Modeling and Simulation — Lesson 1
The Limits to Growth: Basic Population Models

Václav B. Petrák

Faculty of Biomedical Engineering
Czech Technical University in Prague

April 17, 2025

Content

Introduction

Contacts

Lecturer: Václav B. Petrák
Email: vaclav.petrak@fbmi.cvut.cz
Phone/WhatsApp: +420 725 878 390

Consultations:

- In person: after individual request. Please mail me or text me.
- In person: on Friday 2nd on May 2025 for project-related problems
- By email any time

Absence Policy

- **Advance Notice Required:** All absences must be excused in advance whenever possible.
- **Communication Procedure:** Contact me as soon as possible by email with date of absence a reason for the absence.
- **Responsibility for Missed Work:** You will be are responsible for making up missed assignments, by following the content of course materials.

Lectures

- We will have 5 topics in modeling and simulation, each for one day.
- The last lecture will be focused on presentation of your work.

Classification

There is no exam.

Your grade will be based on your independent work.

1. **Journal Club Presentation**
2. **Capstone Project**
3. **Code Review**

Journal Club is a discussion of research papers.

A journal club is a group who meet to critically evaluate articles in the academic literature. Journal clubs:

- Encourages critical thinking.
- Helps in understanding the studied concepts and methods.
- Helps keep up with the latest research.
- Improves communication and presentation skills.
- Presentation simulates experience of a scientist

The main purpose in this course is to explore how methods we learned are applied in current research

Your Task for Journal Club

- You will be assigned a research paper related to the topic of your capstone project.
- Your task is to read the paper and prepare a 15-minute presentation, followed by a 5-minute discussion, to explain the paper to your fellow students.
- The task will be graded by the lecturer and your classmates.

The Presentation Will Cover

- **High-level Overview:** Provide a background of the research topic and the main objectives of the paper.
- **Methods:** Give overview of the methodology used in the paper.
- **Results:** Highlight the key results and conclusions of the paper.
- **Implications and Applications:** Discuss the practical implications of the research and its applications in real-world situations.
- **Limitations:** Discuss any limitations of the paper.
- **Discussion:** Engage in a 5-minute discussion with other students, encouraging questions and opinions.

Grading Criteria for Journal Club Presentation

- **Presentation (35%)**
  - **Design:** The presentation is well-organized and easy to follow. The slides are well-integrated into the flow of the presentation.
  - **Timing:** The presenter adhered to the 15-minute presentation format.
  - **Delivery:** Clear and confident speech with appropriate pacing.

Grading Criteria for Journal Club Presentation

- **Presentation (35%)**
  - **Design:** The presentation is well-organized and easy to follow. The slides are well-integrated into the flow of the presentation.
  - **Timing:** The presenter adhered to the 15-minute presentation format.
  - **Delivery:** Clear and confident speech with appropriate pacing.
- **Content of the presentation (35%)**
  - **Introduction:** The introduction provides a high-level overview of the topic. The presenter explains the research motivation (why) and the research question (what).
  - **Methods:** The presenter explained the experimental approach used by the authors. Unfamiliar methods are explained without going into excessive detail.
  - **Results:** The presenter showed and explained the key results from the paper.

Grading Criteria for Journal Club Presentation

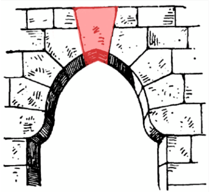
- **Critical Thinking (15%):** The presenter provides a critical evaluation of the paper that may consist of:
  - Critique of methods or experimental design.
  - Ideas for improvement and further research.
  - Discussion of how the results relate to the research question.
  - Significance of the results.

Grading Criteria for Journal Club Presentation

- **Critical Thinking (15%):** The presenter provides a critical evaluation of the paper that may consist of:
  - Critique of methods or experimental design.
  - Ideas for improvement and further research.
  - Discussion of how the results relate to the research question.
  - Significance of the results.
- **Questions (15%):** The presenter is able to respond convincingly to audience questions.

Capstone project

- You will work on independent project that builds on or extends course content.
- Output of your work will be a presentation, presented on Friday 16th of May 2025 — take a maximum effort to be present
- You will also prepare an abstract on on the topic.



*A **capstone** (or keystone) is the wedge-shaped stone at the apex of an arch. **It is the final piece placed during construction and locks all the stones into position**, allowing the arch to bear weigh.*

Avoid using AI to solve entire exercises

- Consider a math problem:
  - *Given that two trains are traveling on parallel tracks in opposite directions with speeds of 56 km/h and 52 km/h, respectively, and lengths of 136 m and 242 m, respectively, after what duration will they cross each other?*
- The primary goal is **not** to find the answer but to practice your reasoning and calculation skills.
- Similarly, the aim of modeling and simulation exercises is to learn and practice new skills.
- Avoid using generative AI like ChatGPT or Gemini to solve the **entire problem** for you.
  - It spoils learning process for you
  - It may disrupt pacing of the lesson.

Using AI During Lessons

During lessons, you may use AI to:

- Ask clarifying questions.
- Debug your code: after you've made your own attempts.
- Get ideas for further work.

Let's discuss what is a fair use of generative AI in the capstone project

Model is a representation of a **system**

- A model is a representation of a **system** that enables us investigate its properties and **predict** future outcomes.
    - A **system** is a set of components that interact with each other within a boundary to function as a whole: Solar system, rabbit and fox populations, a microscope
- In modeling we aim to identify components, relationships and behavior to predict system dynamics.

Model is a representation of a **system**

- A model is a representation of a **system** that enables us investigate its properties and **predict** future outcomes.
  - A **system** is a set of components that interact with each other within a boundary to function as a whole: Solar system, rabbit and fox populations, a microscope
- In modeling we aim to identify components, relationships and behavior to predict system dynamics.
- Modeling always requires simplification (abstraction)

Example of a model: SIR (susceptible, infectious, and recovered) model for dynamics of infectious diseases

Model is a representation of a **system**

- A model is a representation of a **system** that enables us investigate its properties and **predict** future outcomes.
    - A **system** is a set of components that interact with each other within a boundary to function as a whole: Solar system, rabbit and fox populations, a microscope
- In modeling we aim to identify components, relationships and behavior to predict system dynamics.
- Modeling always requires simplification (abstraction)
- **A mathematical model uses mathematical equations to describe a system.**

SIR model equations:

$$\frac{dS}{dt} = -\beta SI$$

$$\frac{dI}{dt} = \beta SI - \gamma I$$

$$\frac{dR}{dt} = \gamma I$$

All models are approximations. Assumptions, whether implied or clearly stated, are never exactly true.

All models are wrong, but some models are useful.

So the question you need to ask is not *Is the model true?*

It never is.

But *Is the model good enough for this particular application?*

— George E. P. Box

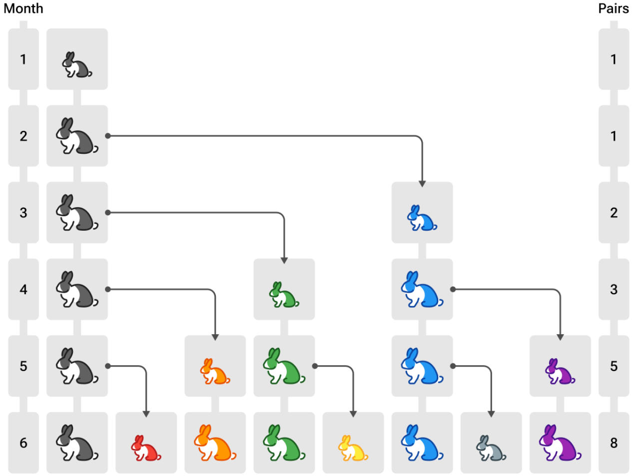
Section 1: Fibonacci Sequence

A very simple population model

Italian mathematician Leonardo Fibonacci (1170 - 1250) considered following conditions when calculated the number of rabbits born over the course of one year:

- a single newly born pair of rabbits (one male, one female) are put in a field;
- rabbits are able to mate at the age of one month so that at the end of its second month a female can produce another pair of rabbits;
- rabbits never die and a mating pair always produces one new pair (one male, one female) every month from the second month on.

Graphical representation of first 6 months



Fibonacci sequence

The problem leads to a series of numbers called the **Fibonacci sequence**:

$$1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, \ldots$$

The first two numbers are given, and subsequent numbers are found by adding the two numbers before them. Thus, we have the following recursive definition (a recursive definition defines elements in a set in terms of other elements in the set) of Fibonacci numbers:

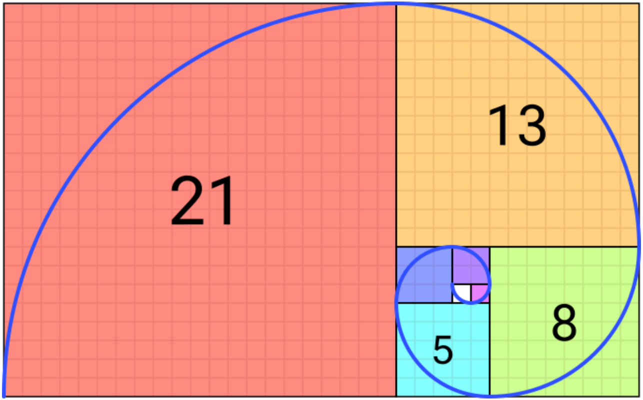
$$F_1 = F_2 = 1$$
$$F_n = F_{n-1} + F_{n-2}$$

The ratio of two consecutive Fibonacci numbers converges to a number known as the **golden ratio**:

$$\psi = \frac{1+\sqrt{5}}{2} \approx 1.618\ldots \tag{1}$$

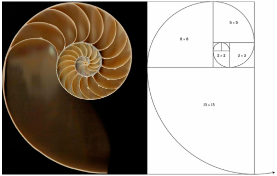
Fibonacci sequence

A spiral created by drawing circular arcs connecting the opposite corners of squares of A tiling with squares whose side lengths are successive Fibonacci numbers



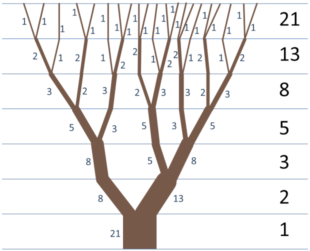
Fibonacci sequence in nature

- **Spiral Patterns**: The growth chambers of a nautilus shell create a distinctive spiral that aproximately follows the Fibonacci sequence and the golden ratio.



Fibonacci sequence in nature

- **Spiral Patterns**: The growth chambers of a nautilus shell create a distinctive spiral that aproximately follows the Fibonacci sequence and the golden ratio.
- **Branches**: Tree branches sometimes follow Fibonacci sequence. Main branch splits into two, then one of those branches splits while the other stays dormant.

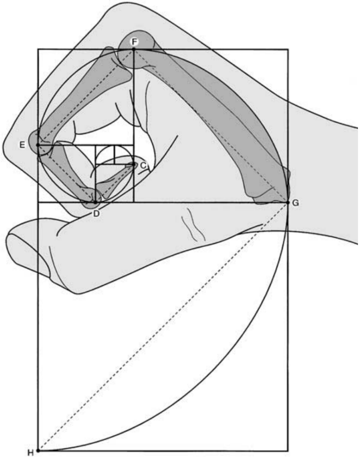


Fibonacci sequence in nature

- **Spiral Patterns**: The growth chambers of a nautilus shell create a distinctive spiral that aproximately follows the Fibonacci sequence and the golden ratio.
- **Branches**: Tree branches sometimes follow Fibonacci sequence. Main branch splits into two, then one of those branches splits while the other stays dormant.
- **Flower Petals:** The number of petals on many flowers corresponds to Fibonacci numbers: lilies (3 petals), buttercups (5 petals), and daisies (often 34 or 55 petals).

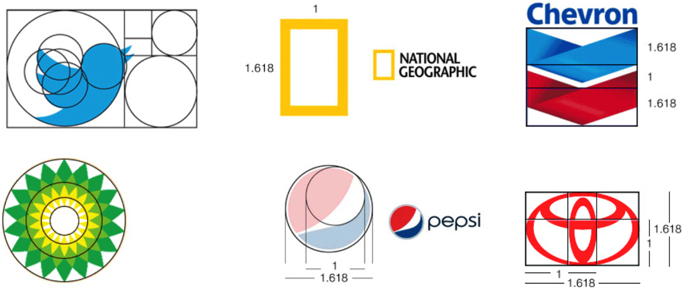
Fibonacci sequence in nature

- **Spiral Patterns**: The growth chambers of a nautilus shell create a distinctive spiral that aproximately follows the Fibonacci sequence and the golden ratio.
- **Branches**: Tree branches sometimes follow Fibonacci sequence. Main branch splits into two, then one of those branches splits while the other stays dormant.
- **Flower Petals:** The number of petals on many flowers corresponds to Fibonacci numbers: lilies (3 petals), buttercups (5 petals), and daisies (often 34 or 55 petals).
- **Why?** Fibonacci patterns in nature are likely related to optimal growth and packing efficiency.
- **While the Fibonacci sequence appears in nature, it is not a universal rule.**



Fibonacci sequence is extensively used in arts

Golden ratio in famous logos



Fibonacci sequence tasks as Matlab refresher

We will work Fibonacci sequence to go review some basic concepts in Matlab. Basic experience with programming is expected.

We will cover

- Loop control statements
- Vector and matrix indexing
- Functions
- Plotting

Task 1.1: Loop control statements

With loop control statements, you can repeatedly execute a block of code. There are two types of loops:

1. **for** statements loop a specific number of times, and keep track of each iteration with an incrementing index variable.
2. **while** statements loop as long as a condition remains true.

Your tasks:

1. Generate first 20 elements of Fibonacci sequence with help of for loop
2. Generate all number of Fibonacci sequence smaller than 500 using while loop

Note: Save generated numbers to a vector, we will need them for the next task

Solution 1.1.1: First 20 Fibonacci numbers

Generate first 20 elements of Fibonacci sequence using **for** loop

```
% require element count
n = 20;

% 1st and 2nd elements are 0 and 1
fib_seq = [1,1];

for i = 3:n
    fib_seq(i) = fib_seq(i-1) + fib_seq(i-2);
end

disp(fib_seq)
```

Solution 1.1.2: Fibonacci numbers smaller than 500

Generate all number of Fibonacci sequence smaller than 5000 using **while** loop

```
fib_seq = [1,1];
max_value = 5000;

%set index to 2
i = 2;

while (fib_seq(i) + fib_seq(i-1)) < max_value
    fib_seq(i+1) = fib_seq(i) + fib_seq(i-1);
    i = i + 1;
end

disp(fib_seq);
```

Task 1.2: Vector Indexing

Vector indexing allows you to access and manipulate specific elements within a vector. Here are the most common techniques:

- **Linear Indexing**
  - Access elements by their position in the vector: `x(3)` gets the third element of x; `x(end)` the last element.
  - Use ranges with the colon operator: `x(2:5)` gets elements 2 to 5 of vector x
- **Logical Indexing**:
  - Create a logical array the same size as your vector to get elements where the condition is true: `x(x > 10)` extracts elements greater than 10).

Task 1.2: Vector Indexing

Your tasks

- Divide 6^{th} and 5^{th} element of vector with Fibonacci numbers.
- Divide last and 2nd last element of vector with Fibonacci numbers.
- Get all Fibonacci numbers smaller than 50
- Get all Fibonacci numbers between 50 and 500
- With help of indexing and `mod` function replace all even Fibonacci numbers in the vector with 0.

Task 1.2: Solution

- Divide 6th and 5th element of vector with Fibonacci numbers.
  `fib_seq(6)/ fib_seq(5)`
- Divide last and 2nd last element of vector with Fibonacci numbers.
  `fib_seq(end)/ fib_seq(end-1)`
- Get all Fibonacci numbers smaller than 50
  `fib_seq(fib_seq < 50)`
- Get all Fibonacci numbers between 50 and 500
  `fib_seq(fib_seq > 50 & fib_seq < 500)`
- With help of indexing and `mod` function replace all even Fibonacci numbers in the vector with 0.
  `fib_seq(mod(fib_seq,2)~= 0)= 0`

Matrix indexing is similar to vector indexing

- Define matrix
  `M = [1,2,3; 4,5,6; 7,8,9]`
- Extract the element in row 2, column 3
  `M(2,3)` One or both of the row and column subscripts can be vectors:
  `M(1:3,2:3)`
- A `":"` is shorthand notation for 1:end
  `M(:,2:3)`
- When put one subscript, MATLAB treats M as if its elements were strung out in a long column.
  `M(8)`
- You use a logical array as subscript. MATLAB extracts true elements in the form of a column vector
  `N = [1.3, 2.0, 3.8; 4.8, 5.0, 6.0]`
  `N(mod(N,1)= 0)`
  `N(N>3)`

Task 1.3: Functions background

- Functions are reusable blocks of code that take inputs, perform calculations, and return outputs. They are essential for code organization and modularity.
- Functions in Matlab can be defined in a separate script or at the end of a script file

```
% call function
my_circile = circle_area(5)

% declare function
function area = circle_area(radius)
    pi = 3.14159; % approx. value of pi
    area = pi * radius^2;
end
```

Task 1.3: Functions

Your task

1. Writer function to return vector with generated n Fibonacci numbers
2. Write function to return random Fibonacci number
3. Check if the number is Fibonacci

Think about your solution fist; after that fell free to search for hints online.

Solution 1.3.1: Fibonacci generator

- We will reuse our previous Task 1.1 code.
- The subsetting is important for n values 1 or 2.
- **What are limitation of this code?**

```
function fib_seq = generate_fibonacci(n)
fib_seq = [1, 1];

for i = 3:n
    fib_seq(i) = fib_seq(i-1) + fib_seq(i-2);
end

fib_seq = fib_seq(1:n); %subset the result
end
```

Solution 1.3.2: Random Fibonacci number

```
function random_fib = random_fibonacci(n)
    fib_seq = [1, 1];
    for i = 3:n
        fib_seq(i) = fib_seq(i - 1) + fib_seq(i - 2);
    end

    fib_seq = fib_seq(1:n); %subsetting

    % Choose a random index within the range
    random_index = randi([1, n]);  % For integer index

    % Return the random Fibonacci number
    random_fib = fib_seq(random_index);
end
```

Solution 1.3.3: Check if number is Fibonacci

One possible approach : Check if the last generated number before overstepping treshold is Fibonacci number.

```
function is_fibonacci = check_fibonacci(input)
    % initialize the first two Fibonacci numbers
    a = 1; b = 1;

    % keep generating Fibonacci numbers until
    % we exceed input value
    while b < input
        memomory = b; % memorize b
        b = a + b;
        a = memomory;  % a is recalled from memory
    end

    % true if we found an exact match
    is_fibonacci = (b == input);
end
```

Task 4: Plotting

Plotting allows you to represent your data graphically, which is always a good idea. MATLAB offers a wide range of plots. Here we cover only basic line plot.
Your tasks:

1. Plot the first 15 elements of Fibonacci sequence
2. Calculate and plot ratio of all `n(i+1)/n(i)` in the first 15 elements of Fibonacci sequence.

Solutions 1.4

1. Plot the first 15 elements of Fibonacci sequence:

```
fib_seq_15 = fib_seq(1:15)
plot(fib_seq_15)
```

2. Calculate and plot ratio of all `n(i+1)/n(i)` in the first 15 elements of Fibonacci sequence.

```
ratio = fib_seq_15(2:end)./fib_seq_15(1:end-1);
display(ratio)
plot(ratio, '-s')
```

Note: Period "." signifies element-wise operation

Section 2: Malthusian Growth Model

Malthusian Growth Model

- The Malthusian growth model is a mathematical model of population growth.
- It is essentially exponential growth of population.
- The model is named after Thomas Malthus, who wrote *An Essay on the Principle of Population* (1798), one of the first and most influential books on population.

Thomas Robert Malthus 1766 - 1834, an English economist and scholar.

Key assumption of Thomas Malthus

- **Population grows exponentially when resources are abundant.**
- Mathus further thought that Agricultural production increases linearly.
- Population growth is limited by available resources and can be reduced by famine, disease, or war.

Malthusian Catastrophe

A situation where population exceeds the agricultural production of its environment. Results in severe consequences like poverty, famine, disease, and war. Malthus believed a Malthusian catastrophe was inevitable unless population growth was controlled

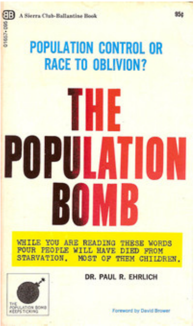
Malthus, Darwin and Natural Selection

- Before reading Malthus, Darwin had thought that living things reproduced just enough individuals to keep populations stable
- Malthus helped him to realize that as populations bred beyond their means, advantageous traits become more prevalent due to competition for resources
- This led to the development of the theory of natural selection.

> *In October 1838, that is fifteen months after I had begun my systematic enquiry, I happened to read for amusement Malthus on Population, and being well prepared to appreciate the struggle for existence which everywhere goes on from long-continued observation of the habits of animals and plants,* ***it at once struck me that under these circumstances favourable variations would tend to be preserved, and unfavourable ones to be destroyed.*** *The result of this would be the formation of a new species.*
>
> — *Charles Darwin*

Neo-Malthusianism:

- A revival of Malthusian concerns in the 20th century, often focused on environmental limits.
- Neo-Malthusians advocate for population control due to concerns about overpopulation, resource depletion, and environmental degradation.
- Widely discussed in 1960s, examples is Paul Ehrlich's book *The Population Bomb* (1968)
- Their influence can be seen in family planning initiatives and advocacy for sustainability.
- Modern environmentalism acknowledges the impact of population growth on resource use, pollution, and environmental strain.



Malthusian Growth Model Equation

Change in population dP/dt is proportional to the current population size P at time t. Parameter r is the intrinsic rate of increase. It is difference between birth rate and death rate.

$$\frac{dP}{dt} = rP$$

To solve the differential equation, we separate the variables. We move terms with P to one side and terms with t to the other side.

$$\frac{dP}{P} = rdt$$

Malthusian Growth Model Equation

We integrate both sides of the equation. Integration introduces a constant of integration, C.

$$\int \frac{dP}{P} = \int r dt$$
$$\ln |P| = rt + C$$

We exponentiate both sides. We remove the absolute value as the population size always is positive,

$$e^{\ln |P|} = e^{rt+C}$$
$$P = e^{rt+C}$$

The integration constant C represents initial population P_0. Substitution leads to the final explicit form

$$P(t) = P_0 e^{rt}$$

Numerical vs explicit solution

- Numerical and explicit solutions are two methods used to solve mathematical problems in modeling and simulation.

- **Explicit solution** is based on an mathematical **exact** solution using algebraic equations.
  - Provide a precise solution
  - Often allow for a deeper understanding of the underlying mathematical principles
  - It can be very difficult or impossible to find explicit solution for more complex problems

Numerical vs explicit solution

- Numerical and explicit solutions are two methods used to solve mathematical problems in modeling and simulation.

- **Numerical solution** use **iterative methods to approximate** a solution using numerical calculations.
  - Allow solving a wider range of problems.
  - Useful for problems that are difficult or impossible to solve explicitly
  - Solutions are only approximate
  - Solutions can be affected by calculation errors
  - Difficult to validate without explicit solutions or real-world data for comparison

Numerical vs explicit solution for Malthusian growth

explicit Solution
Calculation using formula:

$$P(t) = P_0 e^{rt}$$

where

- P_0 is the initial population at time 0
- r is the growth rate
- t is the time

Equation for numerical solution:

$$\frac{dP}{dt} = rP$$

Solution by Euler's Method:

1. **Discretization:**

$$t_i = i\Delta t$$

2. **Iteration:**

$$P(t_{i+1}) = P(t_i) + rP(t_i)\Delta t$$

where

- $P(t_i)$ is the population at time t_i
- r is the growth rate
- Δt is the time step

Task 2.1: Population at discrete times

Calculate population at certain times of interest using explicit formula.

- Initial population size: 2.5 million
- Yearly growth rate: 2.7%
- Equation $P(t) = P_0 e^{rt}$
- Time in years :
  - 1
  - 5
  - 7.6
  - 25.78
- Note: Population size and growth rate correspond to population of Senegal in 1950.

Solution 2.1: Population at discrete times

```
% Clean up
    clear, clc

% Population parameters
    P0 = 2.5;    % Intial popoulation
    r = 0.027;   % Growt rate 2.7%

% Years of interest
    t_task1 = [1, 5, 7.5, 25.78];

% Results
    disp("Years of interest:")
    disp(t_task1)
    disp("Caclulated population size:")
    disp(P0*exp(r*t_task1))
```

Solution 2.1: Population at discrete times

- The results show population size at times of 1, 5, 7.5, and 25.78 years.
- The explicit solution allows us to easily calculate population predictions at any given time.

```
Years of interest:
    1.0000    5.0000    7.5000   25.7800
Calculated population size:
    2.5684    2.8613    3.0612    5.0146
```

Note: this line leads to the same result, but it is much better to maintain your code well organized.

```
2.5*exp(0.027*[1,5,7.5,25.78])
```

Task 2: Continuous modeling with explicit solution

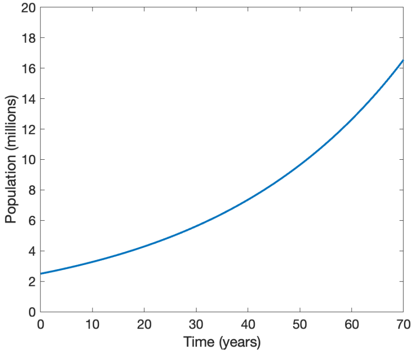
- The population has same parameters as in Task 1
- Calculate population for the next 70 years
- The time step is 1 year.
- Plot the results.

Solution 2.2: Continuous modeling with explicit solution

```
% Population parameters
    P0 = 2.50; % Intial popoulation
    r = 0.027; % Growt rate 2.7%

% Timescale
    t_start = 0;  t_end = 70;  t_diff = 1;

% Years of interest
    t_expl= t_start:t_diff:t_end;

 % Results
    P_expl= P0*exp(r*t_expl);

% Plot
    plot(t_expl, P_expl, LineWidth=2)
    xlabel("Time (years)");
    ylabel("Population (millions)"),
    ylim([0, 20])
```

Solution 2.2: Continuous modeling with explicit solution

Population increase exponentially.



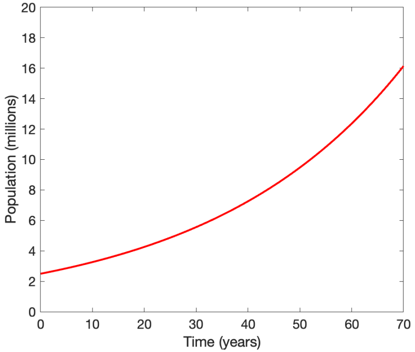
Task 2.3: Numerical approach

- Use numerical approach to calculate population for 0 to 70 years
- The time step should be 1 year.
- Plot the result.

Solution 2.3: Numerical approach

```
% Population parameters
    P0 = 2.50; r = 0.027;
% Timescale
    t_start = 0;  t_end = 70;  t_diff = 1;
% Initial time and population
    t_num = t_start;  P_num = P0;
% Numerical caclulation loop
    for i = 1:t_end/t_diff
       % Calulate new step
          P_new = P_num(i) + r*P_num(i)*t_diff;
          t_new = t_num(i) + t_diff;
       % Append to existing vector
          P_num = [P_num, P_new];
          t_num = [t_num, t_new];
    end
 % Plot
    plot(t_num, P_num, Color="red")
    xlabel("Time (years)");
    ylabel("Population (millions)"),
    ylim([0, 20])
```

Solution 2.3: Numerical approach



Task 2.4: Difference in solutions

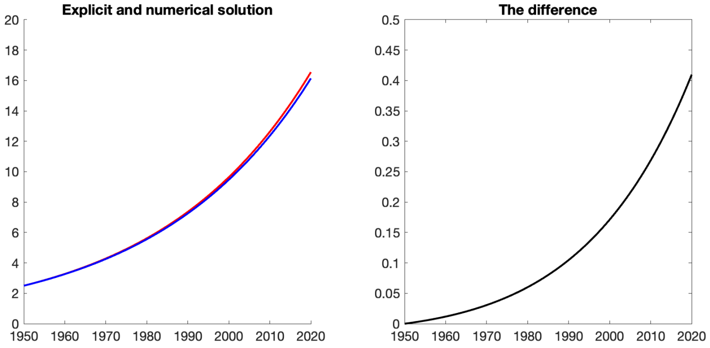
- Create plot that shows results from explicit and numerical approach and their difference.
- Add start year of 1950 to the plot to have correct timeline
- Compare the difference for time step of 1 year, 1 month and 1 day.

Solution 2.4: The difference in solutions

```
% MODELING PARAMETERS
    P0 = 2.50;                      % Intial popoulation in milions
    r = 0.027;                      % Population Growth rate 2.7%
    start_year = 1950;              % For plotting
    t_start = 0;  t_end = 70;       % Start and end years
    t_diff = 1;                     % Step (1/12 is one month, 1/365 is one day)

% EXPLICIT SOLUTION
    t_expl= t_start:t_diff:t_end;
    P_expl= P0*exp(r*t_expl);

% NUMERICAL SOLUTION
   t_num = t_start;  P_num = P0;
    for i = 1:t_end/t_diff
           P_num = [P_num, P_num(i) + r*P_num(i)*t_diff;];
           t_num = [t_num, t_num(i) + t_diff];
    end

% PLOTTING
tiledlayout (1, 2)
nexttile        % Tile with explicit and numerical solution
    hold on
    plot(t_expl + start_year, P_expl , Color = "red")
    plot(t_num + start_year, P_num , Color = " blue")
    title("Explicit and numerical solution")
    ylim([0,20]); xlim([1950,2020])
    hold off
nexttile        % Tile with the difference
    plot(t_expl + start_year, P_expl - P_num, Color =" black")
    title ("The difference")
    ylim([0,0.5]); xlim([1950,2020])
```

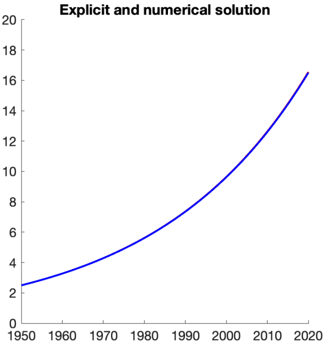
Solution 2.4: The difference in solutions

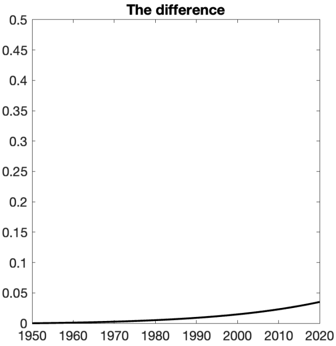
Step size of 1 year (70 iterations) leads to substantial error.



Solution 2.4: The difference in solutions

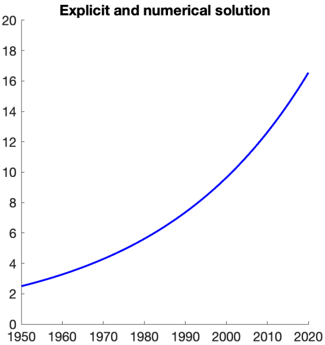
Step size of 1 month (840 iterations) reduces the error

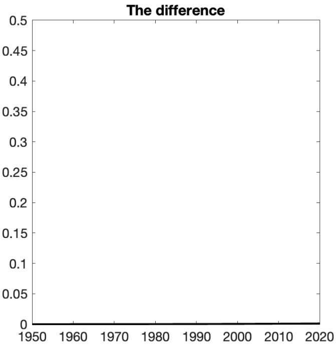




Solution 2.4: The difference in solutions

Step size of 1 day (25 550 iterations) leads to negligible error.





Task 2.5: Real world data

Compare prediction modeled by the explicit formula with real data of Senegalese population

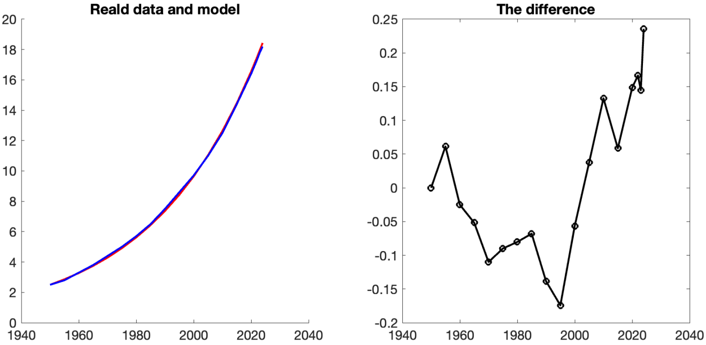
```
% Growth rate
   r = 0.027;

% Start year
   start_year = 1950;

% Time axis
    t_sen    = [0,   5, 10, 15, 20, 25, 30, 35, 40,  ...
              45, 50, 55, 60, 65, 70, 72, 73, 74];

% Senegalese population
    P_sen = [ 2.5,   2.8,   3.3,   3.8,   4.4,   5.0, 5.7,   ...
              6.5,   7.5,   8.6,   9.7, 11.0, 12.5,14.4,   ...
```

Solution 2.5: Real world data

```
% Growth rate
  r = 0.027;

% Start year
  start_year = 1950;

% Time axis
   t_sen    = [0,  5, 10, 15, 20, 25, 30, 35, 40,  ...
              45, 50, 55, 60, 65, 70, 72, 73, 74];

% Senegalese population
   P_sen = [ 2.5,  2.8,  3.3,  3.8,  4.4,  5.0, 5.7,  ...
             6.5,  7.5,  8.6,  9.7, 11.0, 12.5,14.4,  ...
            16.4, 17.3, 17.8, 18.2];

% EXPLICIT SOLUTION
   P_expl= P_sen(1)*exp(r*(t_sen)); %initial populatin is the first value

% PLOTTING
   tiledlayout(1, 2)
   nexttile
       hold on
       plot(t_sen + start_year, P_expl, Color = "red")
       plot(t_sen + start_year, P_sen,  Color = " blue")
       title("Reald data and model")
       ylim([0,20]);
       hold off
   nexttile
       plot(t_sen + start_year, P_expl - P_sen, '-o', Color =" black ")
       title ("The difference")
```

Solution 2.5: Real world data

Even a very simple model can be capture reality very well.



Task 2.6: Real world data — USA

- Compare prediction modeled by the explicit formula for USA with real data.
- Adjust time span start year and the initial population
- Set growth rate to 2.75%.

```
% Growth rate
   r = 0.0275;

% Start year
   start_year = 1790;

% Time axis
    t_usa = 0:10:230;

% US population
   P_usa = [  3.9,    5.3,    7.2,    9.6,   12.9,   17.1,   23.2,   31.4, ...
             38.6,   50.2,   63.0,   76.2,   92.2,  106.0,  123.2,  132.2, ...
            151.3,  179.3,  203.3,  226.5,  248.7,  281.4,  308.7,  331.4];
```

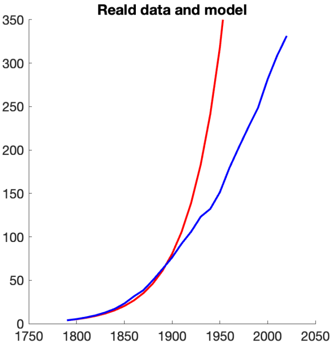
Solution 2.6: Real world data — USA

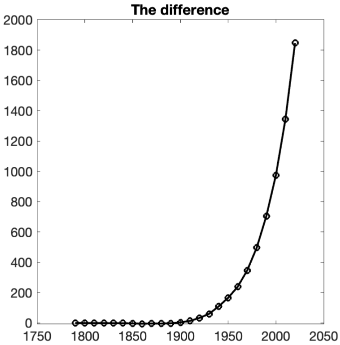
```
% Growth rate
   r = 0.0275;

% Start year
   start_year = 1790;

% Time axis
    t_usa = 0:10:230;

% US population
   P_usa = [  3.9,   5.3,   7.2,   9.6,  12.9,  17.1,  23.2,  31.4, ...
             38.6,  50.2,  63.0,  76.2,  92.2, 106.0, 123.2, 132.2, ...
            151.3, 179.3, 203.3, 226.5, 248.7, 281.4, 308.7, 331.4];

% EXPLICIT SOLUTION
    P_expl= P_usa(1)*exp(r*(t_usa)); %initial populatin is the first value

% PLOTTING
    tiledlayout(1, 2)
    nexttile
        hold on
        plot(t_usa + start_year, P_expl, Color = "red")
        plot(t_usa + start_year, P_usa,  Color = " black")
        title("Real data and model")
        ylim([0,350]);
        hold off
    nexttile       % Tile with the difference
        plot(t_usa + start_year, P_expl - P_usa, '-o', Color =" black ")
        title ("The difference")
```

Solution 2.6: Real world data — USA

Model is reasonably precise for the first one hundered years, then it diverges.

Why?





Section 3: Logistics Growth Model

Logistic growth

The **logistic growth model represents how populations change over time when resources imitated**. The logistic model incorporates a carrying capacity—the maximum population size a specific environment can sustainably support.

Logistic growth

The **logistic growth model represents how populations change over time when resources imitated**. The logistic model incorporates a carrying capacity—the maximum population size a specific environment can sustainably support.

Key Concepts

- **Carrying Capacity (K)**: A ceiling that limits population growth due to limited resources (food, space, water, etc.)

Logistic growth

The **logistic growth model represents how populations change over time when resources imitated**. The logistic model incorporates a carrying capacity—the maximum population size a specific environment can sustainably support.

Key Concepts

- **Carrying Capacity (K)**: A ceiling that limits population growth due to limited resources (food, space, water, etc.)
- **Intrinsic Growth Rate (r)**: The rate at which a population would grow if there were unlimited resources.

Logistic growth

The **logistic growth model represents how populations change over time when resources imitated**. The logistic model incorporates a carrying capacity—the maximum population size a specific environment can sustainably support.

Key Concepts

- **Carrying Capacity (K)**: A ceiling that limits population growth due to limited resources (food, space, water, etc.)
- **Intrinsic Growth Rate (r)**: The rate at which a population would grow if there were unlimited resources.
- **S-Shaped Curve**: The logistic growth model plots as an S-shaped curve. Growth is initially exponential, then slows as it approaches the carrying capacity.

Equation of logistic growth

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Equation of logistic growth

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Iterative Form:

$$P(t_{i+1}) = P(t_i) + rP(t_i)\left(1 - \frac{P_t}{K}\right)\Delta t$$

Equation of logistic growth

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Iterative Form:

$$P(t_{i+1}) = P(t_i) + rP(t_i)\left(1 - \frac{P_t}{K}\right)\Delta t$$

Explicit solution

$$P(t) = \frac{K}{1 + \left(\frac{K - P_0}{P_0}\right)e^{-rt}}$$

Equation of logistic growth

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Iterative Form:

$$P(t_{i+1}) = P(t_i) + rP(t_i)\left(1 - \frac{P_t}{K}\right)\Delta t$$

Explicit solution

$$P(t) = \frac{K}{1 + \left(\frac{K - P_0}{P_0}\right)e^{-rt}}$$

where

- $P(t_i)$ is the population at time t_i

Equation of logistic growth

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Iterative Form:

$$P(t_{i+1}) = P(t_i) + rP(t_i)\left(1 - \frac{P_t}{K}\right)\Delta t$$

Explicit solution

$$P(t) = \frac{K}{1 + \left(\frac{K - P_0}{P_0}\right)e^{-rt}}$$

where

- $P(t_i)$ is the population at time t_i
- P_0 is initial population

Equation of logistic growth

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Iterative Form:

$$P(t_{i+1}) = P(t_i) + rP(t_i)\left(1 - \frac{P_t}{K}\right)\Delta t$$

Explicit solution

$$P(t) = \frac{K}{1 + \left(\frac{K - P_0}{P_0}\right)e^{-rt}}$$

where

- $P(t_i)$ is the population at time t_i
- P_0 is initial population
- r is the growth rate

Equation of logistic growth

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Iterative Form:

$$P(t_{i+1}) = P(t_i) + rP(t_i)\left(1 - \frac{P_t}{K}\right)\Delta t$$

Explicit solution

$$P(t) = \frac{K}{1 + \left(\frac{K - P_0}{P_0}\right)e^{-rt}}$$

where

- $P(t_i)$ is the population at time t_i
- P_0 is initial population
- r is the growth rate
- K is carrying capacity

Equation of logistic growth

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Iterative Form:

$$P(t_{i+1}) = P(t_i) + rP(t_i)\left(1 - \frac{P_t}{K}\right)\Delta t$$

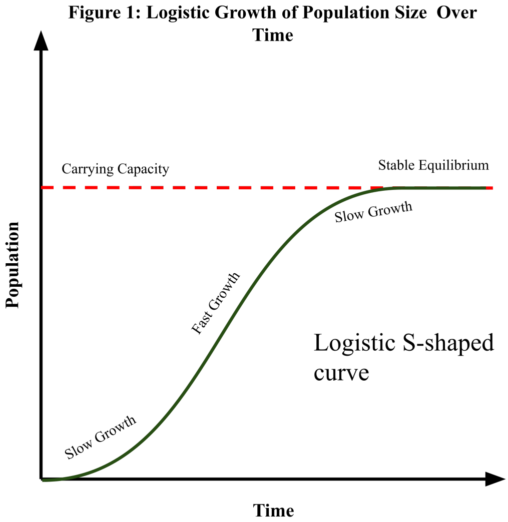
Explicit solution

$$P(t) = \frac{K}{1 + \left(\frac{K - P_0}{P_0}\right)e^{-rt}}$$

where

- $P(t_i)$ is the population at time t_i
- P_0 is initial population
- r is the growth rate
- K is carrying capacity
- t is time, Δt is the time step

Logistic growth of population over time



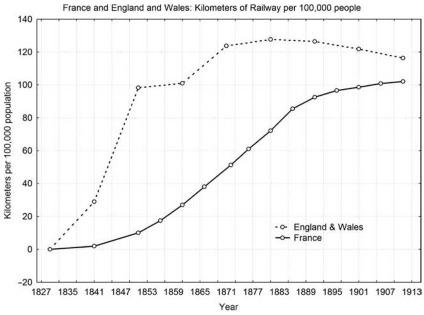
Carrying capacity is the amount of organisms within a region that the environment can support sustainably

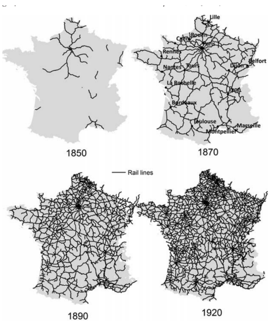
Stable equilibrium is met when the population aligns with the carrying capacity line

Slow growth occurs when natality is slightly above mortality, for fast growth natality is drastically greater than mortality

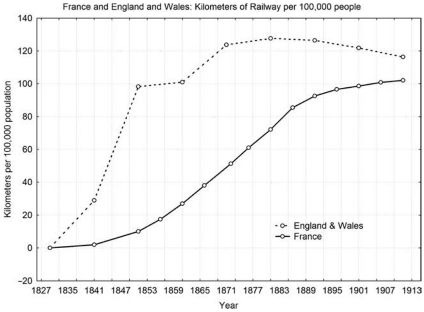
The S-shaped logistic curve is formed when growth rate decreases as carrying capacity is approached by the population

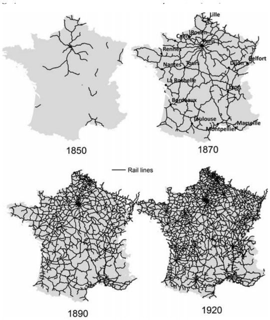
Example — French and English railway construction





Example — French and English railway construction

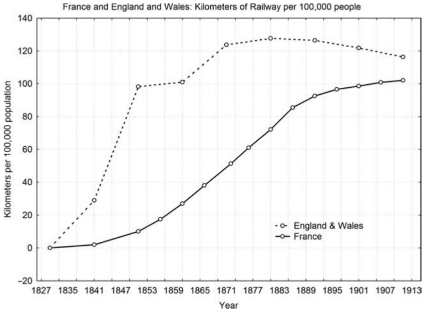


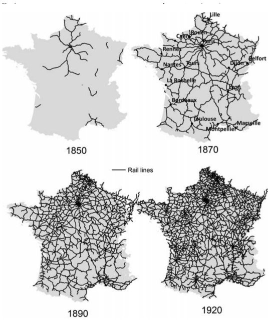


Topics for discussion

- Why do you think that the French railway construction follows logistic growth but English does not?

Example — French and English railway construction

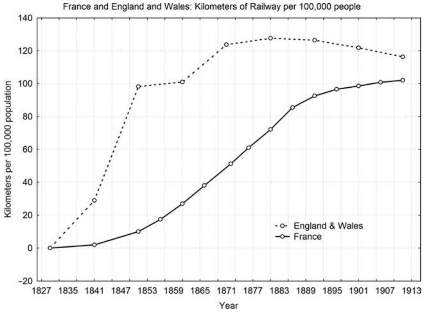


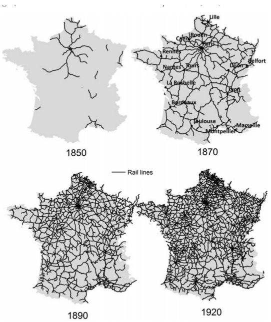


Topics for discussion

- Why do you think that the French railway construction follows logistic growth but English does not?
- How would you explain **carrying capacity** in the context of railroad construction?

Example — French and English railway construction





Topics for discussion

- Why do you think that the French railway construction follows logistic growth but English does not?
- How would you explain **carrying capacity** in the context of railroad construction?
- What may affect carrying capacity for railways? Is the capacity it constant or dynamic?

Closer look at logistic growth equation

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Let's explore the equation

- Try to figure out how carrying capacity affects the population growth. Try different combinations of P an K.

Closer look at logistic growth equation

Differential equation of logistic growth

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right)$$

Let's explore the equation

- Try to figure out how carrying capacity affects the population growth. Try different combinations of P an K.
- What are equilibria solutions where there is no population growth?

Derivation of Equilibrium Solutions:

To find equilibrium solutions, we set $\frac{dP}{dt} = 0$:

$$0 = rP\left(1 - \frac{P}{K}\right)$$

Derivation of Equilibrium Solutions:

To find equilibrium solutions, we set $\frac{dP}{dt} = 0$:

$$0 = rP\left(1 - \frac{P}{K}\right)$$

This equation is satisfied when either

- Population is zero

$$P = 0$$

Derivation of Equilibrium Solutions:

To find equilibrium solutions, we set $\frac{dP}{dt} = 0$:

$$0 = rP\left(1 - \frac{P}{K}\right)$$

This equation is satisfied when either

- Population is zero

$$P = 0$$

- Growth rate is zero

$$r = 0$$

Derivation of Equilibrium Solutions:

To find equilibrium solutions, we set $\frac{dP}{dt} = 0$:

$$0 = rP\left(1 - \frac{P}{K}\right)$$

This equation is satisfied when either

- Population is zero

$$P = 0$$

- Growth rate is zero

$$r = 0$$

- Population size equals carrying capacity

$$1 - \frac{P}{K} = 0$$
$$1 = \frac{P}{K}$$
$$P = K$$

History

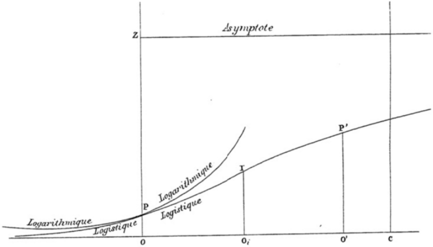
- **Pierre Verhulst**: A Belgian mathematician first proposed the logistic model in the 1830s and 1840s. His model sought to refine the exponential population growth model of Malthus.

History

- **Pierre Verhulst**: A Belgian mathematician first proposed the logistic model in the 1830s and 1840s. His model sought to refine the exponential population growth model of Malthus.
- Verhulst got inspiration from the concept of intraspecific competition —— the idea that as populations get larger, individuals compete more intensely for limited resources.

History

- **Pierre Verhulst**: A Belgian mathematician first proposed the logistic model in the 1830s and 1840s. His model sought to refine the exponential population growth model of Malthus.
- Verhulst got inspiration from the concept of intraspecific competition —— the idea that as populations get larger, individuals compete more intensely for limited resources.
- The equation was published after Verhulst had read Thomas Malthus' *An Essay on the Principle of Population*.



Pierre Verhulst and the first plot of logistic growth curve

The logistic growth model has extensive applications

- **Biology** : Used extensively in modeling animal, plant and bacterial population growth

The logistic growth model has extensive applications

- **Biology** : Used extensively in modeling animal, plant and bacterial population growth
- **Epidemiology and medicine**: Helps analyze how diseases spread within a population or growth of tumor.

The logistic growth model has extensive applications

- **Biology** : Used extensively in modeling animal, plant and bacterial population growth
- **Epidemiology and medicine**: Helps analyze how diseases spread within a population or growth of tumor.
- **Economics**: Model of market adoption of new products or technologies — a diffusion of innovation

The logistic growth model has extensive applications

- **Biology** : Used extensively in modeling animal, plant and bacterial population growth
- **Epidemiology and medicine**: Helps analyze how diseases spread within a population or growth of tumor.
- **Economics**: Model of market adoption of new products or technologies — a diffusion of innovation
- **Linguistics**: an innovation that is at first marginal begins to spread more quickly with time, and then more slowly as it becomes more universally adopted.

The logistic growth model has extensive applications

- **Biology** : Used extensively in modeling animal, plant and bacterial population growth
- **Epidemiology and medicine**: Helps analyze how diseases spread within a population or growth of tumor.
- **Economics**: Model of market adoption of new products or technologies — a diffusion of innovation
- **Linguistics**: an innovation that is at first marginal begins to spread more quickly with time, and then more slowly as it becomes more universally adopted.
- **Statistics and machine learning** Logistic functions are often used in artificial neural networks to introduce nonlinearity in the model — but it is mostly for the shape.

Tasks overview

1. Numerically solve the logistic growth equation and plot:
   - Population size vs time
   - Population increase vs time
   - Per capita growth rate vs population size
   - Population increase vs population size

Tasks overview

1. Numerically solve the logistic growth equation and plot:
   - Population size vs time
   - Population increase vs time
   - Per capita growth rate vs population size
   - Population increase vs population size
2. Adjust the appearance of the plot
   - Add plot title and axis labels
   - Change line thickness and appearance
   - Experiment with other changes.

Tasks overview

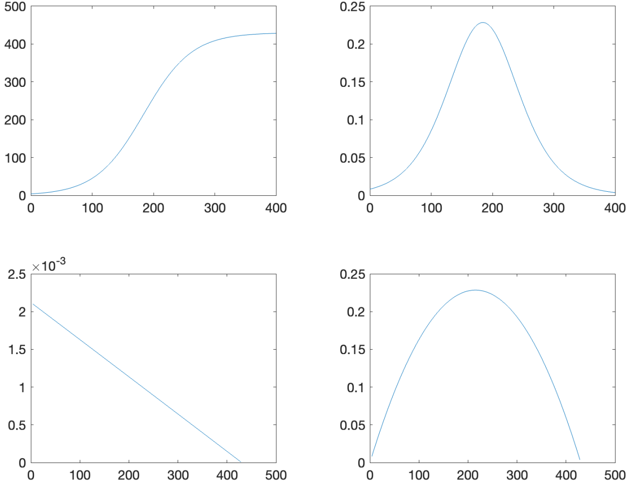
1. Numerically solve the logistic growth equation and plot:
   - Population size vs time
   - Population increase vs time
   - Per capita growth rate vs population size
   - Population increase vs population size
2. Adjust the appearance of the plot
   - Add plot title and axis labels
   - Change line thickness and appearance
   - Experiment with other changes.
3. Experiment with model parameters
   - Try to increase the initial population over the capacity of the environment
4. Model the population* of the USA and compare it with real data.

* *For the population model of the USA, set step size to 1 month, the initial population size to 5.2 million in 1790,the growth rate to 0.0237, and the capacity of the environment to 460 million.*

Solution 3.1: Logistic growth model

```
% Population parameters
    P0 = 3.9;      % Initial population
    r = 0.0255;      % Intrinsic growth rate
    K = 430;       % Carrying capacity
% Timescale
    t_start = 0;  t_end = 400; t_diff = 1/12;
% Initial time and population
    t_num = t_start; P_num = P0; P_diff = NaN;

% Calculation loop with Logistic Model
    for i = 1:t_end/t_diff
        P_diff_step =  r*P_num(end)*(1 - P_num(end)/K)*t_diff;
        P_diff = [P_diff, P_diff_step];
        P_num =  [P_num,  P_num(end) + P_diff_step];
        t_num =  [t_num,  t_num(end) + t_diff];
    end

 % Plot
figure(1); tiledlayout(2, 2)
nexttile
   plot(t_num, P_num)
nexttile
    plot(t_num, P_diff)
nexttile
    plot(P_num, P_diff./P_num)
nexttile
    plot(P_num, P_diff)
```

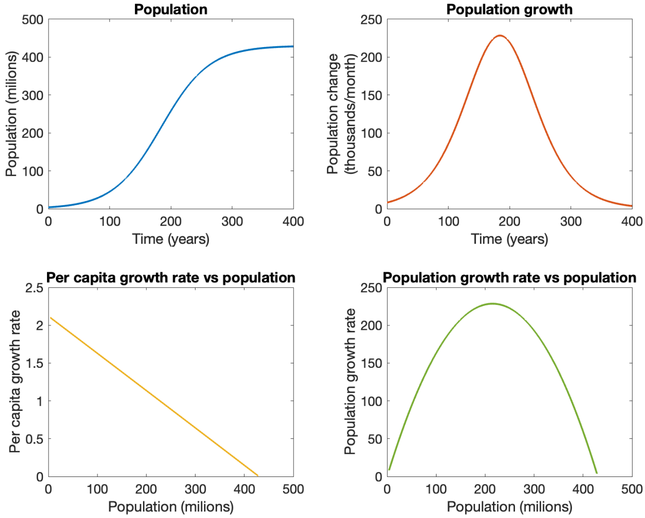
Solution 3.1: Logistic growth model



Solution 3.2: Adjusted plot

```
figure(1)
tiledlayout(2, 2)
nexttile
    plot(t_num, P_num, LineWidth=2, Color = "#0072BD")
    title('Population')
    xlabel('Time (years)')
    ylabel('Population (milions)')

nexttile
    plot(t_num, P_diff*1000, LineWidth=2,Color = "#D95319")
    title('Population growth')
    xlabel('Time (years)')
    ylabel( {'Population change';'(thousands/month)'})

nexttile
    plot(P_num, 1000*P_diff./P_num, LineWidth=2, Color="#EDB120")
     title('Per capita growth rate vs population')
    ylabel('Per capita growth rate')
    xlabel('Population (milions)')

nexttile
    plot(P_num,1000*P_diff, LineWidth=2,Color = "#77AC30")
    title('Population growth rate vs population')
    ylabel('Population growth rate')
    xlabel('Population (milions)')
```

Solution 3.2: Adjusted plot



Solution 3.3: US population

```
% Population parameters
    r = 0.0237; P0 = 5.2; K = 460;
% Start year
    start_year = 1790;
% Time axis
    t_usa = 0:10:230;
% US population
    P_usa = [  3.9,   5.3,   7.2,   9.6,  12.9,  17.1,  23.2,  31.4, ...
              38.6,  50.2,  63.0,  76.2,  92.2, 106.0, 123.2, 132.2, ...
             151.3, 179.3, 203.3, 226.5, 248.7, 281.4, 308.7, 331.4];
 % Timescale for numerical modelling
    t_start = 0;  t_end = 230; t_diff = 1/12;
 % Initial time and population
    t_num = t_start; P_num = P0; P_diff = NaN;

% Calculation loop with Logistic Model
    for i = 1:t_end/t_diff
        P_diff_step =  r*P_num(end)*(1 - P_num(end)/K)*t_diff;
        P_num =  [P_num,  P_num(end) + P_diff_step];
        t_num =  [t_num,  t_num(end) + t_diff];
    end

% Plotting
   hold on
        plot(t_num + start_year, P_num, Color = "red")
        plot(t_usa + start_year, P_usa, 'o', Color = " black")
        ylim([0,350]);   xlim([1790,2020]);
   hold off
```

Solution 3.3: US population