Optimización

Técnicas para NLPs

Docente: Cristian Guarnizo Lemus

Contenido

1. Programas cuadráticos (QPs).
2. Técnicas para NLPs : Eliminación de variables.
3. Métodos de Penalización y de Barrera.

Programa Cuadrático - QP

La forma general de un programa cuadrático (QP) es:

$$\min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{G}\boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{x}$$

$$\text{s.t. } \boldsymbol{a}_i^\top \boldsymbol{x} - b_i = 0, i \in E$$

$$\boldsymbol{a}_i^\top \boldsymbol{x} - b_i \leq 0, i \in I$$

$\boldsymbol{a}_i$ es la fila i-esima de $\boldsymbol{A}$

- $\boldsymbol{G}$ es una matriz $(n \times n)$ simétrica.
- Si $\boldsymbol{G}$ es semidefinida positiva, entonces QP es convexo. Los QPs son típicamente no convexos, si $\boldsymbol{G}$ es indefinida.
- Los problemas cuadráticos con restricciones cuadráticas (QCQP) tienen términos cuadráticos en la función objetivo y las restricciones, y son mucho mas difíciles de solucionar.

Condiciones KKT de optimalidad para QPs

NLP, programa no-lineal

$$\min_{\boldsymbol{x}} f(\boldsymbol{x})$$

$$\text{s.t. } c_i(\boldsymbol{x}) = 0, i \in E$$
$$c_i(\boldsymbol{x}) \leq 0, i \in I$$

QP, programa cuadrático

$$\min_{\boldsymbol{x}} \boldsymbol{x}^\top \boldsymbol{G}\boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{x}$$

$$\text{s.t. } \boldsymbol{a}_i{}^\top \boldsymbol{x} - b_i = 0, i \in E$$
$$\boldsymbol{a}_i{}^\top \boldsymbol{x} - b_i \leq 0, i \in \{1, \dots, n\}$$

Función Lagrangiana

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i \in E \cup I} \lambda_i c_i(\boldsymbol{x})$$

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = \boldsymbol{x}^\top \boldsymbol{G}\boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{x} + \sum_{i \in E \cup I} \lambda_i \left(\boldsymbol{a}_i{}^\top \boldsymbol{x} - b_i\right)$$

Condiciones KKT

$$\nabla_x L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = \boldsymbol{0}$$
$$c_i(\boldsymbol{x}^*) = 0, \forall i \in E$$
$$c_i(\boldsymbol{x}^*) \leq 0, \forall i \in I$$
$$\lambda_i^* \geq 0, \forall i \in I$$
$$\lambda_i^* c_i(\boldsymbol{x}^*) = 0, \forall i \in I$$

$$\Rightarrow$$

$$\boldsymbol{G}\boldsymbol{x}^* + \boldsymbol{d} + \boldsymbol{A}^\top \boldsymbol{\lambda}^* = \boldsymbol{0}$$
$$\boldsymbol{a}_i{}^\top \boldsymbol{x}^* - b_i = 0, i \in E$$
$$\boldsymbol{a}_i{}^\top \boldsymbol{x}^* - b_i \leq 0, i \in I$$
$$\lambda_i^* \geq 0, \forall i \in I$$
$$\lambda_i^* \left(\boldsymbol{a}_i{}^\top \boldsymbol{x}^* - b_i\right) = 0, \forall i \in I$$

Ecuaciones No-lineales!
Bilineales como en los LPs

Solución de QPs (convexos)

- QPs convexos son muy similares a LPs.
  - Las condiciones KKT son necesarias y suficientes para un optimo global.
  - Estacionariedad lineal, factibilidad primaria lineal, holgura complementaria no-lineal, y limites lineales sobre las variables.

- QPs no convexos son difíciles de solucionar
  - Las condiciones KKT son solamente necesarias, no son suficientes para optimalidad.
  - Algoritmos para la optimización global de NLPs no convexos se puede aplicar.
  - Algoritmos especiales existen pero para entenderlos se requiere su algebra lineal cuidadosamente.

Chequeo

Cual es la forma estándar de un QP? Cuando el QP es convexo?

Contenido

1. Programas cuadráticos (QPs).
2. Técnicas para NLPs : Eliminación de variables.
3. Métodos de Penalización y de Barrera.

Problema de optimización no-lineal (NLP)

Formulación general:

$$\min_{\boldsymbol{x} \in D} f(\boldsymbol{x})$$

$$\text{s.t.} \quad c_i(\boldsymbol{x}) = 0, i \in E$$

$$c_i(\boldsymbol{x}) \leq 0, i \in I$$

$\boldsymbol{x} = [x_1, x_2, \dots, x_n]^\top \in D$ un vector (punto n-dimensional)

D conjunto anfitrión

$f: D \to R$ función objetivo

$c_i: D \to R$ funciones de restricción $\forall i \in E \cup I$

E el conjunto índice de las restricciones de igualdad

I el conjunto índice de las restricciones de desigualdad

- Tres estrategias de solución:
  - **Eliminación de variables** (convertir a un problema sin restricciones).
  - Aproximación como una serie de problemas sin restricción.
  - Aproximación como una serie de problemas mas simples con restricciones.

Eliminación de variables: Idea

$$\min_{\boldsymbol{x} \in D} f(\boldsymbol{x})$$

$$\text{s.t.} \quad c_i(\boldsymbol{x}) = 0, i \in E$$

- n variables y m igualdades → $n - m$ grados de libertad para la optimización.
- Idea: simplificar el problema eliminando m variables usando las igualdades.

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{z} \end{bmatrix} \begin{array}{l} \longleftarrow \text{dimension } n - m \\ \longleftarrow \text{dimension } m \end{array} \quad \Longrightarrow \quad \min_{\boldsymbol{y}} \tilde{f}(\boldsymbol{y})$$

Observaciones:

- Se necesita poder solucionar las funciones $c_i(\boldsymbol{x})$ para $\boldsymbol{z}(\boldsymbol{y})$.
  - La eliminación puede ser simbólica o numérica.
  - Posible para igualdades lineales y no-lineales.
  - Algunas veces “formulación de espacio reducido”.

Eliminación de variables: Ejemplo

$$\min_{\boldsymbol{x} \in R^2} f(\boldsymbol{x}) = 4x_1 + 5x_2^2$$
$$\text{s.t.} \quad \sqrt{x_1} + x_2 = 3$$

Solucionar para x_1 e insertar en la función objetivo:

$$x_1 = (3 - x_2)^2$$

$$\min_{x_2} f(x_2) = 9x_2^2 - 24x_2 + 36$$

Solucionar para el problema sin restricciones:

$$\left.\frac{d\tilde{f}}{dx_2}\right|_{x_2} = 18x_2 - 24 \quad \Rightarrow \quad x_2^* = 4/3$$

$$\left.\frac{d^2\tilde{f}}{dx_2^2}\right|_{x_2} = 18 > 0$$

Problema estrictamente convexo: el punto estacionario es un mínimo global. El problema original parecía no-convexo.

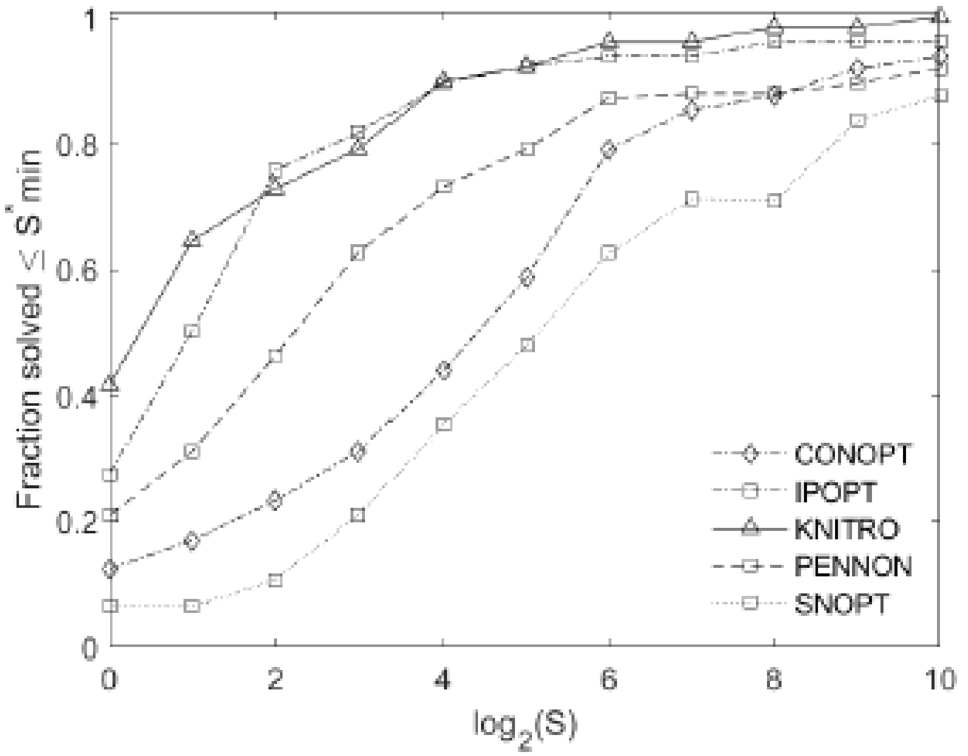
Como seleccionar un solver

- Muchas opciones:
  - Directo vs Indirecto.
  - Punto interior vs conjunto activo.
  - Aproximación de orden, por ejemplo: primer orden (steepest descent), segundo orden (método de Newton).
  - Secuencia de problemas con restricciones o sin restricciones.
  - Espacio completo o reducido?
  - Iteraciones factibles o infactibles?
- Nos interesa: robustez, encontrar un buen mínimo local, bajo tiempo de CPU, memoria aceptable.
- Muchos solvers existentes, la mayoría disponible para múltiples plataformas.
- Recordar la complejidad aritmética: tiempo CPU = # iteraciones* tiempo CPU/iteración
  - Cada factor depende del solver y la estructura del problema!

Comparación cuantitativa de Solvers

- Comparar solvers es una tarea difícil:
  - Métrica de comparación?
  - valores para las tolerancias y opciones de sintonización?.
  - como manejar las instancias fallidas.

- Métricas utilizadas:
  - Tiempo de CPU.
  - Tiempo de CPU escalado al mejor solver.
  - # evaluaciones de la funciones.
  - # iteraciones.
- Perfiles de desempeño de Dolan-Moré: orden de los solver por # de problemas solucionados como función de la métrica seleccionada:
  - No se puede distinguir de forma segura entre los mejores solvers.
  - La métrica seleccionada cambia con el orden.



S = tiempo de CPU en min. Mittelmann NLP benchmarl.

Prof. Mittelmann (http://plato.la.asu.edu/bench.html)

Chequeo

Que estrategias existen para la solución general de NLPs?

La eliminación de variables siempre es segura de aplicar? Cuales son las desventajas?

Cuales son las graficas de desempeño y como se pueden usar?

Contenido

1. Programas cuadráticos (QPs).
2. Técnicas para NLPs : Eliminación de variables.
3. Métodos de Penalización y de Barrera.

Métodos de Penalización y Barrera

- Idea: reemplazar el problema de restricciones por una secuencia de problemas de optimización sin restricciones. Como remover las restricciones?
- Método de penalización cuadrático (QPM): reemplazar las restricciones adicionando una penalización cuadrático en la función objetivo.
  - Aproximación desde los puntos infactibles.

- Método del Langrangiano Aumentado (ALM): Mejora de QPM que evita los mal-condicionamientos por medio de la estimación de los parámetros de Lagrange.

- Método de Log-Barrier (LBM): usar la barrera del logaritmo para forzar una estricta satisfacción de las desigualdades.
  - Aproximación desde los puntos factibles.

QPM – Restricciones de Igualdad

Reemplazar cada restricción por un termino de penalización cuadrático en la función objetivo

$$\min_{\boldsymbol{x} \in D} f(\boldsymbol{x})$$

$$\text{s.t.} \quad c_i(\boldsymbol{x}) = 0, i \in E$$

Función de penalización cuadrática: $Q(\boldsymbol{x}; \mu) = f(\boldsymbol{x}) + \frac{1}{2\mu} \sum_{i \in E} [c_i(\boldsymbol{x})]^2$

- Con parámetro de penalización μ>0.
- Construir una secuencia $\{\mu^{(k)}\}$ con $\mu^{(k)}$→ 0 y minimizar $Q(\boldsymbol{x}; \mu^{(k)})$.
  - $\boldsymbol{x}^{(k)}$ son soluciones aproximadas infactibles del problema original.
  - La solución optima de un paso es la adivinación inicial para la siguiente.

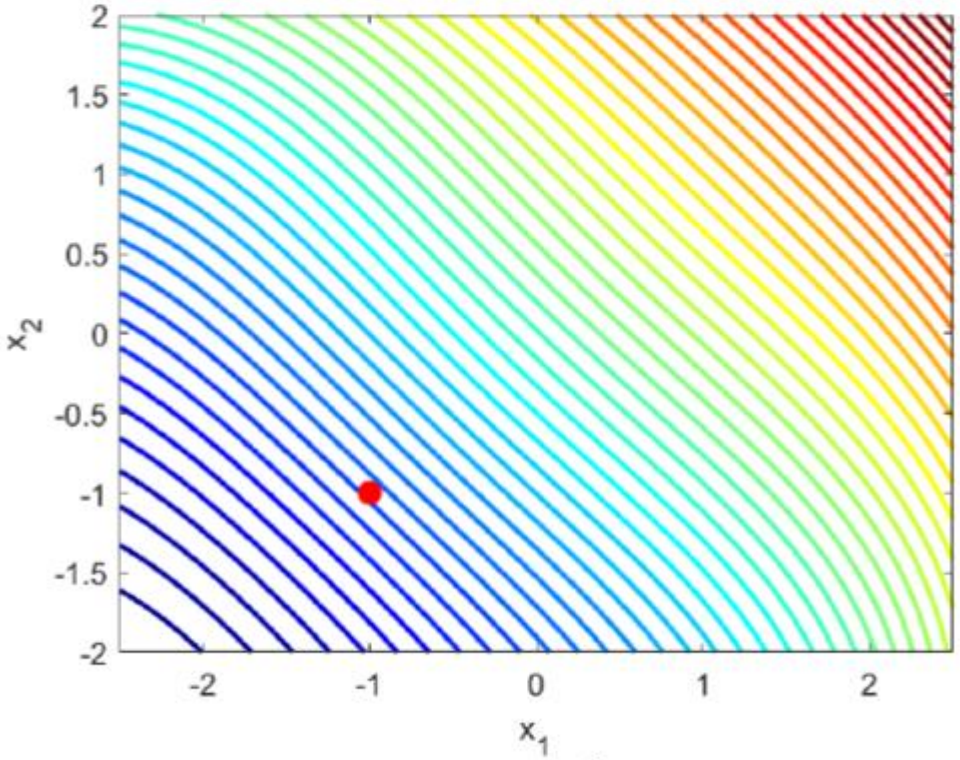
- Para $\mu \rightarrow 0$ la violación de la restricción es incrementalmente penalizada.
  - La aproximación se mejor de forma progresiva.
  - $\boldsymbol{x}^{(k)}$ converge a la solución, sí $Q(\boldsymbol{x}; \mu^{(k)})$ son minimizados globalmente.

QPM – Restricciones de Igualdad

$$\min_{x \in R^2} \; x_1 + x_2$$

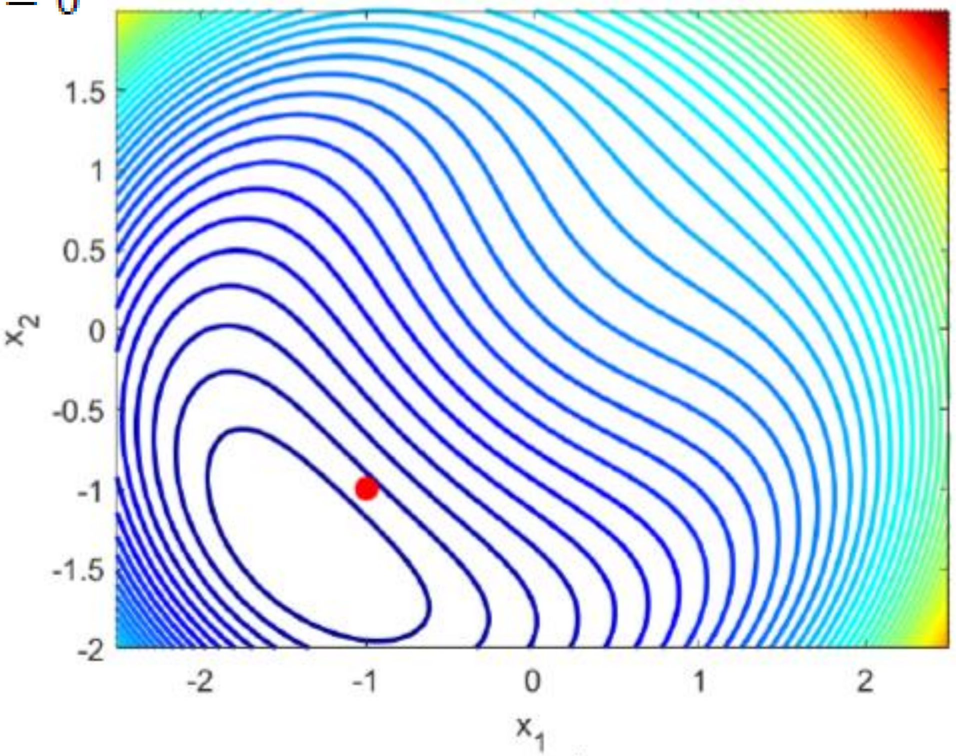
$$\text{s.t.} \;\; x_1^2 + x_2^2 - 2 = 0$$

$\mu = 50$



$$Q(\boldsymbol{x}; 50) = x_1 + x_2 + \frac{1}{100}(x_1^2 + x_2^2 - 2)^2$$

$\mu = 5$



$$Q(\boldsymbol{x}; 5) = x_1 + x_2 + \frac{1}{10}(x_1^2 + x_2^2 - 2)^2$$

QPM – Incluir restricciones de desigualdad

$$\begin{aligned} \min_{\boldsymbol{x} \in R^n} \quad & f(\boldsymbol{x}) \\ \text{s.t.} \quad & c_i(\boldsymbol{x}) = 0,\ i \in E \\ & c_i(\boldsymbol{x}) \leq 0,\ i \in I \end{aligned}$$

- El signo de la desigualdades importa:

$$Q(\boldsymbol{x};\mu) = f(\boldsymbol{x}) + \frac{1}{2\mu}\sum_{i\in E}[c_i(\boldsymbol{x})]^2 + \frac{1}{2\mu}\sum_{i\in I}[\max(0, c_i(\boldsymbol{x}))]^2$$

Algoritmo QPM:

- Dado $\mu^{(1)} > 0$, $\tau^{(1)} > 0$ y un punto inicial $\boldsymbol{x}^{(0)}$
- Para $k = 1,2,\ldots$
  - Usar $\boldsymbol{x}^{(k-1)}$ como punto inicial (aleatorio). Solucionar $\boldsymbol{x}^{(k)} \in Q\left(\boldsymbol{x};\mu^{(k)}\right)$ aproximadamente: $\left\|\nabla_x Q(\boldsymbol{x}^{(k)};\mu^{(k)})\right\| < \tau^{(k)}\mu^{(k)}$.
  - SI el gradiente y la violación de la restricción son suficientemente pequeñas, PARAR: $\boldsymbol{x}^* = \boldsymbol{x}^{(k)}$.
  - DE LO CONTRARIO seleccionar $\mu^{(k+1)} \in \left(0, \mu^{(k)}\right), \tau^{(k+1)}$ (s.t. $\lim_{k\to\infty}\tau^{(k)} = 0$)

Observaciones de QPM

- El parámetro $\mu^{(k)}$ puede ser seleccionado adaptativamente.
  - Sí $\min_{\boldsymbol{x}} Q(x;\mu^{(k)})$ fue difícil, reducir μ de manera modesta: $\mu^{(k+1)} = 0.7\mu^{(k)}$.
  - Si $\min_{\boldsymbol{x}} Q(x;\mu^{(k)})$ fue fácil, reducir μ mas rápidamente: $\mu^{(k+1)} = 0.1\mu^{(k)}$.

- Para restricciones de igualdad, la función de penalización es suave
  - Continua para la primera derivada, pero discontinua para la segunda derivada.

- Como $\mu^{(k)} \to 0$, solucionar $\min_{\boldsymbol{x}} Q(x;\mu^{(k)})$ es cada vez mas difícil.
  - La Hessiana se vuelve mas y mas mal-condicionada.
  - El método del Lagrangiano Aumentado alivia este problema.

ALM: Restricciones de Igualdad

$$\min_{\boldsymbol{x} \in D} f(\boldsymbol{x})$$
$$\text{s.t.} \quad c_i(\boldsymbol{x}) = 0, i \in E$$

Lagrangiano: $$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i \in E} \lambda_i c_i(\boldsymbol{x})$$

Lagrangiano Aumentado: $$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i \in E} \lambda_i c_i(\boldsymbol{x}) + \frac{1}{2\mu} \sum_{i \in E} [c_i(\boldsymbol{x})]^2$$

- Ventaja de ALM con respecto a QPM: pequeña violación de la restricción para un μ relativamente grande.
  - Evita problemas numéricos de mal-condicionamiento.

- Como seleccionar de mantera iterativa los parámetros μ y λ?

ALM: Restricciones de Igualdad

- Gradiente de la función Lagrangiana Aumentada:

$$\nabla_{\boldsymbol{x}} L_A(\boldsymbol{x}, \boldsymbol{\lambda}, \mu) = \nabla_{\boldsymbol{x}} f(\boldsymbol{x}) + \sum_{i \in E} \left(\lambda_i + \frac{c_i(\boldsymbol{x})}{\mu} \right) \nabla_{\boldsymbol{x}} c_i(\boldsymbol{x})$$

- $\operatorname{argmin}_{\boldsymbol{x}} L_A\left(\boldsymbol{x}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)}\right)$ debería ser aproximado $\min_{\boldsymbol{x} \in E} f(\boldsymbol{x})$

- La estacionariedad en $L(\boldsymbol{x}, \boldsymbol{\lambda})$ implica $\lambda_i^* \approx \lambda_i^{(k)} + \frac{c_i(\boldsymbol{x}^{(k)})}{\mu^{(k)}}$
- Si $\lambda_i^* \approx \lambda_i^{(k)}$, la violación de las restricciones es pequeña debido que $c_i(\boldsymbol{x}^{(k)}) \approx \mu^{(k)} \left(\lambda_i^* - \lambda_i^{(k)} \right)$

  - Aun para un $\mu^{(k)}$ grande.

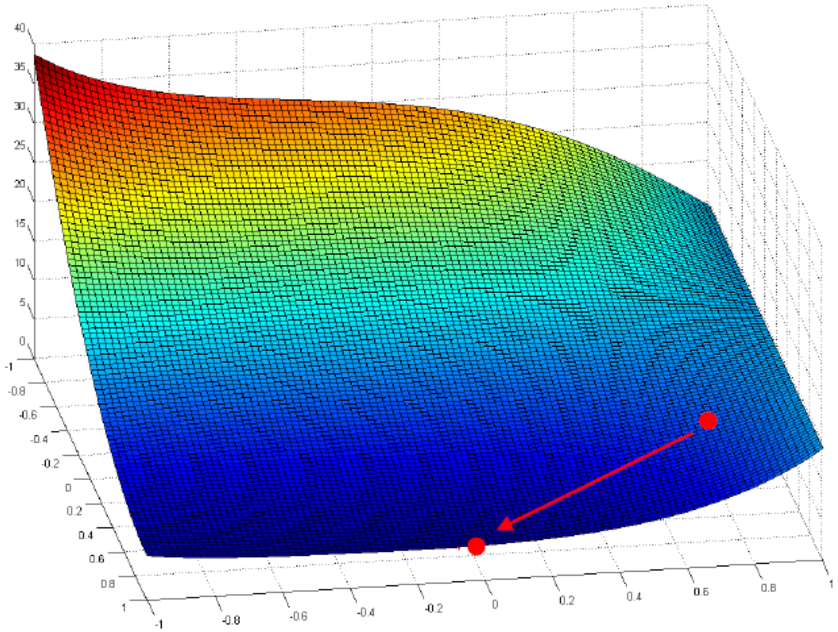
- Como λ_i^* es desconocido, nosotros iterativamente actualizamos: $\lambda_i^{(k+1)} = \lambda_i^{(k)} + \frac{c_i(\boldsymbol{x})}{\mu^{(k)}}$

- Algoritmo similar a QPM pero converge para μ mas grandes. Se espera menos iteraciones y mejor condicionamiento.

QPM vs ALM: Ejemplo

$$\min_{x_1,x_2} [1.5 - x_1(1-x_2)]^2 + [2.25 - x_1(1-x_2^2)]^2 + [2.625 - x_1\left(2.625 - x_1(1-x_2^3)\right)]^2$$

$$\text{s.t. } x_1^2 + x_2^2 - 1 = 0$$



$$\boldsymbol{x}^{(0)} = \frac{\sqrt{2}}{2}\begin{bmatrix}1\\1\end{bmatrix} \qquad \boldsymbol{x}^* = \begin{bmatrix}0.997\\-0.07744\end{bmatrix}$$

$$f(\boldsymbol{x}^*) = 4.42$$

QPM:

- Converge después de 39 iteraciones.
- $\mu^{(39)} = 10^{-8}$

ALM:

- Converge después de 28 iteraciones.
- $\mu^{(28)} = 10^{-4}$
- Estimaciones del multiplicador de Lagrange
  $\lambda = 0 \to \cdots \to -2.63 \to -3.33 \to -3.35$

Chequeo

Cual es la idea principal en los métodos de penalizacion?

Cual es la idea principal en el método de Lagrangiano Aumentado?

Institución Universitaria
Reacreditada en Alta Calidad

Referencias

- Basado en el curso “Applied Numerical Optimization” por el profesor Alexander Mitsos.
- Nocedal J. Wright S. J. Numerical Optimization, 2nd Edition, Springer, 2006.

Institución Universitaria
Reacreditada en Alta Calidad

¡Gracias!

Somos Innovación Tecnológica con Sentido Humano

Alcaldía de Medellín