

Introducción

El cálculo científico es parte de lo que se conoce como ciencia de la computación, cuya investigación incluye las fases:

- 1) Desarrollo de un modelo matemática para un fenómeno de interés
- 2) Desarrollo de un algoritmo
- 3) Implementación del algoritmo en un software
- 4) Simulaciones numéricas del fenómeno con el software
- 5) Representación de los resultados calculados (gráficas o herramientas visuales)
- 6) Interpretación y validación de los resultados

El cálculo científico típicamente se encarga de las fases 2-4: desarrollo, implementación y uso de los algoritmos numéricos y software para aplicaciones en análisis y ajuste de modelos a datos, optimización / machine learning, entre otras.

Análisis numérico y cálculo científico

Nuestro curso involucra temas del análisis numérico.

El análisis numérico se enfoca al diseño y análisis de algoritmos para resolver problemas que surgen en la ciencia de la computación e ingeniería, por esto, al análisis numérico, se le ha llamado cálculo científico, sin embargo se distingue de otras áreas de la ciencia de la computación en que trabaja con cantidades continuas.

Muchos problemas de matemáticas con este tipo de cantidades, no se resuelven, en principio, en un número finito de pasos, por lo que deben resolverse por un proceso iterativo que converja a la solución, teóricamente infinita. Sin embargo, en la práctica, no se realizan procesos infinitos, y además no se resuelven problemas de manera exacta, se obtienen respuestas

aproximadamente correctas, "suficientemente cercanas o precisas" al resultado deseado. Por esto, uno de los aspectos más importantes del cálculo científico es encontrar algoritmos iterativos que converjan rápidamente, así como realizar una estimación de la exactitud de las aproximaciones calculadas.

Como consecuencia de lo anterior, un segundo tema que distingue al análisis numérico, es el relacionado con las aproximaciones y sus efectos:

Durante todo el trayecto que uno realiza para resolver un problema, se realizan aproximaciones y algunas de ellas no las controlamos directamente. Es indispensable entonces, que los algoritmos, además de tener propiedades que los caracterizan como "buenos algoritmos" como la eficiencia (hacer más con menos), sean confiables y

exactos ante la serie de aproximaciones que se han realizado durante el trayecto: estabilidad. La eficiencia se relaciona con el costo computacional, medida típicamente con el número de operaciones que realiza el algoritmo, y la estabilidad se refiere a no amplificar los errores generados por las aproximaciones o resultados calculados durante la ejecución del algoritmo y se relaciona con la pregunta: ¿Si se perturban los datos de entrada, el algoritmo calcula una solución "cercana" a los datos no perturbados? → análisis de sensibilidad

Algunas fuentes de error que nos encontramos al resolver un problema surgen por:

-) Uso de modelos → simplificación u omisión
-) Mediciones con instrumentos
-) Cálculos previos → "input data"

Estas fuentes de **error** típicamente están fuera de nuestro control y en el cálculo científico se estudian dos fuentes del **error** que influyen en el algoritmo utilizado para resolver un problema:

- ⇒) Truncamiento: relacionado con el uso de procesos o cantidades finitas
- ⇒) Redondeo: relacionado con la representación de los números y la aritmética en una máquina

Así, estas fuentes de **error** influyen en la **exactitud** de un cálculo, y las perturbaciones que resulten de estas fuentes, serán amplificadas (o no) por la naturaleza del problema y el tipo de algoritmo utilizado. El estudio de la **exactitud** y estabilidad de un algoritmo por fuentes de error se le llama **análisis del error**:

Un problema típicamente puede verse como evaluar una función

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

Sea x el dato de entrada y $f(x)$ el valor de salida deseado.

Al trabajar con una máquina, nos encontramos con que

la representación de un número real en general no es exacta,

por ello tenemos \hat{x} una aproximación a x (redondeo)

y nuestra forma de calcular f es por medio de un

algoritmo \hat{f} , entonces, el error al evaluar f es:

$$\text{Error total} = \hat{f}(\hat{x}) - f(x)$$

$$= \underbrace{\hat{f}(\hat{x}) - f(\hat{x})}_{\begin{array}{l} \text{Error} \\ \text{Computacional} \end{array}} + \underbrace{f(\hat{x}) - f(x)}_{\begin{array}{l} \text{Error en} \\ \text{datos} \end{array}}$$

El 1º término es generado por el algoritmo utilizado; \hat{f} (cálculos y aproximaciones realizadas) y el 2º término por

debido a mediciones o redondeos realizadas por la máquina,
observa que este término no se ve influenciado por el
algoritmo utilizado.

El error computacional tiene los componentes⁶, truncamiento y
redondeo mencionados anteriormente. El truncamiento
es la diferencia entre el resultado verdadero y el resultado
que se obtiene con un algoritmo usando aritmética exacta,
por ejemplo: reemplazar una serie infinita (resultado verdadero)
con una serie truncada (algoritmo; truncar serie infinita)
y evaluar la serie truncada con aritmética exacta
(usar $\frac{1}{3}$ en lugar de 0.333 p. ej.). El error por redondeo
es la diferencia entre el resultado obtenido por un algoritmo
usando aritmética exacta y el resultado producido por el
mismo algoritmo usando aritmética de máquina

Aunque los errores por truncamiento y redondeo son componentes del error total, un tipo de error típicamente domina sobre otro, y hacer la distinción entre ellos, nos ayuda a entender el comportamiento de los algoritmos y los factores que influyen en su exactitud.

NOTA SOBRE EXACTITUD Y PRECISIÓN

Los errores en cálculos y medidas se pueden caracterizar con respecto a su exactitud y su precisión. La exactitud se refiere a qué tan cercano está el valor calculado o medido del valor verdadero. La precisión se refiere a qué tan cercanos se encuentran, unos de otros, diversos valores calculados o medidos.

