Métodos Numéricos
8 de junio de 2016
Taller de Interpolación e Integración

DEPARTAMENTO DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

Agencia de Inteligencia

La agencia de inteligencia MNI6[1] está planeando instalar cámaras de seguridad en toda la ciudad para controlar los movimientos de individuos potencialmente peligrosos. Un algoritmo de procesamiento de imágenes reconoce a los individuos y los marca con un punto en un plano de la ciudad. Para disminuir el riesgo de que una imagen sea descubierta por un intruso en la red, desean realizar la menor cantidad posible de capturas, pero manteniendo una precisión de metros en la estimación de la posición del individuo en cada momento.

Concretamente, la posición de un individuo en función del tiempo se modela utilizando una curva:

$$f(t) = (X(t), Y(t))$$

donde $X(t) : \mathbb{R} \to \mathbb{R}$, $Y(t) : \mathbb{R} \to \mathbb{R}$ corresponden a la posición (km) en x y en y respectivamente en función del tiempo t (h). Luego $X'(t), Y'(t)$ corresponden a la velocidad ($\frac{km}{h}$) en función del tiempo y $X''(t), Y''(t)$ a la aceleración ($\frac{km}{h^2}$) en función del tiempo. Se estima que el módulo de la aceleración de un ser humano promedio y en cualquier dirección de avance siempre es menor que $1000\frac{km}{h^2}$.

En este contexto se desea determinar el mayor intervalo de tiempo posible (Δt) entre cada par de imágenes capturadas, de manera tal que el error de la interpolación lineal de la posición del individuo en todo momento sea menor a 10^{-3} km (precisión de metros). Para ello:

1. Suponiendo que se tomaron las mediciones $\{(t_0, f(t_0)), \ldots, (t_n, f(t_n))\}$, modelar el problema utilizando interpolación fragmentaria lineal. Calcular analíticamente el máximo valor posible de $\Delta t_i = t_{i+1} - t_i$ para cualquier i, acotando el error de interpolación lineal en cada intervalo.

Se tienen muestras de caminatas de ciudadanos ficticios de 30 minutos de duración cada una. Se desea completar una implementación inicial de la cátedra en Matlab y realizar pruebas empíricas sobre el valor de Δt hallado previamente. Todas las pruebas pueden realizarse ejecutando el script `interpolacion/main.m`, completando el valor de Δt (medido en horas) hallado en el ejercicio anterior. Para simplificar, ya se han realizado todas las conversiones de unidades necesarias en el código, con lo cual podrán trabajar sobre kilómetros y horas.

2. Completar la implementación del algoritmo de interpolación lineal fragmentaria, en el archivo `interpolacion/interpolacionFragmentariaLineal.m`.

3. Verificar que la cota del error hallada se cumpla para el ciudadano "Mareado". ¿La cota era ajustada? ¿Cuál es el mayor valor de Δt para el cual se obtiene precisión de 10^{-3} km?

4. Comparar el uso de interpolación fragmentaria lineal con Splines Cúbicos para los ciudadanos "Mareado" y "Kane", usando $\Delta t = \frac{1}{60}h$ (1 minuto) . ¿Disminuye el error al usar Splines? ¿Mejora gráficamente la estimación de la trayectoria? ¿Por qué?

[1] MetNum Intelligence 2016.

Bit Error Rate

La transmisión de dígitos binarios entre dos extremos de un cable se puede realizar de diversas maneras. Una de ellas, llamada *Polar NRZ*, consiste en representar al dígito `1` como un voltaje positivo (A) y al `0` como un voltaje negativo ($-A$). Ambos voltajes tienen el mismo valor absoluto, llamado *amplitud*.

Cuando el emisor envía una cadena de dígitos binarios al receptor, este último sensa el voltaje del cable y determina el dígito:

$$digito(voltaje) = \begin{cases} 1 & \text{si } 0 \leq voltaje \leq 2A \\ 0 & \text{si } -2A \leq voltaje < 0 \end{cases}$$

A causa del ruido térmico, el voltaje que el emisor pone en el cable no necesariamente es el mismo que el receptor encontrará en el otro extremo. El ruido térmico típicamente se modela con una variable aleatoria de distribución normal (Gaussiana) de media $\mu = 0$, cuya varianza σ^2 representa la intensidad del ruido.

El *Bit Error Rate* (*BER*) es la tasa de errores cometidos por el receptor al interpretar el voltaje descripto.

$$BER = \frac{\text{cantidad de bits interpretados de forma errónea}}{\text{cantidad de bits recibidos}}$$

El *BER* puede calcularse como la probabilidad de que el error Gaussiano altere el voltaje v a tal extremo que $digito(v)$ devuelva el dígito incorrecto o se indefina. Esto sucederá en el caso en que el módulo del error ($|E|$) sea mayor a la amplitud A. Entonces se deduce la siguiente fórmula:

$$BER(A, \sigma) = P(|E| > A) = 1 - \frac{2}{\sqrt{\pi}} \int_0^{\frac{A}{2\sigma\sqrt{2}}} e^{-x^2} dx$$

Se desea estimar el *BER* para un canal con ruido térmico de intensidad $\sigma^2 = \frac{1}{2}$ y amplitud de comunicación $A = 2\ volts$. Esto significa calcular:

$$BER(1, \frac{1}{\sqrt{2}}) = 1 - \frac{2}{\sqrt{\pi}} \int_0^1 e^{-x^2} dx$$

Para esto debemos calcular la integral propia de la función e^{-x^2}, para la cual no existe primitiva (no existe g tal que $g'(x) = e^{-x^2}$).

5. Indicar cuántos puntos se deben tomar en la aproximación de

$$\int_0^1 e^{-x^2} dx$$

por medio de la regla de los Trapecios Compuesta para que el error sea menor que 10^{-6}. Idem con la regla de Simpson Compuesta.

Se tienen funciones y scripts en Matlab para comparar la solución obtenida con la solución calculada por Matlab. En el archivo `integracion/main.m` se encuentra una rutina para experimentar con el valor de n hallado (cantidad de puntos menos 1).

6. Completar la implementación del algoritmo de integración de Simpson Compuesto, en el archivo `integracion/simpson.m`.

7. Obtener el valor de la integral utilizando la cantidad de puntos hallada previamente y comparar con la solución provista por Matlab. ¿Se obtuvo la precisión esperada?

8. ¿Podría haberse obtenido la precisión deseada utilizando menos puntos? ¿Cuántos si usamos trapecios? ¿Cuántos si usamos Simpson?

Ejercicios Opcionales

9. Dados los puntos:

x_i	1	2	4	5
$f(x_i)$	0	2	12	20

   Calcular el polinomio interpolador de grado menor o igual que 3 usando el método de:

   a) Lagrange.

   b) Diferencias divididas.

10. Dados los puntos $(-1, 3), (1, 1), (2, 3), (3, 7)$, determinar cuántos polinomios de grado d existen que pasen por todos los puntos, para

   a) $d = 2$

   b) $d = 3$

   c) $d = 4$

   Para cada valor de d, en caso de ser posible, mostrar uno.

Evaluación

- Coloquio con los docentes durante las clases del miércoles 8 y viernes 10 de junio.
- En caso de no asistir a clase, se debe entregar la resolución por escrito hasta el Viernes 24 de Junio, incluidos los Ejercicios Opcionales.