s)\right)^2 \, \mathrm{d}s = \int_t^{t+T} f(\phi(s))\phi'(s) \, \mathrm{d}s = \int_{\phi(t)}^{\phi(t+T)} f(u) \, \mathrm{d}u = 0,$$

where we applied the change of variables $u = \phi(s)$. The last integral is zero because $\phi(t) = \phi(t + T)$. But the first integral is strictly positive, so we have a contradiction.

Note. For $n = 1$, a more general non-autonomuous ODE $y' = f(t, y)$ *can have* periodic solutions. But this is not a dynamical system, unless we recast it as a system, thus $n = 2$ and periodic solutions are possible.

Example (test problem)

The dynamical system $y' = \lambda y$ has only one equilibrium, $y^* = 0$.

Example (logistic equation)

The dynamical system $y' = y(1-y)$ has two equilibria, $y_1 = 0$ and $y_2 = 1$.

Example (linear system)

Given $\mathbf{A} \in \mathbb{R}^{n\times n}$, the dynamical system $\mathbf{y}' = \mathbf{A}\mathbf{y}$ has equilibria such that $\mathbf{A}\mathbf{y}^* = 0$, that is equilibria belong to the kernel of $\mathbf{A}$, $\mathbf{y}^* \in \ker \mathbf{A}$. If $\mathbf{A}$ is invertible, then we only have $\mathbf{y}^* = 0$.

Example (FitzHugh-Nagumo model)

We consider the dynamical system defined by the ODE

$$\begin{cases} u' = f(u,r) = u(1-u)(u-\alpha) - r, \\ r' = g(u,r) = \beta(u - \gamma r), \end{cases}$$

where $\alpha, \beta, \gamma \in \mathbb{R}$ are parameters of the model. This system is an important phenomenological model of excitability of cells and for modeling the action potential.
We define as *nullclines* associated to u' (resp. r') the implicit curve defined by the equation $f(u,r) = 0$ (resp. $g(u,r) = 0$). Equilibria are found as intersections between nullclines, because at those points all right hand sides simultaneously cancel. For the FitzHugh-Nagumo system, nullclines are:

$$\begin{cases} f(u,r) = 0, \Leftrightarrow r = u(1-u)(u-\alpha), \\ g(u,r) = 0, \Leftrightarrow r = \gamma^{-1}u. \end{cases}$$

The first nullcline is a cubic function with zeros at 0, α, and 1, and such that $r \to -\infty$ for $u \to \infty$. The second nullcline is a line through the origin and with slope γ^{-1}. One equilibrium is certainly the point $(u_1, r_1) = (0,0)$, for all choices of the parameters. The other equilibria depend on the choice of the parameters, specifically on whether the cubic function and the line intersect outside the origin. Those are such that

$$u(1-u)(u-\alpha) = \gamma^{-1}u,$$

so, besides $u = 0$, the number of zeros depends on the sign of the discriminant:

$$\Delta = (\alpha+1)^2 - 4\gamma^{-1}.$$

In conclusion, we have three equilibria for $\Delta > 0$, two equilibria for $\Delta = 0$ and only one for $\Delta < 0$.

A.5.3 Lyapunov stability and attractors

Consider a dynamical system $(\mathbf{f}, \Omega)$ with $\Omega \subseteq \mathbb{R}^n$, with an equilibrium $\mathbf{y}_0 \in \Omega$, that is $\mathbf{f}(\mathbf{y}_0) = 0$. How does the system behave in a neighborhood of the equilibrium? Say, if we start close to $\mathbf{y}_0$, do the stay close to it for long? Please note that the interest is purely qualitative: we are not interested in the specific form of the trajectory, but rather its behavior for $t \to \infty$.

Definition A.13 (Lyapunov stability)**.** We say that $\mathbf{y}_0$ is a *locally stable equilibrium* if for each $\varepsilon > 0$ there exists $\delta = \delta(\varepsilon, \mathbf{y}_0)$ such that for all $\mathbf{y}_1 \in \Omega$ with $\|\mathbf{y}_1 - \mathbf{y}_0\| < \delta$ we have that $\|\Phi(t, \mathbf{y}_1) - \mathbf{y}_0\| < \varepsilon$ for all $t \geq 0$, where $\Phi(t, \mathbf{y})$ is the flow of the system.

Moreover, we say that $\mathbf{y}_0$ is *asymptotically stable* if in the above definition we have $\|\phi(t; \mathbf{y}_1) - \mathbf{y}_0\| \to 0$ for $t \to \infty$.

The difference between simple stability and asymptotic stability is that in the former case orbits stay close to the equilibrium without necessarily approaching it for $t \to \infty$. For instance, the vertical downward position of the frictionless pendulum is only stable, because the orbits of the systems (oscillations at given amplitude) stay close to it with distance equal to the amplitude but never approach it. On the other hand, in the presence of friction, the equilibrium becomes asymptotically stable. (Thermodynamically speaking, asymptotically stable equilibria are quite boring: it often means extinction, death.)

Finally, an *unstable equilibrium* is defined by (logically) negating the above definition. That is, $\mathbf{y}_0 \in \Omega$ is an *unstable equilibrium* if there exists $\varepsilon > 0$ such that for all $\delta > 0$ there exists $\mathbf{y}_1 \in \Omega$ with the property that $\|\mathbf{y}_1 - \mathbf{y}_0\| < \delta$ but $\|\phi(t_n, \mathbf{y}_1) - \mathbf{y}_0\| \geq \varepsilon$ for a sequence $\{t_n\}_n \to \infty$ and all $n \in \mathbb{N}$. In other words, there exists at least one initial condition that brings the associated trajectory arbitrarily far from the equilibrium at some time points t_n approaching infinity. (It would be too much to ask the same for all $t \geq M$, because we could have a diverging trajectory that periodically comes very close to the equilibrium, without really approaching it. Take for instance the function $\phi(t) = t \sin^2 t + \varepsilon/2$, where for $t = t_n = n\pi$ is equal to $\varepsilon/2$, but for $t_n = n\pi/2$ is diverging: it is clearly unstable.)

An interesting concept associated with equilibria is that of *basin of attraction*.

Definition A.14 (Basin of attraction)**.** If an equilibrium is only *locally asymptotically stable*, then we have the basin of attraction defined as

$$\mathcal{B}(\mathbf{y}_0) := \Big\{ \tilde{\mathbf{y}} \in \Omega \colon \lim_{t \to +\infty} \Phi(t, \mathbf{y}_0) = \mathbf{y}_0 \Big\}.$$

An equilibrium is *globally asymptotically stable* when $\mathcal{B}(\mathbf{y}_0) = \Omega$.

Remark. Lyapunov stability is different from the concept of stability to perturbation, or zero stability (see Corollary A.1). In the latter, the aim is check whether we are able to recover the original solution (equilibrium or not) as the perturbation in the dynamical system goes

to zero. The original and perturbed solutions may diverge from each other for long time, but we can always reduce the gap between them by reducing the initial error (the perturbation): when that's not possible, the system is not (zero) stable. Lyapunov stability, on the other hand, concerns the *structural* stability properties of the system, that is we study the long-term behavior of solutions *without* controlling the initial perturbation. Actually, all initial conditions in the basin of attraction of an equilibrium yield solutions that converge to the same equilibrium, irrespective of the gap between them.

i Example (stability of the test problem)

The stability of equilibrium $y = 0$ of $y' = \lambda y$ depends clearly on λ. Suppose that $\lambda \in \mathbb{C}$. Then, the full solution is of the form

$$\phi(t) = e^{\lambda t} y_0.$$

We can expand the exponential to get more insights:

$$\phi(t) = e^{\lambda t} y_0 = e^{t \operatorname{Re}\lambda} e^{t \operatorname{Im}\lambda} y_0 = e^{t \operatorname{Re}\lambda} (\cos(t \operatorname{Im}\lambda) + i \sin(t \operatorname{Im}\lambda)) y_0.$$

Since the equilibrium is 0, we just need to check whether this trajectory, in modulus, stays close (or even approaches) zero over time. That is,

$$|\phi(t)|^2 = \phi^*(t)\phi(t) = e^{2t \operatorname{Re}\lambda} \left(\cos(t \operatorname{Im}\lambda)^2 + \sin(t \operatorname{Im}\lambda)\right)^2 |y_0|^2,$$

so we have:

$$|\phi(t)| = e^{t \operatorname{Re}\lambda} |y_0|.$$

From this expression, we easily deduce that

1. If $\operatorname{Re}\lambda < 0$, then $|\phi(t)| \to 0$ for all $y_0 \in \mathbb{C}$. The equilibrium is therefore globally asymptotically stable.
2. If $\operatorname{Re}\lambda > 0$, then $|\phi(t)| \to \infty$ for at least one non trivial $y_0 \in \mathbb{C}$. The equilibrium is therefore unstable.
3. If $\operatorname{Re}\lambda = 0$, then $|\phi(t)| = |y_0|$. The equilibrium is stable.

i Example (stability of logistic equation)

The logistic equation $y' = y(1-y)$ has two equilibria, $y_1 = 0$ and $y_2 = 1$. We could study their stability by taking advantage of the analytical solution, available in this case, but we will not. We will proceed more generally. The dynamics is determined by the sign of $f(y) = y(1-y)$.

1. If $y_0 = y_1$ or $y_0 = y_2$, there is no dynamics over time.
2. If $y_0 \in (0, 1)$ (boundaries excluded), then the solution $\phi(t; y_0)$ will never leave the

interval $(0, 1)$, because 0 and 1 are barriers (equilibria are also orbits, and orbits cannot intersect). Moreover, in this region $f(y) > 0$. Therefore $y' > 0$, that the solution $\phi(t; y_0)$ increases over time. Since the equilibrium $y_2 = 1$ cannot be crossed, $\phi(t; y_0)$ indefinitely approaches y_2 (from the left) without crossing it.
3. If $y_0 > 1$, with the same reasoning as above we conclude that $\phi(t; y_0) > 1$ indefinitely. But here $f(y) < 0$, so $y' < 0$ and again $\phi(t; y_0)$ approaches y_2 (from the right).
4. If $y_0 < 0$, $y' = f(y) < 0$ and $\phi(t; y_0) \to -\infty$.

With the above analysis, we can easily conclude that

1. $y_1 = 0$ is *unstable*, because orbits diverge from it.
2. $y_2 = 0$ is *asymptotically stable*, but only *locally*, with $\mathcal{B}(y_2) = \{y > 0\}$.

When restricting the dynamical system to $\Omega = \mathbb{R}^+$, as in the case of population dynamics, the equilibrium y_2 is almost globally attractive, except when we start from $y_1 = 0$.

Please note that the above argument is very general, as it can be straightforwardly applied to *any* 1-D dynamical system: it is enough to study the sign and the zeros of $f(y)$.

A.5.4 Stability of linear ODEs

A.5.4.1 Computation of the matrix exponential

The computation of the matrix exponential is not a trivial task, in general. In some cases, however, it is practical. When $\mathbf{A}$ is diagonal we have seen that the matrix exponential is trivially the element-wise exponential.

When $\mathbf{A}$ is diagonalizable, that is there exists a matrix $\mathbf{S}$ such that

$$\mathbf{S}^{-1}\mathbf{A}\mathbf{S} = \Lambda,$$

with Λ diagonal matrix, the exponential matrix follows immediately from the following observation:

$$(\mathbf{S}\Lambda\mathbf{S}^{-1})^k = (\mathbf{S}\Lambda\mathbf{S}^{-1})(\mathbf{S}\Lambda\mathbf{S}^{-1})\cdots(\mathbf{S}\Lambda\mathbf{S}^{-1}) = \mathbf{S}\Lambda^k\mathbf{S}^{-1}.$$

In fact, we have

$$e^{\mathbf{A}} = \sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!} = \sum_{k=0}^{\infty} \frac{(\mathbf{S}\Lambda\mathbf{S}^{-1})^k}{k!} = \mathbf{S}\sum_{k=0}^{\infty} \frac{\Lambda^k}{k!}\mathbf{S}^{-1} = \mathbf{S}e^{\Lambda}\mathbf{S}^{-1}.$$

The matrix e^{Λ} is easy to compute, using the above formula (at least when it is easy to compute eigenvalues and eigenvectors.)

In the general case, when $\mathbf{A}$ is not diagonalizable, the computation is not straightforward. It is based on the so-called Jordan canonical form. In practice, it is always possible to find a matrix $\mathbf{S}$ such that $\mathbf{S}^{-1}\mathbf{A}\mathbf{S} = \mathbf{J}$ is in the Jordan canonical form, that is $\mathbf{J}$ is a block diagonal matrix, each block with a specific structure. The matrix exponential of the Jordan canonical form is again a block diagonal matrix. The matrix exponential of each block can be computed explicitly.

The Jordan canonical form is as follows:

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_1 & & \\ & \ddots & \\ & & \mathbf{J}_r \end{pmatrix}$$

where J_i is a matrix of the form:

$$\mathbf{J}_i = \begin{pmatrix} \lambda_i & 1 & & \\ & \lambda_i & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{pmatrix}$$

The eigenvalues λ_i of the matrix $\mathbf{A}$ appears on the diagonal of the Jordan block $\mathbf{J}_i$. If the matrix is diagonalizable, then there are exactly n Jordan blocks each of dimension 1. In fact, the Jordan canonical form is diagonal. If some eigenvalues have geometric multiplicity strictly less than algebraic multiplicity, then the matrix is not diagonalizable. The Jordan blocks compensate for the difference in multiplicity. For instance, consider the matrix

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -1 & 2 \end{pmatrix}.$$

The eigenvalues are $\lambda_1 = \lambda_2 = 1$. The algebraic multiplicity is 2, but the geometric multiplicity, that is the dimension of the eigenspace $E_\lambda = \ker(\mathbf{A} - \lambda\mathbf{I})$ associated to $\lambda = 1$, is only 1. The Jordan canonical form is

$$\mathbf{J} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

The matrix exponential of the Jordan block follows from the property (for a 2×2 block):

$$\begin{pmatrix} \lambda_i & 1 \\ 0 & \lambda_i \end{pmatrix}^k = \begin{pmatrix} \lambda_i^k & k\lambda_i^{k-1} \\ 0 & \lambda_i^k \end{pmatrix}.$$

So putting everything together we have

$$e^{t\mathbf{J}} = \begin{pmatrix} e^{\lambda_i t} & te^{\lambda_i t} \\ 0 & e^{\lambda_i t} \end{pmatrix}.$$

Please note that the appearance of the term $te^{\lambda_i t}$.

For the sake of completeness, we just recall that the algebraic multiplicity is associated with the characteristic polynomial $\mathcal{P}(\lambda) = \det(\lambda\mathbf{I}-\mathbf{A})$, and corresponds to the number of times a zero of $\mathcal{P}(\lambda)$ appears. More precisely, we can always write $\mathcal{P}(\lambda) = (\lambda-\lambda_1)^{\mu_1} \cdot (\lambda-\lambda_2)^{\mu_2} \cdots (\lambda-\lambda_r)^{\mu_r}$, $\{\lambda_i\}_{i=1}^r$, $r \leq n$, are the eigenvalues, and $1 \leq \mu_i \leq n$ the algebraic multiplicity of λ_i. The geometrical multiplicity μ_i is the dimension of the eigenspace $V_i = \ker(\lambda_i\mathbf{I} - \mathbf{A}) = \{\mathbf{v} \in \mathbb{R}^n \colon \lambda_i\mathbf{v} = \mathbf{A}\mathbf{v}\}$. Since $\nu_i \leq \mu_i$, when at least one eigenvalue has $\nu_i < \mu_i$, the direct sum of all V_is does not fill the whole $\mathbb{R}^n$, and additional (generalized) eigenvectors are required. These are taken from $\mathbf{A}^k$, for some $k = 2, 3, ...$, and leads to terms of the form $t^{k-1}e^{t\lambda_i}$ in the matrix exponential.

A.5.4.2 Stability

For the linear ODEs $\mathbf{y}' = \mathbf{A}\mathbf{y}$, with $\mathbf{A} \in \mathbb{R}^{n\times n}$ invertible, we only have the equilibrium $\mathbf{y}^* = \mathbf{0}$. In the more general case $\mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{b}$, the equilibrium is $\mathbf{y}^* = \mathbf{A}^{-1}\mathbf{b}$. Note that in this case we can define $\mathbf{z}(t) = \mathbf{y}(t) - \mathbf{y}^*$ that satisfies the ODE $\mathbf{z}' = \mathbf{A}\mathbf{z}$ with equilibrium $\mathbf{z}^* = \mathbf{0}$. Thus, we can focus on the homogeneous case with no loss of generality.

We now try to characterize the stability of the equilibrium $\mathbf{y}^* = \mathbf{A}$ for $\mathbf{y}' = \mathbf{A}\mathbf{y}$.

Let's first consider the case of $\mathbf{A}$ diagonalizible. Here, there exists an invertible matrix $\mathbf{V} \in \mathbb{R}^{n\times n}$ such that $\mathbf{V}^{-1}\mathbf{A}\mathbf{V} = \Lambda$ is a diagonal matrix. The entries of this matrix are the eigenvalues of $\mathbf{A}$. The general solution of the ODE reads:

$$\phi(t) = e^{t\mathbf{A}}\mathbf{y}_0 = \mathbf{V}e^{t\Lambda}\mathbf{V}^{-1}\mathbf{y}_0.$$

We need to compute

$$\lim_{t\to\infty} \|\phi(t) - \mathbf{y}^*\|,$$

for an arbitrary initial condition $\mathbf{y}_0$.

Since the matrix exponential of a diagonal matrix is just the component-wise exponentiation, we have that the components of $\mathbf{V}^{-1}\phi$ are a linear combination of terms of the form $e^{\lambda_i t}$, being λ_i the i-th eigenvalue of $\mathbf{A}$. Thus, we have that

1. If $\operatorname{Re}\lambda_i < 0$ for all $i = 1, \dots, n$, then 0 is the only globally attractive equilibrium.
2. If there exists at least one eigenvalue such that $\operatorname{Re}\lambda_i > 0$, the equilibrium 0 is unstable.
3. If $\operatorname{Re}\lambda_i \leq 0$ for all $i = 1, \dots, n$, then 0 is stable.

For a generic $\mathbf{A} \in \mathbb{R}^{n\times n}$ (diagonalizable or not), given the set of eigenvalues λ_i, the solution is some linear combination of terms of the form:

$$e^{\lambda t}, te^{\lambda t}, \dots, t^m e^{\lambda t},$$

depending on the geometric multiplicity of λ. So we have that 1. and 2. above still applies, because the exponential is stronger than any polynomial. The non-trivial case is when $\operatorname{Re}\lambda_i = 0$ for some λ_i. If λ_i is such that we need extra terms of the form $t^j e^{\lambda_i t}$, $j \geq 1$, to complete the solution space, then the equilibrium is clearly unstable, because:

$$t^j e^{\lambda_i t} = t^j(\cos(t \operatorname{Im}\lambda_i) + i \sin(t \operatorname{Im}\lambda_i)) \to \infty$$

as $t \to \infty$. Otherwise, the equilibrium is stable (not asymptotically). The extra polynomial terms are required when the geometrical multiplicity of λ_i is strictly lower than its algebraic multiplicity. So, in the general case we replace 3. above with

3. If $\operatorname{Re}\lambda_i \leq 0$ for all $i = 1, \dots, n$, and for those with $\operatorname{Re}\lambda_i = 0$ the algebraic and geometrical multiplicity coincide, then the equilibrium is stable.

Concerning asymptotic stability, it is possible to find algebraic conditions on the coefficients of $\mathbf{A}$ such that $\operatorname{Re}\lambda_i < 0$ for all eigenvalues, *without* actually computing them.

A.5.4.3 The $n = 2$ case

The characteristic polynomial of the matrix

$$\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

is as follows

$$\mathcal{P}(\lambda) = \lambda^2 - (a+b)\lambda + (ad - bc) = \lambda^2 - \operatorname{tr}(\mathbf{A})\lambda + \det(\mathbf{A}).$$

where we introduced the *trace* $\operatorname{tr}(\mathbf{A}) = a + b$ and the *determinant* $\det(\mathbf{A}) = ad - bc$.

Proposition A.5. *The condition* $\operatorname{Re}\lambda_i < 0$ *for all* $i = 1, \dots, n$ *is equivalent to*

$$\operatorname{tr}(\mathbf{A}) < 0, \quad \textit{and} \quad \det(\mathbf{A}) > 0.$$

The proof is simple. We also know that:

$$\begin{aligned} \operatorname{tr}(\mathbf{A}) &= \lambda_1 + \lambda_2, \\ \det(\mathbf{A}) &= \lambda_1 \lambda_2, \end{aligned}$$

because the above polynomial has always 2 roots on $\mathbb{C}$, so it admits a factorization $\mathcal{P}(\lambda) = (\lambda - \lambda_1)(\lambda - \lambda_2)$ that expanded gives the equality.

Proposition A.5 is one of the most important results for the course, and we will use it very often.

The case $n = 2$ is also very common in applications, thus it is worth studying in depth the equilibria in the phase space. We suppose that the equilibrium is the origin, that is we study the equation $\mathbf{y}' = \mathbf{A}\mathbf{y}$.

i Example (real and distinct eigenvalues)

Suppose that the eigenvalues λ_1 and λ_2 of the matrix $\mathbf{A}$ are real, with eigenvectors $\mathbf{v}_1$ and $\mathbf{v}_2$. Then the solution is:

$$\mathbf{y}(t) = c_1 e^{t\lambda_1}\mathbf{v}_1 + c_2 e^{t\lambda_2}\mathbf{v}_2,$$

where $\mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \mathbf{V}^{-1}\mathbf{y}_0$ and $\mathbf{V}$ is matrix with the eigenvectors. Thus, the solution is a linear combination of the eigenvectors.
When $\mathbf{y}_0 = \alpha\mathbf{v}_1$, then $c_1 = \alpha$ and $c_2 = 0$, so we stay along the line with direction $\mathbf{v}_1$.Along this line, the solution is

$$\mathbf{y}(t) = \alpha e^{t\lambda_1}\mathbf{v}_1,$$

thus $\mathbf{y}(t)$ stays along the line and, when $\lambda_1 < 0$, approaches the equilibrium for $t \to \infty$. In this case, we call the space generated by $\mathbf{v}_1$, the *stable manifold* of the equilibrium. Viceversa, for $\lambda_1 > 0$, the trajectory diverges from the equilibrium; in this case, $\mathbf{v}_1$, is the *unstable manifold*.
The same applies to $\mathbf{v}_2$. So, when $\lambda_1 < 0$ and $\lambda_2 < 0$, the equilibrium is globally stable, with stable manifold is simply $\mathbf{R}^2$. The equilibrium is a *stable node*. When the eigenvalues are both positive, we have the opposite behavior, and the equilibrium is an *unstable node*. Finally, when they have opposite sign, the equilibrium is a *saddle*.

A.5.4.4 The $n > 2$ case: Routh-Hurwitz criteria

In general, for $n > 2$, we can find a set of algebraic conditions ensuring the asymptotic stability, *without* computing the eigenvalues.

We have the following

Theorem A.7. *Given the polynomial*

$$\mathcal{P}(\lambda) = \lambda^n + a_1\lambda^{n-1} + \cdots + a_{n-1}\lambda + a_n,$$

where the coefficients a_i are real constants, $i = 1, \ldots, n$, define the n Hurwitz matrices *using the coefficients a_i of the characteristic polynomial:*

$$\mathbf{H}_1 = (a_1), \quad \mathbf{H}_2 = \begin{pmatrix} a_1 & 1 \\ a_3 & a_2 \end{pmatrix}, \quad \mathbf{H}_3 = \begin{pmatrix} a_1 & 1 & 0 \\ a_3 & a_2 & a_1 \\ a_5 & a_4 & a_3 \end{pmatrix},$$

and in general

$$\mathbf{H}_n = \begin{pmatrix} a_1 & 1 & 0 & 0 & \cdots & 0 \\ a_3 & a_2 & a_1 & 1 & \cdots & 0 \\ a_5 & a_4 & a_3 & a_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & a_n \end{pmatrix},$$

where $a_j = 0$ if $j > n$. All the roots of the polynomial $\mathcal{P}(\lambda)$ are negative or have negative real part if and only if the determinants of all Hurwitz matrices are positive:

$$\det \mathbf{H}_j > 0, \quad j = 1, 2, \dots, n.$$

We can specialize the Theorem for low n. We have

- $n = 2$: $\det \mathbf{H}_1 = a_1 > 0$ and $\det \mathbf{H}_2 = a_1 a_2 > 0$. This is equivalent to $a_1 > 0$ and $a_2 > 0$. Note that $a_1 = -\operatorname{tr} \mathbf{A}$ and $a_2 = \det \mathbf{A}$.
- $n = 3$: as above, $a_1 > 0$, $a_1 a_2 - a_3 > 0$, and $\det \mathbf{H}_3 = a_3 \det \mathbf{H}_2 > 0$. So, this is equivalent to $a_i > 0$ *and* $a_1 a_2 - a_3 > 0$.
- $n = 4$: it is possible to show that we need $a_1 > 0$, $a_3 > 0$, $a_4 > 0$ and $a_1 a_2 a_3 > a_3^2 + a_1^2 a_4$.

We will make use of the case $n = 3$.

A.5.4.5 Other useful cases

Sometimes we are going to deal with 4×4 or larger matrices. Very often, though, they have a block structure. For instance:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & \mathbf{C} \\ \mathbf{0} & \mathbf{A}_2 \end{pmatrix},$$

where $\mathbf{A}_1$ and $\mathbf{A}_2$ can differ in dimension. Then, the eigenvalues of $\mathbf{A}$ are the union of the eigenvalues of $\mathbf{A}_1$ and $\mathbf{A}_2$. A specific case is the triangular matrix, where eigenvalues are on the diagonal.

A.5.5 Linearization method

The analysis of stability quickly becomes impractical for complex non-linearities. The study of *local* stability properties, however, can be carried out fairly easily. Suppose that $\mathbf{y}^* \in \mathbb{R}^n$ is an equilibrium of the dynamical system $(\mathbf{f}, \Omega)$, with $\mathbf{f} \in \mathcal{C}^1(\Omega)$. Then

$$\mathbf{f}(\mathbf{y}) = \mathbf{f}(\mathbf{y}^*) + D\mathbf{f}(\mathbf{y}^*)(\mathbf{y} - \mathbf{y}^*) + \dots,$$

where $D\mathbf{f}(\mathbf{y}^*)$ is the Jacobian of $\mathbf{f}$. Since $\mathbf{f}(\mathbf{y}^*) = 0$, we have the following linear approximation of the dynamical system:

$$\mathbf{y}' = D\mathbf{f}(\mathbf{y}^*)(\mathbf{y} - \mathbf{y}^*) + ...$$

We now consider $\mathbf{z}(t) = \mathbf{y}(t) - \mathbf{y}^*$, and notice that $\mathbf{z}' = \mathbf{y}'$, to obtain:

$$\mathbf{z}' = \mathbf{A}\mathbf{z},$$

where we set $\mathbf{A} = D\mathbf{f}(\mathbf{y}^*)$. The dynamical system in $\mathbf{z}$ is linear, and we know how to analyze the stability of the equilibrium $\mathbf{z} = 0$. (This corresponds to the equilibrium $\mathbf{y}^*$ in the original variables.) In fact, let $\{\lambda_i\}$ be the eigenvalues of $D\mathbf{f}(\mathbf{y}^*)$. Then:

1. If $\operatorname{Re}\lambda_i < 0$ for all $i = 1, ..., r$, then $\mathbf{y}^*$ is locally asymptotically stable.
2. If there exists $i \in \{1, ..., r\}$ such that $\operatorname{Re}\lambda_i > 0$, the equilibrium is locally unstable.

We cannot conclude anything regarding the case $\operatorname{Re}\lambda_i = 0$, because in this case higher order terms in the Taylor expansion dictates the local dynamics of the system.

> **i** Example (logistic equation, alternative analysis)
>
> We consider again the ODE $y' = y(1-y)$. Given $f(y) = y(1-y)$, we have $f'(y) = 1-2y$. At the first equilibrium, $f'(0) = 1 > 0$, so it is locally unstable. For the second one, $f'(1) = -1 < 0$, so it is locally asympotically stable.

A.6 Periodic orbits

So far, we went is more great detail in the study of equilibria of a dynamical system. Equilibria as "simple", in the sense that we can find all of them as solution of the nonlinear system $\mathbf{f}(\mathbf{y}) = \mathbf{0}$. Their local stability follows from the linearization of the ODE around the equilibrium.

Periodic orbits are more difficult to characterize, but they are extremely important in applications.

A.6.1 Hamiltonian systems

Consider a system of 2 ODEs, written in general as

$$\begin{cases} y_1' = f_1(y_1, y_2), \\ y_2' = f_2(y_1, y_2). \end{cases}$$

The phase space is planar. Let us introduce the vector field

$$\mathbf{f}(y_1, y_2) = \begin{pmatrix} f_1(y_1, y_2) \\ f_2(y_1, y_2) \end{pmatrix},$$

and suppose that given $\Omega \subset \mathbb{R}^2$ we have a dynamical system $(\mathbf{f}, \Omega)$. We introduce another vector field:

$$\mathbf{G}(y_1, y_2) = \begin{pmatrix} -f_2(y_1, y_2) \\ f_1(y_1, y_2) \\ 0 \end{pmatrix},$$

with the following associated differential form

$$\omega(y_1, y_2) = -f_2(y_1, y_2)\mathrm{d}y_1 + f_1(y_1, y_2)\mathrm{d}y_2.$$

Recall the following interpretation of a differential form: given an infinitesimal displacement $\mathrm{d}\mathbf{y}$ in $\mathbb{R}^3$, the quantity $\omega = \langle \mathbf{G}, \mathrm{d}\mathbf{y} \rangle$ is the infinitesimal work done by $\mathbf{G}$ along $\mathrm{d}\mathbf{y}$. A $\mathcal{C}^1(\Omega)$ differential form is *exact* when there exists a $\mathcal{C}^2(\Omega)$ function $H : \Omega \to \mathbb{R}$ such that $\mathrm{d}H = \omega$ in Ω. In particular,

$$\mathrm{d}H = \langle \nabla H, \mathrm{d}\mathbf{y} \rangle = \langle \mathbf{G}, \mathrm{d}\mathbf{y} \rangle = \omega,$$

so we conclude that $\nabla H = \mathbf{G}$. We call H potential or Hamiltonian function.

A simple necessary condition for exactness is that the curl of $\mathbf{G}$ cancels, since $\nabla \times \mathbf{G} = \nabla \times \nabla H = \mathbf{0}$, so

$$\nabla \times \mathbf{G}(y_1, y_2) = \begin{pmatrix} 0 \\ 0 \\ \frac{\partial f_1(y_1, y_2)}{\partial y_1} + \frac{\partial f_2(y_1, y_2)}{\partial y_2} \end{pmatrix} = \mathbf{0}.$$

The condition

$$\frac{\partial f_1(y_1, y_2)}{\partial y_1} + \frac{\partial f_2(y_1, y_2)}{\partial y_2} = 0,$$

is also sufficient for ω being exact when Ω is simply connected. In conclusion, if $\nabla \times \mathbf{G} = \mathbf{0}$ and Ω is simply connected, then there exists a Hamiltonian function $H(y_1, y_2)$ such that:

$$\begin{cases} y_1' = \dfrac{\partial H}{\partial y_2}(y_1, y_2), \\ y_2' = -\dfrac{\partial H}{\partial y_1}(y_1, y_2). \end{cases}$$

ODEs of the above form are called *Hamiltonian systems*. They are very common in mechanics.

For a Hamiltonian system, orbits are level sets of the function H. Indeed, given a trajectory $\mathbf{y}(t)$ of the ODE, we have that the Hamiltonian function along the trajectory is constant:

$$\frac{\mathrm{d}}{\mathrm{d}t} H(y_1(t), y_2(t)) = \frac{\partial H}{\partial y_1} y_1'(t) + \frac{\partial H}{\partial y_2} y_2'(t) = \frac{\partial H}{\partial y_1}\frac{\partial H}{\partial y_2} - \frac{\partial H}{\partial y_2}\frac{\partial H}{\partial y_1} = 0.$$

Thus, $H(y_1(t), y_2(t)) = \text{constant}$ for $t \geq 0$. Given $H(y_1, y_2)$, we can easily draw the phase portrait.

Example (conservative systems)

Consider the ODE

$$mx'' = F(x),$$

where m is the mass of a particle and $F(x)$ is a force. We can recast this into a system:

$$\begin{cases} y_1' = \frac{1}{m} y_2, \\ y_2' = F(y_1), \end{cases}$$

where $y_1 = x$ is the position and $y_2 = mx'$ is the momentum.
When the force is conservative, there exists a potential energy $U(x)$ such that $F = -U'$. This implies the existence of a Hamiltonian function

$$H(y_1, y_2) = \frac{1}{2m} y_2^2 + \frac{1}{m} U(y_1),$$

which is exactly the total energy, the sum of kinetic and potential energy. This quantity is conserved and the orbits are level sets of $H = H_0$, where $H_0 = H(y_1(0), y_2(0))$ is the initial energy of the particle.
Take for instance the Lennard-Jones potential below:

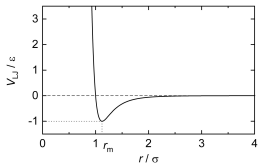


Figure A.4: Lennard-Jones potential

We observe that if the initial energy is between -1 and 0, then we have closed orbits. These corresponds to periodic solutions. When the initial energy is positive, then orbits are open.

Exercise (Double-well potential)

Find a Hamiltonian for the ODE

$$x'' = x - x^3,$$

and show how orbits vary as H_0 increases.

Example (Lotka-Volterra system)

Consider the system:

$$\begin{cases} y_1' = y_1(a - by_2), \\ y_2' = y_2(-c + dy_1). \end{cases}$$

where $y_1(t)$ and $y_2(t)$ are respectively the prey and the predator density. We will study this system more in depth. We can find the Hamiltonian in the positive quadrant as follows. Note that

$$\frac{\mathrm{d}y_1}{\mathrm{d}y_2} = \frac{y_1(a - by_2)}{y_2(-c + dy_1)},$$

so we can separate the variables:

$$\frac{(-c + dy_1)\mathrm{d}y_1}{y_1} = \frac{(a - by_2)\mathrm{d}y_2}{y_2},$$

and integrate:

$$\int_{y_1(0)}^{y_1(t)} \Big(-\frac{c}{y} + d\Big)\mathrm{d}y = \int_{y_2(0)}^{y_2(t)} \Big(\frac{a}{y_2} - b\Big)\mathrm{d}y_2,$$

we obtain that $H(y_1(t), y_2(t)) = H(y_1(0), y_2(0))$ with

$$H(y_1, y_2) = -c \ln y_1 + dy_1 - a \ln y_2 + by_2.$$

This function has a minimum at $(y_1^*, y_2^*) = (\frac{a}{b}, \frac{c}{d})$, since $\nabla H(y_1^*, y_2^*) = 0$ and the Hessian is positive definite. For y_1 or $y_2 \to 0$ we have $H \to +\infty$, as well for $y_{1,2} \to +\infty$. So, the function has level set that are closed curves, and those are associated with periodic orbits.

A.6.2 Isolated periodic orbits

We have seen that periodic orbits are possible in planar systems. In all the above cases, however, periodic orbits are packed, that is if we have an orbit for an energy level H_0, then we can have another periodic orbit infinitesimally close to the original one. Is it possible to have *isolated* periodic orbits? In other words, is it possible that there exists an open set $\Omega' \subseteq \Omega$ such that Ω' contains only *one* periodic orbit? If so, we call this orbit a *limit cycle*.

Example

The ODE

$$\begin{cases} y_1' = y_2, \\ y_2' = y_2(1 - y_1^2 + y_2^2) - y_1. \end{cases}$$

has an isolated limit cycle for $y_1(t)^2+y_2(t)^2 = 1$. To see this, first note that $(y_1(t), y_2(t)) = (\cos t, \sin t)$ is a (periodic) solution. Now we check the stability.
Take the function

$$E(y_1, y_2) = \frac{1}{2}(y_1^2 + y_2^2),$$

and note that

$$\frac{\mathrm{d}}{\mathrm{d}t}E(y_1(t), y_2(t)) = (1 - y_1(t)^2 - y_2(t)^2)y_2(t)^2.$$

Thus, if $y_1^2 + y_2^2 \leq 1$ we have $E'(t) \geq 0$, whereas if $y_1^2 + y_2^2 \geq 1$ we have $E'(t) \leq 0$. Hence, the limit cycle is attracting orbits.

A.7 Limit cycles

We have seen that periodic orbits are possible in planar systems. In all the above cases, however, periodic orbits are packed, that is if we have an orbit for an energy level H_0, then we can have another periodic orbit infinitesimally close to the original one. Is it possible to have *isolated* periodic orbits? In other words, is it possible that there exists an open set $\Omega' \subseteq \Omega$ such that Ω' contains only *one* periodic orbit? If so, we call this orbit a *limit cycle*.

A.7.1 Dulac's criterium

We have the following result of non-existence of limit cycles:

Theorem A.8 (Dulac's criterium)**.** *Let $\mathbf{y}' = \mathbf{f}(\mathbf{y})$ a planar dynamical system with $\mathbf{f} \in \mathcal{C}^1(\Omega)$, $\Omega \subset \mathbb{R}^2$ open. If there exists a function $h \in \mathcal{C}^1(\Omega)$ such that $\operatorname{div}(h\mathbf{f})$ does not change sign in $\Omega' \subseteq \Omega$ simply connected, then there exists no limit cycle in Ω'.*

Note that the theorem applies also with $h \equiv 1$. The proof is simple. Take a closed orbit entirely lying in the region Ω'. We denote by $C \subset \Omega'$ the interior of the orbit whose boundary ∂C is the orbit itself. Then

$$\int_C \operatorname{div}(h\mathbf{f})\mathrm{d}y = \int_{\partial C} h\langle \mathbf{f}, \mathbf{n}\rangle \mathrm{d}s.$$

Since ∂C is an orbit, $\langle \mathbf{f}, \mathbf{n}\rangle = 0$, so the second integral is zero, no matter of the choice of h. But the first integral cannot be zero, because $\operatorname{div}(h\mathbf{f})$ has a constant sign in C. The contraction yields to the non-existence of such a periodic orbit.

Unfortunately there is no algorithm for finding the function h. Usually is of the form $(y_1 y_2)^{-1}$, e^{y_1}, e^{y_2}, ...

Example

Consider the ODE:

$$\begin{cases} y_1' = y_1(2 - y_1 - y_2), \\ y_2' = y_2(4y_1 - y_1^2 - 3). \end{cases}$$

We can show that the system has no closed orbits in the positive quadrant. We take $h(y_1, y_2) = 1/(y_1 y_2)$. Then:

$$\nabla \cdot (h\mathbf{f}) = \frac{\partial}{\partial y_1}\left(\frac{2 - y_1 - y_2}{y_2}\right) + \frac{\partial}{\partial y_2}\left(\frac{4y_1 - y_1^2 - 3}{y_1}\right) = -\frac{1}{y_2} < 0.$$

The region R^2_+ is simply connected, and the functions are smooth, so there are no limit cycles in it.

A.7.2 Poincaré-Bendixson existence theorems

Theorem A.9 (Poincaré-Bendixson)**.** *Suppose that* $(\mathbf{f}, \Omega)$ *is a planar dynamical system,* $\Omega \subset \mathbb{R}^2$*, with* $\mathbf{f} \in \mathcal{C}^1(\Omega)$*. If there exists a closed and bounded region* $R \subseteq \Omega$ *that does not contain any equilibrium and there exists a trajectory* C *that is "confined" in* D*, in the sense that it starts in* D *and stays in* D *for all future time, then either* C *is a closed orbit, or it spirals toward a closed orbit at* $t \to \infty$*. In either case,* D *contains a closed orbit.*

In order to apply the theorem, we need to find a "trapped" orbit in D. How to proceed? The idea is the following: we can construct a region D that is closed and connected, and such that its boundary is *inflow* for the dynamical system. That is, the vector field $\mathbf{f}$ only enters into D. So, orbits cannot leave.

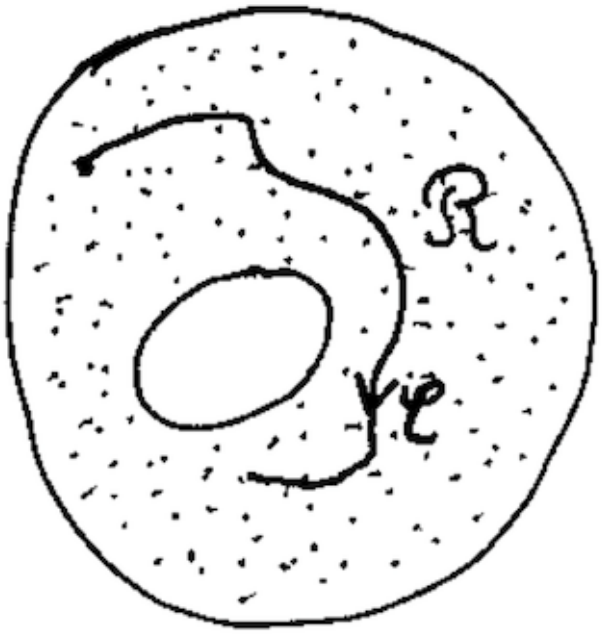


Figure A.5: Trapping region

Some comments:

- The requirement for no equilibria is important. Very often we do have an equilibrium inside (as we are going to see in a moment, a limit cycle always has an equilibrium inside of it), but we can remove it by "digging" a hole around it.
- Equilibria cannot be on the border of the region. Consider a homoclinic orbit as in the figure below. We see that the "punctured" region traps an orbit C, but there are no limit cycles inside. The orbit tends to the homoclinic.

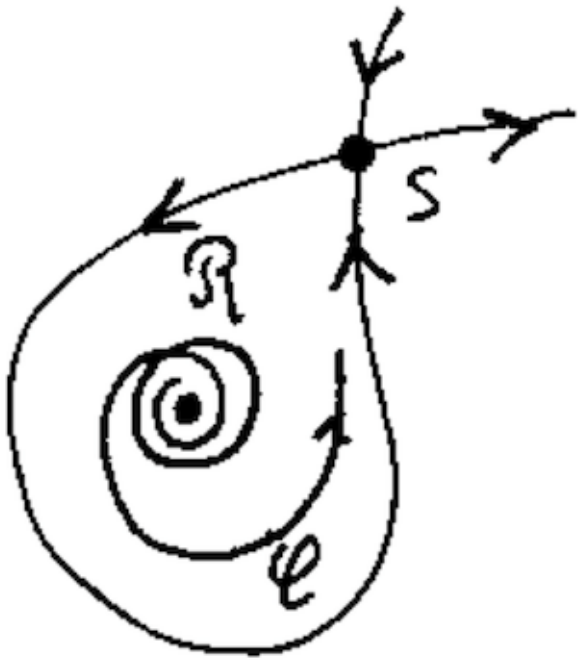


Figure A.6: Homoclinic orbit

We can be more precise. Let us define

Definition A.15 (α-limit and ω-limit sets)**.** We define the α-limit and the ω-limit of an initial condition $\mathbf{y}_0$ as the set of Ω that the trajectory approches as $t \to -\infty$ and $t \to \infty$, respectively:

$$\alpha(\mathbf{y}_0) = \lim_{t \to -\infty} \Phi(t, \mathbf{y}_0),$$
$$\omega(\mathbf{y}_0) = \lim_{t \to +\infty} \Phi(t, \mathbf{y}_0).$$

Then, we have the following

Theorem A.10 (Poincaré-Bendixson Trichotomy)**.** *Suppose that* $(\mathbf{f}, \Omega)$ *is a planar dynamical system,* $\Omega \subset \mathbb{R}^2$*, with* $\mathbf{f} \in \mathcal{C}^1(\Omega)$*. Let* $C^+(\mathbf{y}_0) = \{\Phi(t, \mathbf{y}_0), t \geq 0\}$ *a positive orbit that remains in a closed, bounded region* $R \subseteq \Omega$ *that contains only a finite number of equilibria. Then the* ω*-limit set takes one of the following three forms:*

1. $\omega(\mathbf{y}_0)$ *is an equilibrium,*
2. $\omega(\mathbf{y}_0)$ *is a periodic orbits,*

3. $\omega(\mathbf{y}_0)$ *is a* singular cycle*, that is* $\omega(\mathbf{y}_0)$ *contains a finite number of equilibria and a set of orbits whose* α*- and* ω*-limit sets consist of one these equilibria for each orbit.*

The point 3. is for instance a heteroclinic, which contains two saddle equilibria connected by the stable and unstable manifolds, or a homoclinic orbit, which is a single saddle equilibrium where the stable and unstable manifolds correspond (for $t \geq 0$).

The proofs of these theorems heavily rely on concepts from topology. Here, we only recall *index theory* for a planar dynamical system.

i Example (glycolysis)

We consider a model for glycolysis as proposed by Sel'kov (1968). In yeast cells glycolysis can proceed in a oscillatory fashion. Here is a model:

$$\begin{cases} x' = -x + ay + x^2 y, \\ y' = b - ay - x^2 y, \end{cases}$$

where $x(t)$ is the concentration of ADP (adenosine diphosphate) and $y(t)$ is the concentration of F6P (fructose-6-phosphate). The parameters $a > 0$ and $b > 0$ comes from the kinetic of the reaction.
The phase portrait is as follows:

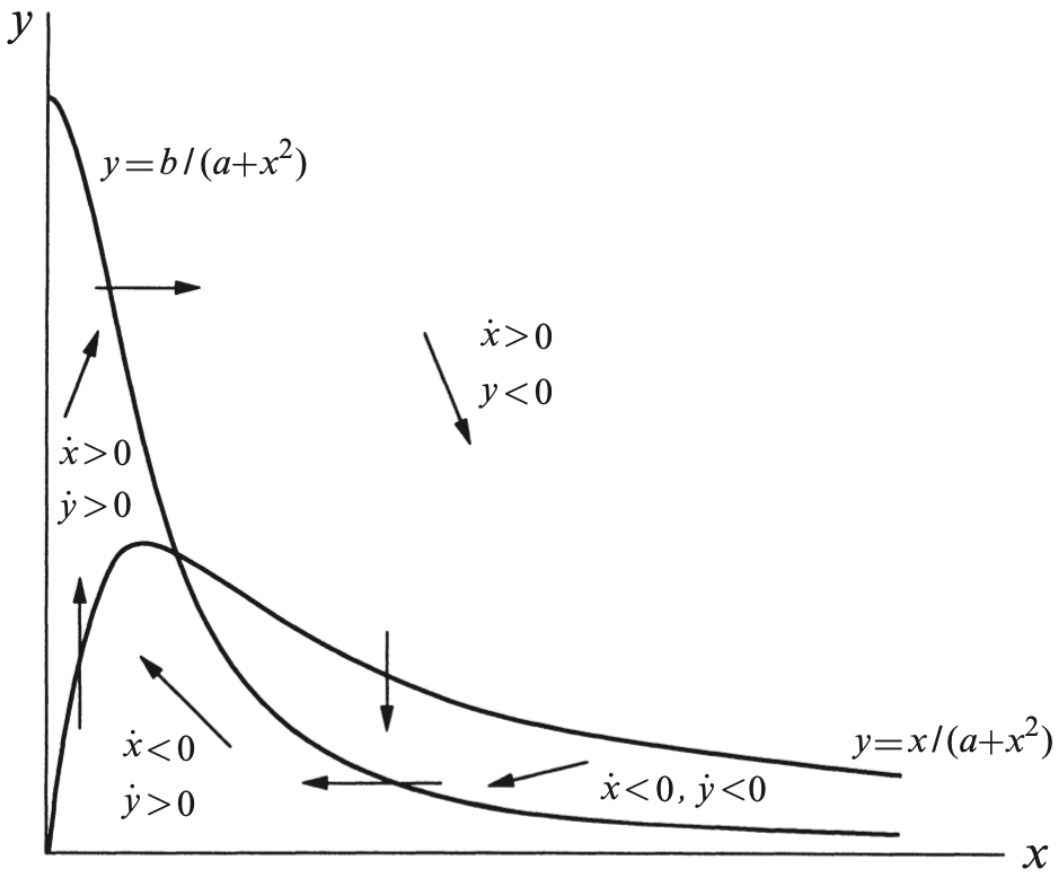


Figure A.7: Trapping region

By looking at the arrows, it looks like we could have a limit cycle or some stable equilibrium. We construct the trapping region as follows:

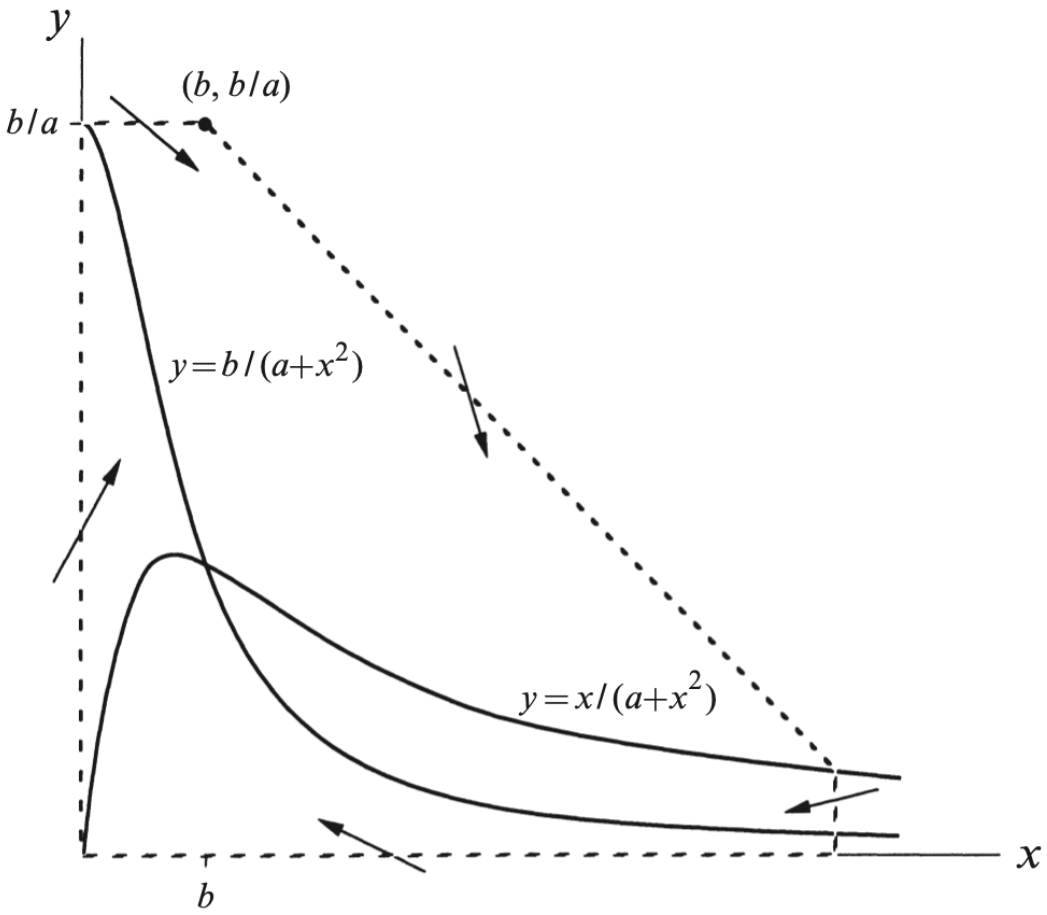


Figure A.8: Trapping region

Now we check. Graphically, we note that

- Above the nullcline $x' = 0$ we have $x' > 0$, below $x' < 0$.
- On the left of the nullcline $y' = 0$ we have $y' > 0$, on the right $y' < 0$.

Thus, the horizontal and vertical parts in of the region are easy to check, just by looking at the arrows. The diagonal part is more subtle. Intuitively, when x and y are large, the ODE is approximately $x' \approx x^2 y$ and $y' \approx -x^2 y$. So, $y'/x' \approx -1$, which is tangent to the trajectories. This suggests to take:

$$x' - (-y') = -x + ay + x^2 y + (b - ay - x^2 y) = b - x.$$

Hence, for $x > b$ we have $-y' > x'$, and trajectories are entering into the diagonal part, because their slope is more negative than -1.

We conclude that R is a trapping region. However, we cannot apply the Theorem, because we still need to rule out equilibria. We have only one, at the intersection between nullclines. We can do so just by digging a hole around it, obtaining a "punctured" region.

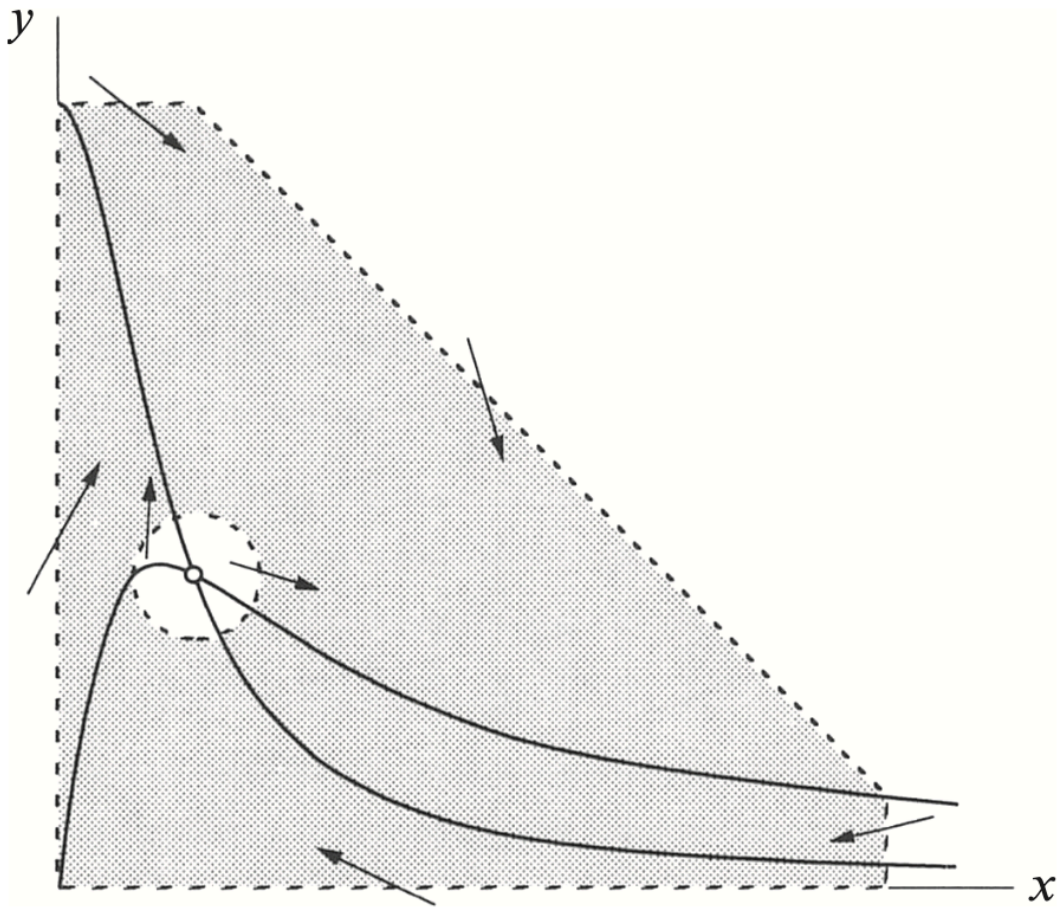


Figure A.9: Punctured region

For a sufficiently small hole, we can ensure that orbits are leaving or entering the region just by looking at the stability of the equilibrium. If it is unstable, then orbits are diverging, and thus entering into R. That is, R traps orbits and it does not contain equilibria. Therefore, there must exist a closed orbit in it!
The stability of the equilibrium is easy to check. First, the equilibrium is

$$(x^*, y^*) = \left(b, \frac{b}{a + b^2}\right).$$

The Jacobian is:

$$J_{\mathbf{f}}(x,y) = \begin{pmatrix} -1+2xy & a+x^2 \\ -2xy & -a-x^2 \end{pmatrix}.$$

The trace and the determinant at $(x,y) = (x^*, y^*)$:

$$\det J = a + b^2 > 0, \quad \operatorname{tr} J = -\frac{b^4 + (2a-1)b^2 + (a+a^2)}{a+b^2}.$$

Depending on a and b, the trace can be either positive or negative. Solving the numerator for b^2 we have that the trace is zero for

$$b^2 = \frac{1}{2}\Big(1 - 2a \pm \sqrt{1-8a}\Big).$$

For choice of the parameters such that the trace is positive, the equilibrium is unstable and the region contains a closed orbit.

A.7.3 Index theory

Definition A.16 (index of a closed curve)**.** We define the *index* of a closed curve C, denoted by i_C, as the net number of counterclockwise revolutions made by the vector field $\mathbf{f}$ around C. More precisely, take the angle $\phi(t)$ between the tangent to the curve C and the vector field $\mathbf{f}$. Then, $\phi(t)$ has changed by an integer multiple of 2π. Such integer is the index.

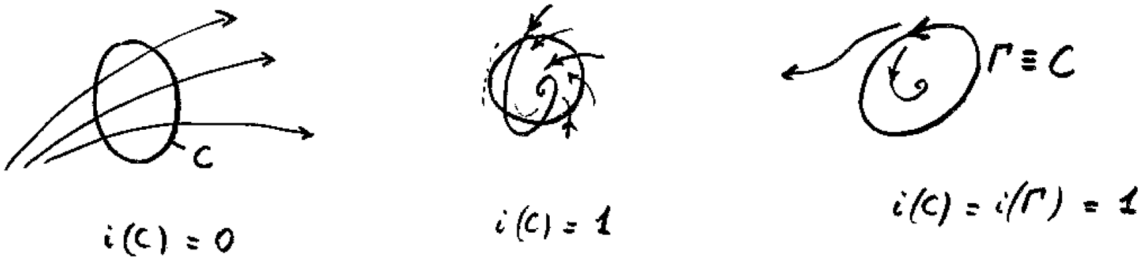


Figure A.10: Index of a curve

The index closely resembles residue theory in complex analysis. For instance, we have that

1. If C can be continuously deformed into C' without passing through an equilibrium, then $i_C = i_{C'}$. (Note that i_C is an integer-valued function of C. The only way i_C can be a continuous function of C is being constant!)

2. If C does not enclose any equilibrium, then $i_C = 0$. (Since C does not contain any zero of $\mathbf{f}$, we can shrink C to a tiny circle where $\mathbf{f}$ is almost constant, so the index is zero.)
3. If we reverse the arrows of the vector field, that is $t \mapsto -t$, the index is unchanged.
4. Suppose that C is a closed orbit. Then $i_C = +1$.

Mathematically, given the vector field $\mathbf{f}(x, y) = (f(x, t), g(x, y))$ in $\mathbb{R}^2$, the angle given by

$$\phi = \tan^{-1}(g/f),$$

so in terms of differentials we have:

$$\mathrm{d}\phi = \mathrm{d}(\tan^{-1}(g/f)) = \frac{f\mathrm{d}g - g\mathrm{d}f}{f^2 + g^2}.$$

The index is, therefore,

$$i_C = \frac{1}{2\pi}\int_C \mathrm{d}\phi = \frac{1}{2\pi}\int_C \frac{f\mathrm{d}g - g\mathrm{d}f}{f^2 + g^2}.$$

Definition A.17 (Index of an equilibrium)**.** The index of an equilibrium $\mathbf{y}^*$, denoted by $i_{\mathbf{y}^*}$ is the index of a curve containing the equilibrium.

Note that thanks to 1. the definition of $i_{\mathbf{y}^*}$ does not depend on the choice of C.

Theorem A.11. *All hyperbolic equilibria, that is the Jacobian of* $\mathbf{f}$ *is non-singular at the equilibrium, have* $i_{\mathbf{y}^*} = \pm 1$*. In particular, only the saddle has* $i_{\mathbf{y}^*} = -1$*.*

A non-hyperbolic equilibrium can also have index 0, for instance a saddle-node.

Theorem A.12. *If a closed curve* C *surrounds* n *isolated equilibria* $\mathbf{y}_1^*, \mathbf{y}_2^*, \dots, \mathbf{y}_n^*$*, then*

$$i_C = \sum_{k=1}^{n} i_{\mathbf{y}_n^*}.$$

The proof is familiar from multivariate calculus. We shrink the curve C so to surround the equilibria with tiny circles and corridors connecting them. The index does not change. Then we show that the contribution of corridors cancels out, because of symmetry reasons. So, we are only left with circles around the equilibria, which is the sum of the indices of each equilibrium.

Theorem A.13 (Poincaré)**.** *Any closed periodic orbit in the phase plane must enclose equilibria whose indices sum to* $+1$*. If all equilibria are hyperbolic, then the limit cycle contains exactly* $2s + 1$ *equilibria,* s *saddles and* $s + 1$ *non saddles, with* $s \geq 0$ *integer.*

Example

Consider the system:

$$\begin{cases} y_1' = -y_1 - y_1^2 y_2^2, \\ y_2' = y_1 + y_2. \end{cases}$$

We show now that there exists no limit cycle around the origin.
First, notice that the divergence of $\mathbf{f}$ is:

$$\operatorname{div} \mathbf{f} = -2y_1 y_2^2,$$

which changes sign in $\mathbb{R}^2$, so we cannot exclude the existence of a limit cycle.
We have 2 equilibria, $\mathbf{y}_1^* = (0,0)$ and $\mathbf{y}_2^* = (-1,1)$. In particular, $\mathbf{y}_1^*$ is a saddle. A limit cycle around the origin must contain the saddle, but the index of the cycle is $+1$, whereas the index of the saddle is -1. It cannot be. Even the case of a limit cycle surrounding both equilibria is not possible, because the sum of the indices is 0. We conclude that we cnnot have a limit cycle around the origin.

A.7.4 Poincaré maps

We still need to discuss the *stability* of limit cycles. Since they are isolated by definition, they can attract or repulse nearby orbits.

Consider a periodic orbit $\Gamma \subset \mathbb{R}^n$, and take any point $\mathbf{y}_0 \in \Gamma$. We introduce a cross-section Σ to the cycle at this point as a smooth hypersurface of dimension $n-1$ that crosses Γ at non-zero angle. A non-zero angle means that the normal of the hypersurface at $\mathbf{y}_0$ is not orthogonal to the tangent of Γ at $\mathbf{y}_0$.

Note that the limit cycle reduces to a point on the cross-section Σ. Other closeby orbits will also intersect the cross-section Σ at some angle (by continuity the angle remains non-zero in a neighborhood of $\mathbf{y}_0$). A point $\mathbf{z} \in \Sigma$, close to $\mathbf{y}_0$, will give some orbit through the flow of the ODE, and eventually return to Σ after some time at a point $\tilde{\mathbf{z}}$. This procedure is encoded in a map $\mathbf{z} \mapsto \tilde{\mathbf{z}} = \mathcal{P}(\mathbf{z})$.

Definition (Poincaré map). We call $\mathcal{P} : \Sigma \to \Sigma$ the Poincaré map of the periodic orbit Γ.

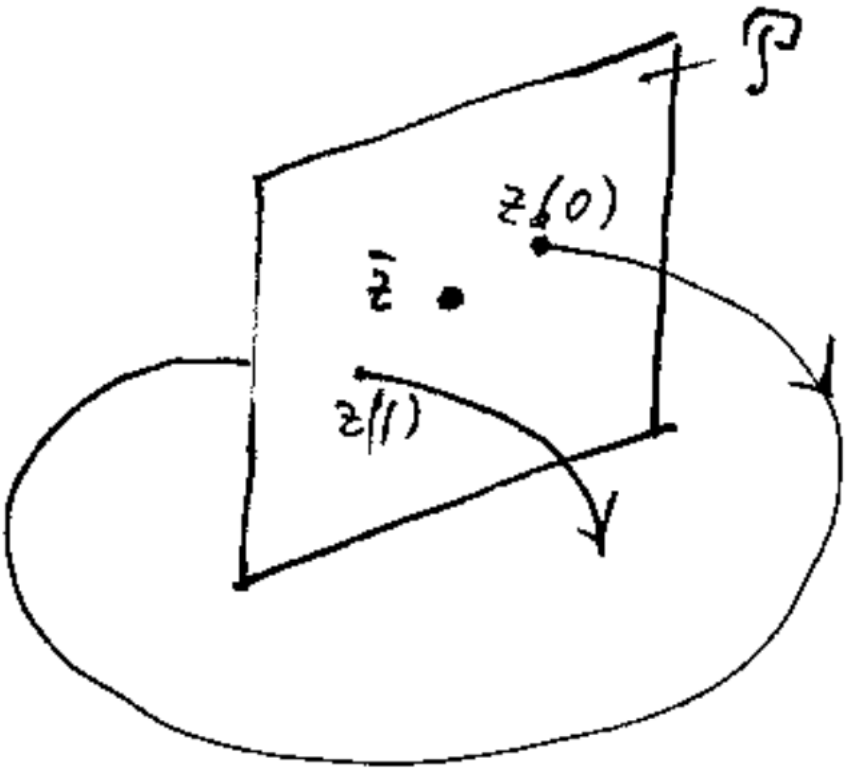


Figure A.11: Poincaré maps

A fixed point $\mathbf{z}^*$ of the map $\mathcal{P}$ corresponds to a limit cycle for the original system. Specifically, if $\mathbf{z}^*$ is a stable equilibrium for the *discrete map*

$$\mathbf{z}^{(k+1)} = \mathcal{P}(\mathbf{z}^{(k)}),$$

that is, $\mathbf{z}^* = \mathcal{P}(\mathbf{z}^*)$, then the limit cycle is stable. If $\mathbf{z}^*$ is asympotically stable, then the limit cycle is asymptotically stable. Otherwise, it is unstable.

Example

Consider the following ODE is polar coordinates:

$$\begin{cases} r' = r(1 - r^2), \\ \theta' = 1. \end{cases}$$

Since $\theta' = 1$, we make a turn after a period of 2π. Thus, if we start from r_0, we return

at r_1 after $\Delta t = 2\pi$. Hence:

$$\int_{r_0}^{r_1} \frac{\mathrm{d}r}{r(1-r^2)} = \int_0^{2\pi} \mathrm{d}t = 2\pi.$$

Integrating we get:

$$\mathcal{P}(r) = \frac{1}{\sqrt{1 + e^{-4\pi}(r^{-2} - 1)}}.$$

For $r^* = 1$ we have a fixed point. We can show (graphically with the cobweb plot) that the equilibrium is attractive.

Given a parametrization of $\mathcal{P}$ in some local coordinate system $\xi = (\xi_1, \dots, \xi_{n-1})$ such that $\xi = \mathbf{0}$ corresponds to $\mathbf{z}^*$, the origin is a fixed point of the map: $\mathcal{P}(\mathbf{0}) = \mathbf{0}$. Thanks to the contraction Theorem, the fixed-point is asymptotically stable if $\mathcal{P}$ is a contraction, which means that

$$\mathbf{B} = \left.\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}\xi}\right|_{\xi=\mathbf{0}}$$

has eigenvalues (multipliers) $\mu_1, \dots, \mu_{n-1}$ with modulus strictly less than 1, that is $|\mu_i| < 1$. It is possible to show that the multipliers are independent on the choice of the cross-section, the local parametrization, and the point $\mathbf{y}_0 \in \Gamma$. In other words, the multipliers are a characteristic of the limit cycle, from which we can deduce the stability.

We can now formalize the stability for a limit cycle. Consider a periodic solution $\phi(t)$ of the dynamical system

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}),$$

that is $\phi(t)$ is a solution and $\phi(t + T_0) = \phi(t)$ for some $T_0 > 0$, called the *period* of the limit cycle. Now we perturb the periodic orbit:

$$\tilde{\phi}(t) = \phi(t) + \mathbf{u}(t),$$

for some perturbation $\mathbf{u}(t)$. Now notice that:

$$\mathbf{u}'(t) = \tilde{\phi}(t) - \phi(t) = \mathbf{f}(\phi(t) + \mathbf{u}(t)) - \mathbf{f}(\phi(t)) = \mathbf{A}(t)\mathbf{u}(t) + \mathcal{O}(\|\mathbf{u}(t)\|^2),$$

where $\mathbf{A} = J_{\mathbf{f}}(\phi(t))$, and $\mathbf{A}(t + T_0) = \mathbf{A}(t)$. The system $\mathbf{u}' = \mathbf{A}(t)\mathbf{u}$ is non-autonomous and linear. The set of solutions form the Wronskian matrix $\mathbf{W}(t)$ and are such that

$$\mathbf{W}' = \mathbf{A}(t)\mathbf{W},$$

with initial condition $\mathbf{W}(0) = \mathbf{I}$. Any solution of $\mathbf{u}' = \mathbf{A}(t)\mathbf{u}$ is of the form $\mathbf{u}(t) = \mathbf{W}(t)\mathbf{u}(0)$. In particular,

$$\mathbf{u}(T_0) = \mathbf{W}(T_0)\mathbf{u}(0).$$

We call $\mathbf{W}(T_0)$ *monodromy matrix* for the limit cycle Γ.

Theorem A.14. *The monodromy matrix has eigenvalues* $1, \mu_1, \ldots, \mu_{n-1}$*, where* μ_i *are the multipliers of the Poincaré map associated to the limit cycle* Γ*.*

Thus, the size of the perturbation $\mathbf{u}(t)$*, measured with* $\|\mathbf{u}(t)\|$*, will tend to zero if* $|\mu_i| < 1$*. The unitary eigenvalue corresponds to the tangent direction to the limit cycle.*

B Introduction to bifurcations

B.1 Background

As we have seen, dynamical system often depend on parameters. For instance, the spruce budworm model

$$u' = \rho\Big(1 - \frac{u}{q}\Big) - \frac{u^2}{1+u^2},$$

depends on two parameters, $\rho > 0$ and $q > 0$. The number of equilibria and their stability will depend on the parameters. However, most of the time a change of a parameter has little effect on the phase portrait: equilibria barely move, their stability is unchanged. In this case we say that the system is *topologically stable*. Otherwise, like for $\rho = \rho_1$ or $\rho = \rho_2$, we have a *bifurcation*.

Another example we have seen is the Gause-type prey-predator model:

$$\begin{cases} u' = \rho u(1-u) - \dfrac{\alpha\delta uv}{1+\delta u}, \\ v' = -v + \dfrac{\alpha\delta uv}{1+\delta u}, \end{cases}$$

where we have 3 parameters. For $\delta = \frac{1}{\alpha-1}$ or $\delta = \frac{\alpha+1}{\alpha-1}$ we also have bifurcations.

In general, a parametric dynamical system is an equation of the form

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}, \mathbf{p}),$$

where $\mathbf{p} \in \mathbb{R}^m$ is the set of parameters. For the spruce budworm equation, $\mathbf{p} = (\rho, q)$. Here, we are only focusing on the case $m = 1$, when $\mathbf{p}$ is a scalar. Bifurcations occuring in this case are of co-dimension 1.

B.2 Structural stability

Consider the one-parameter ODE

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}, p),$$

and suppose that the problem is well-posed for $p \in \mathcal{P} \subseteq \mathbb{R}$ and $\mathbf{f}$.

Given $p = \bar{p}$, the system will have some phase portrait. Is it stable to small perturbations of the parameter p? In other words, if we consider a new system:

$$\tilde{\mathbf{y}}' = \mathbf{f}(\tilde{\mathbf{y}}, p),$$

with $p \in (\bar{p} - \varepsilon, \bar{p} + \varepsilon)$, $\varepsilon \ll 1$, is the "error" between $\tilde{\mathbf{y}}(t)$ and $\mathbf{y}(t)$ going to zero as $\varepsilon \to 0$ for any choice of the initial condition? That is, is the phase portrait of the perturbed system "converging" to the phase portrait of the original ODE? This is typically the case.

Theorem B.1 (stability to perturbations)**.** *Suppose that $\mathbf{f}(\mathbf{y}, p)$ is continuous and locally Lipschitz in $\mathbf{y}$, uniformly in p, and that the ODE is well-posed for any choice of p in the interval. Then the solution of the ODE $\phi(t, \mathbf{y}_0, p) \to \phi(t, \mathbf{y}_0, \bar{p})$ uniformly as $p \to \bar{p}$.*

This theorem gives a zero-stability result, in the limit $p \to \bar{p}$. How about the case of $p \in (\bar{p} - \varepsilon, \bar{p} + \varepsilon)$ but $\varepsilon > 0$? Is the perturbed ODE still "close" to the original one? We have the following definition

Definition B.1 (structural stability)**.** The original ODE is *structurally stable* in some (closed) region of the phase space if and only if there exists $\varepsilon > 0$ such that the perturbed system is *topologically equivalent* to the original ODE for all $p \in (\bar{p} - \varepsilon, \bar{p} + \varepsilon)$.

Definition B.2 (topological equivalence)**.** Two ODEs are *topologically equivalent* if there exists a diffeomorphism that smoothly maps the phase portrait of one system into the other, preserving the direction of time.

For instance, the two phase portraits below are topologically equivalent, because we can smoothly deform one into the other. (Put a fork at the center, then twist like you would do with spaghetti.)

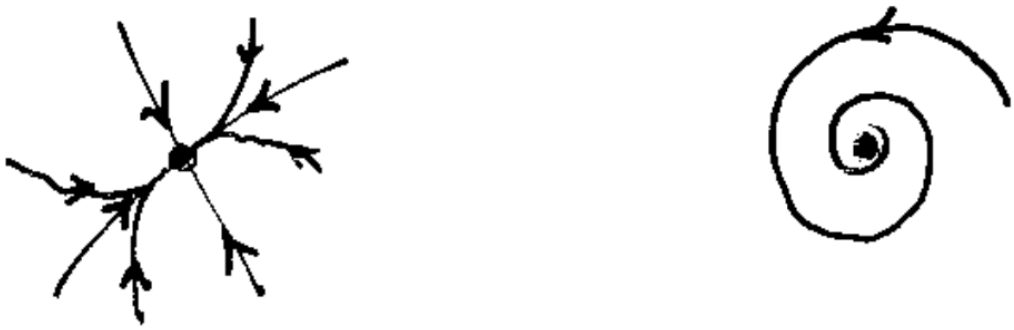


Figure B.1: Topological equivalence

B.3 Bifurcations

On the other hand, these below are *not* topologically equivalent, because if we shrink the limit cycle down to a point, we collide with another equilibrium at the center, making the map non-invertible.

Figure B.2: Phase portraits that are not topologically equivalent

When we are a situation like the one above, we have a *bifurcation*.

Definition B.3 (bifurcation)**.** If for some value $p = \bar{p}$ a system is *not* structurally stable, then for $p = \bar{p}$ we have a *bifurcation*.

Bifurcations result from the *collision* of invariant sets of the dynamical system, for instance 2 equilibria or 2 limit cycles colliding one into the other, for some value of the parameter. We can visualize this in a *bifurcation plot*, like the one below:

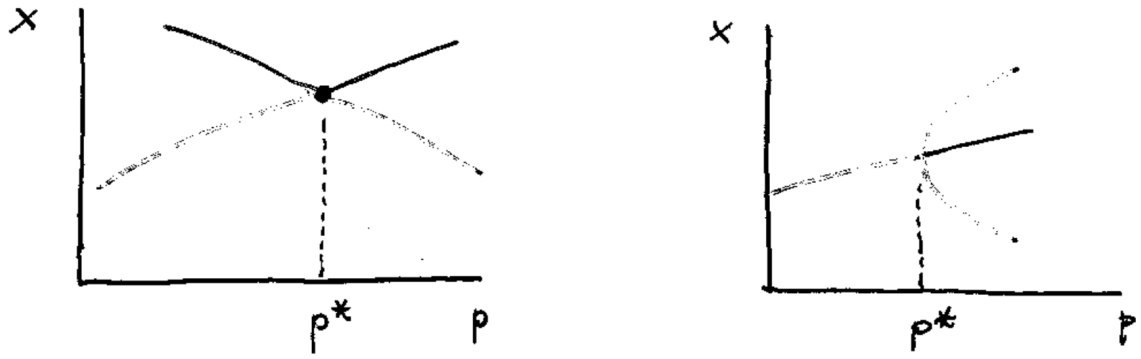


Figure B.3: bifurcation in 1D

where we have on the abscissa the parameter, and on the ordinate the phase space. The curves are equilibria, and for $\bar{p}$ we have a bifurcation.

The bifurcation plot is just a "stack" of phase portraits for various values of p. In some sense, a smooth transition between one layer to the other implies topological equivalence, otherwise we have bifurcations. For instance, for planar system we could have:

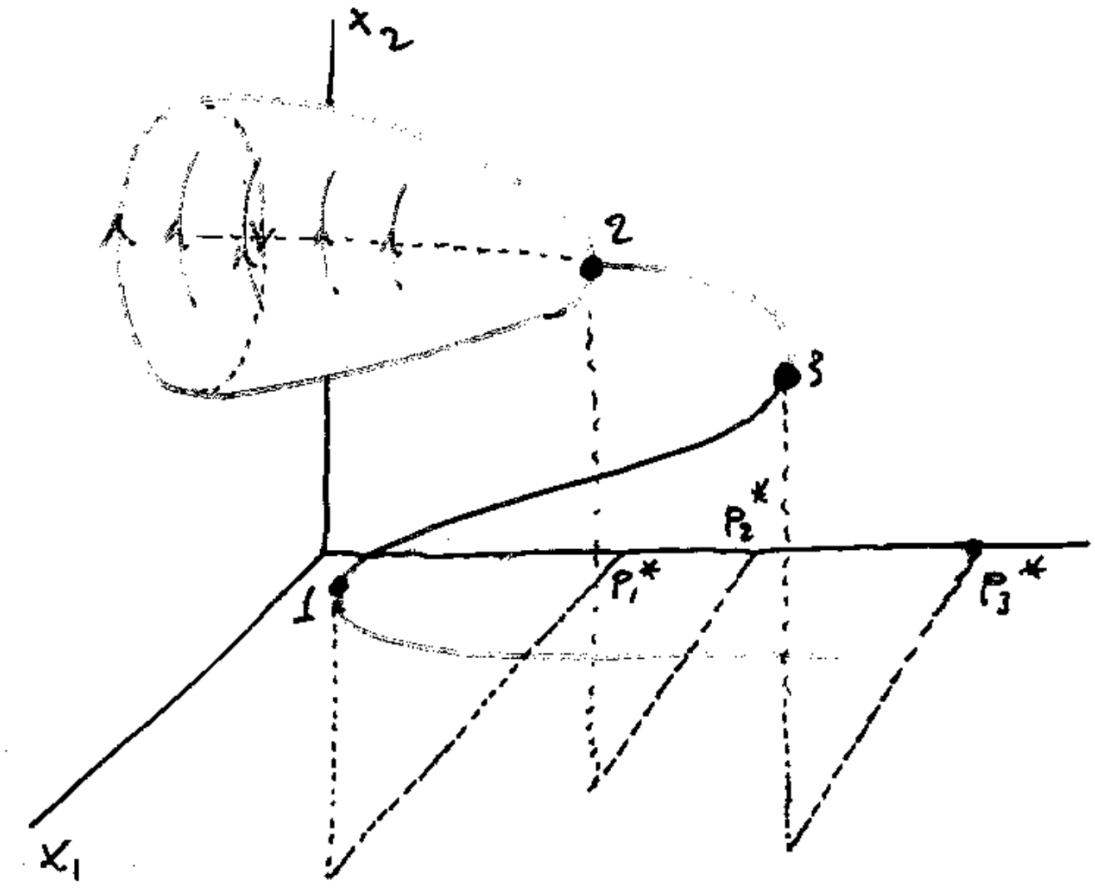


Figure B.4: bifurcation in 2D

The "cone" is formed by limit cycles. Here, we have 3 bifurcations.

B.4 Continuation of equilibria

Consider the dynamical system in 1D:

$$y' = f(y, p),$$

with $p \in \mathbb{R}$. An equilibrium $\bar{y}$ of the ODE for a given parameter $\bar{p}$ is a solution to the equation:

$$0 = f(\bar{y}, \bar{p}).$$

Suppose that f is a smooth function in (y, p). Thanks to the implicit function theorem, if $\partial_p f(\bar{y}, \bar{p}) \neq 0$ we can locally define a represent the curve $f(\bar{y}, \bar{p}) = 0$ in the space $(y, p) \in \mathbb{R}^2$ with a function $y = \phi(p)$ such that $f(\phi(p), p) = 0$ for $p \in \mathcal{B}_\varepsilon(\bar{p})$, some neighborhood of $\bar{p}$.

The curve $(\phi(p), p)$ represents the *continuation* of the equilibrium $\bar{y}$ as we change the parameter $p = \bar{p}$. All points along $(\phi(p), p)$ are equilibria, by construction, for different values of the parameter p.

We can extend the same construction in several dimensions, say $\mathbf{y} \in \mathbb{R}^n$ and $p \in \mathbb{R}$. Let us by $\mathbf{f}_\mathbf{y}(\mathbf{y}, p)$ the Jacobian of $\mathbf{f}$ with respect to $\mathbf{y}$. Suppose that $\bar{\mathbf{y}}$ is an equilibrium for the ODE

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}, \bar{p}),$$

that is, $\mathbf{f}(\bar{\mathbf{y}}, \bar{p}) = \mathbf{0}$, and that

$$\det \mathbf{f}_\mathbf{y}(\bar{\mathbf{y}}, \bar{p}) \neq 0,$$

then there exists a function $\mathbf{y} = \phi(p)$ such that $\mathbf{f}(\phi(p), p) = \mathbf{0}$ for $p \in \mathcal{B}_\varepsilon(\bar{p})$.

The *stability* of the equilibria along the curve is also preserved: the eigenvalues of $\mathbf{f}_\mathbf{y}(\mathbf{y}, p)$ depends continuously on p so the branch of equilibria $\mathbf{y} = \phi(p)$ will inherit the same stability properties of $\bar{\mathbf{y}}$.

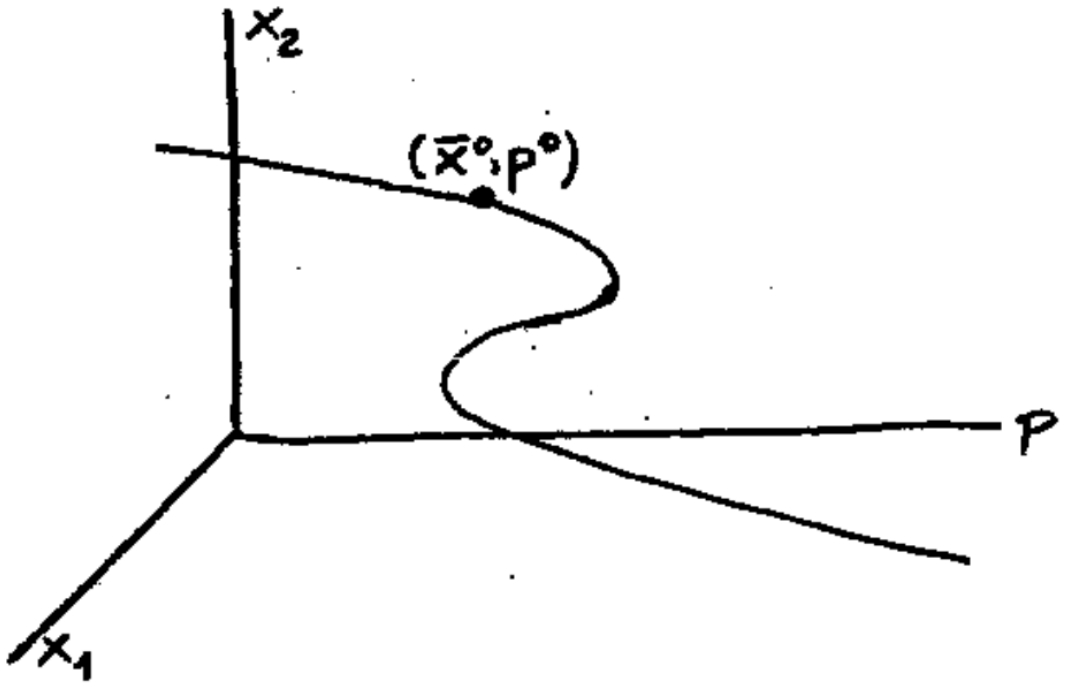


Figure B.5: Curve of equilibria

What if an eigenvalue of $\mathbf{f}_{\mathbf{y}}(\phi(p), p)$ *will end up with zero real part along the curve?*

Then we have a bifurcation. We still denote this point with $\bar{p}$ and the corresponding equilibrium with $\bar{\mathbf{y}}$. We have different options:

- **Tangent bifurcation.** $\mathbf{f}_{\mathbf{y}}(\bar{\mathbf{y}}, \bar{p})$ as eigenvalue 0 and $\mathbf{f}_p(\bar{\mathbf{y}}, \bar{p}) \neq 0$.
- **Transcritical bifurcation.** $\mathbf{f}_{\mathbf{y}}(\bar{\mathbf{y}}, \bar{p})$ as eigenvalue 0 and $\mathbf{f}_p(\bar{\mathbf{y}}, \bar{p}) = 0$.
- **Hopf bifurcation.** $\mathbf{f}_{\mathbf{y}}(\bar{\mathbf{y}}, \bar{p})$ as eigenvalue $\pm i\omega$ (only $n \geq 2$).

Note that if one real eigenvalue is zero, then we cannot apply the implicit function theorem, and the curve ϕ cannot be defined at $p = \bar{p}$ (this is the case of the first 2 bifurcations.)

B.5 Tangent bifurcation

The simplest bifurcation is perhaps the tangent bifurcation. Consider the following ODE:

$$y' = f(y, p) = p - y^2,$$

for $p \in \mathbb{R}$. We have the following situation:

- If $p > 0$ there are 2 equilibria, $\bar{y}^{\pm} = \pm\sqrt{p}$. Since $\partial_y f = -2y$, we have that $\bar{y}^-$ is unstable and $\bar{y}^+$ is asympotically stable.
- If $p < 0$ there is no real equilibrium.
- If $p = \bar{p} = 0$ we have a single equilibrium $\bar{y} = 0$. The stability cannot be deduced from the linearization (why?), but from the sign of f we see that $\bar{y}$ is attractive for $y > 0$ and repulsive for $y < 0$. It is called *saddle-node*.

Since $\partial_y f(\bar{y}, \bar{p}) = 0$ but $\partial_p f(\bar{y}, \bar{p}) = 1 \neq 0$, for $p = \bar{p} = 0$ we have a *tangent bifurcation*. Tangent bifurcations are also called *limit points* (in MatCont denoted by `LP`). The reason for the name should be clear from the bifurcation plot.

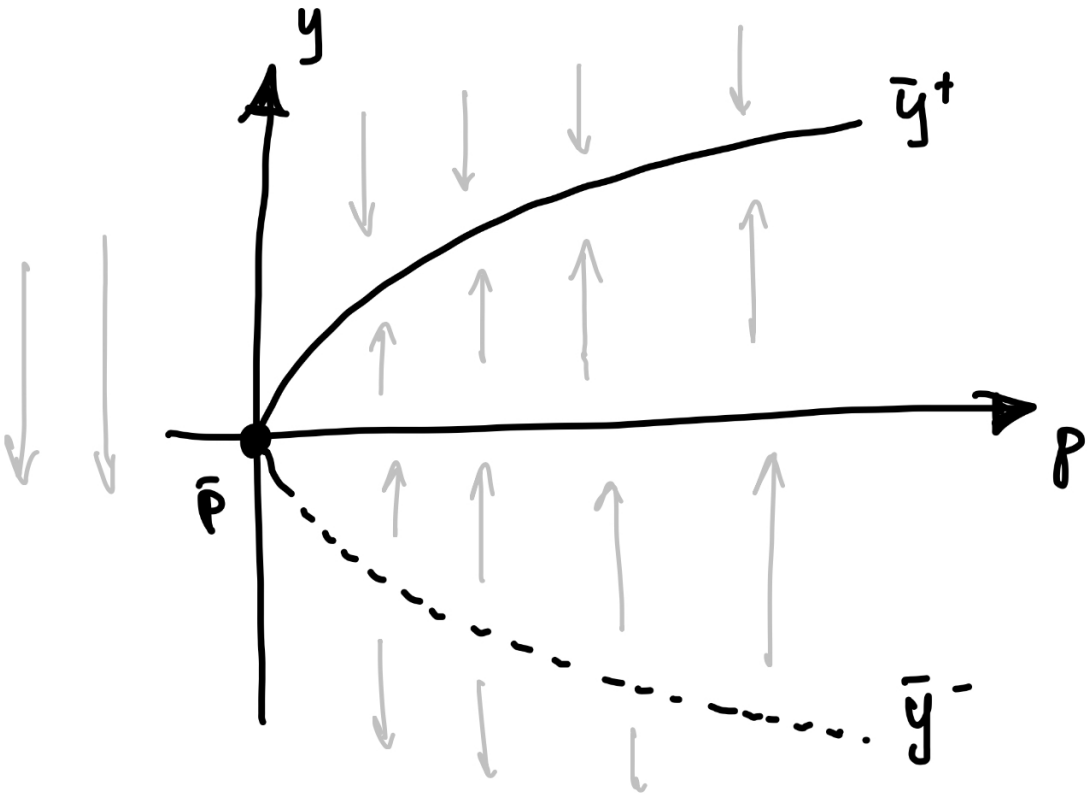


Figure B.6: tangent bifurcation

Tangent bifurcations are *catastrophic*, in the sense that a point at equilibrium right before the bifurcation will diverge from its state after the bifurcation (imagine to move a point along the stable branch from right to left.) Note that the curve is smooth, because $\partial_p f(\bar{y}, \bar{p}) \neq 0$. Check the next figure, where we visualize $f(y, p)$ around the bifurcation (in black the zero levelset, that is the curve of equilibria):

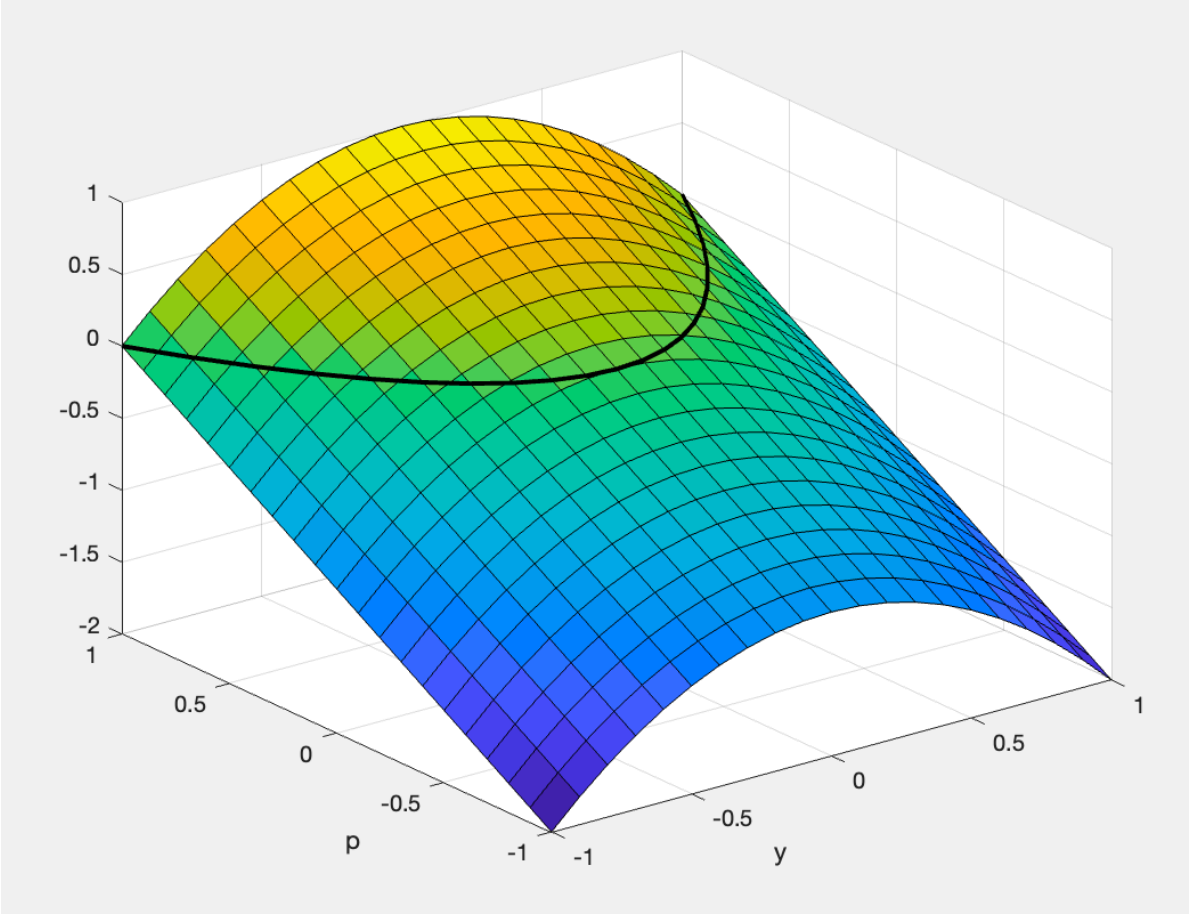


Figure B.7: image

B.6 Transcritical bifurcation

The transcritical bifurcation occurs when an equilibrium has 0 real part and also $\partial_p f(\bar{y}, \bar{p}) = 0$. The normal form is:

$$y' = py - y^2.$$

The derivative of f is $f_y(y, p) = p - 2y$. We have 2 equilibria,

- $y_0 = 0$, stable for $p > 0$ and unstable for $p < 0$.
- $\bar{y} = p$, unstable for $p > 0$ and stable for $p < 0$.

For $p = 0$ we only have one equilibrium $\bar{y} = 0$, which is a saddle-node because it is attractive for $y < 0$ and repulsive for $y > 0$.

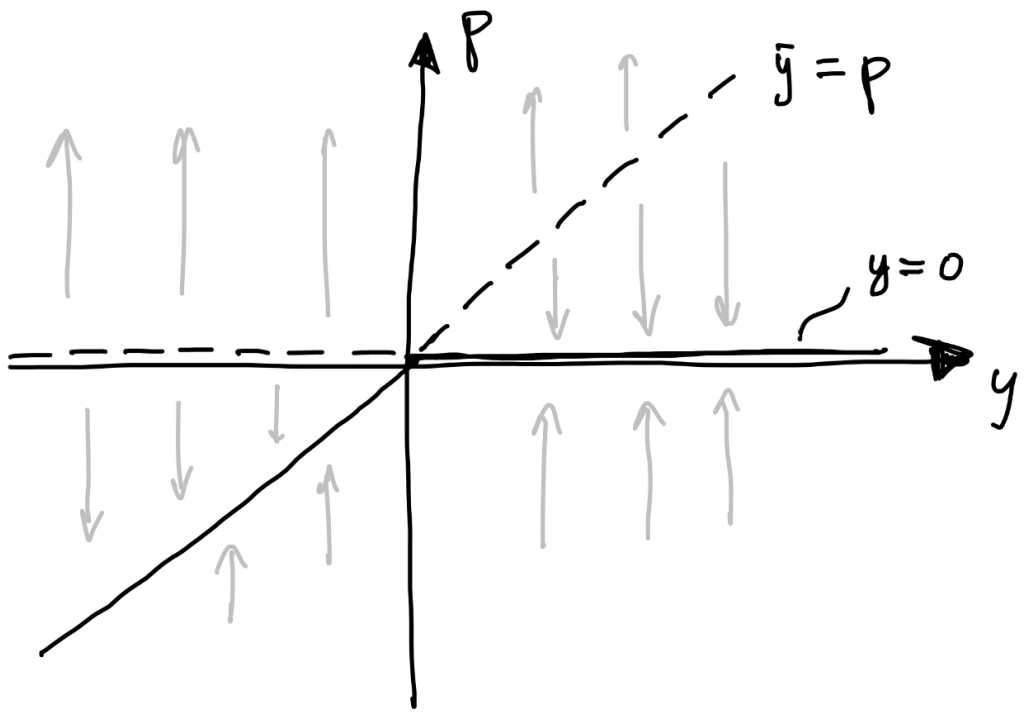


Figure B.8: image

Transcritical bifurcations are very common in biological modeling, because some equilibria (like the origin) do not depend on the parameters and are always present. Therefore, in the bifurcation plot their curve do not change, and can be intersected by other curves of equilibria. In fact, if $f(0, p) = 0$ for all p, then also $f_p(0, p) = 0$. Note that the above model is the logistic equation.

From the 3d visualization of f:

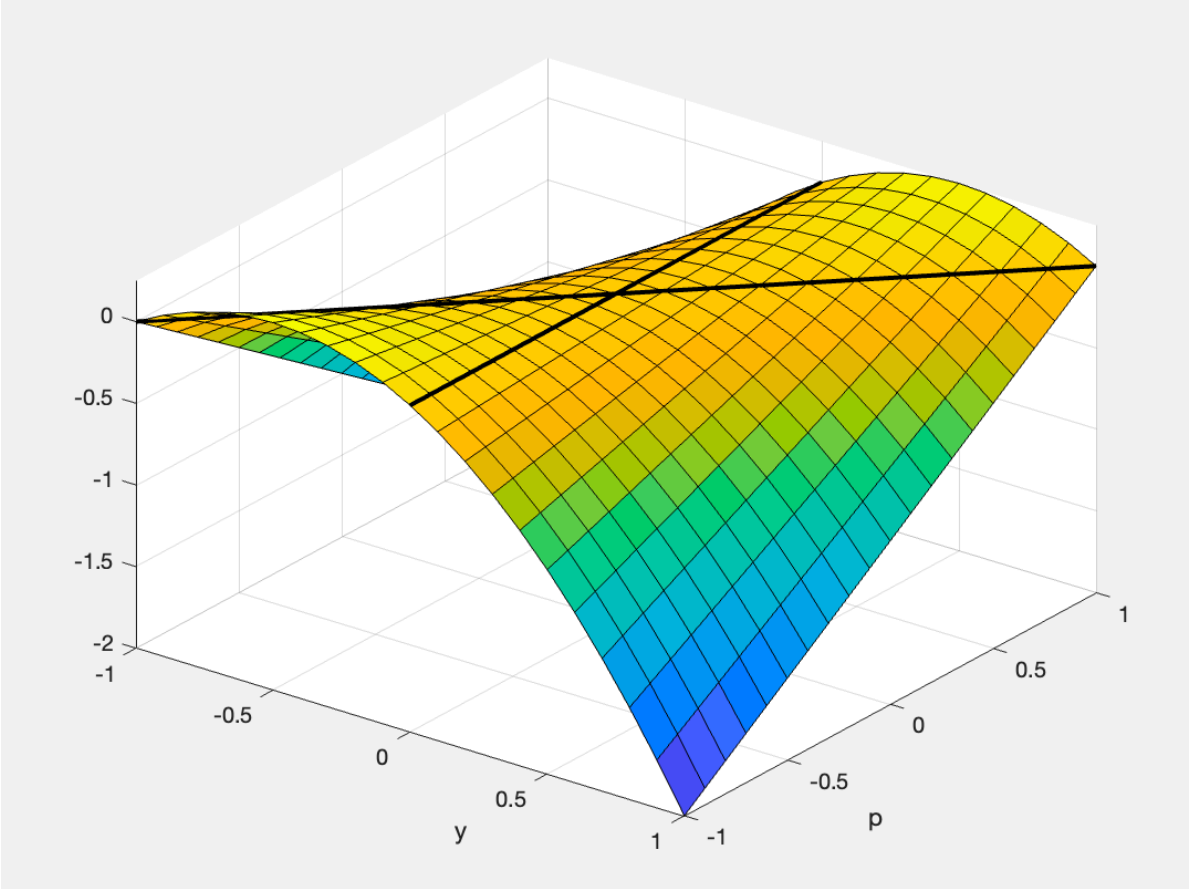


Figure B.9: image

we see a critical point at the bifurcation point. The eigenvectors give the tangents to the 2 curves of equilibria. In this way, it is possible to switch curve at the bifurcation point.

B.7 Hopf bifurcation

The Hopf bifurcation occurs when an equilibrium changes stability and a limit cycle appears. The normal form is:

$$\begin{cases} x' = \sigma x + \omega y + cx(x^2 + y^2), \\ y' = \sigma y - \omega x + cy(x^2 + y^2), \end{cases}$$

Suppose that σ, μ, and c depend on some parameter $p \in \mathbb{R}$.

We have that $(x, y) = (0, 0)$ is an equilibrium for all choices of p. The Jacobian at $(0, 0)$ has eigenvalues

$$\lambda^{\pm} = \sigma \pm i\omega,$$

When $\sigma(p) = 0$, the real part of the complex conjugate eigenvalues becomes zero: we have a Hopf bifurcation. To check the existence of a limit cycle, we consider the system in polar coordinates:

$$\begin{cases} x = r\cos\theta, \\ y = r\sin\theta, \end{cases} \quad \Rightarrow \quad \begin{cases} r = \sqrt{x^2 + y^2}, \\ \theta = \arctan(y/x). \end{cases}$$

To change the variables, simply note that:

$$\begin{aligned} r\mathrm{d}r &= \tfrac{1}{2}\mathrm{d}(r^2) = x\mathrm{d}x + y\mathrm{d}y \\ &= (\sigma(x^2 + y^2) + c(x^2 + y^2)(x^2 + y^2))\mathrm{d}t = \\ &= (\sigma r^2 + cr^4)\mathrm{d}t, \end{aligned}$$

and

$$\begin{aligned} \mathrm{d}\theta &= \frac{\mathrm{d}y/x - y\mathrm{d}x/x^2}{1 + (y/x)^2} = \frac{x\mathrm{d}y - y\mathrm{d}x}{x^2 + y^2} \\ &= \frac{x(\sigma y - \omega x + cy(x^2 + y^2) - y(\sigma x + \omega y + cx(x^2 + y^2)}{x^2 + y^2}\mathrm{d}t \\ &= -\frac{\omega(x^2 + y^2)}{x^2 + y^2}\mathrm{d}t = -\omega\mathrm{d}t. \end{aligned}$$

The system in polar coordinates is:

$$\begin{cases} r' = \sigma r + cr^3, \\ \theta' = -\omega. \end{cases}$$

Assuming $\omega(p) \neq 0$, we have two equilibria:

- $r = 0$, which corresponds to the origin and we know it is stable for $\sigma(p) < 0$ and unstable for $\sigma(p) > 0$.
- $r = \bar{r} = \sqrt{-\sigma/c}$, which exists only for $c(p) \neq 0$ and $\sigma(p)c(p) < 0$. When there exists, it is stable if $\sigma(p) > 0$ and unstable if $\sigma(p) < 0$. This corresponds to a limit cycle of radius $\bar{r}$.

In conclusion, we have 2 types of Hopf bifurcations.

- *Supercritical Hopf bifurcation* H^S: $\sigma'(\bar{p}) > 0$ and $c(\bar{p}) < 0$. The stable equilibrium at the origin becomes unstable and a stable limit cycle is born. The bifurcation is not catastrophical.

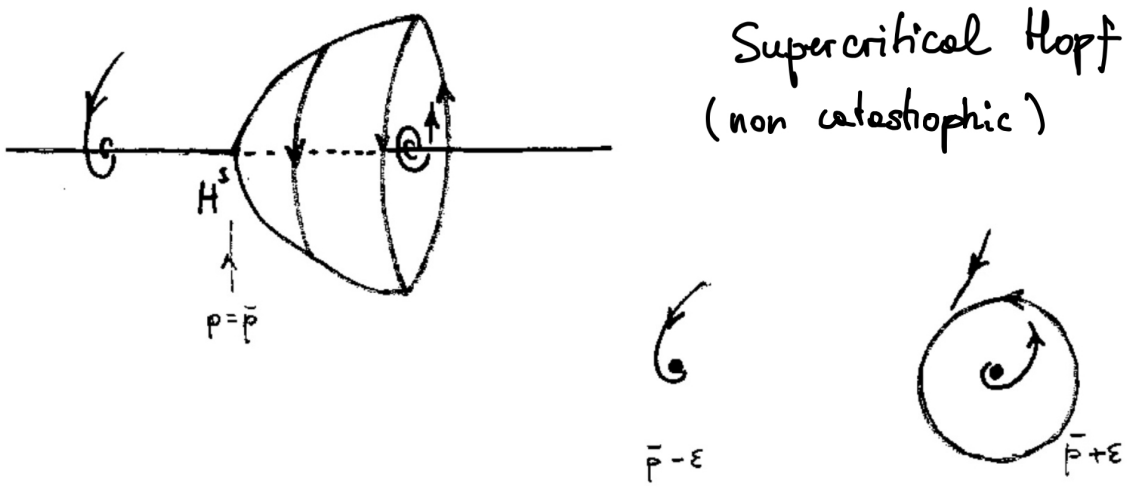


Figure B.10: image

- *Subrcritical Hopf bifurcation* H_S: $\sigma'(\bar{p}) < 0$ and $c(\bar{p}) > 0$. The unstable equilibrium at the origin becomes stable and an unstable limit cycle is born. The bifurcation is catastrophical.

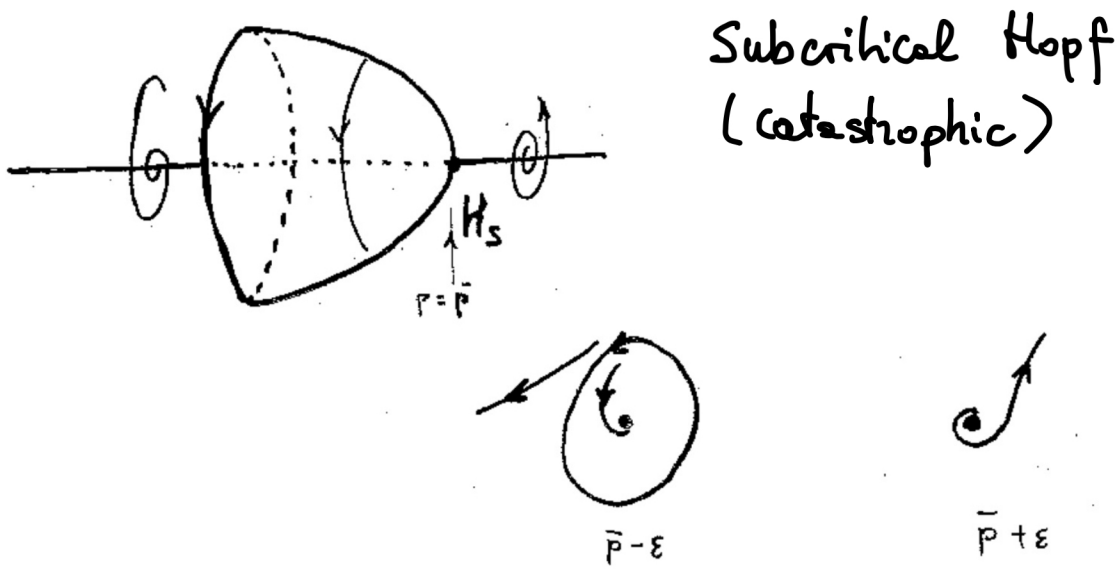


Figure B.11: image

Note that for $c = 0$ at the bifurcation point we have an infinite number of limit cycles (non-isolated periodic orbit), like the classical oscillator. This is a *degenerate Hopf bifurcation* so the condition $c(\bar{p}) \neq 0$ is called *non-degeneracy condition*. The condition $\sigma'(\bar{p}) \neq 0$ is called *trasversality condition*.

For a general system of ODEs:

$$\begin{cases} x' = f(x, y, p), \\ y' = g(x, y, p), \end{cases}$$

such that $(\bar{x}, \bar{y}) = (0, 0)$ is an equilibrium (with no loss of generality we can take the equilibrium at the origin), thanks to linearization we can always write the system around the equilibrium in the form:

$$\begin{cases} x' = \sigma(p)x + \omega(p)y + \tilde{f}(x, y, p), \\ y' = \sigma(p)y - \omega(p)x + \tilde{g}(x, y, p), \end{cases}$$

with $\tilde{f}(0, 0, p) = \tilde{g}(0, 0, p) = 0$. It is possible to further expand $\tilde{f}$ and $\tilde{g}$ so to find a term of the form cr^3 as above. The coefficient c is called *first Lyapunov exponent* and it reads:

$$\begin{aligned} 16c = {} & f_{xxx} + f_{xyy} + g_{xxy} + g_{yyy} \\ & + \frac{1}{\omega}\Big(f_{xy}(f_{xx} + f_{yy}) - g_{xy}(g_{xx} + g_{yy}) - f_{xx}g_{xx} - f_{yy}g_{yy}\Big). \end{aligned}$$

Then we have the following:

Theorem B.2 (Hopf)**.** *Suppose that $\sigma'(p) \neq 0$ (trasversality) and $c \neq 0$ (non-degeneracy). Then there exists a branch Γ_p of periodic solutions with period $T(p)$ for p such that $|p - \bar{p}|$ is small and $p > \bar{p}$ if $\sigma'(\bar{p})c < 0$ (resp. $p < \bar{p}$ if $\sigma'(\bar{p})c > 0$). Furthermore, $\Gamma_p \to \bar{y}$ for $p \to \bar{p}$ and $T(p) \to \frac{2\pi}{\omega(\bar{p})}$. If $c < 0$ Γ_p are attracting; if $c > 0$ Γ_p are repelling.*

We observed the Hopf bifurcation in Gause-type prey-predator models and in negative feedback networks.

B.8 Limit cycle tangent bifurcation

Consider the following model:

$$\begin{cases} r' = r\sigma(1 + cr^2 - r^4), \\ \theta' = -\omega. \end{cases}$$

We have 3 equilibria (note that necessarily $r \geq 0$): the origin $r = 0$ and:

$$\bar{r} = \sqrt{\frac{c \pm \sqrt{c^2 + 4\sigma}}{2}}.$$

If we study the existence of limit cycles (LCs) in the (σ, c) plane we have:

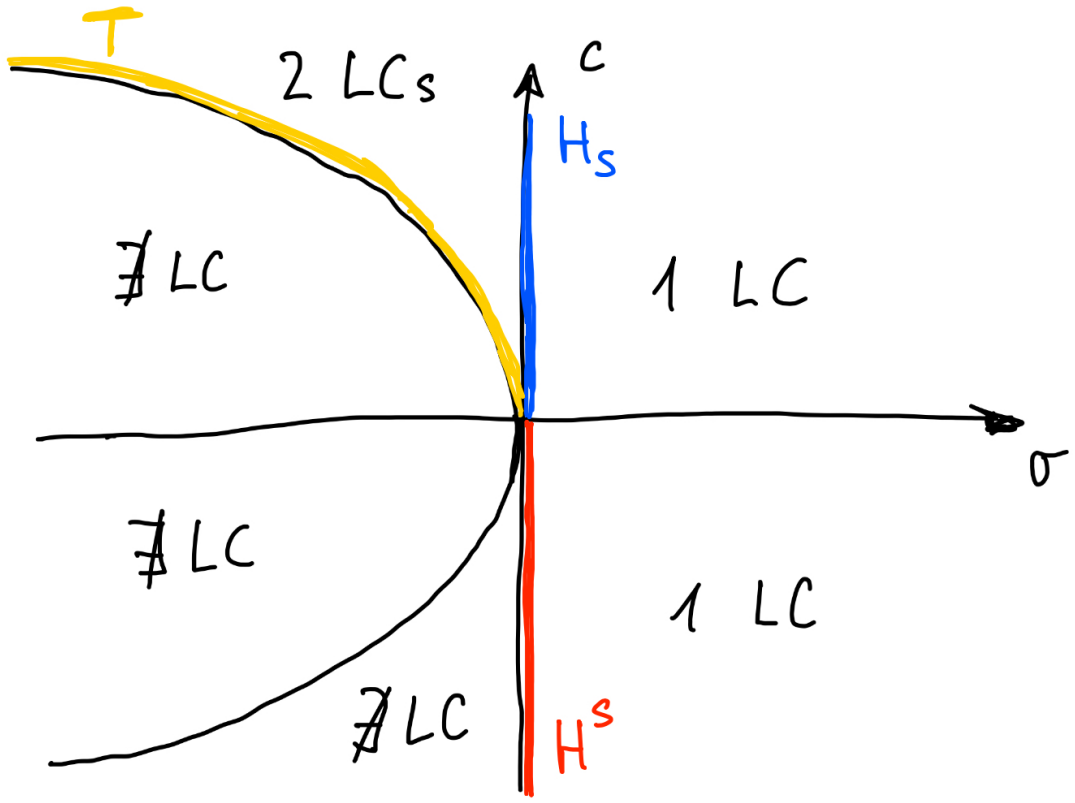


Figure B.12: image

where:

- H^S: supercritical Hopf bifurcation,
- H_S: subcritical Hopf bifurcation,
- T: tangent bifurcation between limit cycles.

In terms of stability we have:

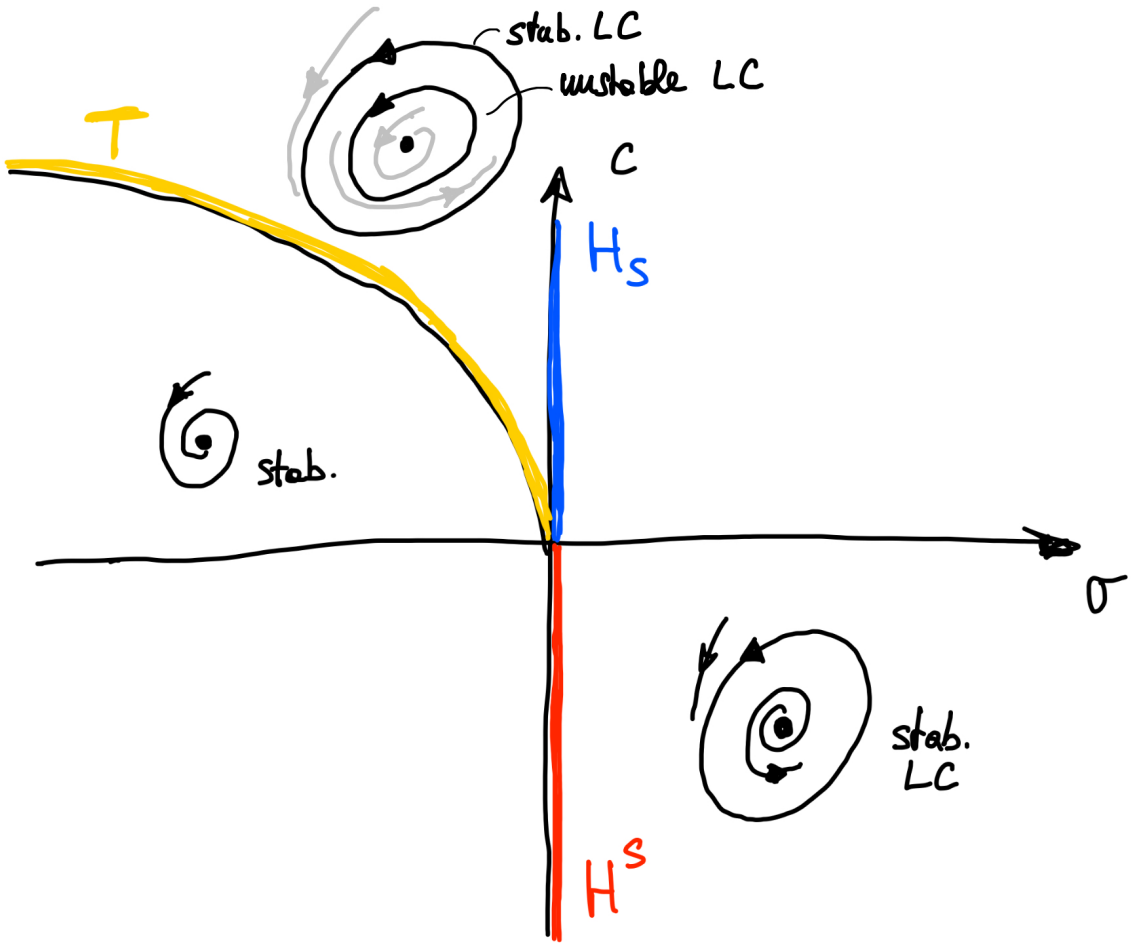


Figure B.13: image

As we cross T we have 2 limit cycles colliding and disappearing, exactly like the tangent bifurcation between equilibria. (In fact, this is what happens in polar coordinates.)

The tangent bifurcation can be visualized as follows:

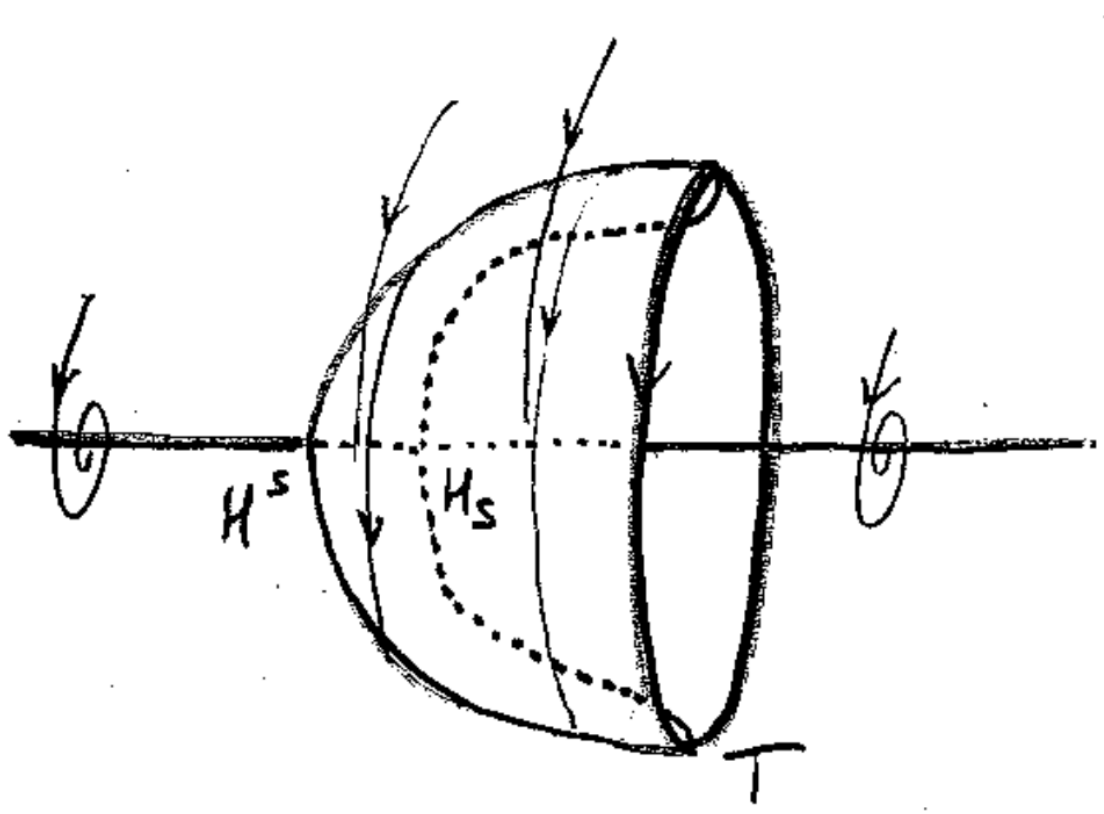


Figure B.14: image

We have seen this type of bifurcation in a negative feedback loop. It is catastrophic.

C Introduction to MATLAB

C.1 Overview

MATLAB is a software for scientific computing. It is a self-contained environment, with thousands of optimized functions for many tasks. MATLAB is a proprietary software, so you need a license to use it. Most universities offer an educational license (see this website for UniTrento.)

There are free alternatives too:

- GNU Octave, an open-source software aiming at 1:1 compatibility with Matlab. You can use it, but beware that some MATLAB packages may not work (e.g., MatCont).
- Python, probably the most popular programming language, has many libraries for scientific computing, such as numpy, scipy, matplotlib, jax, and many others. It would be the best alternative to Matlab, if it weren't for MatCont, that we will extensively use.
- Julia, it is a python competitor, with a similar syntax but much better performance. Many libraries have a Julia wrapper, but again not MatCont.

You are free to use any Matlab alternative, and rely on it only for MatCont.

C.2 Installation

Matlab installation is straightforward on most OSes. Please follow the instruction on the website. We do not need special toolboxes. In case, you can install them later.

You can also use the online version of MATLAB. It comes with an online storage called MATLAB Drive that you can use to sync your files.

C.3 Quick tour

Once open, Matlab will look like this:

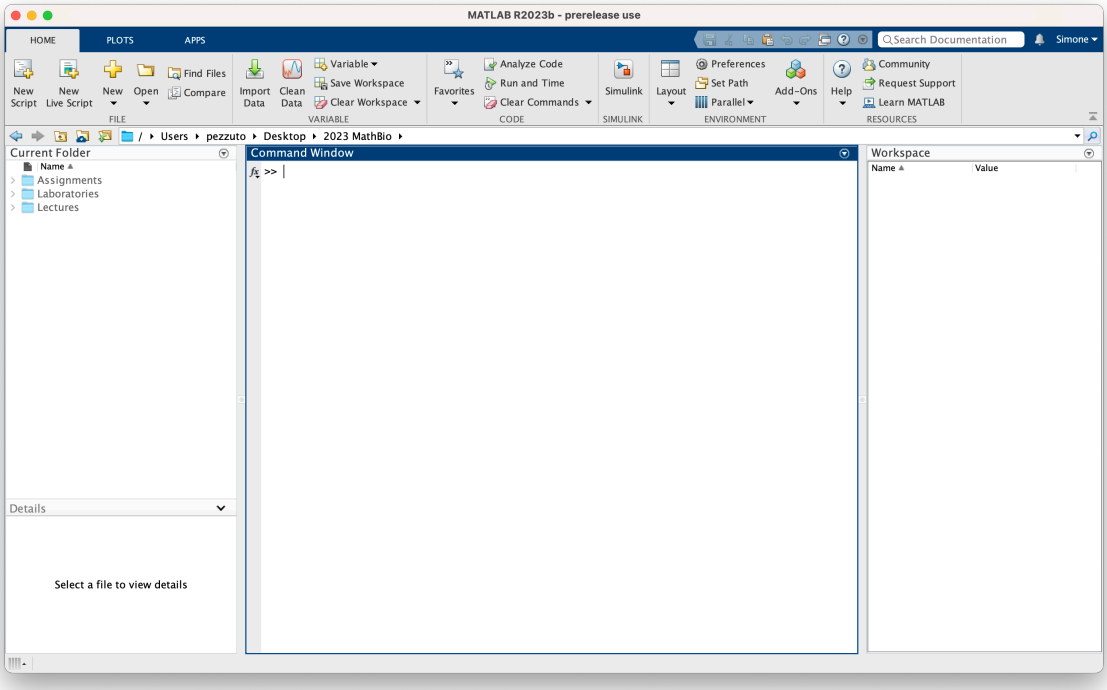


Figure C.1: The MATLAB interface

You can notice different panels:

- **Command Window** is where we type commands.
- **Current Folder** is the current working directory.
- **Workspace** is a summary of all variables.

The system is waiting for an input `>>`.

C.3.1 Basics

MATLAB stands for **MA**trix **LAB**oratory, so its philosophy is to store everything numeric as a matrix. (This is not entirely true anymore, recent versions of Matlab introduced many new types.) A scalar is a 1×1 matrix. Matlab a *weakly typed* (and interpreted) programming language. A variable is a generic, and no type is needed like in C:

```
>> a = 2.45
a =
    2.4500
```

As we inout `a = 2.45`, the system responds with its representation. Note that this is not the internal precision! All numbers with a period are stored as `double`. To suppress the output, just put a semicolon:

```
>> a = 2.45;
```

We can execute more commands per line (note the comma):

```
>> a = 2.45 , b = 3.1; A = 1.2;
```

> **Warning**
>
> Matlab is case-sensitive: the variable `a` and `A` are different. This is the standard behavior also for file names in POSIX systems like UNIX, but not for Windows. Also on macOS you need to be careful: by default the file system is case-insensitive like Windows.

If a number is assigned to nothing, it will go to the variable `ans`:

```
>> 2.45;
>> whos
  Name      Size             Bytes  Class     Attributes
  ans       1x1                  8  double
>> ans
ans =
    2.4500
```

Some variables are predefined, like `pi`, π, and `1i` or `1j`, the imaginary unit i. Note that also `i` and `j` are valid for Matlab, but dangerous: `i` is often an iteration variable in a for-loop:

```
>> a = 5 + 2*i
a =
   5.0000 + 2.0000i

>> i = 2;
>> b = 5 + 2*i
b =
     9

>> c = 5 + 2*1i
c =
   5.0000 + 2.0000i
```

You can clear a variable by using `clear myvar`, or all variables with `clear`. The command `clc` clears the screen.

If you are not sure of a command, use tab-completion: type the beginning of a command, then type "TAB". You can also use `help` and `doc`.

C.3.2 Output format

By default, numbers are visualized in short format. The command `format` can change the behavior:

```
>> format long
>> 1/7
ans =
   0.142857142857143
```

Below in the table different options:

Format	ans
`format rat`	`1/7`
`format short`	`0.1429`
`format short e`	`1.4286e-01`
`format short g`	`0.14286`
`format long`	`0.142857142857143`
`format long e`	`1.428571428571428e-01`
`format long g`	`0.142857142857143`

C.3.3 Vector and matrices

We have many ways to generate a vector in Matlab. The most direct is:

```
>> b = [1 5 6 7];
>> c = [1, 5, 6, 7];
```

The two vectors are identical: the comma is optional. The dimension is 1×4, that is 1 row and 4 columns. For column vector, we need to use the semi-colon (now mandatory):

```
>> d = [1; 5; 6; 7];
>> whos
  Name      Size             Bytes  Class     Attributes
  b         1x4                 32  double
  c         1x4                 32  double
  d         4x1                 32  double
```

Note that the memory requirement is 32 bytes, that it $4 \cdot 8$ bytes, where 8 bytes is a 64-bit floating-point number (`double`).

We can generate a vector also as follows:

```
<start> : <increment> : <end>
```

If we omit the increment, it is assumed one:

```
>> 1:8
ans =
     1     2     3     4     5     6     7     8

>> 1:2:8
ans =
     1     3     5     7

>> 0.5:-0.1:0
ans =
    0.5000    0.4000    0.3000    0.2000    0.1000         0
```

The command above is quite powerful: it also compensates for rounding off errors. Alternatively, we can use `linspace`:

```
>> linspace(0,1,5)
ans =
         0    0.2500    0.5000    0.7500    1.0000
```

Matrices easily follows:

```
>> A = [ 1 2 3 4; 5 6 7 8 ; 9 10 11 12; 13 14 15 16 ]
A =
     1     2     3     4
     5     6     7     8
     9    10    11    12
    13    14    15    16
```

which is returning a 4×4 matrix. More complex matrices can be created with *ad-hoc* functions like `eye`, `ones`, `zeros`, `reshape`, and so on.

```
>> reshape(A, 2, 8)
ans =

     1     9     2    10     3    11     4    12
     5    13     6    14     7    15     8    16

>> B = [ 1:3 0; 4:-1:1 ]
B =
     1     2     3     0
     4     3     2     1

>> eye(2,3)
ans =
     1     0     0
     0     1     0

>> ones(2,2)
ans =
     1     1
     1     1
```

> **Exercise**
>
> Try to create the following matrix:

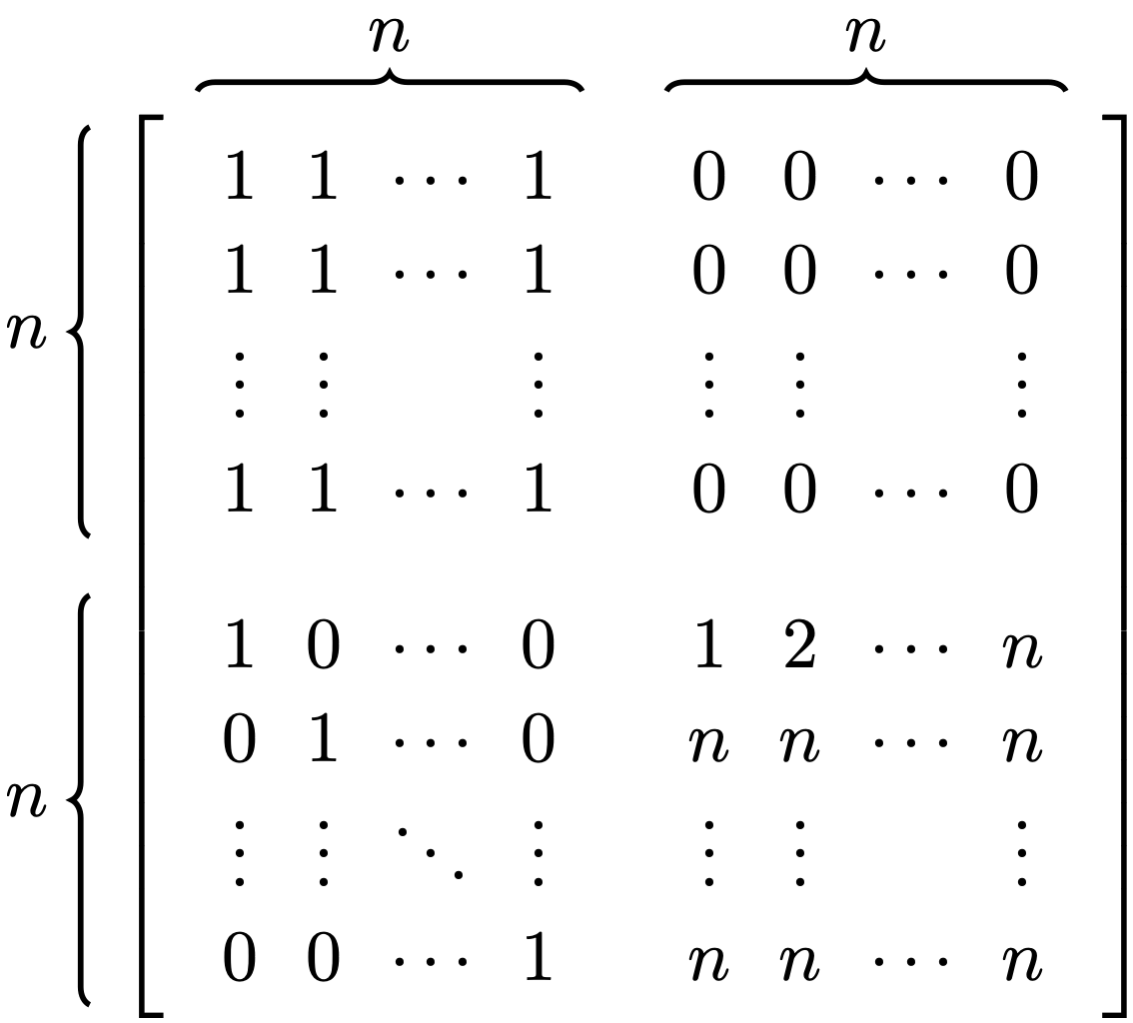


Figure C.2: image

Solution

```
>> n = 3;
>> A = [ones(n), zeros(n); ...
        eye(n), [1:n; n*ones(n-1,n) ];
```

C.3.4 Matrix manipulation

We can access and modify matrices quite naturally:

```
>> v = linspace(0, 1, 5);
>> v(2)
ans =
    0.2500
```

The operator `( )` is much more poweful: we can use slices:

```
>> v([1 3 5])
ans =
         0    0.5000    1.0000

>> v(1:3)
ans =
         0    0.2500    0.5000

>> v(2:end)
ans =
    0.2500    0.5000    0.7500    1.0000

>> v([1 2; 3 2])
ans =
         0    0.2500
    0.5000    0.2500
```

We also can use vectors or matrices of indices. For instance, if indices are `[1 3 5]`, the output is `[v(1) v(3) v(5)]`. With matrices is similar:

```
>> A = [ 1 2 3 ; 4 5 6 ; 7 8 9]
A =
     1     2     3
     4     5     6
     7     8     9

>> A(1,1)
ans =
     1

>> A(1:3,1:2)
ans =
     1     2
     4     5
     7     8
```

```
>> A(1:end-1,2:end)
ans =
     2     3
     5     6

>> A([1 2],[2 3])
ans =
     2     3
     5     6
```

In the last case we extract the first and the second row, and then crossed with the second and third columns.

```
>> A(1:end)
ans =
     1     4     7     2     5     8     3     6     9

>> A([2 3; 9 6])
ans =
     4     7
     9     8
```

The operation `A(1:end)` is equivalent to `A(:)`, except for returning a column vector instead of a row vector.

C.3.5 Operations on Matrices

Let's now discuss algebraic operations on matrices. The basic rules are as follows, for two matrices $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\mathbf{B} \in \mathbb{C}^{p \times q}$:

- Addition is defined if and only if $n = p$ and $m = q$, and $\mathbf{A} + \mathbf{B} \in \mathbb{C}^{n \times m}$.
- Multiplication is defined if and only if $m = p$, and $\mathbf{AB} \in \mathbb{C}^{n \times q}$.
- Transpose $\mathbf{A}^T \in \mathbb{C}^{m \times n}$, with $(\mathbf{A}^T)_{ij} = (\mathbf{A})_{ji}$.
- Conjugate transpose $\mathbf{A}^H \in \mathbb{C}^{m \times n}$, with $(\mathbf{A}^H)_{ij} = \overline{(\mathbf{A})_{ji}}$.

The last two operations can be performed as follows:

```
>> A = [ 1+1i, 3; 1, 1-2i ];
>> A.'
ans =
   1.0000 + 1.0000i   1.0000
```

```
   3.0000             1.0000 - 2.0000i
>> A'
ans =
   1.0000 - 1.0000i   1.0000
   3.0000             1.0000 + 2.0000i
```

If the matrix is real, these two operations are the same. In fact, it is common to use ' as the transpose of a matrix:

```
>> a = [ 1 2 5 ];
>> b = [ 4 6 4 ];
>> H = [ 1 2 3 ; 2 4 7 ; 1 4 3 ];
>> G = [ 4 6 2 ; 8 4 1 ; 3 2 9 ];
>> a + b
ans =
     5     8     9
>> a - b
ans =
    -3    -4     1
>> H + G
ans =
     5     8     5
    10     8     8
     4     6    12
>> H * G
ans =
    29    20    31
    61    42    71
    45    28    33
>> G * a
Error using *
Inner matrix dimensions must agree.
```

This is the most common error. In this case, the vector `a` is 1×3, while `H` is 3×3, which is not valid.

```
>> G * a'
ans =
    26
    21
    52
```

```
>> a * G
ans =
    35    24    49
>> 2 * a
ans =
     2     4    10
>> a' * b
ans =
     4     6     4
     8    12     8
    20    30    20
>> a * b'
ans =
    36
>> A^2
ans =
     8    22    26
    17    48    55
    12    30    40
>> a^2
Error using ^
Inputs must be a scalar and a square matrix.
To compute elementwise POWER, use POWER (.^) instead.
```

The last message tells us that exponentiation is intended as matrix multiplication and is only defined for square matrices. It suggests using the dot version instead. In fact, scalar binary operations like multiplication, division, and exponentiation become element-wise operations for vectors and matrices when you put a dot in front of the operation symbol:

```
>> a .* b
ans =
     4    12    20
>> a ./ b
ans =
    0.2500    0.3333    1.2500
>> a .^ 2
ans =
     1     4    25
>> a .^ b
ans =
     1    64   625
```

The only thing to keep in mind is that the two operands should have the same dimensions, or at least one of them should be a scalar (in which case it is repeated).

Exercise

Try to create the following vector, for arbitrary n:

$$\Big[\overbrace{\underbrace{1,1,\dots,1}_{n},\underbrace{2,2,\dots,2}_{n},\dots,\underbrace{n,n,\dots,n}_{n}}^{n}\Big].$$

Solution

```
>> n = 5;
>> v = reshape(ones(n,1)*(1:n),1,[])
```

C.3.6 Elementary Mathematical Functions

One of the strengths of MATLAB is the vast number of mathematical functions (both elementary and advanced) it offers. Unless specified otherwise, almost all of these functions operate element-wise. For example, given a matrix `A`, `exp(A)` is not the matrix exponential (which is calculated with `expm(A)`), but rather a matrix whose elements are the exponentials of the original elements. Here are some elementary mathematical functions:

- `abs(A)`: Absolute value of the elements of `A`
- `sqrt(A)`: Square root of the elements of `A`
- `exp(A)`: Exponential function applied to the elements of `A`
- `log(A)`: Natural logarithm of the elements of `A`
- `log10(A)`: Base 10 logarithm of the elements of `A`
- `log2(A)`: Base 2 logarithm of the elements of `A`
- `sin(A)`: Sine of the elements of `A`
- `cos(A)`: Cosine of the elements of `A`
- `tan(A)`: Tangent of the elements of `A`
- `asin(A)`: Arcsine of the elements of `A` (in radians)
- `acos(A)`: Arccosine of the elements of `A` (in radians)
- `atan(A)`: Arctangent of the elements of `A` (in radians)
- `sinh(A)`: Hyperbolic sine of the elements of `A`
- `cosh(A)`: Hyperbolic cosine of the elements of `A`
- `tanh(A)`: Hyperbolic tangent of the elements of `A`

Exercise

1. Compute $\dfrac{e^5 + \sin(\pi)}{\sqrt{\log_2 30} - 10}$ and $e^{\log_{10} 50} + e^{\log 30} + e^{\log_2 40}$.
2. Define $\mathbf{x} = [1, 3, 4]$ and $\mathbf{y} = [1, 1, 2]$. Compute $2x_i \log_2(|y_i| + 1) - y_i \log_{10}(x_i + 2)$ and $\arctan\left(\dfrac{x_i}{y_i}\right) - \sin^2\left(x_i \sqrt[3]{|y_i|^2}\right)$.

Solution

```
>> % Point 1
>> (exp(5) + sin(pi))/(sqrt(log2(30))-10)
>> exp(log10(50)) + exp(log(30)) + exp(log2(40))
>> % Point 2
>> x = [ 1 3 4 ];
>> y = [ 1 1 2 ];
>> 2*x.*log2(abs(y)+1) - y.*log10(x+2)
>> atan(x./y) - sin(x.*abs(y).^(2/3)).^2
```

C.3.7 Elementary Mathematical Functions (Continued)

Let `A`, `B` be two matrices, and `b` be a vector.

- `size(A)`: Returns a two-element vector, where the first element is the number of rows in `A`, and the second element is the number of columns.
- `size(A, 1)`: Returns the first element of `size(A)`, which is the number of rows.
- `size(A, 2)`: Returns the second element of `size(A)`, which is the number of columns.
- `length(b)`: Returns the number of elements in the vector `b`.
- `max(b)`: Returns the largest element in the vector `b`.
- `min(b)`: Returns the smallest element in the vector `b`.
- `max(A)`: Returns a row vector containing the maximum element of each column of `A`.
- `min(A)`: Returns a row vector containing the minimum element of each column of `A`.
- `max(A, B)`: Returns a matrix of the same dimensions as `A` and `B`, containing the element-wise maximum.
- `min(A, B)`: Returns a matrix of the same dimensions as `A` and `B`, containing the element-wise minimum.
- `max(A, [], 2)`: Returns a column vector containing the maximum element of each row of `A`. If you replace 2 with 1, you get `max(A)`.
- `min(A, [], 2)`: Returns a column vector containing the minimum element of each row of `A`. If you replace 2 with 1, you get `min(A)`.

- `sum(b)`: Returns a scalar equal to the sum of the elements in the vector `b`.
- `sum(A)`: Returns a row vector whose elements are the column-wise sum of the elements in matrix `A`.
- `sum(A, 2)`: Returns a column vector whose elements are the row-wise sum of the elements in matrix `A`.
- `diag(A)`: Returns the vector of the diagonal elements of matrix `A`.
- `diag(A, k)`: Returns the vector of the k-th super-diagonal of matrix `A`.
- `diag(A, -k)`: Returns the vector of the k-th sub-diagonal of matrix `A`.
- `diag(b)`: Returns a matrix with the elements of vector `b` on its diagonal.
- `diag(b, k)`: Returns a matrix with the elements of vector `b` on its k-th super-diagonal.
- `diag(b, -k)`: Returns a matrix with the elements of vector `b` on its k-th sub-diagonal.
- `tril(A)`: Returns the lower triangular part of matrix `A`, making all elements strictly above the diagonal zero (even in a rectangular matrix).
- `triu(A)`: Returns the upper triangular part of matrix `A`, making all elements strictly below the diagonal zero (even in a rectangular matrix).

There are many other important functions such as `det` (determinant of a square matrix), `trace` (trace of a matrix), `norm` (for calculating norms), and so on. We will introduce these functions as needed.

Exercise

Try to use the function `magic(n)`. Then sum by row, column, diagonal, etc. What do you observe?

Solution

```
>> n = 4;
>> A = magic(n);
>> sum(A, 2)
>> sum(A, 1)
>> sum(diag(A))
```

We notice that the sum is always 34, at least for $n = 4$. In general, the magic square contains numbers from 1 to n^2, and the magic number is $\frac{1}{2}n(n^2 + 1)$. This is true even if we sum over the antidiagonal

```
>> sum(diag(fliplr(A)))
```

The function `fliplr` (*flip left–right*) inverts the right with the left in a given matrix.

C.3.8 Functions and Scripts

Many of the commands we have introduced are defined as functions, which are procedures that take input data and provide output. In MATLAB, there are several ways to define a function.

Suppose we want to create a function `fun(x)` that returns the value of $f(x) = x\sin(x)+\cos^2(x)$ for a given x. The "legacy" method involves defining a string corresponding to the function and then evaluating it using the `eval` command:

```
>> fun = 'x*sin(x) + (cos(x))^2';
>> x = 1.0;
>> eval(fun)
ans =
    1.1334
```

However, there is an issue when `x` is a vector:

```
>> x = [ 1 2 3 ];
>> eval(fun)
Error using *
Inner matrix dimensions must agree.
```

To handle vector input correctly, we need to "vectorize" the expression by using element-wise operations:

```
>> fun = 'x.*sin(x) + cos(x).^2';
>> x = [ 1 2 3 ];
>> eval(fun)
ans =
    1.1334    1.9918    1.4034
```

Alternatively, you can use the `vectorize` command to transform the function into its vectorized version.

However, a more convenient way is to define functions using "anonymous functions":

```
>> fun = @(x) x.*sin(x) + cos(x).^2;
>> x = [ 1 2 3 ];
>> fun(x)
ans =
    1.1334    1.9918    1.4034
```

The `@` symbol is called a "function handle." Note that the `vectorize` command does not work with anonymous functions.

There are at least two other methods for defining functions, such as using `inline` and the Symbolic Math Toolbox. We won't delve into these methods here.

Instead, let's explore using scripts to define functions. An "script" is a text file with a `.m` extension containing a list of commands that MATLAB will execute when the file is called. You can edit the script using any text editor or with MATLAB's built-in editor by running the following command:

```
>> edit
```

Suppose, for example, that the file `myscript.m` contains the following:

```
clear; clc;
% This is a comment:
% sum of squares of the diagonal
% of the magic matrix
n = 4;
A = magic(n);
summa = sum(diag(A).^2)
```

If the file is in the working directory, you can run it as follows:

```
>> myscript
summa =
   414
```

The script is executed as if you had entered the commands directly in the terminal.

You can also have scripts that contain only a function with the following syntax:

```
function y = myfun(x)
  y = x.*sin(x) + (cos(x)).^2;
end
```

In this case, you must save the file with the same name as the function (e.g., `myfun.m`).

Now you can use the function:

```
>> x = [ 1 2 3 ];
>> myfun(x)
ans =
    1.1334    1.9918    1.4034
```

Function scripts can have multiple arguments and return multiple outputs:

```
function [A, b] = vecmat(n, m)
  % Generate a matrix A of size m x n and
  % a vector b of size n x 1 with random values.
  A = rand(n, m);
  b = rand(n, 1);
end
```

You can call this function as follows:

```
>> [K, f] = vecmat(2, 3)
K =
    0.4387    0.7655    0.1869
    0.3816    0.7952    0.4898
f =
    0.4456
    0.6463
```

Script functions can contain at most one function. It is also a good practice to use lowercase names without spaces. You can use functions with multiple arguments and return values as needed.

There are many other advanced features and techniques for creating and using functions in MATLAB, but these are the basics to get you started.

C.3.9 Loops and Control Structures

MATLAB is a full-fledged interpreted programming language, and it provides many control flow operations.

```
>> for i = 1:3, magic(i), end;
ans =
     1
ans =
     1     3
     4     2
ans =
     8     1     6
     3     5     7
     4     9     2
```

In the example above, a `for` loop iterates from 1 to 3, and for each iteration, the `magic` function is called.

```
>> n = 1; while n < 4, magic(n), n = n + 1; end;
ans =
     1
ans =
     1     3
     4     2
ans =
     8     1     6
     3     5     7
     4     9     2
```

In this example, a `while` loop is used. It continues to execute as long as the condition `n < 4` is true. Inside the loop, the `magic` function is called, and `n` is incremented.

```
>> if rand(1) > 0.5, disp('Greater than 0.5'); else, disp('Less than 0.5'); end;
Greater than 0.5
```

The `if` statement is used to perform conditional execution. In this example, it checks if a random number is greater than 0.5 and displays a message accordingly.

As we need more details, we will add them gradually. Typically, writing these control structures directly in the command line can be cumbersome and less readable. It's preferable to write everything in a script with proper indentation and execute the script as needed.

> **Exercise**
>
> - Write a function named `cubediag` that takes a matrix `A` as input and returns the sum of the cubes of the diagonal elements.
> - Write another function named `cubeantidiag` that returns the sum of the cubes of the anti-diagonal elements (the diagonal that is symmetric to the main diagonal).
>
> You can define these functions in separate script files or directly in the MATLAB environment.

Solution

```
function s = cubediag(A)
  s = sum(diag(A).^2);
end

function s = cubeantidiag(A)
  % we could also use fliplr
  Aflip = A(:,end:-1:1);
  s = cubediag(Aflip);
end

function s = cubeantidiag(A)
  % case of square matrices
  n = size(A,1);
  s = sum(A(end-n+1:1-n:n));
end
```

C.3.10 Plots

To create a plot of a real-valued function of a real variable in MATLAB, you can use the `plot` command. In its basic form, you provide two vectors, `x` and `y`, of the same size that represent the data points.

```
>> x = linspace(-1, 1, 100);
>> parab = @(x) x.^2;
>> plot(x, parab(x));
```

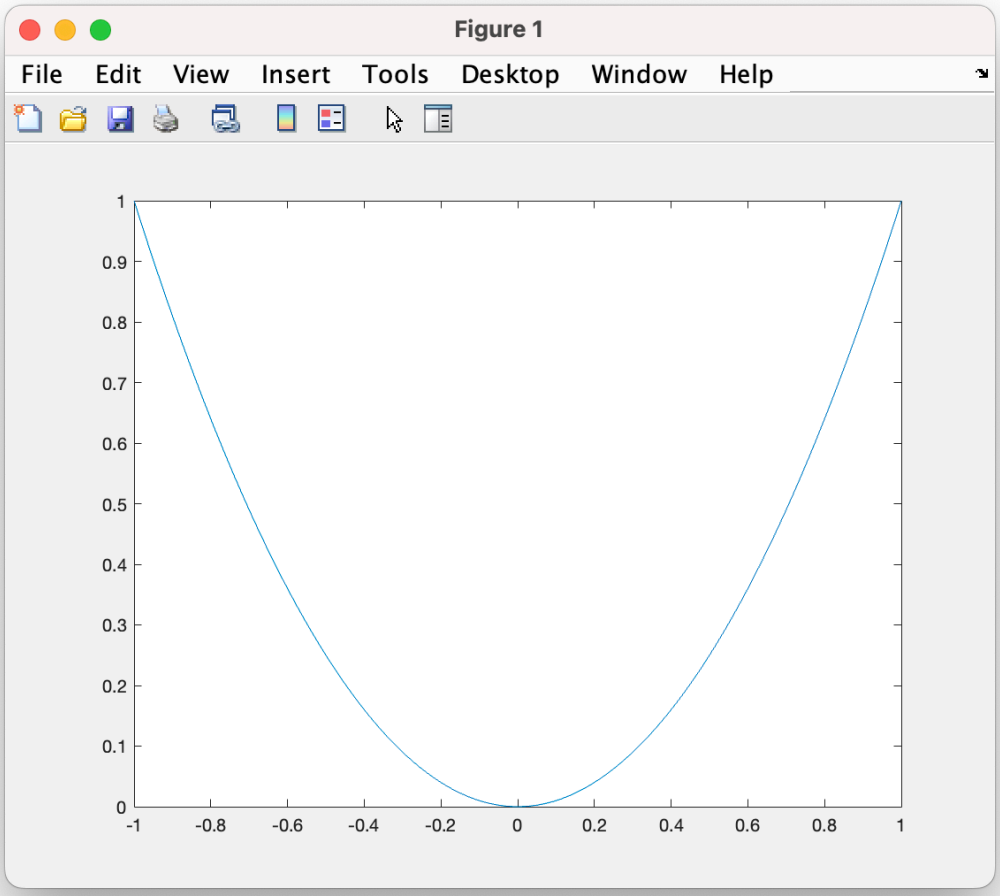


Figure C.3: Parabola Plot

You can customize the style, size, and color of the line using various options typically provided after the data points:

```
>> x = linspace(-1, 1, 20);
>> plot(x, parab(x), 'r*--', 'LineWidth', 2.0, 'MarkerSize', 15.0);
>> grid on;
>> xlabel('X-Axis');
>> ylabel('Y-Axis');
>> title('A Parabola', 'FontWeight', 'bold');
```

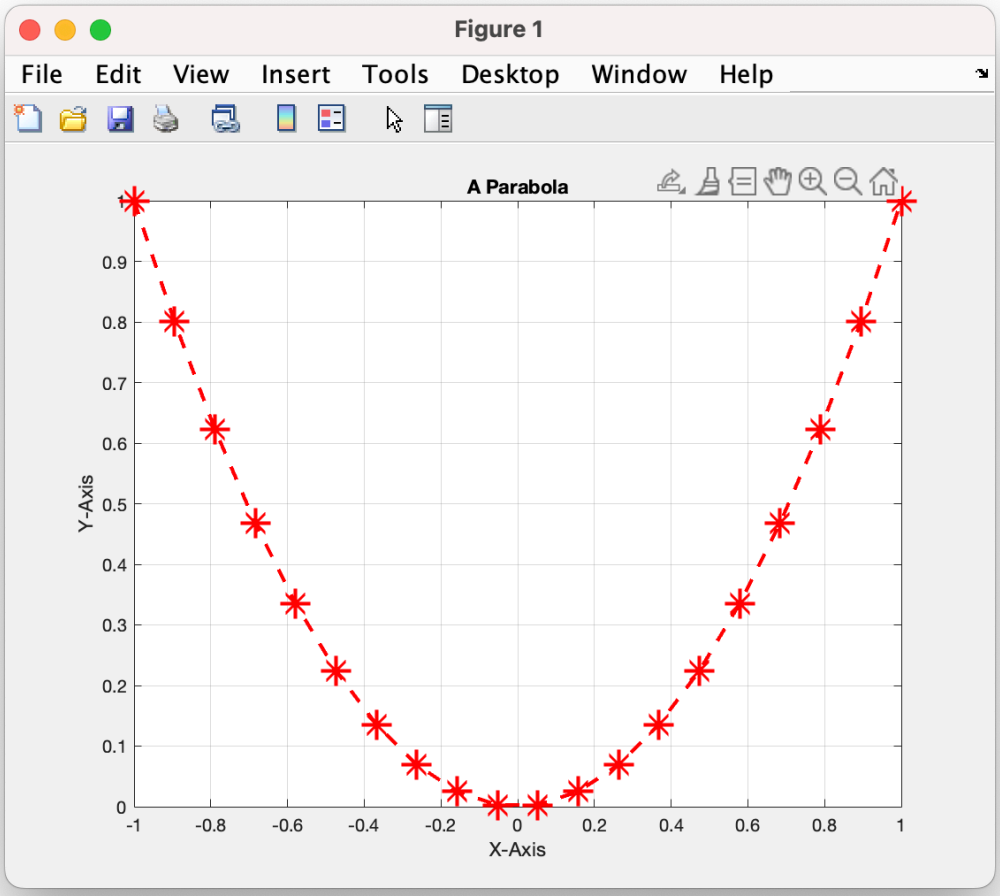


Figure C.4: Styled Parabola Plot

To see the possible combinations of line styles and markers, you can type `help plot` or `doc plot`. However, the most commonly used is the third argument, which is a string like `r*--`, which translates to:

- `r`: Red line
- `*`: Asterisk-shaped markers
- `--`: Dashed line

The order of styles is fixed, but each of them is optional. For example, by simply typing `y`, you will get a yellow line.

To overlay multiple plots, you can follow several approaches:

1. Direct Overlay:

```
>> x = linspace(-1, 1, 20);
>> parab = @(x) x.^2;
>> cubic = @(x) x.^3;
>> plot(x, parab(x), x, cubic(x));
```

2. Creating a Data Matrix:

```
>> plot(x, [parab(x); cubic(x)]);
```

3. Using the `hold` Command:

```
>> plot(x, parab(x));
>> hold all;
>> plot(x, cubic(x));
>> hold off;
```

You can also add a legend to distinguish the plots:

```
>> legend('Parabola', 'Cubic');
```

You can add multiple plots to the same graph by using the `hold` command:

```
>> plot(x, parab(x));
>> hold on;
>> plot(x, cubic(x));
>> hold off;
```

You can add a legend to differentiate between the plots:

```
>> legend('Parabola', 'Cubic');
```

Another important type of plot, especially when evaluating the behavior of errors concerning parameters, is the use of logarithmic or semilogarithmic scales. For a given function $f(x)$ with the graph $y = f(x)$:

- `semilogx` represents the points after changing the variable $x \mapsto \log x$.
- `semilogy` represents the points after changing the variable $y \mapsto \log y$.
- `loglog` performs both of the above changes.

For example, the graph $y = e^{\alpha x}$, in a y-logarithmic scale, becomes $\tilde{y} := \log y = \alpha x$, which is a straight line.

```
>> x = linspace(1, 10, 100);
>> semilogy(x, exp(x), x, exp(2*x), 'LineWidth', 2.0);
>> grid on;
>> legend('Slope 1', 'Slope 2', 'Location', 'NorthWest');
```

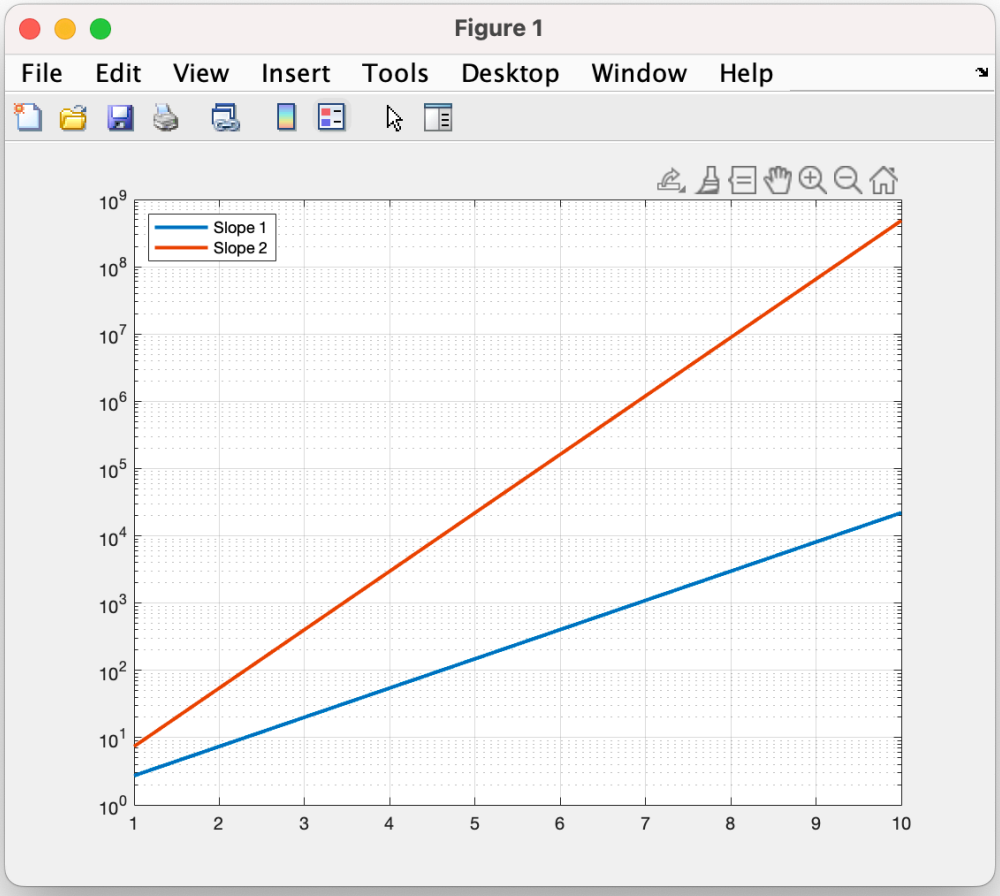


Figure C.5: Logarithmic Plot

Exercise

1. Plot the graph of the function $f(x) = 2+(x-3)\sin(5(x-3))$ for $0 \leq x \leq 6$. Overlay dashed lines that bound this function.

2. Consider the function $f(x) = (\log x)^2$ for $0.1 \leq x \leq 10$. What do you expect from the graph in logarithmic scaling? Plot and verify.

Solution

```
>> x = linspace(0, 6, 100);
>> f = 2 + (x-3).*sin(5*(x-3));
>> r = [ 2 + (x-3); 2 - (x-3) ];
>> plot(x, f, 'k-', x, r, 'k--', 'LineWidth', 2.0);
```

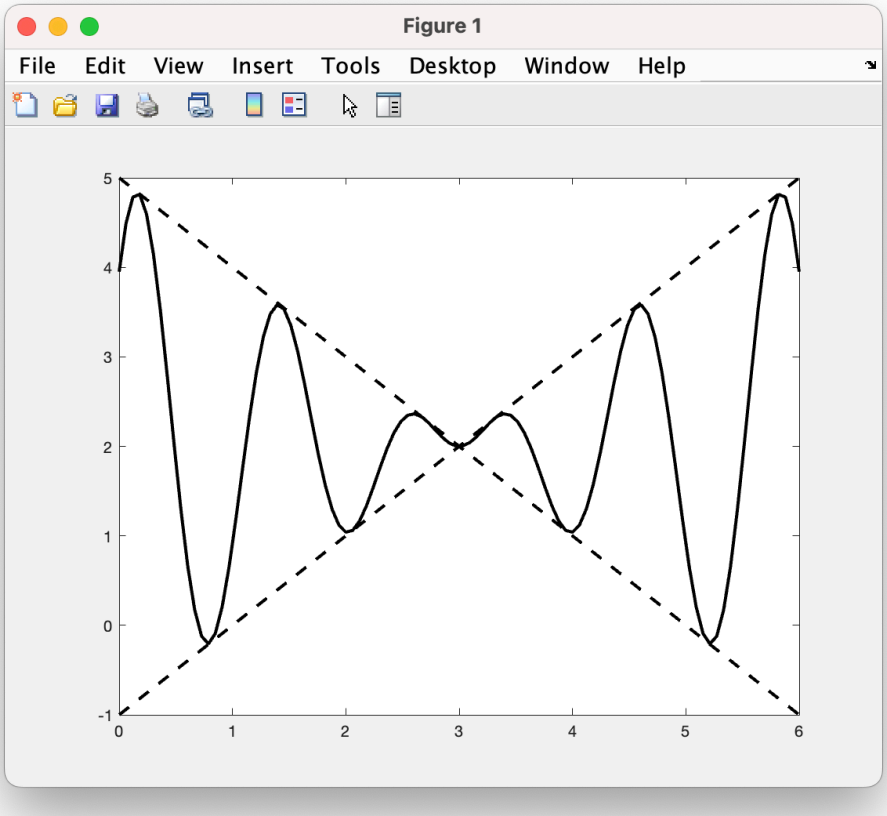
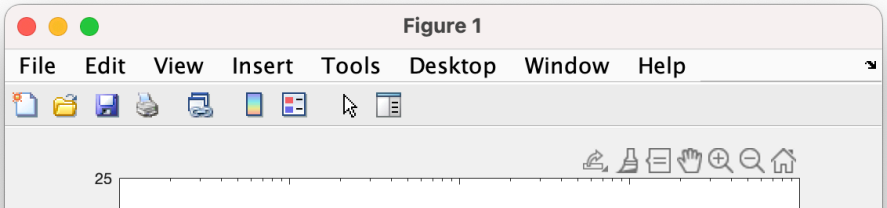


Figure C.6: image

Given $\hat{x} = \log x$ and $\hat{y} = \log y$, the plot in x-log scale is $y = (\log 2)^2 = \hat{x}^2$. We expect a parabola.

```
>> x = 10.^linspace(-2,2,100);
>> y = log(x).^2;
>> semilogx(x, y, 'LineWidth', 2.0);
```

C.4 Approximation error

When working with approximations, it's essential to measure the accuracy of an approximation $\hat{x}$ to a number $x \in \mathbb{R}$. Two common metrics are:

- Absolute Error: $E_{\text{abs}}(\hat{x}) = |x - \hat{x}|$
- Relative Error: $E_{\text{rel}}(\hat{x}) = \frac{|x-\hat{x}|}{|x|}$ for $x \neq 0$

Significant figures are a way to represent the accuracy of an approximation. Given an approximation $\hat{x}$ to x, it has p significant figures if:

$$E_{\text{abs}}(\hat{x}) \leq \frac{1}{2} \times 10^{s-p+1}$$

Where s is the largest integer such that $|x| \geq 10^s$.

When x is not known (common in many practical applications), significant figures can be determined by considering successive approximations. For a sequence $\{x_i\}_{i=0}^{\infty}$ that converges to x, you can calculate the absolute error at each step $|x_i - x_{i-1}|$.

Exercise

Complete the following table (use `format short` for relative error):

x	$\hat{x}$	Relative Error	Absolute Error	Significant Figures
1.6925	1.69285	–	–	–
23.130	23.129	–	–	–
23.130	23.1299	–	–	–
23.130	23.129999	–	–	–
0.00345	0.00343	–	–	–
0.01008	0.01012	–	–	–
0.01008	0.01002	–	–	–
0.01008	0.0102	–	–	–

Solution

x	$\hat{x}$	Relative Error	Absolute Error	Significant Figures
1.6925	1.69285	$0.21 \cdot 10^{-3}$	$0.4 \cdot 10^{-3}$	4
23.130	23.129	$0.43 \cdot 10^{-4}$	$0.1 \cdot 10^{-2}$	4
23.130	23.1299	$0.43 \cdot 10^{-5}$	$0.1 \cdot 10^{-3}$	5
23.130	23.129999	$0.43 \cdot 10^{-7}$	$0.1 \cdot 10^{-5}$	7
0.00345	0.00343	$0.58 \cdot 10^{-2}$	$0.1 \cdot 10^{-4}$	2
0.01008	0.01012	$0.40 \cdot 10^{-2}$	$0.4 \cdot 10^{-4}$	3
0.01008	0.01002	$0.60 \cdot 10^{-2}$	$0.6 \cdot 10^{-4}$	2
0.01008	0.0102	$0.12 \cdot 10^{-1}$	$0.1 \cdot 10^{-3}$	1

C.4.1 Floating-Point Arithmetic

Natural numbers ($\mathbb{N}$), although infinite, can be represented exactly on a computer within a predefined range of values. For example, a 32-bit integer can represent numbers from 0 to $2^{32} - 1$. Similarly, it can represent signed integers ($\mathbb{Z}$) from -2^{31} to $2^{31} - 1$.

Real numbers ($\mathbb{R}$), on the other hand, are too numerous to be represented exactly within any chosen range. There are at least two possibilities: the first involves fixing the number of digits after the decimal point (fixed-point). The second considers the proper subset $\mathbb{F} \subset \mathbb{R}$ of floating-point numbers:

$$y = \pm m \times \beta^{e-t},$$

where β is the base, t is the precision, and $e \in [e_{\min}, e_{\max}]$ is the exponent. For example, with $\beta = 2$, $t = 3$, $e_{\min} = -1$, and $e_{\max} = 3$, you can represent numbers like:

$$0, 0.25, 0.3125, 0.3750, 0.4375, 0.5, 0.625, 0.750, 0.875,$$
$$1.0, 1.25, 1.50, 1.75, 2.0, 2.5, 3.0, 4.0, 5.0, 6.0, 7.0.$$

The following graph illustrates the spacing of these numbers:

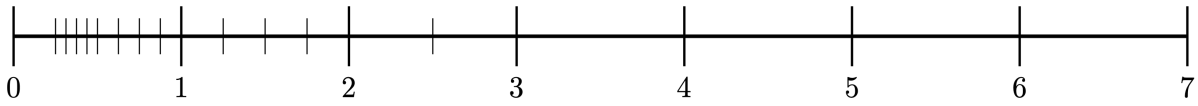


Figure C.8: Graphical representation of floats

The IEEE-754 standard sets values for these parameters in two significant cases: single precision (`float` in C), where $\beta = 2$, $t = 24$, $e_{\min} = -125$, and $e_{\max} = 128$, and double precision (`double` in C), where $\beta = 2$, $t = 53$, $e_{\min} = -1021$, and $e_{\max} = 1024$.

Machine epsilon (ε_{M}) is the smallest positive number such that $1 + \varepsilon_{\mathrm{M}} \neq 1$ in the floating-point system. It represents the distance between 1 and the next representable number in the system. For single precision, ε_{M} is approximately 1.19×10^{-7}, while for double precision, it's approximately 2.22×10^{-16}.

In MATLAB, you can check these values using the `eps` function:

```
>> eps
ans =
   2.2204e-16
```

The limits of representable numbers can also be checked with `realmin` and `realmax`:

```
>> realmax
ans =
  1.7977e+308
>> realmin
ans =
  2.2251e-308
```

As observed from the previous graph, the spacing between numbers in $\mathbb{F}$ is not constant. MATLAB's `eps` command allows you to determine these spacings:

```
>> eps(1)
ans =
   2.2204e-16
>> eps(10)
ans =
   1.7764e-15
>> eps(100)
ans =
   1.4211e-14
>> eps(1000)
```

```
ans =
   1.1369e-13
```

The set $\mathbb{F}$ also includes some exceptional cases:

```
>> 1e400
ans =
   Inf
>> 1e-400
ans =
     0
>> 1/0
ans =
   Inf
>> 0/0
ans =
   NaN
```

- The number `Inf` represents infinity, meaning it's beyond `realmax`. It's not considered an error, and arithmetic operations involving infinity are still valid, although they can often lead to undesirable results.
- When dealing with numbers smaller than `realmin`, they are approximated to 0.
- The situation `NaN` (Not-a-Number) indicates that the result is undefined or has no meaningful value.

Floating-point arithmetic has its limitations, including a lack of associativity for some operations, which can lead to significant errors in some cases. For example, when subtracting two nearly equal numbers, you can experience loss of precision due to the finite precision of the representation:

```
>> x = 1.0e-15;
>> (1+x)-1
ans =
   1.1102e-15
>> (1-1)+x
ans =
   1.0000e-15
```

This issue is known as **cancellation error** and should be avoided whenever possible in numerical computations. The problem is indeed quite serious, and it's a common issue in numerical computations, especially when dealing with small numbers. For example, you have two functions:

$$f(x) = \frac{1 - \cos x}{x^2}, \quad g(x) = \frac{1}{2}\left(\frac{\sin(x/2)}{x/2}\right)^2.$$

Both of these functions are identical. However, when you evaluate them for $x = 1.2 \times 10^{-8}$ in MATLAB, you get different results:

```
>> x = 1.2e-8;
>> (1 - cos(x)) / x^2;
ans =
    0.7710

>> 0.5 * (sin(x / 2) / (x / 2))^2;
ans =
    0.5000
```

The first result is clearly incorrect, and it violates the bounds of the function, which should be between 0 and 0.5. This is a classic example of the problem of numerical precision and the limitations of finite-precision arithmetic. Small values of `x` lead to significant errors due to rounding and truncation.

To address such issues, numerical analysts often use techniques like Taylor series expansions, higher precision arithmetic, or specialized algorithms to improve the accuracy of computations involving small numbers.

It is good practice to avoid cases like these as much as possible, as they are quite common. For example:

$$\begin{aligned} f(x) &= \frac{e^x - 1}{x}, && \text{for } x \approx 1, \\ x_{1,2} &= \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}, && \text{for } b^2 \approx 4ac \text{ and } b \approx \sqrt{b^2 - 4ac}, \\ s_n^2 &= \frac{1}{n-1}\left(\sum_{i=1}^n x_i^2 - \frac{1}{n}\left(\sum_{i=1}^n x_i\right)^2\right) \end{aligned}$$

In the last case, when calculating variance, it is even possible to obtain negative results, which have no mathematical sense.

Exercise (Archimedes' Method for π Approximation)

The strategy used by Archimedes to approximate π involves considering regular polygons inscribed and circumscribed within the unit circle. In fact, if n is the number of sides, the perimeter P_n approaches 2π as $n \to \infty$.

Starting with a hexagon and successively doubling the number of sides, it can be found that as $i \to \infty$, $6 \cdot 2^i \cdot t_i$ approaches π, where t_i satisfies the following relation:

$$t_0 = \frac{1}{\sqrt{3}}, \qquad t_{i+1} = \frac{\sqrt{t_i^2 + 1} - 1}{t_i}.$$

Here are the tasks related to this approximation:

1. Implement this algorithm in MATLAB and compare the approximation with π provided by the `pi` constant. What do you observe?
2. Replace the recursive formula with its equivalent:

$$t_0 = \frac{1}{\sqrt{3}}, \qquad t_{i+1} = \frac{t_i}{\sqrt{t_i^2 + 1} + 1}.$$

Comment on any differences observed.
Note: Use `format long` to display the differences.

Solution

The value given by Matlab for π is:

```
>> format long
>> pi
ans =
   3.141592653589793
```

We compute now π with method 1. `pia` and method 2. `pib`:

```
>> format long;
>> n = 30;
>> ta = 1/sqrt(3);
>> tb = 1/sqrt(3);
>> for i = 1:n
     % first method
     ta = (sqrt(ta^2+1)-1)/ta;
     pia = 6*2^i*ta;
     % second method
     tb = tb/(sqrt(tb^2+1)+1);
     pib = 6*2^i*tb;
>> end
>> fprintf('Method 1, pi = %f, err = %e\n', pia, abs(pia-pi));
>> fprintf('Method 2, pi = %f, err = %e\n', pib, abs(pib-pi));
```

We see that the first method suffers of cancellation errors.

Exercise (Polynomial Expansion)

Consider the polynomial $p(x) = (x-1)^7$ for $x \in [0.998, 1.012]$. Expand the polynomial using the binomial formula and compare it to the original unexpanded polynomial. Comment on any differences.

Solution

Expanding we have

$$p(x) = x^7 - 7x^6 + 21x^5 - 35x^4 + 35x^3 - 21x^2 + 7x - 1.$$

Now we compare

```
>> x = 0.998:0.0001:1.012;
>> plot(x, x.^7-7*x.^6+21*x.^5-35*x.^4+35*x.^3-21*x.^2+7*x-1, ...
        x, (x-1).^7, 'LineWidth', 2.0);
>> grid on;
>> legend('Expanded', 'Original');
```

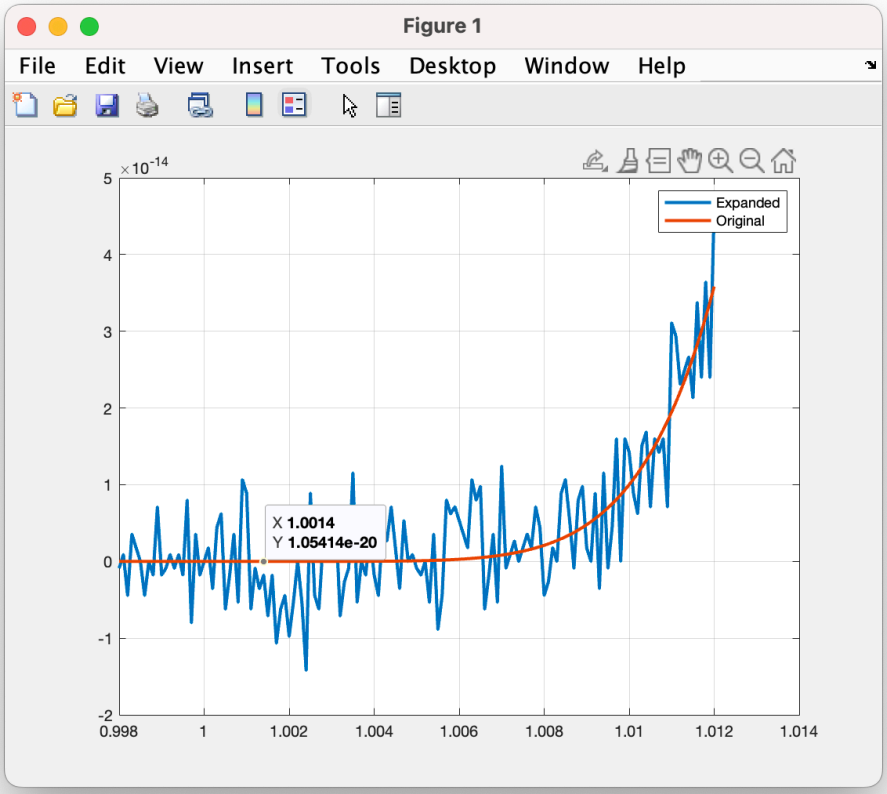


Figure C.9: image