

ANÁLISIS APLICADO PROYECTO II

CONDICIONES PARA ENTREGAR EL PROYECTO

1. Cada equipo debe tener de 2 a 3 miembros.
Deben registrar un conjunto de parámetros para la 2^a función (página 2).
2. Fecha de entrega: [Domingo 1 de Diciembre a las 23:55 en comunidad.itam.](#)
3. Resolver las tareas abajo.
4. Entregar el código completo en un archivo comprimido de formato **.zip**.
 - Para cada tabla de resultados en su documentación un *script* que reproduce el los números en la tabla.
 - El código debe reproducir ciertos datos en su documentación **.pdf**. (No espero que tiempos coincidan.)
 - Todo lo que se requiere para correr el código de ser contenido en el **zip**, *e.g.* si aproximan el Gradiente con nuestra función hecha en Lab. 1, entonces hay que incluir esa función.
5. Entregar un documento en formato **.pdf** que junta y comenta los resultados obtenidos. No quiero código en el documento.

EL PROYECTO

1. Funciones en MatLab u Octave.

Dicho en clase (el 6 de Noviembre de 2019):

En este proyecto deben comparar los siguientes métodos:

- **lineBGFS**: búsqueda en línea con “BGFS updates” de la inversa de la Hessiana (Lab. 8).

$$(n \leq 200)$$

- **mRCSR1**: región de confianza usando formulas de tipo: “symmetric rank one update” (Lab. 9)

$$(n \leq 200)$$

- **lineLMBGFS**: búsqueda en línea con “limited memory BGFS updates” de la inversa de la Hessiana (Lab. 10), para $m \in \{1, 3, 5, 17, 29\}$ y

$$(n \leq 1000)$$

Para obtener un máximo de 9.0 como calificación (de este proyecto) pueden usar todo lo que hicimos juntos en los laboratorios. Si quieren como máximo un 10, entonces pido que implementan este método con memoria cíclica. La idea de esto fue explicada en el laboratorio del 13 de Noviembre.

Ayuda: En este caso el más nuevo elemento ya no tiene posición constante. Se pueden ocupar funciones como `mod` para definir el orden en que utilizar los vectores anteriores. Requiere pensar un poco, pero solo es una aplicación de Álgebra.

2. Lo que se debe hacer. Se deben comparar tiempos e iteraciones que requerían los métodos para las problemas abajo con las dimensiones especificadas. Para cada método las dimensiones n deben respetar los límites arriba. Además, se debe documentar los siguientes parámetros de la computadora que usan:

RAM, CPU y cache de la CPU.

2.1. Parámetros centrales:

$$c_1 = 10^{-4}, \quad c_2 = 0.99, \quad \eta = 0.1, \quad tol = 10^{-5}, \quad \Delta_{max} = 1.5.$$

2.2. Problemas.

En community.itam.edu encuentran el siguiente documento:

CUTE_UnconstrainedOptimizationTestFunctionsCollection.pdf

De este documento se deben implementar las siguientes dos funciones y hacer los experimentos

- Rosenbrock (extended, p. 149) para

$$n \in \{2, 10, 100, 200, 1000\}$$

- DIXMAANA (p. 155) para una letra que tienen que reservar por correo y para

$$n \in \{200, 1000\} \quad \text{y} \quad m = n/3 \quad (\text{la parte entera de la división}).$$

Las siguientes letras (conjuntos de parámetros) ya están reservadas:

A, C, E, G, I, D, H