Clase 5

Análisis del Error

Ejemplo 1: Cálculo de e

La fórmula para calcular una aproximación de la función $f\left(x\right)=e^{x}$ a través de los **Polinomios de Maclaurin** de grado n es la siguiente:

$$P_n\left(x\right)=\sum_{k=0}^{n}\frac{f^{(k)}\left(0\right)}{k!}\cdot x^{k}=\sum_{k=0}^{n}\frac{x^{k}}{k!}$$

El pseudocódigo para hallar la aproximación es el siguiente:

```
var
  n, grado, i: integer;
  tolerancia, x: real;
begin
  { Tomar el grado del polinomio }
  grado := InputInteger(edGrado);

  { Tomar el valor de la tolerancia }
  tolerancia := InputReal(edTolerancia);

  { Iniciar el grado del polinomio en 2 }
  n := 2;

  {Crear un ciclo condicional que aumente el valor del
  grado del polinomio a calcular mientras se verifica
  que la aproximación sea menor que la tolerancia}
  while ((3 / fact(n) >= tolerancia) and (n <= grado))
    do
  begin
```

```
      { Aumentamos el grado del polinomio }
      n := n + 1;
   end;

   { Determinar si se ha superado el grado pedido }
   if n >= grado then
   begin
      console.log('Error', 'Mensaje', 'No se puede
        alcanzar la aproximación pedida');
      Exit;
   end
   else
   begin
      { En caso de lograr la convergencia calcular la
        aproximación }
      x := 1; {Se inicia el valor de la aproximación en 1}

      { Crear el ciclo para calcular la aproximación
        mediante la serie de maclaurin }
      for i := 1 to n do
      begin
         { Calcular los valores de la sumatoria }
         x := x + (1 / fact(i));
      end;

      { Mostrar el valor calculado }
      console.log('Cálculo', 'Valor e', x);
   end;
end;
```

Ejercicio 1:

Crear un *programa* en **Python** que calcule una aproximación del valor de e dados el grado del polinomio y la tolerancia.

Programa de Google Colab

Este programa se ha desarrollado usando Google Colab. Para ver el programa haz clic en el logo.

5.1 Tipos de Algoritmos

Los algoritmos se dividen en dos clases bien diferenciables:

- **Estables**: un algoritmo en el que pequeños cambios en los valores iniciales provoca pequeños cambios en los resultados finales, se dice que es **estable**.
- **Inestables:** un algoritmo que en cambio, cambios pequeños en las condiciones iniciales provocan grandes diferencias en los resultados finales, se dice que es **inestable**.

Ejemplo 2:

Resolver la siguiente integral

$$I_n = \int_0^1 x^{2n} \sin(\pi x)\, dx$$

con $n = 0, 1, 2, \ldots, N$ donde N es un valor que nos dan.

Para resolver esta integral debemos integrar por partes. Sea $u = x^{2n}$ entonces $du = 2nx^{2n-1}dx$ y sea $dv = \sin(\pi x)\, dx$ entonces $v = -\frac{1}{\pi}\cos(\pi x)$, así

$$\begin{aligned} I_n &= \left[x^{2n}\left(-\frac{1}{\pi}\cos(\pi x)\right)\right]_0^1 - \int_0^1 -\frac{1}{\pi}\cos(\pi x)\, 2nx^{2n-1}dx \\ &= \left[-\frac{1}{\pi}x^{2n}\cos(\pi x)\right]_0^1 + \frac{2n}{\pi}\int_0^1 x^{2n-1}\cos(\pi x)\, dx \end{aligned}$$

Observación

Calculemos la expresión

$$\left[-\frac{1}{\pi}x^{2n}\cos\left(\pi x\right)\right]_0^1 = -\frac{1}{\pi}\left(1\right)^{2n}\cos\left(\pi\right) - \left[-\frac{1}{\pi}\left(0\right)^{2n}\cos\left(0\right)\right]$$
$$= \frac{1}{\pi}$$

Entonces la expresión se simplifica a

$$I_n = \frac{1}{\pi} - \int_0^1 -\frac{1}{\pi}\cos\left(\pi x\right) 2nx^{2n-1}dx$$

de nuevo sea $u = x^{2n-1}$ entonces $du = \left(2n-1\right)x^{2n-2}dx$ y $dv = \cos\left(\pi x\right)dx$ entonces $v = \frac{1}{\pi}\sin\left(\pi x\right)$, así,

$$\begin{aligned} I_n &= \frac{1}{\pi} + \frac{2n}{\pi}\left[\left[x^{2n-1}\cdot\frac{1}{\pi}\sin\left(\pi x\right)\right]_0^1 - \int_0^1 \frac{1}{\pi}\sin\left(\pi x\right)\left(2n-1\right)x^{2n-2}dx\right] \\ &= \frac{1}{\pi} + \frac{2n}{\pi}\left[\left[x^{2n-1}\cdot\frac{1}{\pi}\sin\left(\pi x\right)\right]_0^1 - \frac{2n-1}{\pi}\int_0^1 x^{2n-2}\sin\left(\pi x\right)dx\right] \\ &= \frac{1}{\pi} + \left[\frac{2n}{\pi^2}x^{2n-1}\sin\left(\pi x\right)\right]_0^1 - \frac{2n\left(2n-1\right)}{\pi^2}\int_0^1 x^{2n-2}\sin\left(\pi x\right)dx \end{aligned}$$

Observación

Calculemos la expresión

$$\left[\frac{2n}{\pi^2}x^{2n-1}\sin\left(\pi x\right)\right]_0^1 = \frac{2n}{\pi^2}\left(1\right)^{2n-1}\sin\left(\pi\right) - \frac{2n}{\pi^2}\left(0\right)^{2n-1}\sin\left(0\right)$$
$$= 0$$

Entonces la expresión se simplifica como

$$I_n = \frac{1}{\pi} - \frac{2n\left(2n-1\right)}{\pi^2}\int_0^1 x^{2n-2}\sin\left(\pi x\right)dx$$

de la siguiente expresión si calculamos el término $n-1$ tenemos lo siguiente,

$$I_n = \int x^{2n}\sin\left(\pi x\right)dx$$

entonces

$$I_{n-1} = \int x^{2(n-1)} \sin(\pi x)\, dx$$
$$= \int x^{2n-2} \sin(\pi x)\, dx$$

así en la integración que llevamos se transforma en

$$I_n = \frac{1}{\pi} - \frac{2n(2n-1)}{\pi^2} I_{n-1}$$

esto quiere decir que para calcular la integral I_n debemos usar la recurrencia.

Ejemplo 3:

Calcular el valor de I_0

$$I_0 = \int_0^1 x^{2(0)} \sin(\pi x)\, dx = \int_0^1 \sin(\pi x)\, dx$$
$$= \left[-\frac{1}{\pi}\cos(\pi x)\right]_0^1$$
$$= \left(-\frac{1}{\pi}\cos(\pi)\right) - \left(-\frac{1}{\pi}\cos(0)\right)$$
$$= \frac{1}{\pi} + \frac{1}{\pi} = \frac{2}{\pi} \approx 0.6366197$$

Ejercicio 2:

Se calculan los valores aproximados de I_n mediante la recurrencia. Determinar si el algoritmo planteado es estable o inestable.

Como tenemos el valor de $I_0 \approx 0.6366$ nuestras condiciones iniciales están perturbadas, es decir, tenemos un pequeño cambio respecto con el valor exacto de I_0. ¿Cómo afectará este pequeño cambio a los valores futuros I_n?.

Veamos

$$\begin{aligned} I_1 &= \frac{1}{\pi} - \frac{2}{\pi^2} \cdot I_0 \\ &= \frac{1}{\pi} - \frac{2}{\pi^2} \cdot \frac{2}{\pi} \\ &\approx 0.1893037 \end{aligned}$$

Los demás valores los podemos observar en la siguiente tabla

n	I_n	Aproximación
1	$\frac{1}{\pi} - \frac{4}{\pi^3}$	0.1893037
2	$\frac{\pi^4-12\pi^2+48}{\pi^5}$	0.0881441
3	$\frac{\pi^6-30\pi^4+360\pi^2-1440}{\pi^7}$	0.0503838
4	$\frac{\pi^8-56\pi^6+1680\pi^4-20160\pi^2+80640}{\pi^9}$	0.0324325
5	$\frac{\pi^{10}-90\pi^8+5040\pi^6-151200\pi^4+1814400\pi^2-7257600}{\pi^{11}}$	0.0225607

Ejercicio 3:

Crear un *programa* en **Python** que calcule los valores de la integral para los primeros 20 valores de n usando al recurrencia.

Programa de Google Colab

Este programa se ha desarrollado usando Google Colab. Para ver el programa haz clic en el logo.

Vemos que los valores I_{15} e I_{17} son negativos pero la función que integramos $x^{2n}\sin(\pi x)$, es positiva para todo valor de $x \in (0,1)$ en el intervalo de integración. Por tanto, no puede haber valores negativos.

Ejercicio 4:

Crear un *programa* en **Python** en el que se grafique la función dada para n valores en el intervalo para $x \in (0,1)$

¿Nos hemos equivocado?

No, el algoritmo aplicado es inestable, veamos porqué.

Sea ΔI_n el error que tenemos del valor de I_{n-1}, como

$$I_n = \frac{1}{\pi} - \frac{2n(2n-1)}{\pi^2} I_{n-1}$$

lo que calculamos en la práctica es

$$I_n + \Delta I_n = \frac{1}{\pi} - \frac{2n(2n-1)}{\pi^2}\left(I_{n-1} + \Delta I_{n-1}\right)$$

Para simplificar nuestro análisis, vamos a suponer que el número π no tiene error, entonces de la igualdad anterior tenemos que

$$\Delta I_n = -\frac{2n(2n-1)}{\pi^2}\Delta I_{n-1}$$

Es decir, en el paso $n-$ésimo el error de I_{n-1} se multiplica por $\frac{2n(2n-1)}{\pi^2}$. Dicha sucesión tiende a infinito a medida que n crece, de lo que deducimos que

$$\lim_{n\to\infty} \Delta I_n = \infty$$

y por lo tanto, el algoritmo es inestable.

Ejercicio 5:

Crea un programa en Python que grafique la sucesión de términos para

$$s(n) = \frac{2n(2n-1)}{\pi^2}$$

para los primeros 100 valores de n

¿Cómo podemos calcular de forma eficiente los valores de I_n?.

Fijémonos que si escribimos la recurrencia "al revés", obtenemos

$$I_{n-1} = \frac{\pi}{2n\left(2n-1\right)} - \frac{\pi^2}{2n\left(2n-1\right)} I_n$$

Observación

Sea la expresión, deseamos despejar I_{n-1}

$$\begin{aligned} I_n &= \frac{1}{\pi} - \frac{2n\left(2n-1\right)}{\pi^2} I_{n-1} \\ I_n - \frac{1}{\pi} &= -\frac{2n\left(2n-1\right)}{\pi^2} I_{n-1} \\ \pi^2 I_n - \frac{\pi^2}{\pi} &= -2n\left(2n-1\right) I_{n-1} \\ \frac{\pi^2 I_n}{-2n\left(2n-1\right)} + \frac{\pi}{2n\left(2n-1\right)} &= I_{n-1} \end{aligned}$$

así,

$$I_{n-1} = \frac{\pi}{2n\left(2n-1\right)} - \frac{\pi^2}{2n\left(2n-1\right)} I_n$$

Si realizamos el análisis del error que hemos hecho antes, tenemos que si aplicamos la recurrencia anterior, los errores ΔI_n verifican:

$$\Delta I_{n-1} = -\frac{\pi^2}{2n\left(2n-1\right)} \Delta I_n$$

Es decir, se van multiplicando por $-\frac{\pi^2}{2n(2n-1)}$ y como dicha sucesión tiende a cero, los errores van disminuyendo y el algoritmo sería estable.

Ahora bien, aplicar la recurrencia anterior no tiene mucho sentido ya que tendríamos que calcular I_{n-1} en función de I_n.

Recordemos la definición de los valores I_n

$$I_n = \int_0^1 x^{2n} \sin\left(\pi x\right) dx$$

Podemos demostrar que

$$\lim_{n\to\infty} I_n = 0$$

Es decir,

$$|I_n| = \left| \int_0^1 x^{2n} \sin(\pi x)\, dx \right| \leq \int_0^1 x^{2n} dx = \left[\frac{x^{2n+1}}{2n+1} \right]_0^1 = \frac{1}{2n+1}$$

Aplicando el criterio del "Emparedado" para limites de sucesiones tenemos que como

$$\lim_{n\to\infty} \frac{1}{2n+1} = 0$$

Entonces

$$\lim_{n\to\infty} I_n = 0$$

Entonces para aplicar el algoritmo anterior estable, escojamos un valor de n tal que $|I_n| < \epsilon$, donde ϵ sería la tolerancia. Por ejemplo $\epsilon = 0.001$, el valor de n sería

$$\begin{aligned} |I_n| \leq \frac{1}{2n+1} &\leq 0.001 \\ \Rightarrow 2n+1 &\geq \frac{1}{0.001} = 1000 \\ n &= \frac{1000-1}{2} = 499.5 \end{aligned}$$

entonces $n \geq 500$. Empezaríamos con $n = 500$, usando que $I_{500} \approx 0$ yendo hacia atrás usando la recurrencia anterior calcularíamos los valores para I_n para $n = 499, \ldots, 0$

Ejercicio 6:

Crear un programa en Python que determine los valores de I_n desde $n = 499, \ldots, 0$

5.2 Velocidad de Convergencia

Muchos de los algoritmos basados en métodos numéricos consisten en hallar una sucesión de números reales convergente. Por tanto, es importante de cara a estudiar la eficiencia del algoritmo en cuestión saber a qué velocidad converge la sucesión.

En este apartado, vamos a definir formalmente la **velocidad de convergencia** de una sucesión.

Definición 1: Definición de Equivalencia de Velocidad de Convergencia

Sea $(x_n)_n$ una sucesión de números reales que converge a L e $(y_n)_n$ otra sucesión convergente a cero. Diremos que $(x_n)_n$ tiene **orden de convergencia** $(y_n)_n$ si existe una constante $K > 0$ y un natural n_0 tal que:

$$|x_n - L| \leq K\,|y_n|\,,$$

para todo $n \geq n_0$. En este caso, se escribe que $x_n = L + O\left(y_n\right)$

La definición anterior nos permite reducir la convergencia de una sucesión a un valor cualquiera a sucesiones que convergen a cero.

Un tipo de sucesiones $(y_n)_n$ que convergen a cero que se suelen usar son $y_n = \frac{1}{n^k}$, con k natural.

Ejemplo 4:

La sucesión $x_n = \frac{P_p(n)}{Q_q(n)}$, donde $P_p\left(n\right)$ y $Q_q\left(n\right)$ son polinomios de grados p y q, respectivamente, con $q > p$ tienen orden de convergencia $\frac{1}{n^{q-p}}$.

En primer lugar, escribimos

$$P_p\left(n\right) = a_p n^p + a_{p-1} n^{p-1} + \cdots + a_0,$$
$$Q_q\left(n\right) = b_q n^q + b_{q-1} n^{q-1} + \cdots + b_0.$$

Entonces

$$\lim_{n\to\infty} x_n = \frac{P_p\left(n\right)}{Q_q\left(n\right)} = 0$$

Ahora bien,

$$\begin{aligned}\frac{P_p(n)}{Q_q(n)} &= \frac{a_p n^p + a_{p-1}n^{p-1} + \cdots + a_0}{b_q n^q + b_{q-1}n^{q-1} + \cdots + b_0} \\ &= \frac{a_p n^p \left(1 + \frac{a_{p-1}}{a_p n} + \cdots + \frac{a_0}{a_p n^p}\right)}{b_p n^q \left(1 + \frac{b_{p-1}}{b_p n} + \cdots + \frac{b_0}{b_p n^q}\right)} \\ &= \frac{a_p}{b_p} \cdot \frac{1}{n^{q-p}} \cdot \left(1 + \frac{a_{p-1}}{a_p n} + \cdots + \frac{a_0}{a_p n^p}\right) \cdot \left(1 - O\left(\frac{1}{n}\right)\right) \\ &= \frac{a_p}{b_p} \cdot \frac{1}{n^{q-p}} \cdot \left(1 + O\left(\frac{1}{n}\right)\right)\left(1 - O\left(\frac{1}{n}\right)\right) \\ &= \frac{a_p}{b_p} \cdot \frac{1}{n^{q-p}} \cdot \left(1 + O\left(\frac{1}{n}\right)\right)\end{aligned}$$

Entonces, podemos escribir que existen valores $K = \left|\frac{a_p}{b_q}\right|$ y n_0 tal que:

$$\left|\frac{P_p(n)}{Q_q(n)}\right| \leq K \frac{1}{n^{q-p}}$$

para $n \geq n_0$, tal como queríamos ver.

Observación

$1 - O\left(\frac{1}{n}\right)$ proviene del desarrollo de Maclaurin. Si desarrollamos $\frac{1}{1+x}$ se da origen al término anterior.

5.2.1 Error en los algoritmos

Existen algoritmos que, en lugar de darnos una sucesión como resultado, nos dan una expresión cuyo error depende de un parámetro h, supuestamente pequeño. Interesa que el error de los algoritmos tenga la forma $K \cdot h^t$, con K y $k > 0$ constantes. Cuanto mayor sea la constante k, mejor es el algoritmo o más rápido converge a la solución.

Por dicho motivo, vamos a definir el orden de convergencia para funciones numéricas que dependen de un valor continuo h:

Ejemplo 5: Definición de equivalencia de velocidad de convergencia

Sean F y G dos funciones reales de variable real. Supongamos que existe un valor L tal que

$$\lim_{h\to 0} F(h) = L$$

y

$$\lim_{h\to 0} G(h) = 0.$$

Diremos que $F(h)$ tiene orden de convergencia $G(h)$ si existen constantes $K > 0$ y $c > 0$ tal que:

$$|F(h) - L| \leq K\,|G(h)|\,,$$

Si $|h| \leq c$. En este caso, se escribe que $F(h) = L + O(G(h))$. Un tipo de funciones $G(h)$ que convergen a cero que suelen usar son $G(h) = h^k$, con $k > 0$

Ejemplo 6:

Veamos que la función $F(h) = e^h$ tiene orden de convergencia $G(h) = h$.

En este caso, el valor de L sería $\lim\limits_{h\to 0} F(h) = \lim\limits_{h\to 0} e^h = 1$. Entonces, usando la expresión del error del desarrollo de MacLaurin de la función e^h tenemos que:

$$F(h) = e^h = 1 + e^{\xi} h$$

Con $\xi \in \langle 0, h\rangle$. Por tanto, existe un valor $c > 0$ tal que si $|h| \leq c$,

$$|F(h) - 1| = \left|e^h - 1\right| = e^{\xi}\,|h| \leq \max\left\{e^h, 1\right\}|h| \leq e^c\,|h| = K\,|h|$$

Donde hemos usando que $e^{\xi} \leq e^h$ si $h > 0$ y $e^{\xi} \leq 1$, si $h < 0$

En general, si consideramos $F_k(h) = e^h - \left(1 + h + \frac{h^2}{2} + \cdots + \frac{h^k}{k!}\right)$ tenemos que $F_k(h)$ tiene orden de convergencia h^{k+1}. Usando la expresión del error del desarrollo de MacLaurin de la función e^h de orden $k+1$, tenemos que:

$$F_k(h) = e^h - \left(1 + h + \frac{h^2}{2} + \cdots + \frac{h^k}{k!}\right) = \frac{e^{\xi}}{(k+1)!} h^{k+1},$$

Con $\xi \in \langle 0, h \rangle$, Por tanto, existe un valor $c > 0$ tal que si $|h| \leq c$,

$$\begin{aligned} |F_k(h)| &= \left| e^h - \left(1 + h + \frac{h^2}{2} + \cdots + \frac{h^k}{k!} \right) \right| \\ &= \left| \frac{e^\xi}{(k+1)!} h^{k+1} \right| \leq \frac{\max\left\{ e^h, 1 \right\}}{(k+1)!} \left| h^{k+1} \right| \\ &\leq \frac{e^c}{(k+1)!} \left| h^{k+1} \right| = K \left| h^{k+1} \right| \end{aligned}$$

Tal como queríamos ver.

5.3 Taller del Capítulo

1. Usando aritmética de tres dígitos significativos, da el valor que resulta de realizar la operación siguiente: $\frac{\sqrt{15}+\sqrt{5}}{\sqrt{15}-\sqrt{5}}$
2. Escribir el número -2.5 en formato de 64 bits
3. Escribe el número decimal que es representado por el siguiente formato de 64 bits.

   0 | 1001001000011111100111110000000110111000011001101111 | 100000

4. Calcular los errores absoluto y relativo para las aproximaciones siguientes $\widehat{x}$ y x

   a) $x = \pi$ y $\widehat{x} = \frac{22}{7}$

   b) $x = e$ y $\widehat{x} = 2.7183$

   c) $x = \sqrt{3}$ y $\widehat{x} = 1.732$

5. Realizar las operaciones siguientes de tres maneras: exactamente, cortando usando aritmética de 4 dígitos significativos y redondeando usando aritmética de 4 dígitos significativos. Comparar los errores relativos en las dos últimas formas.

   a) $\frac{2}{5} + \frac{1}{7}$

   b) $\left(\frac{1}{5} - \frac{2}{11}\right) - \frac{5}{9}$

6. El número e se define como $e = \sum_{n=0}^{\infty} \frac{1}{n!}$. Hallar los errores absoluto y

relativos en la aproximación siguiente de e

$$e \approx \sum_{n=0}^{6} \frac{1}{n!}$$

7. Dada la ecuación de segundo grado $ax^2 + bx + c = 0$, para hallar sus raíces se pueden usar las fórmulas

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \text{ y } x_{1,2} = \frac{2c}{b \pm \sqrt{b^2 - 4ac}}$$

   a) Demostrar la equivalencia de las fórmulas anteriores.

   b) Usando aritmética de 4 dígitos significativos redondeando, resolver la ecuación de segundo grado

$$1.002x^2 + 11.01x + 0.01265 = 0$$

8. Queremos calcular $x = \left(7 - 4\sqrt{3}\right)^4$, usando 1.73 como aproximación a $\sqrt{3}$, ¿Cuál de las siguientes expresiones es más adecuada para calcular x?

   a) $f_1(x) = \frac{1}{\left(7+4\sqrt{3}\right)^4}$

   b) $f_2(x) = \left(97 - 56\sqrt{3}\right)^2$

   c) $f_3(x) = \frac{1}{\left(97+56\sqrt{3}\right)^2}$

   d) $f_4(x) = 18817 - 10864\sqrt{3}$

   e) $f_5(x) = \frac{1}{18817+10864\sqrt{3}}$

9. Queremos resolver la ecuación de segundo grado $x^2 - \frac{2001}{2000}x + \frac{1}{100} = 0$, usando aritmética de 3 dígitos significativos. ¿Cuál es el error relativo cometido al calcular la raíz menor usando la fórmula $\frac{-b-\sqrt{b^2-4ac}}{2a}$?

10. Calcular la velocidad de convergencia de las sucesiones siguientes:

   a) $\lim_{n\to\infty} \sin\left(\frac{1}{n}\right) = 0$

   b) $\lim_{n\to\infty} \ln(n+1) - \ln(n) = 0$

11. Calcular la velocidad de convergencia de las funciones siguientes cuando $h \neq 0$

a) $\lim_{h \to 0} \frac{\sin(h)}{h} = 1$

b) $\lim_{h \to 0} \frac{\sin(h) - h\cos(h)}{h} = 0$

12. Consideremos la sucesión de Fibonacci $F_{n+2} = F_{n+1} + F_n$, con $F_0 = F_1 = 1$. Los primeros términos de dicha sucesión serán los siguientes:

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

Dicha sucesión F_n puede escribirse explícitamente de la forma siguiente donde dicha fórmula puede demostrarse por inducción:

$$F_n = \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^n - \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^n$$

a) Supongamos que tenemos $\sqrt{5}$ con un error absoluto acotado por E. Usando la fórmula de propagación del error y suponiendo que no cometemos error en las operaciones aritméticas básicas (suma, resta, multiplicación y división), estime en función de e el error absoluto que cometemos al aplicar la fórmula explicita anterior cuando calculamos F_n.

b) Supongamos ahora que usamos la fórmula recurrente $F_{n+2} = F_{n+1} + F_n$ y suponiendo que F_0 y F_1 tienen un error acotado por E, estime el valor del error absoluto de F_n

c) Considere $n = 0, 1, \ldots, 50$ y $E = 0.000001$

- Calcule la sucesión $\hat{F}_n$ del apartado a), para $n = 1, 2, \ldots, 50$ suponiendo que $\sqrt{5}$ vale aproximadamente 2.236069 en el primer caso
- Calcule la sucesión $\tilde{F}_n$ del apartado b), para $n = 1, 2, \ldots, 50$ suponiendo que $F_0 = F_1 = 1.000001$ en el segundo caso.
- Realice un gráfico de los errores cometidos usando los dos métodos y compruebe qué método es el mejor.
- ¿Podrías dar una explicación teórica el fenómeno observado en el paso anterior?