

OPTIMIZACIÓN - 1ER CUATRIMESTRE 2014

Trabajo Práctico 2 - Estudio de métodos de Descenso

Objetivo del Trabajo:

El objetivo del trabajo es realizar un estudio exhaustivo de los métodos de descenso trabajados en clase. En este caso, el trabajo requerirá de la confección de un informe final con los resultados obtenidos. Las funciones y rutinas de matlab no necesitan ser diseñadas para el usuario, aunque si deberán mantener los principios de programación que fueron delineados en las sucesivas clases y el código deberá estar comentado.

Se deberán implementar los siguientes algoritmos de descenso:

- A0 Algoritmos de Matlab (`fminsearch`, `fminunc`)
- A1 Gradiente: En donde $d_k = -\nabla f(x_k)$
- A2 Newton generalizado: Como en el Ejercicio 16 de la Práctica 3.
- A3 Gradiente Conjugado: Como en el Ejercicio 22 de la Práctica 3.
- A4 Quasi-Newton (BFGS): Como se vió en la clase. Ver libro de Martinez.
- A5 Algun algoritmo extra para cada grupo.

Los algoritmos anteriores deben tener las siguientes opciones con respecto al gradiente y al hessiano:

- D1 Ingresarlos: Con lo cual el input seria algo de la forma $[f, \nabla f, \nabla^2 f]$.
- D2 Calcularlos numéricamente (por ejemplo con las rutinas del TP0)

También se deberán implementar rutinas de búsqueda lineal:

- B1 Búsqueda lineal exacta: Usando `fminbnd`.
- B2 Búsqueda lineal inexacta con corrección de Armijo y Backtracking: Como el paso (5) Del Ejercicio 15 de la Práctica 3.
- B3 Ajuste cuadrático: Como se vio en la clase. Ver libro de Luenberger.
- B4 Ajuste cúbico: Como se vio en la clase. Ver libro de Luenberger.

Estos algoritmos deberán ser implementados de manera que todos tengan como Inputs la función objetivo f y un punto inicial x_0 . Además habrá parámetros que pueden resumirse todos en una variable de *opciones*.

Un ejemplo de opciones podría ser:
opciones=[MaxNumIter, tolGrad, tolIter, gradHess, alpha, beta, theta]
en donde:

- Op1 MaxNumIter: Máximo número de iteraciones.
- Op2 tolGrad: tolerancia para determinar si el gradiente es cero.
- Op3 tolIter: tolerancia para determinar si dos iteraciones son iguales.
- Op4 gradHess: opcion para ingresar el gradiente y el hessiano o para usar una rutina que los calcule.
- Op5 alpha,beta,theta: parámetros propios de un método
- Opn alguna otra opción que se considere pertinente.

Intentar sí, que si diferentes métodos usan iguales parámetros estén codificados de la misma manera.

En el archivo `testFunctions.zip` hay un gran conjunto de funciones para implementar los algoritmos de minimización. El mismo está explicado Aquí,
<http://www.mathworks.com/matlabcentral/fileexchange/23147-test-functions-for-global-optimization-algorithms>

También se pueden encontrar funciones Aquí,
http://en.wikipedia.org/wiki/Test_functions_for_optimization

Cada grupo tendrá asignado un conjunto de funciones para testear.

Informe:

Para la confección del informe la sugerencia es usar la funcion Publish dentro de Matlab. Publish se aplica a un script que no requiere ninguna entrada de usuario. Dicho informe debería incluir tiempos de corrida de los algoritmos, distancia relativa al mínimo (mejor aproximación conocido), parametros que se usaron para las corridas. Un breve análisis comparativo (cualitativo y cuantitativo) de los métodos para las funciones estudiadas.