Universidade de Brasília

A brief introduction to Sturm-Liouville problems

Matheus Horácio

Professor: Yuri Dumaresq Sobral

Department of Mathematics,
University of Brasília

Final project for the course Introduction to Computational Methods in ODEs

Brasília, July 14, 2025

English translation by Claude Opus 5.5 (Anthropic) of the original Portuguese text by Matheus Horácio, translated from the author's LaTeX source. The Portuguese original is the authoritative version.

Contents

Introduction **3**

1 Solving the Sturm-Liouville problem numerically **6**

1.1 Regular problems 6

1.1.1 The marching shooter method 9

1.1.2 Using the Newton-Raphson method 13

1.1.3 Numerical results of the marching shooter method 13

1.2 Singular problems of type 1 18

1.3 Singular problems of type 2 19

1.4 Numerical results of the adapted marching shooter method 19

1.5 Numerical results of the Newton-Raphson method 29

1.6 Analysis of the order of the shooting method with Newton-Raphson 30

Bibliography **33**

Introduction

A regular Sturm-Liouville problem is (in dimension one) a second-order boundary value problem of the form

$$\begin{aligned} -\frac{\mathrm{d}}{\mathrm{d}t}\left(p(t)\frac{\mathrm{d}y}{\mathrm{d}t}\right) + q(t)y = \lambda r(t)y, \quad \forall t \in [a,b], \\ g\left(y(a), y'(a), y(b), y'(b)\right) = 0; \end{aligned} \tag{1}$$

where $g : \mathbb{R}^4 \longrightarrow \mathbb{R}^2$ is given by

$$g(u_1, u_2, u_3, u_4) \;=\; \begin{pmatrix} \alpha\, u_1 + \beta\, u_2 \\ \gamma\, u_3 + \delta\, u_4 \end{pmatrix}$$

Here, the data of the problem are the real functions p, q, r (with appropriate continuity/smoothness hypotheses) and the constants $\alpha, \beta, \gamma, \delta$ (which, for evident reasons, we require to satisfy $\{\alpha^2 + \beta^2, \gamma^2 + \delta^2\} \neq \{0\}$). Finding $\lambda \in \mathbb{R}$ such that problem (1) admits a solution y that is not identically zero is what we call the *Sturm-Liouville eigenvalue problem*, a nomenclature justified by the fact that if $0 \neq \lambda$ is such that problem (1) admits a solution that is not identically zero, then λ is an eigenvalue of the operator

$$\mathbb{H} \ni u \mapsto L(u) = -\frac{1}{w(t)}\left(\frac{\mathrm{d}}{\mathrm{d}t}\left[p(t)\frac{\mathrm{d}u}{\mathrm{d}t}\right] + q(t)u\right) \in \mathbb{H};$$

where $\mathbb{H}$ is an appropriate Hilbert space that it is not convenient for us to define here.

The goal of this introduction is to convince the reader of the great relevance of such problems, both in mathematics (pure or applied) and in physics. More specifically, we will concretely illustrate the extreme strength of the following theorem, which is the crown jewel of Sturm-Liouville theory:

Theorem (T.0.1) (Properties of the regular Sturm-Liouville problem). *(a) There are infinitely many real eigenvalues, which can be arranged in increasing order:*

$$\lambda_0 < \lambda_1 < \lambda_2 < \cdots < \lambda_n < \cdots$$

and in such a way that $\lambda_n \to \infty$ as $n \to \infty$.

(b) Each eigenspace has dimension one.

(c) Eigenfunctions corresponding to distinct eigenvalues are linearly independent.

(d) The eigenfunctions are orthogonal with respect to the inner product induced by $p(t)$ and form an orthonormal basis.

The Helmholtz equation and the Dirichlet eigenvalues: can one hear the shape of a drum?

One can show that the displacement F of a flat membrane Ω, held fixed along its boundary, satisfies the wave equation

$$\frac{\partial^2 F}{\partial t^2} = c^2 \nabla^2 F,$$

where c depends on the physical properties of the membrane and on the applied tension. Modeling the membrane of a drum as a bounded open set $\Omega \subset \mathbb{R}^2$, we have the eigenvalue problem

$$\begin{cases} -\Delta u = \lambda u \quad \text{in } \Omega, \\ u|_{\partial\Omega} = 0. \end{cases} \tag{2}$$

in which we seek $\lambda \in \mathbb{R}$ and $u \not\equiv 0$. Note that such a problem can be seen as an n-dimensional Sturm-Liouville problem. Such eigenvalues λ are called the *natural frequencies* of vibration of the membrane, and the functions u are called the *normal modes* of vibration. The general solution of the problem (or, equivalently, the sound of the drum) is a combination of the normal modes.

Mark Kac, in 1966, posed the following question: *Can one hear the shape of a drum?*: is it possible to determine the shape of Ω knowing only its natural frequencies of vibration, i.e, the spectrum of the Laplacian?

The investigation goes back to 1911, when Weyl showed that the area of a planar domain is determined by its spectrum (that is, it is possible to "hear the area of a drum"!). Later it was verified that the spectrum also determines the perimeter and the number of connected components. Combining these facts with the isoperimetric inequality

$$\text{perimeter}(\Omega) > 4\pi \text{ area}(\Omega),$$

The general answer to Kac's question is negative: there exist non-congruent isospectral domains. However, Kac proved that a domain isospectral to a disk of radius r must be congruent to the disk itself, showing that at least some domains are completely determined by their spectrum (that is, one can "hear" the shape of *some* drums).

Physical motivations and applications

The autonomous Schrödinger equation in dimension one is given by

$$\frac{-\hbar^2}{2m}\frac{d^2\psi(t)}{\mathrm{d}t^2} = [E - V(t)]\,\psi(t),$$

which is clearly a Sturm-Liouville problem. The non-autonomous Schrödinger equation, namely,

$$\frac{-\hbar^2}{2m}\frac{\partial^2\psi(x,t)}{\partial x^2} + V(t)\psi(x,t) = i\hbar\frac{\partial\psi(x,t)}{\partial t};$$

can also be seen as a Sturm-Liouville problem via separation of variables. Theorem **(T.0.1)** can be seen as a mathematical validation (with certain caveats pertaining to dimensionality, infinite domains and singularities, and with subtleties that depend on how closely the Sturm-Liouville problem relates to the original Schrödinger problem) of the following facts:

- There is a *discrete* sequence of energy levels, labeled by a quantum number n. This discrete nature is the reason quantum physics is called "quantum". The energy values of the states form the increasing sequence $\lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \ldots$, which tends to infinity. Thus, there is a well-defined notion of *ground state* (minimal energy): it is the state with $n = 1$.
- The wave functions of two energy levels with distinct values can be taken orthonormal.
- Any general wave function u can be written as a linear combination $u = \sum_n c_n u_n$ of the pure energy level wave functions u_n. This is the statement of *completeness* in Sturm–Liouville theory. In the language of physicists, the state u is a *superposition* of the pure energy level states.

Mathematical motivations

Consider any second-order linear ODE, say

$$\psi(t)y'' + \varphi(t)y' + \eta(t)y = f(t), \text{ where } \psi \neq 0. \tag{3}$$

Defining

$$\begin{aligned} p(t) &= \exp\left(\int \frac{\varphi(t)}{\psi(t)}\,\mathrm{d}t\right), \\ q(t) &= p(t)\frac{\eta(t)}{\psi(t)}, \\ F(t) &= p(t)\frac{f(t)}{\psi(t)}; \end{aligned} \tag{4}$$

we see that **(3)** is equivalent to an ODE in Sturm-Liouville form. Indeed, we can rewrite **(3)** as

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(p(t)\frac{dy}{\mathrm{d}t}\right) + q(t)y = F(t), \tag{5}$$

This proves one of the strengths of Sturm-Liouville theory: we have reduced the study of all second-order linear ODEs to a single one! Moreover, one can show that any solution y of **(5)** is a linear combination of eigenfunctions u_n of the Sturm-Liouville problem

$$-\frac{\mathrm{d}}{\mathrm{d}t}\left(p(t)\frac{\mathrm{d}u}{\mathrm{d}t}\right) + q(t)u = \lambda r(t)u \tag{6}$$

Chapter 1

Solving the Sturm-Liouville problem numerically

We will use the shooting method to numerically solve Sturm-Liouville problems, that is, to numerically determine the eigenvalues and eigenfunctions of such problems.

Remark (O.1). The codes developed to numerically solve all the examples presented here are available in the following repositories

- **https://github.com/SaganGromov/SL_REGULAR_EXEMPLOS**
- **https://github.com/SaganGromov/SL_SINGULAR_EXEMPLOS**
- **https://github.com/SaganGromov/SL_NEWTON_RAPHSON**

1.1 Regular problems

Definition (D.1.1). A Sturm-Liouville equation of the form

$$-(p(x)y'(x))' + q(x)y(x) = \lambda r(x)y(x), \ \forall x \in [a,b], \tag{1.1}$$

is said to be *regular* if the functions p, q and r are smooth on $[a,b]$, with p, q strictly positive on $[a,b]$.

Definition (D.1.2). A *regular* Sturm-Liouville problem is a problem of the form **(1.1)** subject to separated linear boundary conditions, namely,

$$\alpha \cdot y(a) + \beta \cdot y'(a) = 0; \tag{1.2}$$
$$\gamma \cdot y(b) + \delta \cdot y'(b) = 0; \tag{1.3}$$

where $\alpha, \beta, \gamma, \delta$ are real constants such that $(\alpha, \beta) \neq (0,0)$ and $(\gamma, \delta) \neq (0,0)$

Before making the shooting method explicit, we will state several classical results of Sturm-Liouville theory that will be essential for creating our algorithm that numerically determines the spectrum and

the eigenfunctions of a regular Sturm-Liouville problem. The proof of such results is beyond the scope of this work, but can be found in [2].

Theorem (T.1.1).

- *The eigenvalues of a regular problem are real, simple and can be ordered as*

$$\lambda_0 < \lambda_1 < \lambda_2 < \cdots$$

- *The eigenfunction y_n corresponding to λ_n has exactly n nodal zeros (interior zeros where a change of sign occurs) and no other interior zero. Moreover, the nodes of y_{n-1} and y_n strictly interlace, that is, between any two consecutive zeros of y_n there is exactly one zero of y_{n-1}.*
- *If $\alpha\beta \leq 0$, $\gamma\delta \geq 0$, $p > 0$, $q \geq 0$, and $r(x) > 0$ on $[a, b]$, then all eigenvalues are positive, except in the case where the eigenvalue problem is $-(py')' = \lambda ry$, $y'(a) = 0$, $y'(b) = 0$, in which case 0 is an eigenvalue and all other eigenvalues are positive. (See Theorem 124.)*
- *If $\alpha\beta \leq 0$, $\gamma\delta \geq 0$, $p > 0$, $q \geq 0$, and $r(x) > 0$ on $[a, b]$, then $\lambda_0 \geq \min\limits_{\forall x\in[a,b]} \dfrac{q(x)}{r(x)}$ and $\lambda_0 > 0$ unless the eigenvalue problem is $-(py')' = \lambda ry$, $y'(a) = 0$, $y'(b) = 0$, in which case 0 is an eigenvalue and all other eigenvalues are positive.*

Remark (O.2). Note that we are starting the count at $n = 0$. This means that the eigenvalue λ_0 is the smallest eigenvalue, and the eigenfunction y_0 is the eigenfunction corresponding to λ_0. The eigenfunction y_0 has no nodal zeros; in particular, it never changes sign in (a, b).

Let us now consider the regular eigenvalue problem, i.e

$$\begin{cases} -(p(x)y'(x))' + (q(x) - \lambda r(x))y(x) = 0, \quad \forall x \in [a, b], \\ \alpha y(a) + \beta y'(a) = 0, \\ \gamma y(b) + \delta y'(b) = 0, \end{cases} \tag{1.4}$$

with the coefficients and constants satisfying the hypotheses already mentioned. The method we will use to solve this BVP consists in transforming the BVP into an IVP. Since the eigenvalues λ_i of (1.4) are all simple and the corresponding eigenfunctions satisfy $\alpha y(a) + \beta y'(a) = 0$, the vector $(y(a), y'(a))$ is parallel to $(-\beta, \alpha)$. It follows that there is a unique normalized eigenfunction corresponding to each eigenfunction, which is exactly the solution of the following IVP:

$$\begin{cases} -(p(x)u'(x))' + (q(x) - \lambda r(x))u(x) = 0, \quad \forall x \in [a, b], \\ u(a) = -\dfrac{\beta}{\sqrt{\alpha^2 + \beta^2}}, \quad u'(a) = \dfrac{\alpha}{\sqrt{\alpha^2 + \beta^2}}. \end{cases} \tag{1.5}$$

Let us then denote the global solution of the IVP by $u = u(x) = u(x, \lambda)$, which is defined on $[a, b]$ whatever our choice of λ. Now, recall that λ is an eigenvalue of the Sturm-Liouville problem (1.4) if and only if the IVP has a solution $u(x, \lambda)$ defined on $[a, b]$ that satisfies $F(\lambda) = 0$, where

$$F(\lambda) = \gamma u(b, \lambda) + \delta u'(b, \lambda),$$

In this case, $y(x) = u(x, \lambda)$ is the corresponding normalized eigenfunction.

We then know that the roots of $F(\lambda)$ are exactly the eigenvalues of the Sturm-Liouville eigenvalue problem. This then motivates us to use the following algorithm:

(i). Step 1. We determine an initial guess for an eigenvalue of interest (or, even better, an interval that contains an eigenvalue of interest).

(ii). Step 2. We solve the IVP with the initial guess λ_k^0 and obtain the solution $u(x, \lambda_k^0)$.

(iii). Step 3. We compute $F(\lambda_k^0) = \gamma u(b, \lambda_k^0) + \delta u'(b, \lambda_k^0)$.

(iv). Step 4. If $|F(\lambda_k^0)| < \varepsilon$, then λ_k^0 is an (approximate) eigenvalue of the Sturm-Liouville problem. Otherwise, we use some root-finding method (such as the bisection method) to update the guess λ_k^0 and return to step 2.

To use the bisection method, we need to know a priori that F changes sign at each eigenvalue. This is indeed true, as shown in [2]. We can also use the Newton-Raphson method: indeed, denoting

$$w(x, \lambda) \ := \ \frac{\partial u}{\partial \lambda}(x, \lambda);$$

and differentiating term by term using the IVP, we obtain

$$-\Big(p\, w'\Big)' + (q - \lambda\, r)\, w \ - \ r\, u \ = \ 0.$$

Now, since the initial data do *not* depend on λ, it follows that

$$w(a, \lambda) = 0, \quad w'\Big(a, \lambda\Big) = 0.$$

This gives us another IVP, namely,

$$\begin{aligned} -\Big(p\, w'\Big)' + (q - \lambda\, r)\, w = r\, u, \ \forall x \in [a, b], \\ w(a) = 0, \quad w'(a) = 0. \end{aligned}$$

Such an IVP allows us to numerically determine $F'(\lambda)$,

$$F'(\lambda) = \gamma w(b, \lambda) + \delta w'(b, \lambda).$$

As shown in [2], $F'(\lambda) \neq 0$. Thus, we can use the Newton-Raphson method to update the guess λ_k^n, by setting

$$\lambda_k^{(n+1)} = \lambda_k^{(n)} - \frac{F\Big(\lambda_k^{(n)}\Big)}{F'\Big(\lambda_k^{(n)}\Big)}.$$

We will start by applying this strategy (initially using the bisection method) to determine the first eigenvalue λ_0 and its corresponding eigenfunction y_0. For many problems we will consider, one can show a priori that $\lambda_0 > 0$, which makes an initial guess $\lambda_0^0 = \varepsilon$ with some small $\varepsilon > 0$ reasonable. Choosing a small increment $\Delta\lambda$, we can then iterate the bisection method on intervals of the form $[\lambda_k^i + k\Delta\lambda, \lambda_k^i + (k+1)\Delta\lambda]$ until we find as many eigenvalues and eigenfunctions as we want. Evidently,

this approach is very expensive and inefficient, but it has two advantages compared to the Newton-Raphson method: it does not depend on a good initial guess for whichever k-th eigenvalue we want to find, and it can be iterated successively to determine any number of eigenvalues. Note that the same approach equipped with the Newton-Raphson method faces serious problems: the speed/quadratic convergence of the method holds only locally, which makes divergence or the finding of spurious values possible. The technique of incrementing λ in small fixed steps $\Delta\lambda$ and iterating the bisection method to sweep the whole spectrum also obviously does not work for the Newton-Raphson method.

1.1 The marching shooter method

In this section we will explain, step by step, a program developed in Fortran whose goal is to numerically solve any Sturm-Liouville problem using the shooting method, as described above. The program is capable of finding the eigenvalues and the corresponding eigenfunctions for a regular Sturm-Liouville problem with separated linear boundary conditions. To solve the associated IVP, we use the classical fourth-order Runge-Kutta method, and to find the roots of F we use the bisection method. Next, we present a step-by-step analysis of the implemented algorithm.

STEP 1. Problem Setup and Preparation

SUBSTEP 1.1. Initialization of the domain and boundary conditions - `constantes.f90`

We define the domain $[a, b]$ of the problem (in the current version of the code, $[0, 1]$) and initialize $\alpha, \beta, \gamma, \delta$ as real constants. In the initial example implemented, we use $\alpha = \gamma = 1$ and $\beta = \delta = 0$, corresponding to the boundary conditions $y(0) = y(1) = 0$.

SUBSTEP 1.2. Initialization of various numerical parameters - `constantes.f90`

- `passos = 100000`: Number of integration steps used to solve the IVP via RK4
- `tol = 1e-12`: Tolerance for the convergence of the bisection method
- `n_eigen = 10`: Number of eigenvalues to be found
- `verbose_level = 3`: Desired verbosity level when running the compiled program, mainly useful for debugging

SUBSTEP 1.3. Problem definition - `problema.f90`

We specify which Sturm-Liouville problem we will consider by defining the functions $p(x)$, $q(x)$ and $r(x)$ as Fortran functions. For example, for the classical Dirichlet problem, we define:

$$p(x) = 1$$
$$q(x) = x$$

$$r(x) = 0$$

corresponding to the ODE

$$y'' + \lambda y = 0$$

STEP 2. 2nd-order ODE as a first-order system - `definicao_EDO.f90`

We convert the initial second-order ODE into a first-order system by setting

$$\begin{cases} y_1(x) = y(x), \\ y_2(x) = p(x)y'(x) \end{cases}$$

so that

$$\begin{cases} y_1' = \dfrac{y_2}{p(x)}, \\ y_2' = \Big(q(x) - \lambda\, r(x)\Big)\, y_1, \end{cases} \tag{1.6}$$

STEP 3. Definition of the RK4 method - `integrador_rk4.f90`

The classical 4th-order Runge-Kutta method is implemented with step size

$$h = \frac{b-a}{\text{passos}} = 10^{-5},$$

where, for the system $\mathbf{y}' = \mathbf{f}(x, \mathbf{y}, \lambda)$, we have:

$$\mathbf{k}_1 = \mathbf{f}(x_n, \mathbf{y}_n, \lambda), \tag{1.7}$$

$$\mathbf{k}_2 = \mathbf{f}\left(x_n + \frac{h}{2}, \mathbf{y}_n + \frac{h}{2}\mathbf{k}_1, \lambda\right), \tag{1.8}$$

$$\mathbf{k}_3 = \mathbf{f}\left(x_n + \frac{h}{2}, \mathbf{y}_n + \frac{h}{2}\mathbf{k}_2, \lambda\right), \tag{1.9}$$

$$\mathbf{k}_4 = \mathbf{f}\left(x_n + h, \mathbf{y}_n + h\mathbf{k}_3, \lambda\right), \tag{1.10}$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right). \tag{1.11}$$

STEP 4. Implementation of the Shooting Method - `atirador.f90`

SUBSTEP 4.1. Solving the IVP

We now numerically solve the IVP **(1.5)**, namely,

$$\begin{cases} -\left(p(x)u'(x)\right)' + \left(q(x) - \lambda r(x)\right)u(x) = 0, & \forall x \in [a,b], \\ u(a) = -\dfrac{\beta}{\sqrt{\alpha^2+\beta^2}}, \quad u'(a) = \dfrac{\alpha}{\sqrt{\alpha^2+\beta^2}}. \end{cases}$$

We store the numerical solution y and its derivative y' in a vector of size `passos`. Moreover, we do the "bookkeeping" of the nodes, that is, we count how many times the function $y_1(x)$ changes sign in the interval $[a,b]$. The system is integrated from $x = a$ to $x = b$, counting the sign changes of $y_1(x)$ (nodes):

- We initialize `nodes = 0`, `yprev = y_1(a)`
- For each step/iteration of the integration loop:
  - If $y_1(x_{i+1}) \cdot \text{yprev} < 0$, increment `nodes`
  - Update $y_{\text{prev}} = y_1(x_{i+1})$

Finally, we compute the residual associated with the boundary condition at b:

$$F(\lambda) = \gamma y_1(b) + \delta \frac{y_2(b)}{p(b)} = \gamma y(b) + \delta y'(b) \qquad \textbf{(1.12)}$$

STEP 5. The march of the eigenvalue search - `espectro_melhorado.f90`

SUBSTEP 5.1. Sweeping values of λ

The algorithm sweeps values of λ starting from 0 with an adaptive increment $\Delta\lambda$:

- The initial $\Delta\lambda$ is set to 0.01.
- After finding eigenvalues, we refine $\Delta\lambda = 0.01 \cdot (\lambda_n - \lambda_{n-1})$

SUBSTEP 5.2. Detection of eigenvalue candidates

An eigenvalue candidate is detected when one of the following conditions occurs:

1. Sign change in $F(\lambda)$: $F(\lambda_{\text{prev}}) \cdot F(\lambda) < 0$
2. Increase in nodes: $\text{nodes}(\lambda) > \text{nodes}(\lambda_{\text{prev}})$

STEP 6. Refined bisection - `modulo_bisseccao.f90`

Summary of the Bisection Methods:

- **Method 1: Bisection by Sign (bissec_sig)**
  - Used when the function $F(\lambda)$ has a sign change in the interval.
  - Performs the standard bisection to find the root of $F(\lambda) = 0$.
  - Operates with a tolerance of 10^{-12}.
- **Method 2: Bisection by Nodes (bissec_nos)**
  - Employed when the node count increases by 1 in the interval.
  - Divides the interval to identify where the change in the node count occurs.
  - Issues a warning if the variation in the nodes is not exactly 1.
- **Method 3: Robust Bisection (bissec_robusta)**
  - Evaluates both endpoints of the interval to determine the appropriate condition.
  - Selects the suitable method (by sign or by node count) based on the observed conditions.
  - If none of the previous methods applies, uses the midpoint as the bisection criterion.

STEP 7. Eigenvalue validation - `espectro_melhorado.f90`

SUBSTEP 7.1. Checking the validation criteria

Each eigenvalue candidate λ_i is validated by checking that the two conditions below are satisfied:

1. $|F(\lambda)| < \varepsilon$ (residual approximately zero)
2. $F(\lambda - \varepsilon) \cdot F(\lambda + \varepsilon) < 0$ (sign change)

Candidates that fail this validation are rejected as spurious.

STEP 8. Saving the solution - `solucao.f90`

For each eigenvalue found, we compute the corresponding eigenfunction:

- We reintegrate the IVP from $x = a$ to $x = b$ using the eigenvalue found
- We save the values $(x, y_1(x))$ in the files `solucao_autovalor_N.dat`

1.1 Using the Newton-Raphson method

We also developed a program that implements the Newton-Raphson method to find the eigenvalues and corresponding eigenfunctions of a regular Sturm-Liouville problem. Since the implementation in this case is much simpler, we will not explain the program step by step now. In contrast with the marching algorithm using the bisection method that we described above, in this case we can only find one eigenvalue at a time, and we need a good initial guess for each eigenvalue we want to find.

Recall that Newton-Raphson updates the initial guess via

$$\lambda_k^{(n+1)} = \lambda_k^{(n)} - \frac{F\left(\lambda_k^{(n)}\right)}{F'\left(\lambda_k^{(n)}\right)}.$$

where $F(\lambda)$ is the residual associated with the IVP **(1.5)** and $F'(\lambda)$ is the derivative of F with respect to λ, which can be obtained by solving the IVP

$$\begin{cases} -(pw')' + (q - \lambda r)w = ru, \quad \forall x \in [a, b], \\ w(a) = 0, \quad w'(a) = 0. \end{cases}$$

Indeed,

$$F'(\lambda) = \gamma w(b, \lambda) + \delta w'(b, \lambda).$$

1.1 Numerical results of the marching shooter method

Example (E.1.1). We will test the marching shooter approach with the simplest regular Sturm-Liouville problem there is, the classical Dirichlet problem:

$$\begin{cases} y'' + \lambda y = 0, \ \forall x \in [0, 1]; \\ y(0) = y(1) = 0. \end{cases}$$

By the construction of the method, we should find the eigenvalues $\lambda_n = (n + 1)^2\pi^2$, with the corresponding normalized eigenfunctions

$$y_n(x) = \frac{\sin((n + 1)\pi x)}{(n + 1)\pi}, \quad n = 0, 1, 2, \ldots$$

Note that the normalization is due to the fact that we are solving the IVP **(1.5)**, i.e

$$\begin{cases} -(1 \cdot u'(x))' + (0 - \lambda \cdot 1)u(x) = 0, \quad \forall x \in [0, 1], \\ u(0) = 0, \quad u'(0) = 1. \end{cases} \tag{1.13}$$

Below is a table with the eigenvalues found by the program, as well as the number of bisection iterations needed to find them. Due to the careful approach of the method of incrementing λ in very small steps,

we need to search for eigenvalues in thousands of intervals, the vast majority of them (in fact, all except one, the last) being discarded.

n	λ_n	Intervals searched	Bisection iterations
0	9.86960440	987	34
1	39.47841760	3938	34
2	88.82643961	4105	39
3	157.91367042	4245	39
4	246.74011003	4374	40
5	355.30575844	4497	40
6	483.61061565	4616	40

We also plot the corresponding eigenfunctions $y_n = \dfrac{\sin((n+1)\pi x)}{(n+1)\pi}$ and the eigenfunctions numerically determined by the program. As we will see below, there is no humanly perceptible difference between the eigenfunctions numerically determined by the program and the theoretical eigenfunctions.

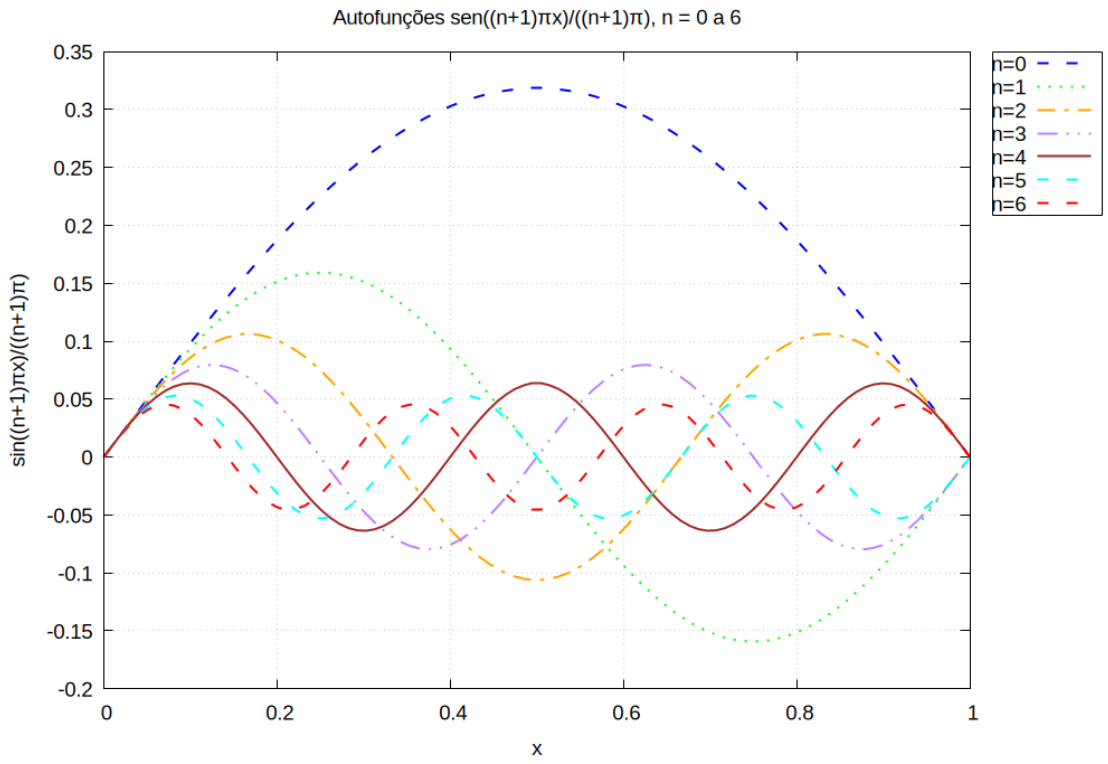


Figure 1.1: Exact eigenfunctions

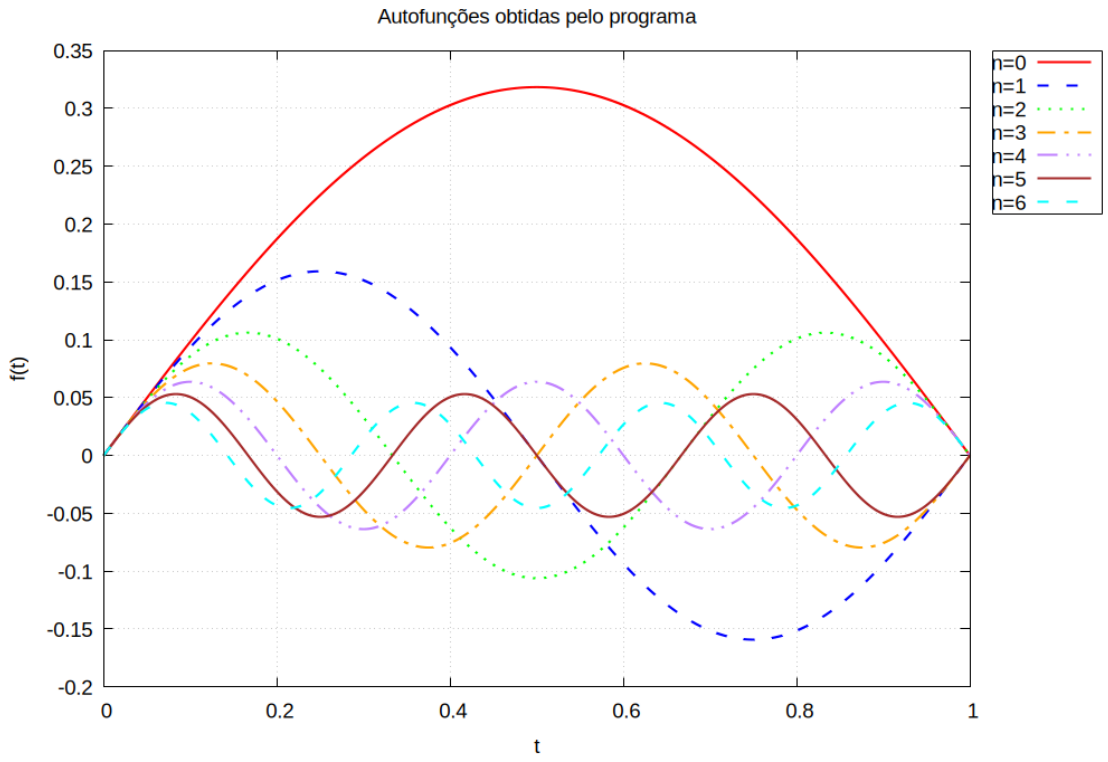


Figure 1.2: Eigenfunctions numerically determined by the program

Example (E.1.2). We will now numerically determine the first five eigenvalues and eigenfunctions of the regular Sturm-Liouville problem

$$\begin{cases} y'' + \lambda y = 0, & 0 \leq x \leq 1, \\ y(0) = 0, \\ y(1) + y'(1) = 0. \end{cases}$$

Numerically, we obtain the following eigenvalues:

n	λ_n	Intervals searched	Bisection iterations
0	4.11585837	412	34
1	24.13934203	2405	34
2	63.65910655	2602	38
3	122.88916176	2752	39
4	201.85125830	2886	40

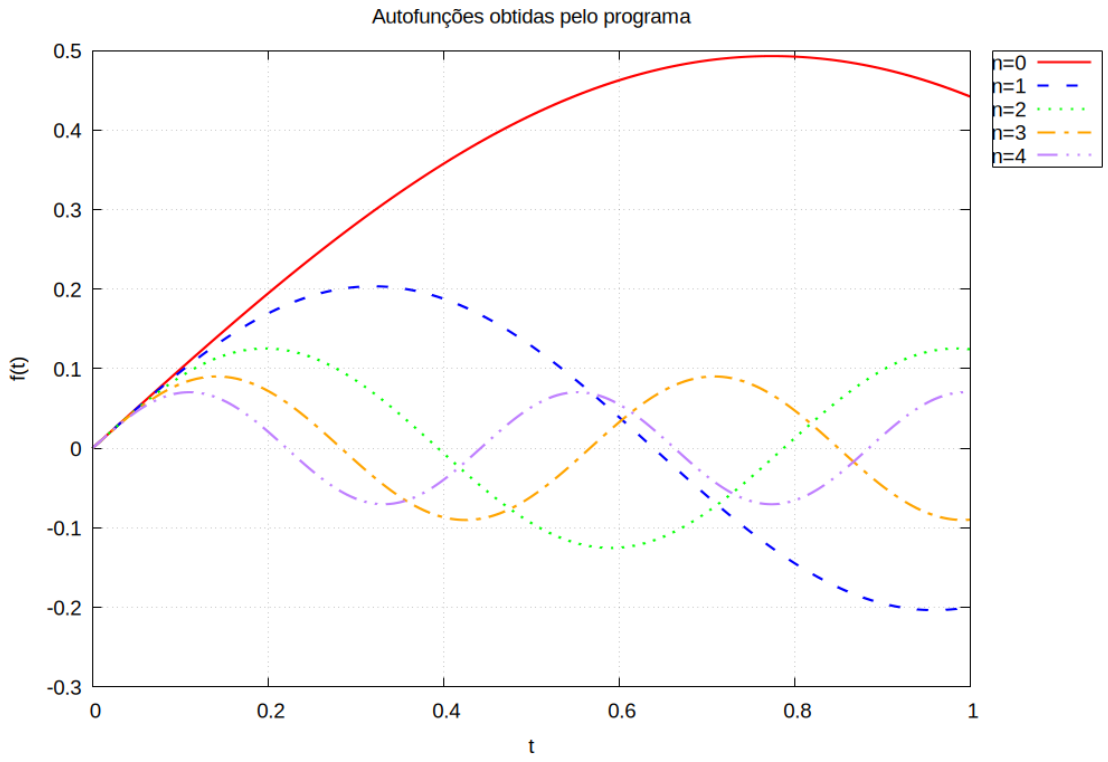


Figure 1.3: Eigenfunctions numerically determined by the program

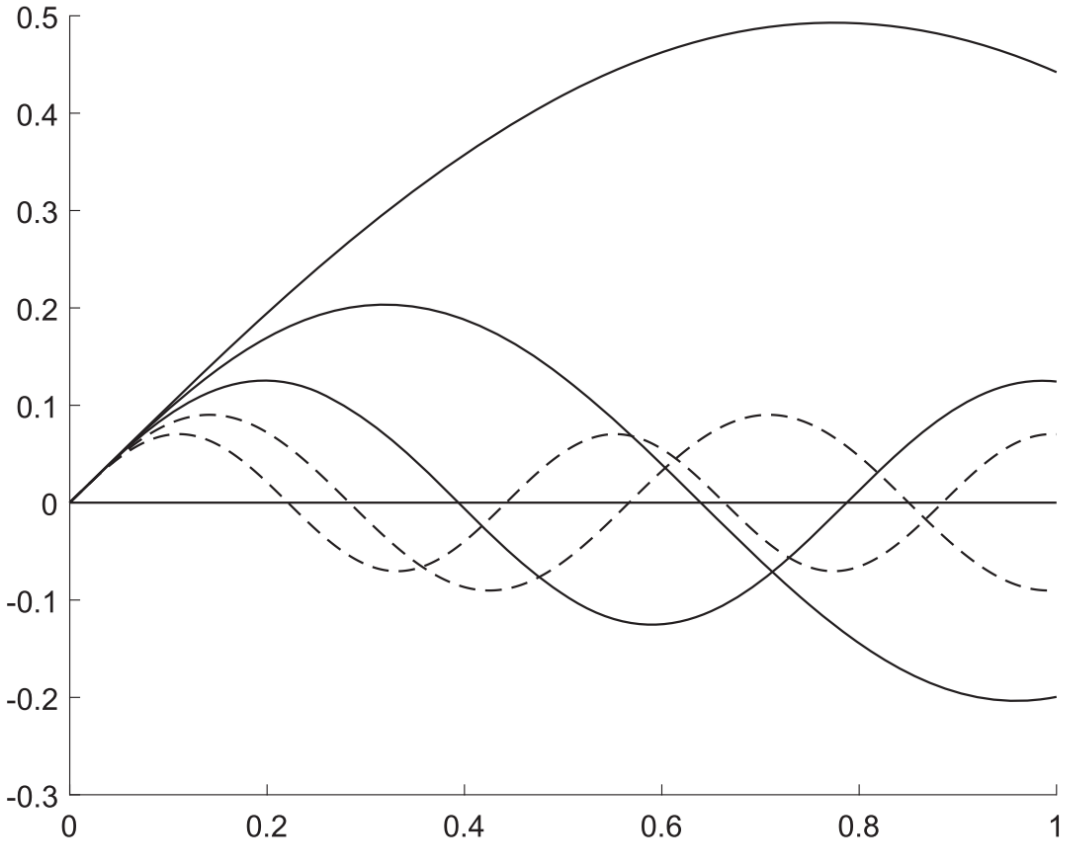


Figure 1.4: Graphs of the eigenfunctions as shown in the book [2]

Example (E.1.3). We will now numerically determine the first five eigenvalues and eigenfunctions of the regular Sturm-Liouville problem

$$\begin{cases} -(1 \cdot u'(x))' + (x - \lambda \cos(x))u(x) = 0, \quad \forall x \in [0,1], \\ u'(0) = 0, \quad u'(1) = 0. \end{cases} \tag{1.14}$$

Numerically, we obtain the following eigenvalues:

n	λ_n	Intervals searched	Bisection iterations
0	0.57824838	58	34
1	13.01659499	1292	34
2	48.48917784	1577	37
3	107.57497472	1744	39
4	190.28375117	1884	40

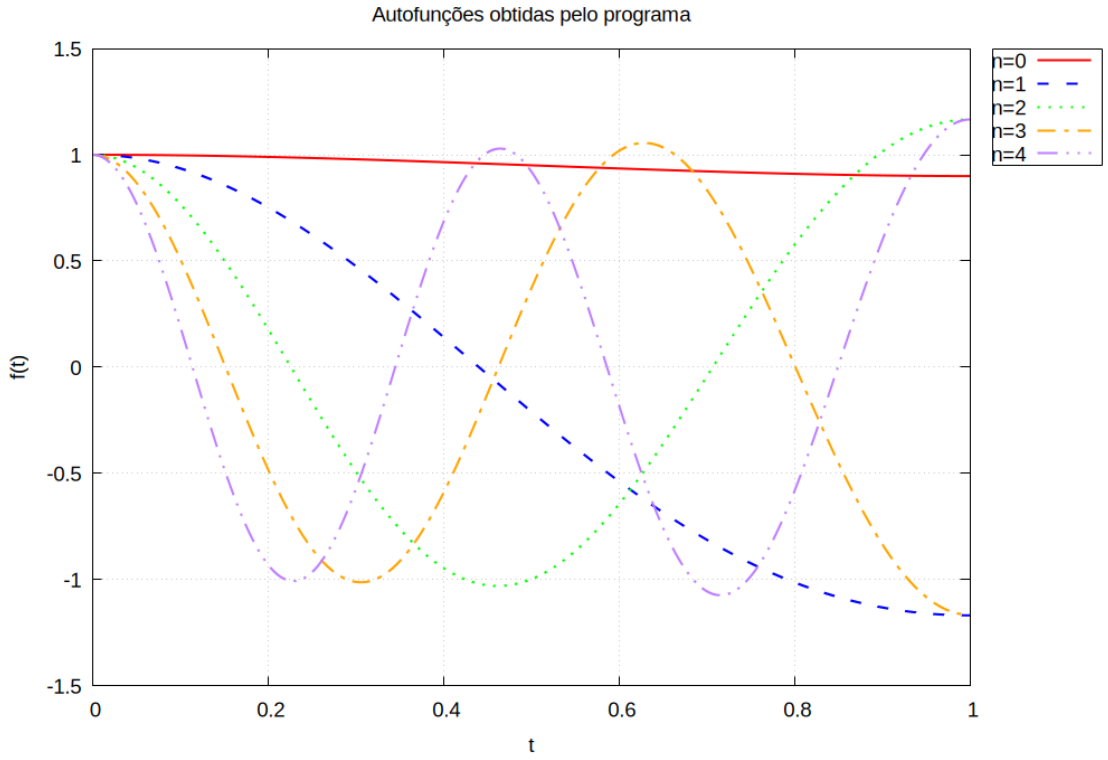


Figure 1.5: Eigenfunctions numerically determined by the program

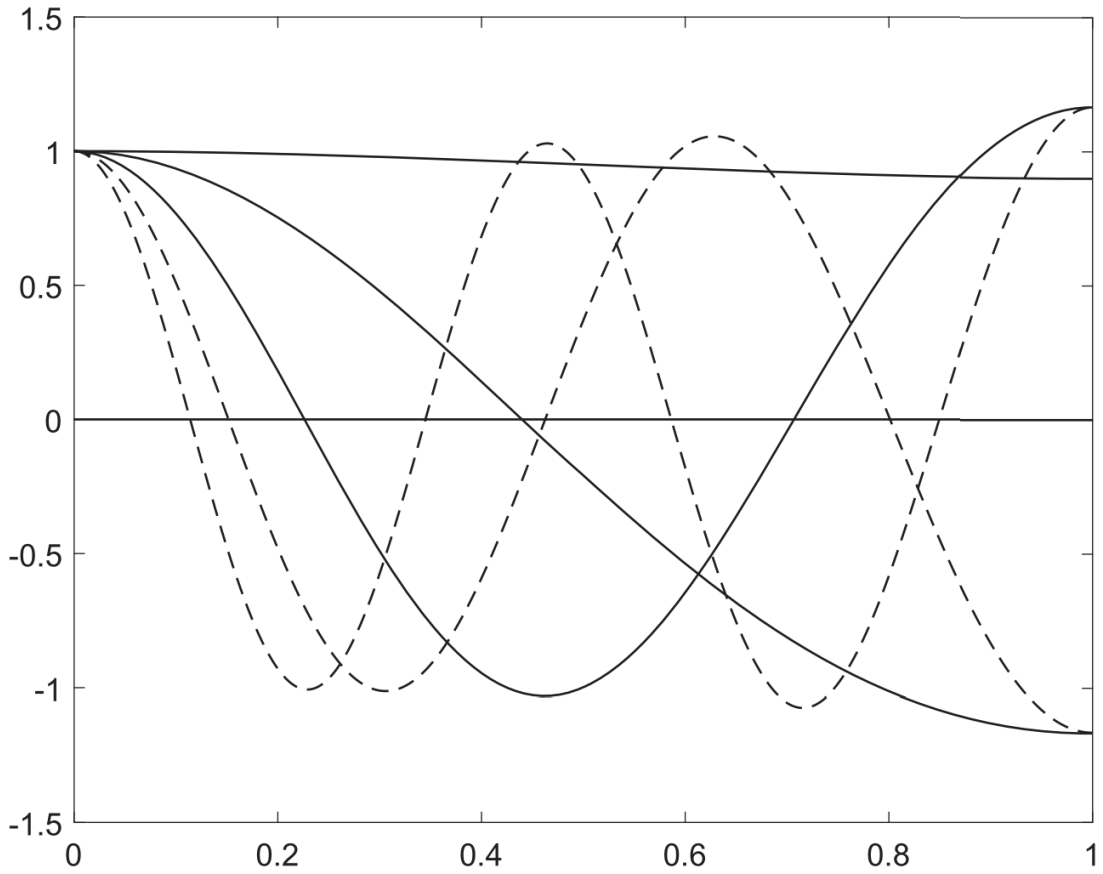


Figure 1.6: Graphs of the eigenfunctions as shown in the book [2]

1.2 Singular problems of type 1

We will now adjust the approach developed above to deal with singular Sturm-Liouville problems, that is, problems that are not regular. A singular Sturm-Liouville problem of type 1 is a problem of the form:

$$\begin{cases} -\left(p(x)y'\right)' + q(x)y = \lambda r(x)y, & a < x < b, \\ |y(a)| < \infty, & \\ \gamma y(b) + \delta y'(b) = 0, & |\gamma| + |\delta| \neq 0, \end{cases} \tag{1.15}$$

where now

(1) $p(x) \geq 0$ is smooth on $[a, b]$, nonzero for $a < x \leq b$, and satisfies $p(a) = 0$, $p'(a) \neq 0$.

(2) $q(x)$ is smooth on $[a, b]$.

(3) $p(x)$ and $q(x)$ are real functions, γ and δ are real constants.

(4) $r(x)$ is smooth on $[a, b]$, with either $r(x) > 0$ on $[a, b]$ or $r(x) = (x - a)^m \rho(x)$ where $m > 0$ and $\rho(x) > 0$ is smooth on $a \leq x \leq b$.

Let us now observe that the fact that p vanishes at a fatally dooms the approach defined in (1.6), introducing a division by zero at $x = a$. To deal with this, we will start integrating from $a + \varepsilon$, ε being a small positive number. Evidently, it is also necessary to change the IVP we are solving.

As shown in [2], the appropriate IVP and normalization to be solved is

$$\begin{cases} -\left(p(x)u'(x)\right)' + \left(q(x) - \lambda r(x)\right)u(x) = 0, \quad a \leq x \leq b, \\ u(a) = 1, \quad u'(a) = \dfrac{q(a) - \lambda r(a)}{p'(a)}. \end{cases}$$

The boundary condition at b remains the same, that is, $\gamma u(b) + \delta u'(b) = 0$. Moreover, it is also possible to determine $F'(\lambda)$ via another IVP, as we will see next

Theorem (T.1.2) (Theorem 190, [2]). *If λ is an eigenvalue of the Sturm-Liouville problem (1.16), then $F(\lambda) = 0$ and*

$$F'(\lambda) = \gamma w(b, \lambda) + \delta w'(b, \lambda) \neq 0$$

where $w = w(x, \lambda) = \dfrac{\partial u(x, \lambda)}{\partial \lambda}$ and $y = u(x, \lambda)$ is the corresponding normalized eigenfunction.

1.3 Singular problems of type 2

A singular Sturm-Liouville problem of type 2 is a problem of the form:

$$\begin{cases} -\left(p(x)y'\right)' + q(x)y = \lambda r(x)y, & a < x < b, \\ |y(a)| < \infty, & \\ \gamma y(b) + \delta y'(b) = 0, & |\gamma| + |\delta| \neq 0, \end{cases} \tag{1.16}$$

where now

(1) $p(x) = (x - a)\varphi(x)$, where $\varphi > 0$ is smooth on $[a, b]$.

(2) $q(x) = \dfrac{q_1(x)}{x - a}$, where q_1 is real-valued and smooth on $[a, b]$, and moreover satisfies $q_1(a) > 0$.

(3) $r(x)$ is smooth on $[a, b]$, γ and δ are real constants such that $\gamma\delta \geq 0$.

Note that now $p(a) = 0$ also holds, and moreover, it is possible that $|q(x)| \to \infty$ as $x \to a$. However, we can still use the same approach as for singular problems of type 1. This will make the eigenfunctions we find not normalized as in the convention adopted in [2], which is evidently not a big problem, since they will differ from the normalized ones only by a constant multiplicative factor.

1.4 Numerical results of the adapted marching shooter method

We will use almost exactly the same approach we used for regular problems, the only differences being that we now solve another IVP when shooting and we integrate from $x = a + 10^{-6}$ to $x = b$.

Example (E.1.4). Let us consider the singular Sturm-Liouville problem of type 1

$$\begin{cases} -(xy')' = \lambda xy, & 0 < x < 1, \\ |y(0)| < \infty, \quad y(1) + y'(1) = 0. & \end{cases}$$

Numerically, we obtain the following eigenvalues:

n	λ_n	Intervals searched	Bisection iterations
0	1.57699273	158	24
1	16.64213839	1655	24
2	51.20546183	1884	28
3	105.49314030	2041	29
4	179.51705520	2178	30

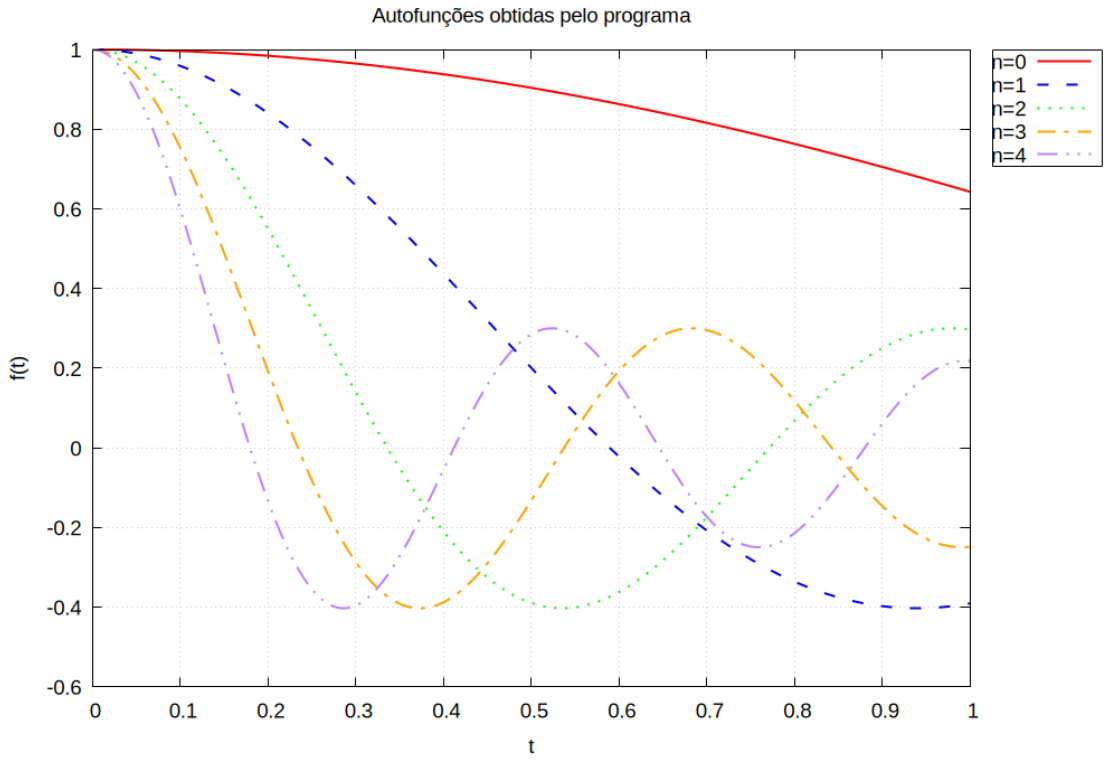


Figure 1.7: Eigenfunctions numerically determined by the program

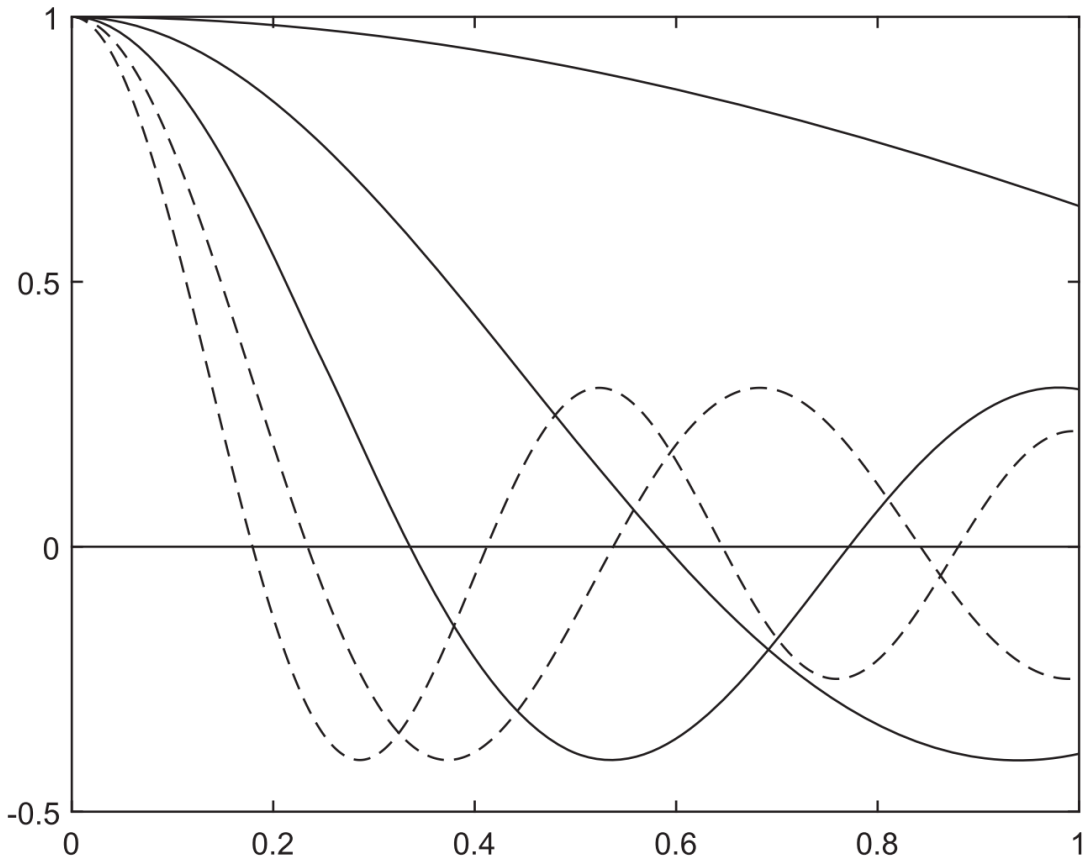


Figure 1.8: Graphs of the eigenfunctions as shown in the book [2]

Example (E.1.5). Let us consider the singular Sturm-Liouville problem of type 1

$$\begin{cases} -\left((\sin x)y'\right)' + xy = \lambda(\cos x)y, & 0 < x < 1, \\ |y(0)| < \infty, \quad y(1) = 0. \end{cases}$$

Numerically, we obtain the following eigenvalues:

n	λ_n	Intervals searched	Bisection iterations
0	1.63556063	164	24
1	8.31310754	822	24
2	20.27440115	1000	26
3	37.55132388	1144	27
4	60.14601837	1275	28

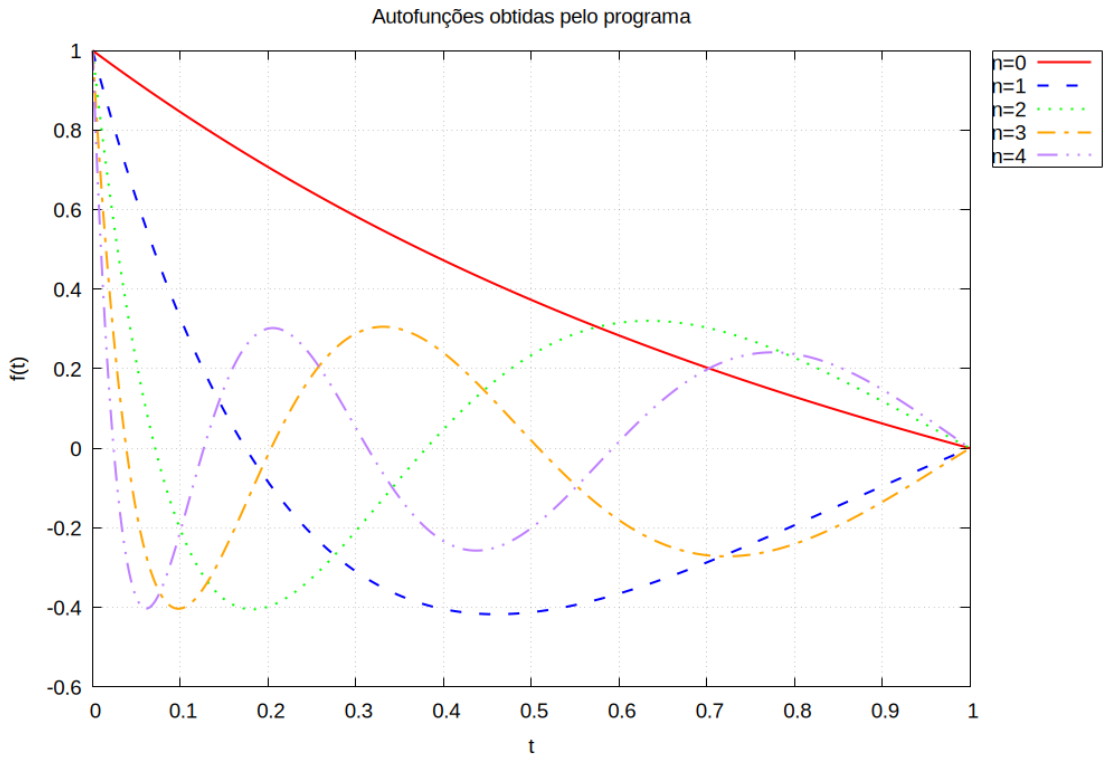


Figure 1.9: Eigenfunctions numerically determined by the program

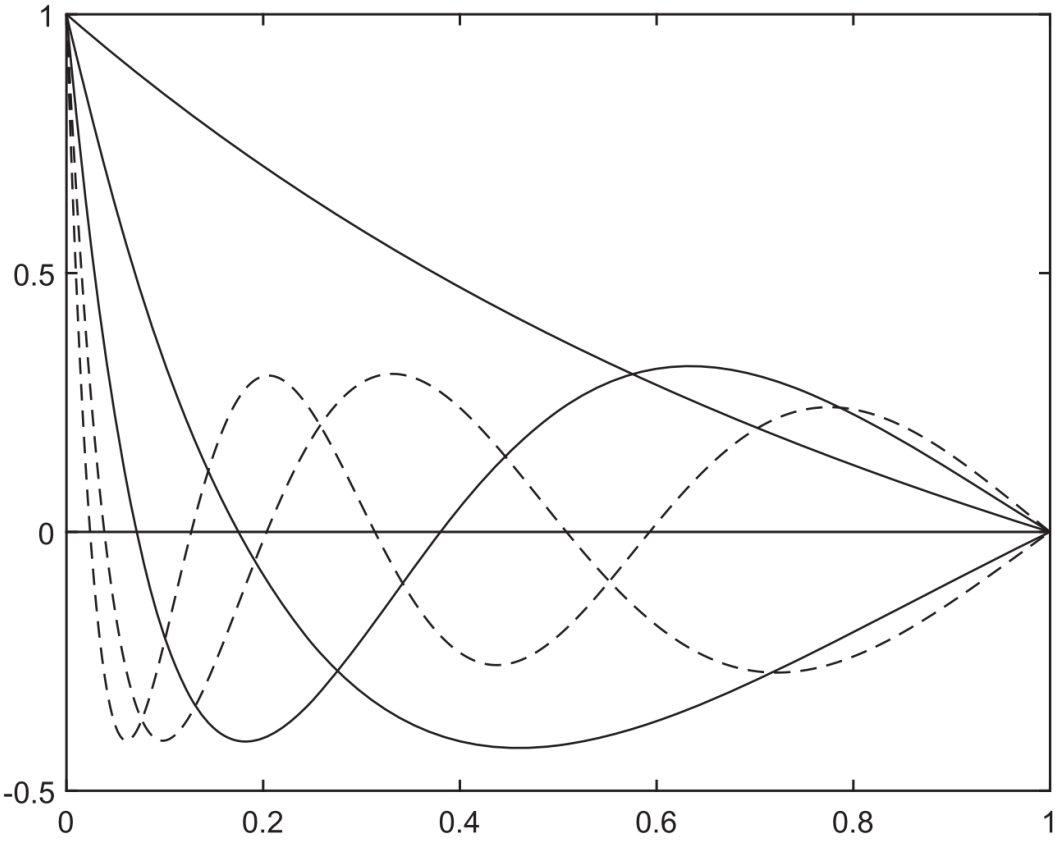


Figure 1.10: Graphs of the eigenfunctions as shown in the book [2]

Example (E.1.6). Let us consider the singular Sturm-Liouville problem of type 1

$$\begin{cases} -(xy')' + \sin(\pi x)\, y = \lambda x y, & 0 < x < 1, \\ |y(0)| < \infty, \quad y'(1) = 0. \end{cases}$$

Numerically, we obtain the following eigenvalues:

n	λ_n	Intervals searched	Bisection iterations
0	1.22123592	123	24
1	16.63599690	1655	24
2	51.09474141	1878	28
3	105.36250987	2036	29
4	179.37909191	2173	30

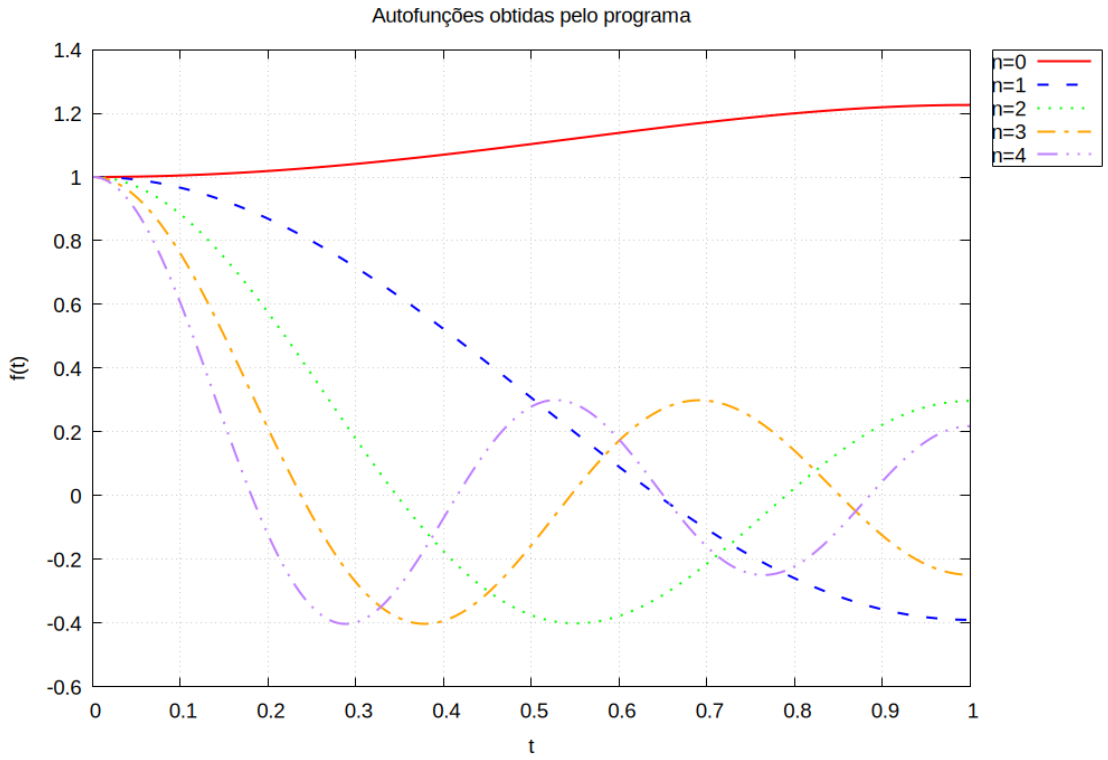


Figure 1.11: Eigenfunctions numerically determined by the program

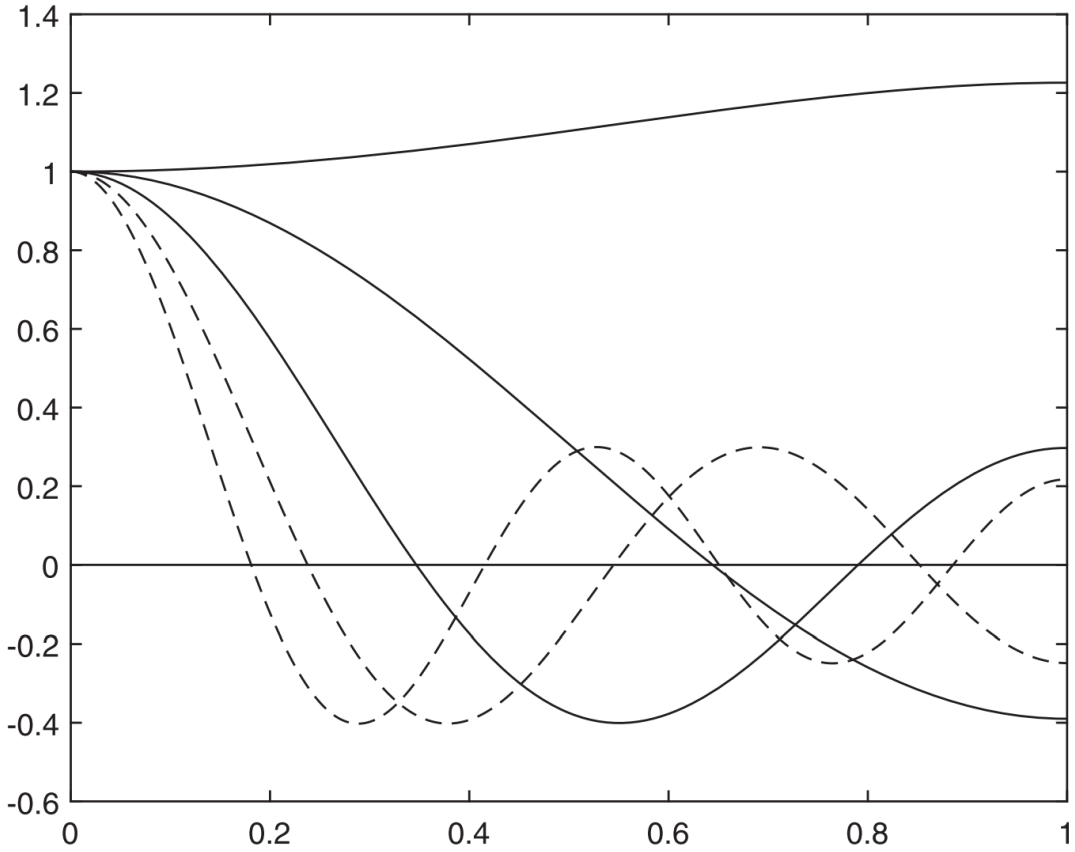


Figure 1.12: Graphs of the eigenfunctions as shown in the book [2]

Example (E.1.7). Let us consider the singular Sturm-Liouville problem of type 2

$$\begin{cases} -(xy')' + \dfrac{1}{x}y = \lambda xy, & 0 < x < 1, \\ |y(0)| < \infty, \quad y(1) + y'(1) = 0. \end{cases}$$

Numerically, we obtain the following eigenvalues:

n	λ_n	Intervals searched	Bisection iterations
0	5.78318596	579	24
1	30.47126234	3038	24
2	74.88700679	3218	28
3	139.04028443	3363	29
4	222.93230362	3494	30

It is interesting to observe here that, in order to find the eigenvalues, we need to relax the tolerance of the residual, F, from 10^{-6} to 10^{-3}, otherwise the program discards true eigenvalues as spurious. It was also necessary to significantly increase the number of integration steps to determine the eigenfunctions, in addition to decreasing the value of ε when integrating starting from $x = a + \varepsilon$.

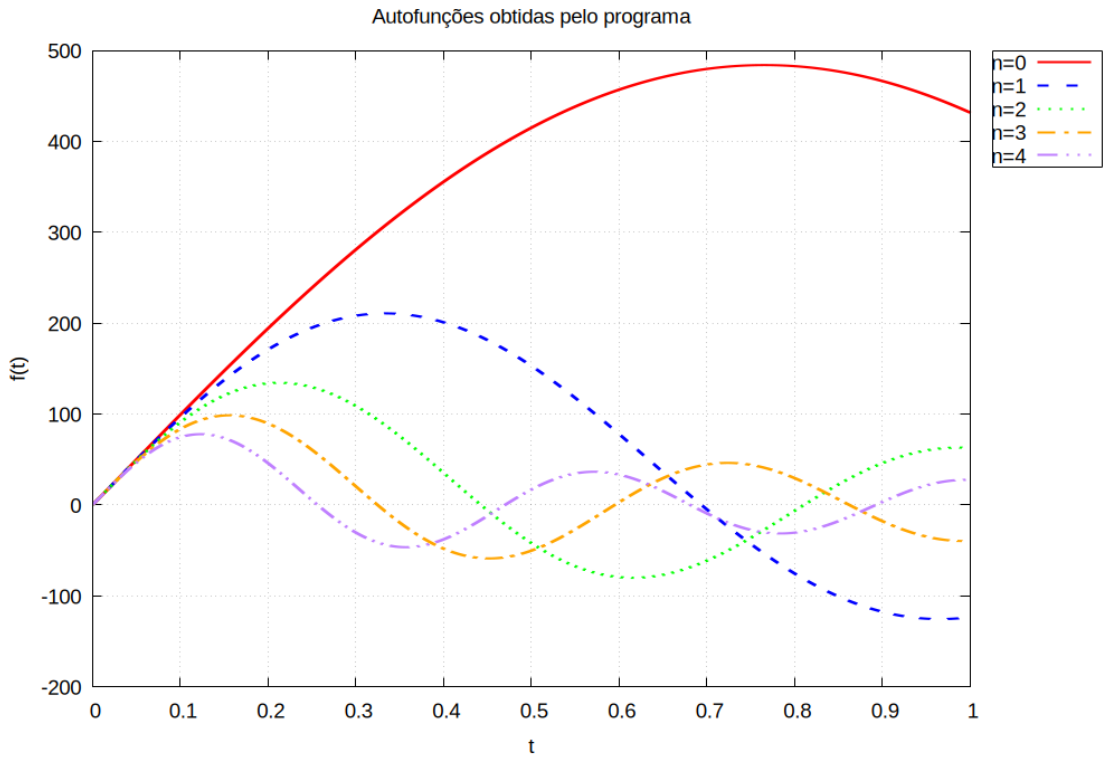


Figure 1.13: Eigenfunctions numerically determined by the program

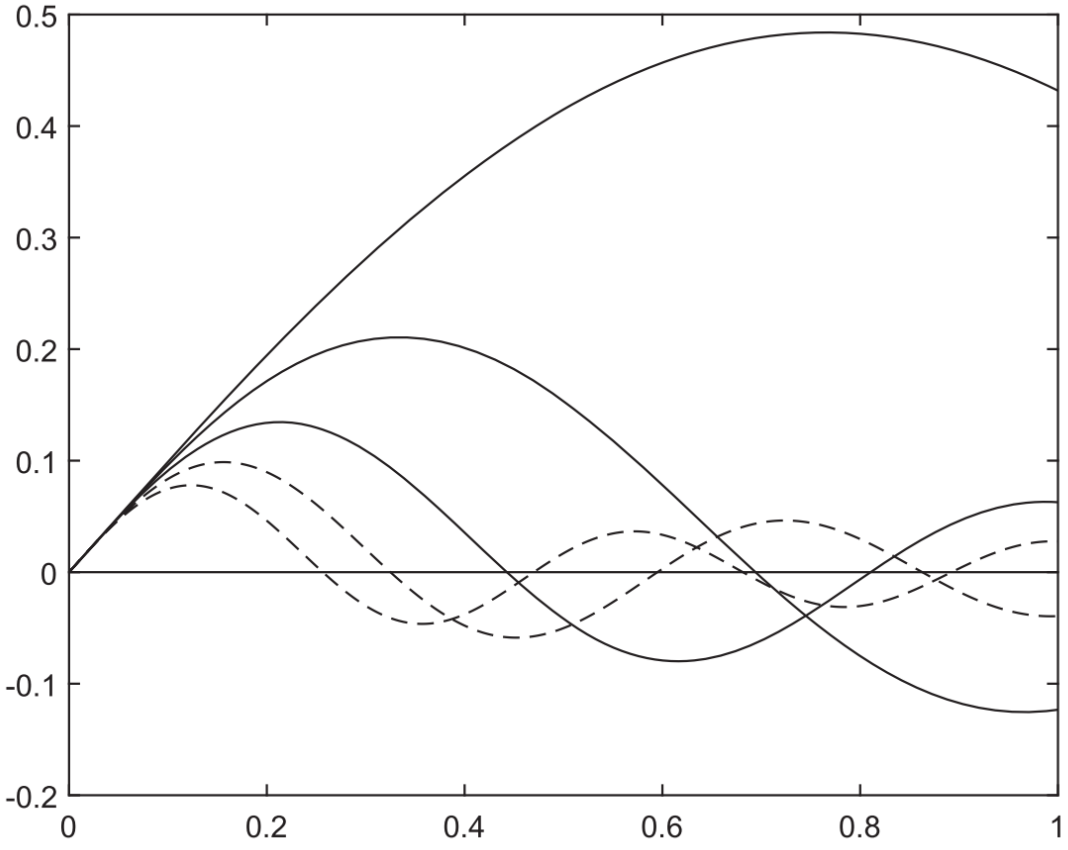


Figure 1.14: Graphs of the eigenfunctions as shown in the book [2]

Example (E.1.8). Let us consider the singular Sturm-Liouville problem of type 2

$$\begin{cases} -(xy')' + \dfrac{\cos(x)}{4x} y = \lambda(\sin x)y, \quad 0 < x < 1, \\ |y(0)| < \infty, \quad y(1) = 0. \end{cases}$$

Numerically, we obtain the following eigenvalues:

n	λ_n	Intervals searched	Bisection iterations
0	10.21624703	1022	24
1	41.53026555	4144	24
2	93.72409193	4311	29
3	166.79600338	4451	29
4	260.74576040	4580	30

Table 1.1: Results for the second set of eigenvalues

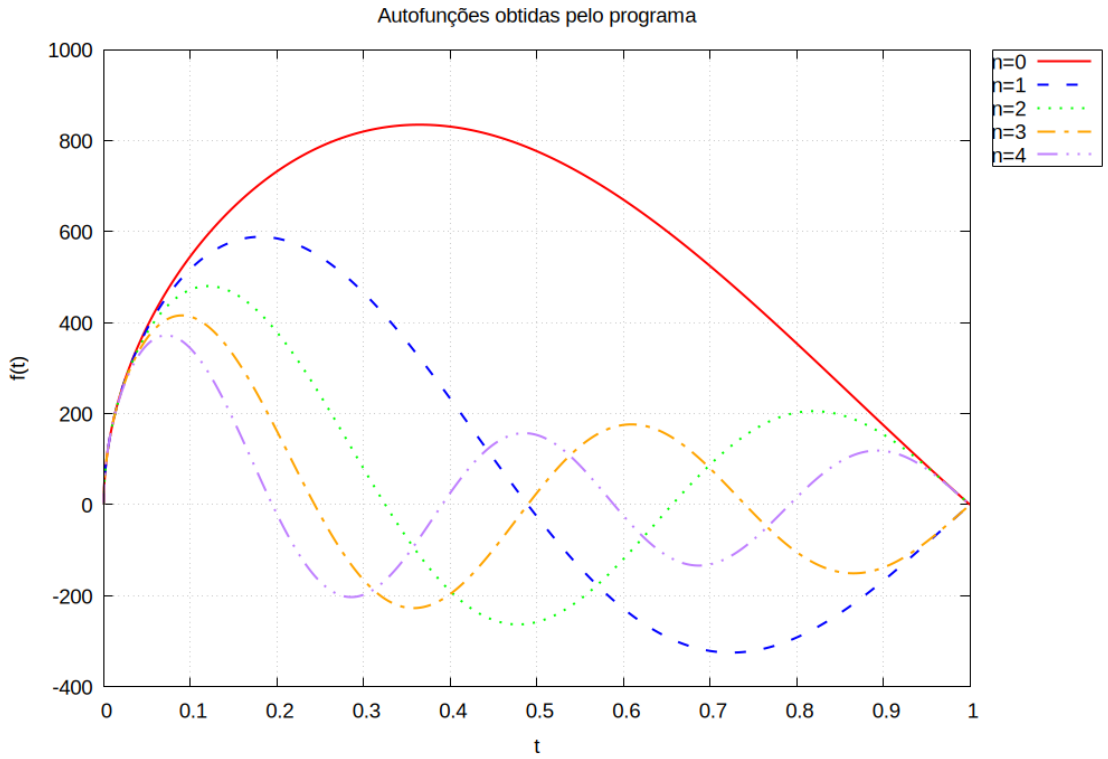


Figure 1.15: Eigenfunctions numerically determined by the program

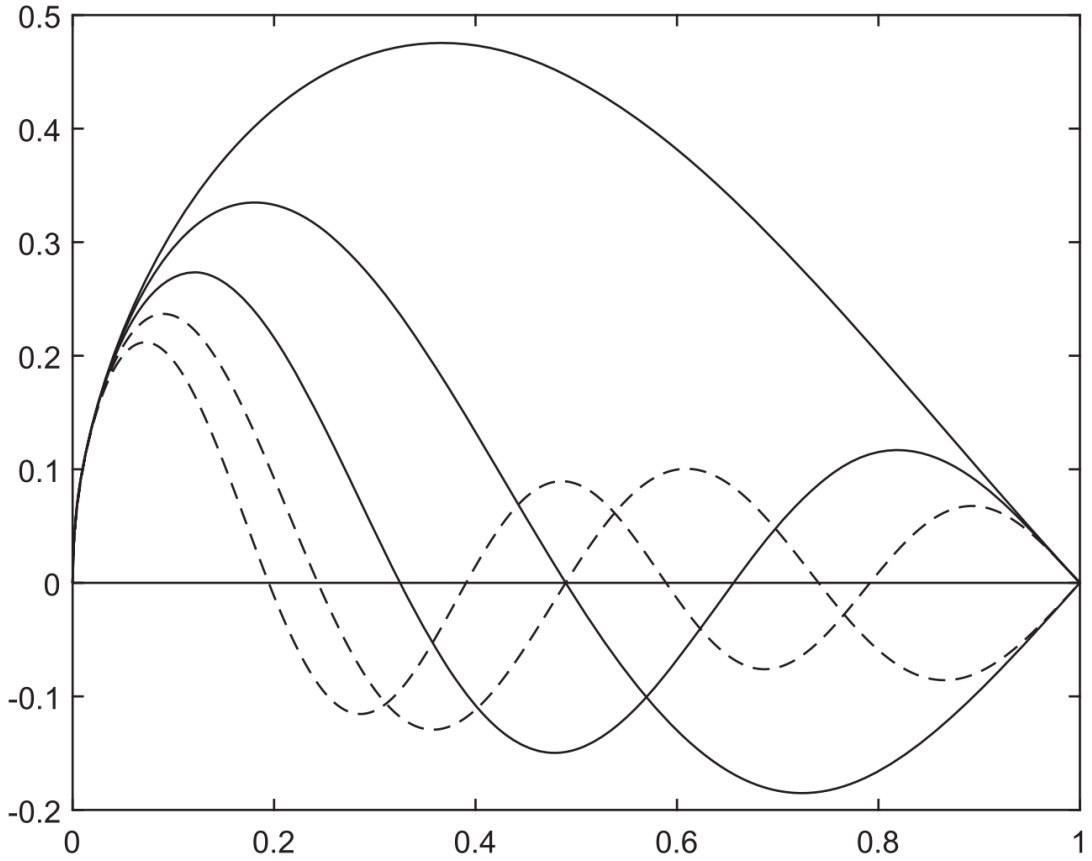


Figure 1.16: Graphs of the eigenfunctions as shown in the book [2]

Example (E.1.9). The Legendre equation can be seen as a singular Sturm-Liouville problem:

$$\begin{cases} -\dfrac{\mathrm{d}}{\mathrm{d}t}\left((1-t^2)\dfrac{\mathrm{d}P_n}{\mathrm{d}t}\right) = \lambda P_n, & \forall t \in [-1,1], \\ P(-1) = (-1)^n, \quad P(1) = 1. \end{cases}$$

It is known that the eigenvalues are given by $\lambda_n = n(n+1)$. The corresponding eigenfunctions from $n = 0$ to $n = 4$ are given by

n	Legendre polynomial
0	1
1	x
2	$\frac{1}{2}(3x^2 - 1)$
3	$\frac{1}{2}(5x^3 - 3x)$
4	$\frac{1}{8}(35x^4 - 30x^2 + 3)$

Numerically, we obtain the following eigenvalues:

n	λ_n	Intervals searched	Bisection iterations
0	0.00007518	1	24
1	2.00022555	192	24
2	6.00037594	387	25
3	12.00052635	535	26
4	20.00067675	667	26

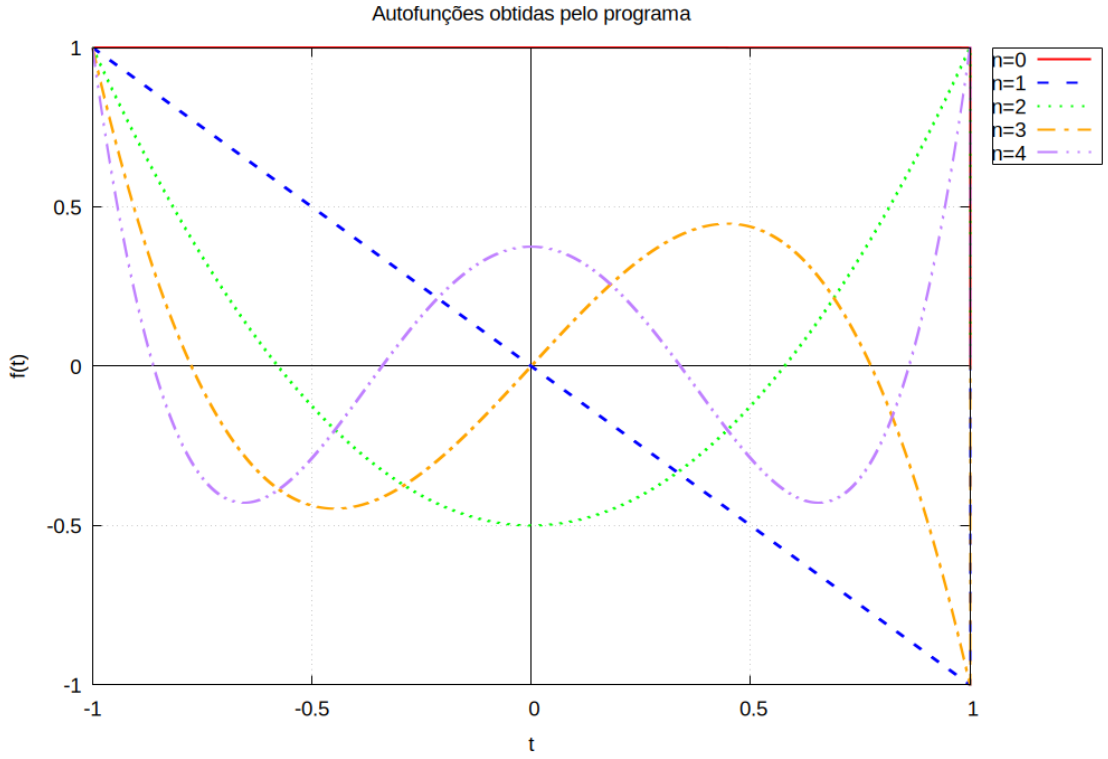


Figure 1.17: Eigenfunctions numerically determined by the program

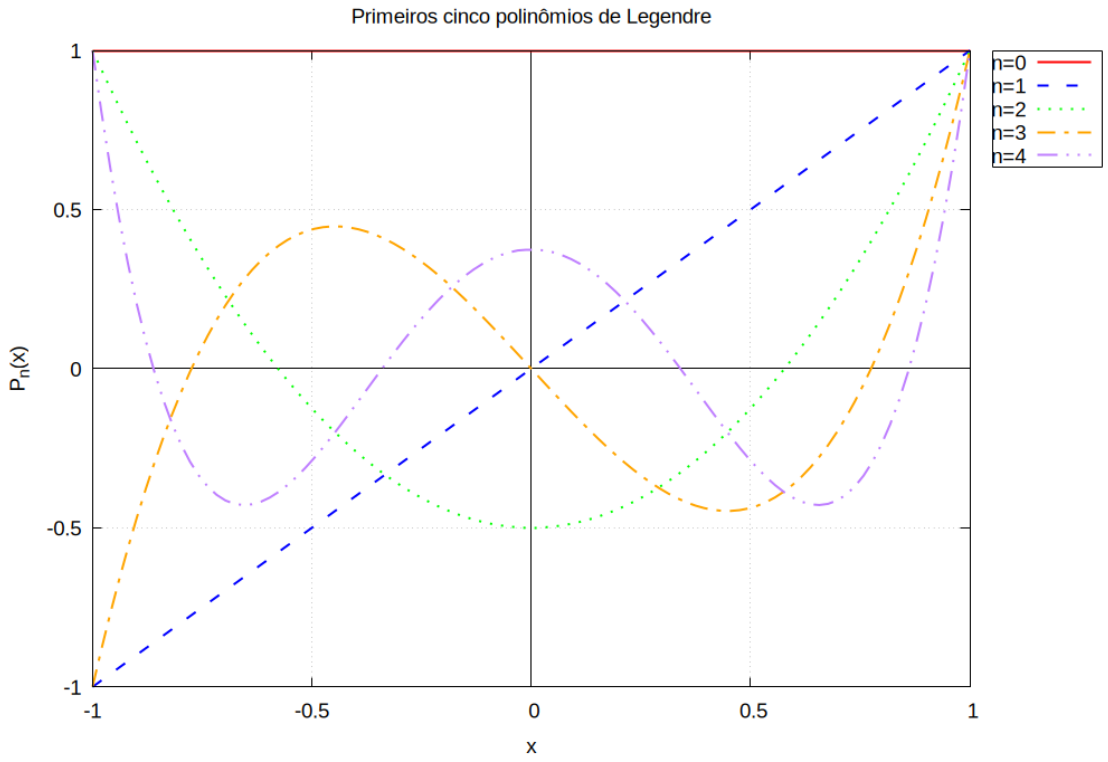


Figure 1.18: Graphs of the exact Legendre polynomials

We observe that the odd-degree polynomials numerically determined differ by the multiplicative constant -1 from the corresponding exact Legendre polynomials, which is expected, since the Legendre polynomials satisfy $P_n(-1) = (-1)^n$ and $P_n(1) = 1$, whereas in our program we always force $P_n(-1) = 1$.

1.5 Numerical results of the Newton-Raphson method

As we mentioned earlier, the marching approach using the bisection method faces great difficulties in being adapted to the Newton-Raphson method. In the program we developed, we implemented the Newton-Raphson method to find one eigenvalue at a time, with a good initial guess. The implementation is much simpler than that of the marching shooter method, but it cannot systematically determine the spectrum as before.

Example (E.1.10). For the problem

$$\begin{cases} -(1 \cdot u'(x))' + (x - \lambda \cos(x))u(x) = 0, \quad \forall x \in [0,1], \\ u'(0) = 0, \quad u'(1) = 0. \end{cases} \tag{1.17}$$

and with the initial guess $\lambda_0 = 0$, Newton-Raphson converged in only 4 iterations:

Iteration	λ	$F(\lambda)$
1	0.53761681266555983	-0.53403483428546394
2	0.57802783474609609	-0.034945050888573075
3	0.57824837086275171	-0.00018864919931886801
4	0.57824837740937718	-0.000000055997283388719053

With the initial guess $\lambda_0 = 140$, on the other hand, Newton-Raphson converged in 5 iterations:

Iteration	$\boldsymbol{\lambda}$	$\boldsymbol{F(\lambda)}$
1	60.773415429648637	-9.8908269771386994
2	46.353458505247033	4.6605507929687588
3	48.528804125931671	-0.83760614199634920
4	48.489185568458076	0.015769242002305423
5	48.489177842533614	0.00000307392121547741 14

It is somewhat surprising that the initial guess $\lambda_0 = 140$, closer to the theoretical eigenvalue

$$\lambda_3 = 107.57497472$$

converged to λ_2 instead of λ_3.

1.6 Analysis of the order of the shooting method with Newton-Raphson

We carried out an analysis of the order of the shooting method with Newton-Raphson. We set the tolerance of the residual $F(\lambda)$ to 10^{-16} and used RK4 again. The analysis was done using the Dirichlet problem:

$$\begin{cases} y'' + \lambda y = 0, \ \forall x \in [0,1]; \\ y(0) = y(1) = 0. \end{cases}$$

with initial guess $\lambda = 10$, and the evaluated point was $x_0 = 0.2$. We also observed that the closer to 0 we take x_0

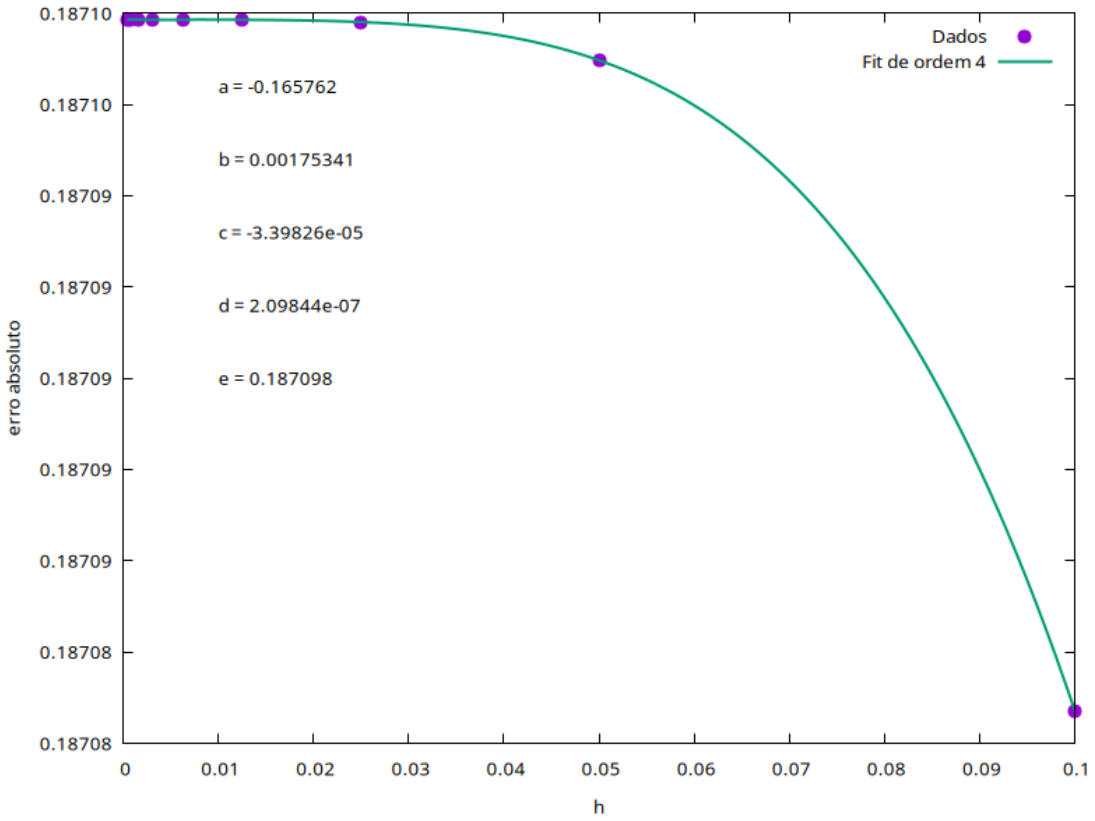


Figure 1.19: Convergence plot, h vs $y(0.2)$. Note that the coefficient e is indeed approximately $y_0(0.2)$.

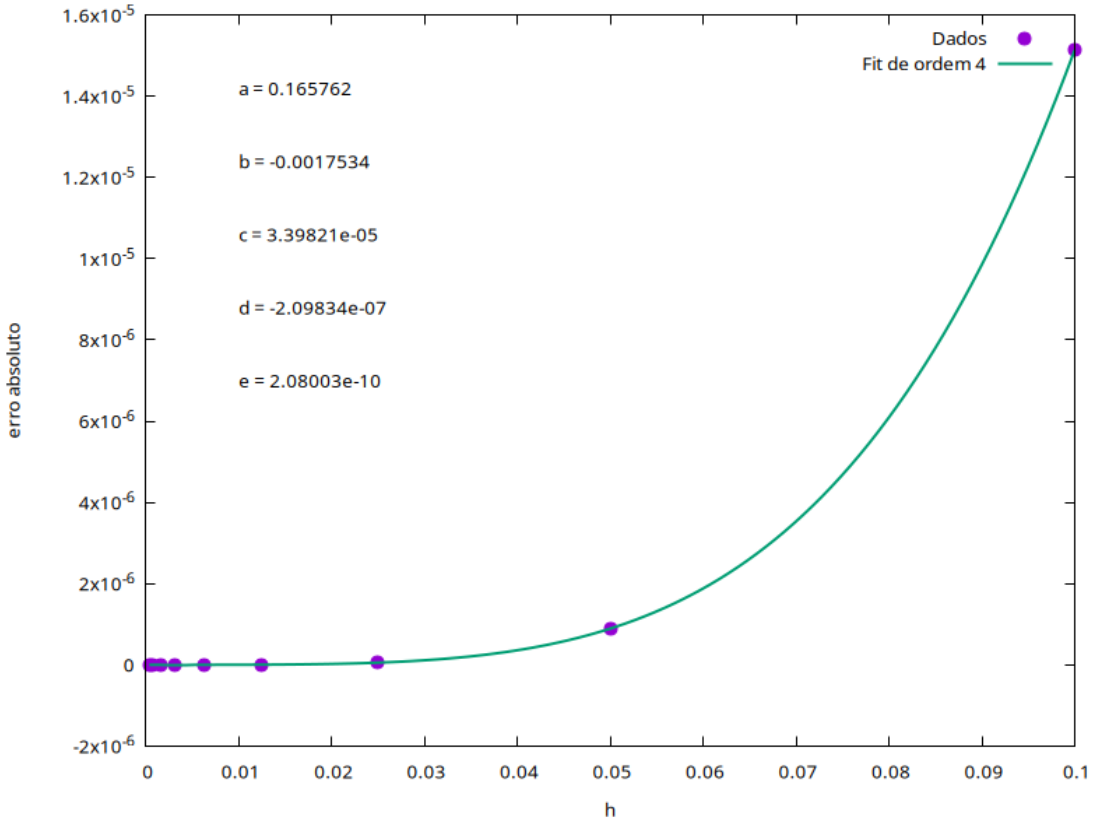


Figure 1.20: Fit of the local truncation error.

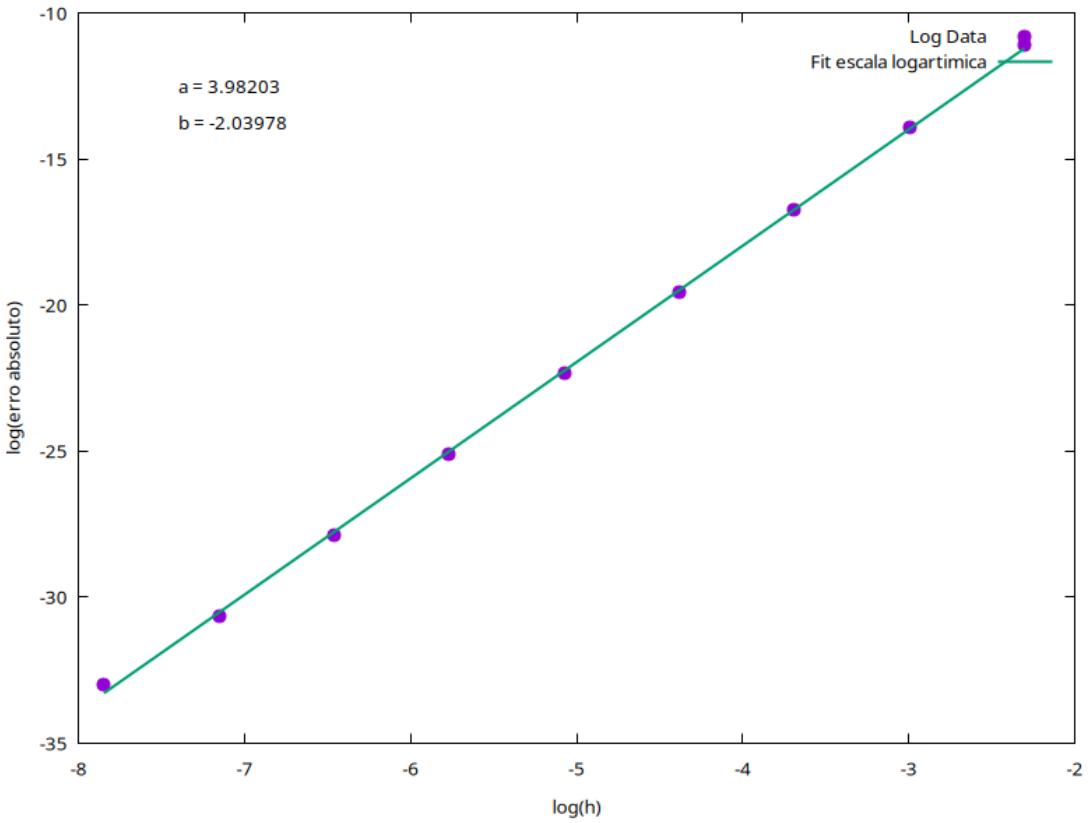


Figure 1.21: Linear fit of $\log(h)$ vs log of the local truncation error. Note that the slope is indeed approximately 4.

Bibliography

[1] **Kac, Marc.** "Can One Hear the Shape of a Drum?". *The American Mathematical Monthly*, vol. 73, no. 4, 1966, pp. 1–23. *JSTOR*, **https://doi.org/10.2307/2313748**.

[2] **Guenther, R.B; Lee, J.W.** (2019) *Sturm-Liouville problems: Theory and numerical implementation.*, Boca Raton: CRC Press, Taylor & Francis Group.

[3] **https://math.stackexchange.com/questions/1915313/what-is-the-motivation-for-the-equation-of-the-sturm-liouville-problem**.

[4] **https://math.stackexchange.com/questions/2268037/can-sturm-liouville-theory-actually-solve-odes**.

[5] **https://physics.stackexchange.com/questions/176041/the-sturm-liouville-equations-the-schrodinger-equation-and-the-wave-equation**.

[6] **https://github.com/SaganGromov/SL_REGULAR_EXEMPLOS**.

[7] **https://github.com/SaganGromov/SL_SINGULAR_EXEMPLOS**.

[8] **https://github.com/SaganGromov/SL_NEWTON_RAPHSON**.