Análisis numérico

Clase 4: Preliminares, algoritmos y computación

Joaquin Cavieres

Instituto de Estadística, Universidad de Valparaíso

Outline

Error de análisis

Fuentes de error

- Formulación del modelo matemático.
    - Número de datos
    - Incertidumbre asociada a los datos
    - Planteamiento del problema
- Formulación del modelo discretizado
    - Tipo de modelo discretizado
    - Aproximación de datos y variables
    - El problema está mal condicionado
- Selección de métodos resolutivos
    - Errores en algoritmos
    - Errores en máquinas

Error de análisis

Los análisis desarrollados con una calculadora o una computadora difieren de la aritmética que se imparte en los cursos de álgebra y cálculo. Por ejemplo, si expresamos $2 + 2 = 4$, $5 * 5 = 25$ y $(\sqrt{3})^2$ siempre sean verdaderas, sin embargo, en la aritmética *computacional* esperamos resultados exactos para $2 + 2 = 4$ y $5*5 = 25$, pero no obtendremos los resultados exáctos para $(\sqrt{3})^2$.

Error de análisis

Los análisis desarrollados con una calculadora o una computadora difieren de la aritmética que se imparte en los cursos de álgebra y cálculo. Por ejemplo, si expresamos $2+2=4$, $5*5=25$ y $(\sqrt{3})^2$ siempre sean verdaderas, sin embargo, en la aritmética *computacional* esperamos resultados exactos para $2+2=4$ y $5*5=25$, pero no obtendremos los resultados exáctos para $(\sqrt{3})^2$.

Para comprender lo expuesto anteriormente debemos explorar la aritmética de los dígitos finitos.

Error de análisis

Generalmente, dentro de nuestro mundo matemático tradicional, trabajamos con números con una cantidad infinita de dígitos. Por ejemplo, si consideramos a $\sqrt{3}$ sabemos que hay un único número positivo que cuando se multiplica por sí mismo produce el número 3. Pero, dentro de la arimética computacional, cada número puede ser representado a través de un número dijo y finito de digítos.

Para nuestro ejemplo, con $\sqrt{3}$ que es un número racional, se establece una solución aproximada en donde el cuadrado de será exactamente 3 y aceptable en la mayoría de las situaciones.

Definición: Precisión

La precisión proporciona una medida de lo específico que puede llegar a ser un número. Es decir, la precisión explica el nivel de detalle en una medición.

Números binarios de máquinas

Los computadores tienen la funcionalidad para los números enteros y de punto flotante (o de máquina) para representar números reales (números enteros y números reales respectivamente). Para representar un número de punto flotante se permiten números de gran tamaño pero con limitaciones de magnitud como en el número de dígitos. La representación de punto flotante está relacionada específicamente con la notación científica, la cual facilita la representación de este tipo de números reales con una cierta precisión para así simplificar las operaciones aritméticas usuales que con ellos se realizan.

Números binarios de máquinas

Un digíto binario de 64 bits es utilizado para representar un número real. El primer bit es un indicador de signo y lo representamos con la letra **s**, luego le sigue un exponente de 11 bits al cual le asignamos la letra **c** y se le conoce como la **característica**, y una fracción binaria de 52 bits, asignado con la letra f que es la **mantisa**. La base para el exponente es 2.

Podemos asumir que los 52 digítos binarios corresponden a un número, representado en este sistema, que tiene al menos 16 digítos decimales de precisión.

Para ahorrar almacenamiento y generar una representación única para cada número de punto flotante, imponemos una normalización de la forma:

$$(-1)^s 2^{c-1023}(1+f)$$

Ejemplo

Considere el siguiente número:

0 10000000011 1011100100010000000000000000000000000000000000000000

El bit de más a la izquierda es $\mathbf{s} = 0$ y nos indica un número positivo. Los siguientes 11 números (bits) 10000000011 son la característica ($\mathbf{c}$) y representa al número decimal:

$$c = 1 * 2^{10} + 0 * 2^{9} + ... + 0 * 2^{2} + 1 * 2^{1} + 1 * 2^{0} = 1024 + 2 + 1 = 1027$$

La parte exponencial del número es $2^{1027-1023} = 2^4$. Finalmente, los siguientes 52 bits representan la mantisa, la cual se representa como:

$$f = 1 * (\frac{1}{2})^{1} + 1 * (\frac{1}{2})^{3} + 1 * (\frac{1}{2})^{4} + 1 * (\frac{1}{2})^{5} + 1 * (\frac{1}{2})^{8} + 1 * (\frac{1}{2})^{12}$$

Ejemplo

Por lo tanto, el número de máquina representa el siguiente número decimal

$$\begin{aligned}(-1)^s 2^{c-1023}(1+f) &= \\ &= (-1)^0 * 2^{1027-1023}\left(1+\frac{1}{2}+\frac{1}{8}+\frac{1}{16}+\frac{1}{32}+\frac{1}{256}+\frac{1}{4096}\right) \\ &= 27.56640625\end{aligned}$$

Error de análisis

¿Que pasa si consideramos al siguiente número de máquina más pequeño?

0 10000000011 1011100100001111111111111111111111111111111111111111

o al siguiente número de máquina más grande?

0 10000000011 1011100100010000000000000000000000000000000000000001

¿Que pasa si consideramos al siguiente número de máquina más pequeño?

0 10000000011 1011100100001111111111111111111111111111111111111111

o al siguiente número de máquina más grande?

0 10000000011 1011100100010000000000000000000000000000000000000001

Lo anterior nos indica que el número de máquina inicial representa:

- La mitad de los números reales que se encuentran entre 27.56640625 y el siguiente número de máquina más pequeño.
- La mitad de los números reales que se encuentran entre 27.56640625 y el siguiente número de máquina más grande.

Error de análisis

El número positivo normalizado más pequeño se puede escribir mediante $s = 0$, $c = 1$ y $f = 0$ como:

$$2^{-1022} * (1 + 0) \approx 0.22251 \times 10^{-307}$$

y el más grande con $s = 0$, $c = 2046$ y $f = 1 - 2^{-52}$:

$$2^{1023} * (2 - 2^{-52}) \approx 0.17977 \times 10^{309}$$

- Los números con magnitud menor a $2^{-1022} * (1 + 0)$ resultan en un **subdesbordamiento** y generalmente son asignados a 0.
- Los números con magnitud mayor a $2^{1023} * (2 - 2^{-52})$ resultan en un **desbordamiento** y generalmente son utilizados para parar un calculo.

Números decimales de máquinas

Cuando usamos digítos binarios generalmente se ocultan dificultades computacionales, como por ejemplo, cuando usamos una colección finita de números de máquina para representar los números reales. Así, en este caso, es más simple utilizar números decimales que una representación binaria.

Números decimales de máquinas

Cuando usamos digítos binarios generalmente se ocultan dificultades computacionales, como por ejemplo, cuando usamos una colección finita de números de máquina para representar los números reales. Así, en este caso, es más simple utilizar números decimales que una representación binaria.

Particularmente, vamos a representar los números de máquina en una estructura normalizada de punto flotante decimal de la siguiente manera:

$$\pm 0.d_1 d_2, ..., d_k \times 10^n, \;\; 1 \leq d_1 \leq 9 \;\; \text{y} \;\; 0 \leq d_i \leq 9$$

para cada $i = 1, ..., k$. Este tipo de números se les conoce como números de máquina decimales de digíto k

Así, cualquier número real positivo que esta contenido en el intervalo numérico de la máquina se puede normalizar de la forma:

- Método de **corte**: Consiste en cortar los digítos $d_{k+1}d_{k+2}...$ y produce el punto flotante:

$$fl(y) = 0.d_1 d_2, ..., d_k \times 10^n$$

- Método de **redondeo**: Consiste en sumar $5 \times 10^{n-(k+1)}$ a y y luego cortar el resultado para obtener:

$$fl(y) = 0.\delta_1 \delta_2, ..., \delta_k \times 10^n$$

Para redondear, considerando a $d_{k+1} \geq 5$, sumamos 1 a d_k y así se obtiene $fl(y)$ (redondeo hacia arriba). Si consideramos $d_{k+1} < 5$, entonces se corta todo excepto los primeros digítos k, así redondeamos hacia abajo.

Ejemplo

Calcule el valor de corte y de redondeo de cinco digítos del numéro π.

El número π tiene una estructura infinita de la forma $\pi = 3.14159265....$ Si lo normalizamos tendría la siguiente forma:

$$\pi = 0.314159265... \times 10^1$$

Ejemplo

Calcule el valor de corte y de redondeo de cinco digítos del numéro π.

El número π tiene una estructura infinita de la forma $\pi = 3.14159265....$ Si lo normalizamos tendría la siguiente forma:

$$\pi = 0.314159265... \times 10^1$$

- Método de corte: usando el este método y asumiendo 5 digítos tenemos:

$$fl(\pi) = 0.31415 \times 10^1 = 3.1415$$

- Método de redondeo: El sexto digíto del número π es 9, por tanto mediante este método tenemos que:

$$fl(\pi) = (0.31415 + 0.00001) \times 10^1 = 3.1416$$

Errores de aproximación

Definiciones: *Accuracy*

La accuracy mide qué tan cerca está una estimación del valor real. Supongamos que p^* es una aproximación de p. De lo anterior podemos definir a:

- error real $= p - p^*$
- error absoluto $= |p - p^*|$
- error relativo $= \frac{|p-p^*|}{|p|}$, siempre y cuando $p \neq 0$.

Ejemplo

Encuentre el error real, absoluto y relativo de:

- $p = 0.3000 \times 10^1$ y $p^* = 0.3100 \times 10^1$
- $p = 0.3000 \times 10^{-3}$ y $p^* = 0.3100 \times 10^{-3}$

Ejemplo

Encuentre el error real, absoluto y relativo de:

- $p = 0.3000 \times 10^1$ y $p^* = 0.3100 \times 10^1$
- $p = 0.3000 \times 10^{-3}$ y $p^* = 0.3100 \times 10^{-3}$

- Para $p = 0.3000 \times 10^1$ y $p^* = 0.3100 \times 10^1$ el error real es -0.1, error aboluto 0.1 y el error relativo $0.333\bar{3} \times 10^{-1}$
- Para $p = 0.3000 \times 10^{-3}$ y $p^* = 0.3100 \times 10^{-3}$ el error real es -0.1×10^{-4}, error absoluto -0.1×10^{-4} y el error relativo es $0.333\bar{3} \times 10^{-1}$

Error de análisis

Lo anterior nos indica que, como medida de precisión, el error absoluta puede resultar engañoso en muchas ocaciones y el error relativo más significativo devido a que considera el tamaño del valor en cuestion. Pero, ¿como podemos evaluar el límite aceptable de error?

Error de análisis

Lo anterior nos indica que, como medida de precisión, el error absoluta puede resultar engañoso en muchas ocaciones y el error relativo más significativo devido a que considera el tamaño del valor en cuestion. Pero, ¿como podemos evaluar el límite aceptable de error?

El límite aceptable de error es un número no negativo y que es mayor que el valor error absoluto.

Definición

p^* se aproxima a p para t digítos significativos si t es el entero no negativo más grande cuando se cumple:

$$\frac{|p - p^*|}{|p|} \leq 5 \times 10^{-t}$$

Error de análisis

Si bien una mayor precisión podría parecer tener mayores beneficios, también puede causar problemas. Un cálculo puede inducir una mayor precisión en el resultado final de lo que se justifica. A esto se le llama falsa precisión. Esto surge cuando se presenta una estimación muy precisa para una medida.

Ejemplo: Una caja que puede contener 5 kilogramos podría convertirse en 11.0231 libras. En este caso, la precisión es generada por el factor de conversión, 2.20462 libras por kilogramo. Pero decir que la caja tiene capacidad para 11.0231 libras agrega una medida de precisión que probablemente no está justificada y, ciertamente, es fácil de malinterpretar.

Resumen

- Un cálculo debe tener como objetivo la mayor precisión y exactitud sin desperdiciar recursos computacionales
- Los números introducidos en un algoritmo para los calculos pueden ser solo aproximaciones cercanas.
- La cantidad de error que podemos permitir en un cálculo depende completamente de las circunstancias. A esto lo llamamos tolerancia al error.

Burden, R. L., & Faires, J. D. (2011). Numerical analysis.

Howard, J. P. (2017). Computational Methods for Numerical Analysis with R. CRC Press.

Banerjee, S., & Roy, A. (2014). Linear algebra and matrix analysis for statistics. Crc Pr