**Universidad de Chile**
Facultad de Ciencias Físicas y Matemáticas
Departamento de Ingeniería Matemática
MA3711-1 Optimización Matemática

# Programación Para Problemas de Optimización

## Métodos Cuasi-Newton y de Barrera

Integrantes: Sebastián Cobaise

Arturo Lazcano

Profesor: Héctor Ramírez
Auxiliares: María José Alfaro
Javier Madiaraga
Rolando Roggers
2 de Agosto de 2020

# 1. Introducción

Los problemas de optimización son un campo muy estudiado en las matemáticas , son problemas de utilidad cotidiana y muy cercanos a la realidad. Es por esto que en este informe se estudiarán problemas de optimización irrestrictos donde el objetivo será minimizar funciones, al menos dos veces continuamente diferenciables, mediante programación de métodos de descenso y luego incluyendo restricciones de tipo desigualdad para implementar el método de barrera. Se procederá a usar los famosos métodos de Davidson-Fletcher-Powell (DFP) y Broyden-Fletcher-Goldfarb-Shanno (BFGS) para minimizar las siguientes funciones:

- $f(x, y) = x^2 + y^2 \qquad x, y \in \mathbb{R}$
- $f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)^2 \qquad x, y \in \mathbb{R}$
- $f(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2 \qquad x, y \in \mathbb{R}$

En la última parte de este informe se implementará un método que permita aproximar la solución de problemas de optimización con una restricción. Así, se estudiará la eficacia del método en tres problemas, según los parámetros necesarios.
Para observar la eficacia del método de barrera se trabajarán con los siguientes problemas de optimización:

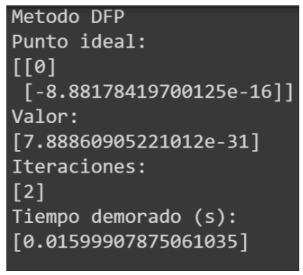
- $\begin{cases} \min\limits_{(x,y)} & f(x, y) = x^2 + y^2 \\ \text{s.a} & x + y \leq -100 \end{cases}$
- $\begin{cases} \min\limits_{(x,y)} & f(x, y) = (1 - x)^2 + 100(y - x^2)^2 \\ \text{s.a} & (x - 1)^3 - y + 1 \leq 0 \end{cases}$
- $\begin{cases} \min\limits_{(x,y)} & f(x, y) = (1 - x)^2 + 100(y - x^2)^2 \\ \text{s.a} & x + y \leq 2 \end{cases}$

## 2. DFP

Se trabajarán funciones dos veces continuamente diferenciables, que en este caso serán funciones de $\mathbb{R}^2$ a $\mathbb{R}_+$, donde para los métodos de Cuasi-Newton se aproximará la matriz Hessiana por una matriz simétrica definida positiva en cada iteración.

Para la implementación de este método, se usó que la condición de parada del programa fuera que el gradiente de la función objetivo fuera cercana a 0, en términos de programación, que la norma del gradiente de la función fuera menor que un cierto $\varepsilon$ positivo dado.

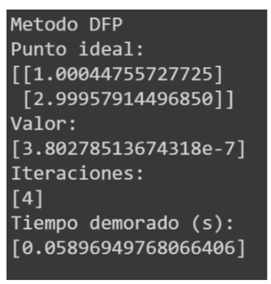
1) Se aplica el programa numérico para minimizar la función $f(x,y) = x^2 + y^2 \quad x, y \in \mathbb{R}$ vía DFP:



```
Metodo DFP
Punto ideal:
[[0]
 [-8.88178419700125e-16]]
Valor:
[7.88860905221012e-31]
Iteraciones:
[2]
Tiempo demorado (s):
[0.01599907875061035]
```

Figura 1: DFP para el primer problema

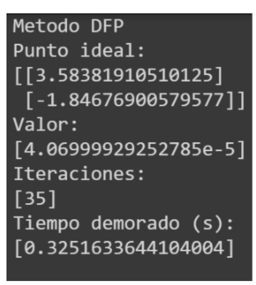
2) Se aplica el programa numérico para minimizar la función $f(x,y) = (x + 2y - 7)^2 + (2x + y - 5)^2 \quad x, y \in \mathbb{R}$ vía DFP:



```
Metodo DFP
Punto ideal:
[[1.00044755727725]
 [2.99957914496850]]
Valor:
[3.80278513674318e-7]
Iteraciones:
[4]
Tiempo demorado (s):
[0.05896949768066406]
```

Figura 2: DFP para el segundo problema

3) Se aplica el programa numérico para minimizar la función $f(x,y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2 \quad x, y \in \mathbb{R}$ vía DFP:



```
Metodo DFP
Punto ideal:
[[3.58381910510125]
 [-1.84676900579577]]
Valor:
[4.06999929252785e-5]
Iteraciones:
[35]
Tiempo demorado (s):
[0.3251633644104004]
```

Figura 3: DFP para el tercer problema

En las figuras 1, 2 y 3 el "punto ideal" es el vector $(\bar{x}, \bar{y}) \in \mathbb{R}^2$ que optimiza el problema. Y "valor", es la función objetivo evaluada en $(\bar{x}, \bar{y})$, es decir, $f(\bar{x}, \bar{y}) \in \mathbb{R}$. En estas mismas figuras esta detallada la cantidad de iteraciones que le tomó al problema encontrar un punto considerado óptimo y el tiempo demorado.

# 3. BFGS

Al igual que antes, las hipótesis sobre las funciones son las mismas y se aproximará el Hessiano por una matriz simétrica semidefinida positiva. Y de la misma forma, el criterio de parada del programa es que la norma del gradiente de la función objetivo sea menor que un cierto $\varepsilon$ positivo.

1) Se aplica programa numérico para minimizar la función $f(x, y) = x^2 + y^2 \quad x, y \in \mathbb{R}$ vía BFGS:

```
Metodo BFGS
Punto ideal:
[[0]
 [0]]
Valor:
[0]
Iteraciones:
[2]
Tiempo demorado (s):
[0.008603096008300781]
```

Figura 4: BFGS para el primer problema

2) Se aplica programa numérico para minimizar la función $f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)^2 \quad x, y \in \mathbb{R}$ vía BFGS:

```
Metodo BFGS
Punto ideal:
[[1.00034614623616]
 [3.00252815484470]]
Valor:
[3.95578109494544e-5]
Iteraciones:
[4]
Tiempo demorado (s):
[0.029375076293945312]
```

Figura 5: BFGS para el segundo problema

3) Se aplica programa numérico para minimizar la función $f(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2 \quad x, y \in \mathbb{R}$ vía BFGS:

```
Metodo BFGS
Punto ideal:
[[3.58455529796635]
 [-1.84809099698903]]
Valor:
[8.94343347640957e-7]
Iteraciones:
[17]
Tiempo demorado (s):
[0.18687105178833008]
```

Figura 6: BFGS para el tercer problema

En las figuras 4, 5 y 6 el "punto ideal" es el vector $(\bar{x}, \bar{y}) \in \mathbb{R}^2$ que optimiza el problema. Y "valor", es la función objetivo evaluada en $(\bar{x}, \bar{y})$, es decir, $f(\bar{x}, \bar{y}) \in \mathbb{R}$. En estas mismas figuras esta detallada la cantidad de iteraciones que le tomó al problema encontrar un punto considerado óptimo y el tiempo que necesitó.

# 4. Método de Barrera

Para los siguientes problemas de minimización, habrán restricciones de tipo desigualdad. El objetivo de este método es resolver de forma numérica este tipo de problemas, para ello se hace uso de una función denominada función barrera, la cual está definida en el interior del conjunto de las restricciones de desigualdad tal que cumple lo siguiente: $b(x) \to +\infty$ cuando $\max\limits_{i=1,\dots,m} g_i(x) \to 0^-$

Para los siguientes problemas se hará uso de una función barrera logarítmica $b(x) = -\sum\limits_{i=1}^{m} ln(-g_i(x))$ que en este caso sería simplemente $b(x) = -ln(-g(x))$ dado que solo hay una restricción. Se puede observar que esta función claramente cumple la condición descrita más arriba. Por último, en cada iteración se usó el método BFGS y la condición de parada de que la solución entre iteraciones tengan una diferencia de sus módulos suficientemente cercana a 0.

$$\begin{cases} \min\limits_{(x,y)} & f(x,y) = x^2 + y^2 \\ \text{s.a} & x + y \leq -100 \end{cases}$$

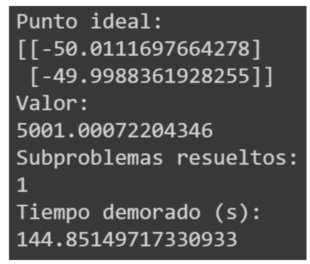



Figura 7: Método barrera para el primer problema

$$\begin{cases} \min\limits_{(x,y)} & f(x,y) = (1-x)^2 + 100(y - x^2)^2 \\ \text{s.a} & (x-1)^3 - y + 1 \leq 0 \end{cases}$$

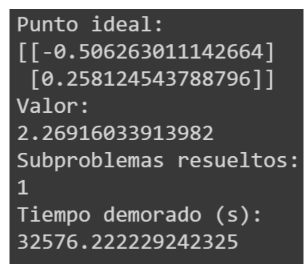



Figura 8: Método barrera para el segundo problema

Al estudiar el desempeño del algoritmo, se presentaron problemas debido a la función barrera, dado que esta no está bien definida para valores positivos de $g(x)$, esto significó que el paso entre iteraciones del método BFGS no podía ser grande, por lo que se probó con varios valores del alfa inicial para el paso de Armijo, lo que aumentó el tiempo que el algoritmo tardaría. En la primera función (Figura 7), se observa que se llega confiablemente a un aproximado del punto ideal $(-50, -50)$ en un tiempo razonable, con la resolución de un solo subproblema (además del problema final, donde se cumple la condición de parada).

En la figura 8 se puede ver la repercusión de elegir un alfa muy pequeño, sí se llegó a una solución del segundo problema que se puede considerar suficientemente cercana al óptimo $(1, 1)$, pero el tiempo que demoró en encontrarlo fue de unas 9 horas,pese a que el punto inicial factible dado era cercano a la solución, de esto se piensa que efectivamente el paso de Armijo juega un rol importante en el desempeño del algoritmo.

Para el tercer problema, se probó con diversos valores para el alfa inicial y omega en el paso de Armijo, pero el tiempo que demora superó las 12 horas y no se pudo encontrar solución.

## Conclusión

Como se pudo apreciar en este informe, los método numéricos son una buena herramienta para afrontar problemas de optimización mucho más para problemas dificiles de resolver de otra forma.

Con respecto a los primeros métodos (DFP y BFGS), estos fueron muy eficientes, pues la cantidad de iteraciones y de tiempo que les tomó no fue excesiva y entregaron resultados muy aproximado a los óptimos teóricos de cada problema. También se puede apreciar que BFGS, para estas funciones trabajadas, es el doble de eficaz que DFP, siendo que la implementación en términos de programación es bastante similar, y ambos tienen cierta versatilidad al poder cambiar, por ejemplo, los criterios de parada, el punto inicial, iteraciones máximas, entre otros.

Con respecto al método de barrera, se vió que la eficacia temporal que tuvo este método no fue la mejor. Esto se puede apreciar claramente el la Figura 8. A pesar de esto, el programa demuestra funcionar pertinentemente para el primer problema, lo que implica que la falla no se evidenció siempre, que en este caso está relacionada con la condición de Armijo. Esto pues todo el programa era muy sensible al valor de $\alpha_0$ inicial, ya que de ser muy grande indefinía la función barrera al tener logaritmos de valores negativos y al ser muy pequeño el programa demoraba mucho.

Por último, para mejorar la implementación de este metodo se pudo haber considerado agregar una filtración ciertos parámetros, es decir, para cada iteración verificar si el punto es factible, que $b(x)$ no se indefina y en tal caso no considerar esa iteración y pasar a la siguiente, para así no tener que recurrir a un alfa muy pequeño que haga que el algoritmo demore mucho, otra opción pudo haber sido cambiar la condición de Armijo por otra búsqueda del alfa.

## Referencias

1. Ramirez Héctor. 2020. Método de Newton [Archivo PDF].
2. Madariaga Javier. 2020. Auxiliar 10 [Archivo PDF].