

Implémentation d'ENLSIP en Julia

Pierre Borie

12 octobre 2020

- 1 Introduction
 - Présentation du problème
 - Idée générale de la résolution
- 2 Calcul de la direction de descente
 - Méthode de Gauss Newton
 - Méthode de Newton
- 3 Mise à jour de l'ensemble de travail
- 4 Calcul du pas
 - Fonction de mérite et pénalités
 - Méthode de calcul du pas
 - Commentaires sur la méthode
- 5 Critères de convergence
- 6 Résultats avec des contraintes d'égalité

1 Introduction

- Présentation du problème
- Idée générale de la résolution

2 Calcul de la direction de descente

- Méthode de Gauss Newton
- Méthode de Newton

3 Mise à jour de l'ensemble de travail

4 Calcul du pas

- Fonction de mérite et pénalités
- Méthode de calcul du pas
- Commentaires sur la méthode

5 Critères de convergence

6 Résultats avec des contraintes d'égalité

Présentation du problème

- m données (t_i, y_i)
- modèle h de paramètres $x \in \mathbb{R}^n$
- $r_i : x \mapsto y_i - h(x, t_i)$ i -ème résidu
- l contraintes à respecter

Problème à résoudre

$$\left\{ \begin{array}{l} \min_{x \in \mathbb{R}^n} \frac{1}{2} \|r(x)\|^2 \\ \text{sous contraintes} \\ c_i(x) = 0 \text{ pour } i = 1, \dots, q \\ c_j(x) \geq 0 \text{ pour } j = q + 1, \dots, l \end{array} \right.$$

Avec $r = (r_1, \dots, r_m)$ multi-fonction des résidus

$f : x \mapsto \frac{1}{2} \|r(x)\|^2$ la fonction objectif

Un algorithme itératif

- On part de $x_0 \in \mathbb{R}^n$

Un algorithme itératif

- On part de $x_0 \in \mathbb{R}^n$
- Itération k , calcul de :
$$\begin{cases} p_k \in \mathbb{R}^n \text{ direction de descente} \\ \alpha_k \in \mathbb{R} \text{ longueur de pas} \end{cases}$$

Un algorithme itératif

- On part de $x_0 \in \mathbb{R}^n$
- Itération k , calcul de : $\begin{cases} p_k \in \mathbb{R}^n \text{ direction de descente} \\ \alpha_k \in \mathbb{R} \text{ longueur de pas} \end{cases}$
- Mise à jour : $x_{k+1} = x_k + \alpha_k p_k$ jusqu'à arrêt

1 Introduction

- Présentation du problème
- Idée générale de la résolution

2 Calcul de la direction de descente

- Méthode de Gauss Newton
- Méthode de Newton

3 Mise à jour de l'ensemble de travail

4 Calcul du pas

- Fonction de mérite et pénalités
- Méthode de calcul du pas
- Commentaires sur la méthode

5 Critères de convergence

6 Résultats avec des contraintes d'égalité

1 Introduction

- Présentation du problème
- Idée générale de la résolution

2 Calcul de la direction de descente

- Méthode de Gauss Newton
- Méthode de Newton

3 Mise à jour de l'ensemble de travail

4 Calcul du pas

- Fonction de mérite et pénalités
- Méthode de calcul du pas
- Commentaires sur la méthode

5 Critères de convergence

6 Résultats avec des contraintes d'égalité

Linéarisations des résidus et des contraintes (1/2)

Au voisinage p d'une approximation \tilde{x} de la solution,

$$r_i(\tilde{x} + p) \approx r_i(\tilde{x}) + \left(\frac{\partial r_i(\tilde{x})}{\partial x_1}, \dots, \frac{\partial r_i(\tilde{x})}{\partial x_n} \right) p$$
$$c_j(\tilde{x} + p) \approx c_j(\tilde{x}) + \left(\frac{\partial c_j(\tilde{x})}{\partial x_1}, \dots, \frac{\partial c_j(\tilde{x})}{\partial x_n} \right) p$$

Linéarisations des résidus et des contraintes (2/2)

On a alors :

$$r(\tilde{x} + p) \approx J(\tilde{x})p + r(\tilde{x})$$

$$c(\tilde{x} + p) \approx A(\tilde{x})p + c(\tilde{x})$$

J et A jacobiennes de r et c .

Détermination de l'ensemble de travail

$$\text{Lagrangien : } \mathcal{L}(x, \lambda) = f(x) - \sum_{i \in \mathcal{W}} \lambda_i c_i(x)$$

\mathcal{W} : ensemble de travail, comprend contraintes d'égalité et contraintes d'inégalité actives

\Rightarrow C'est une prédiction des contraintes qui resteront actives après la descente

Restriction du nombre de contraintes

Suite à estimation des multiplicateurs λ

- On évalue les inégalités actives et on les ajoute à l'ensemble de travail

Restriction du nombre de contraintes

Suite à estimation des multiplicateurs λ

- On évalue les inégalités actives et on les ajoute à l'ensemble de travail
- Contraintes d'inégalité actives \rightarrow contraintes d'égalité

Restriction du nombre de contraintes

Suite à estimation des multiplicateurs λ

- On évalue les inégalités actives et on les ajoute à l'ensemble de travail
- Contraintes d'inégalité actives \rightarrow contraintes d'égalité
- Restriction à $t \geq q$ contraintes actives
- $\hat{c} : \mathbb{R}^n \rightarrow \mathbb{R}^t$ multi-fonction des contraintes actives, \hat{A} sa jacobienne

Sous-problème de l'itération k

A x_k fixé,
linéarisations + ensemble actif

$$\Rightarrow \begin{cases} \min_{p \in \mathbb{R}^n} \frac{1}{2} \|J_k p + r(x_k)\|^2 \\ \text{s.c.} \\ \hat{A}_k p + \hat{c}(x_k) = 0 \end{cases}$$

Première factorisation

Factorisation QR de \hat{A}_k^T

- $\hat{A}_k^T P_1 = Q_1 \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$
- Q_1 matrice orthogonale $n \times n$
- R_1 matrice triangulaire supérieure $t \times t$
- P_1 matrice de permutation $t \times t$ telle que $|r_{11}| \geq |r_{22}| \geq \dots \geq |r_{tt}|$
(les r_{ii} sont les éléments diagonaux de R_1)

Factorisation LQ de \hat{A}_k

Par transposition, $\hat{A}_k = P_1 \begin{pmatrix} L_1 & 0 \end{pmatrix} Q_1^T$

Posant $Q_1^T p = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}$ avec $p_1 \in \mathbb{R}^t$ et $p_2 \in \mathbb{R}^{n-t}$, $\hat{A}_k Q_1 = \begin{pmatrix} Y & Z \end{pmatrix}$:

$$\hat{A}_k p = -\hat{c}(x_k) \Rightarrow \begin{cases} Y p_1 = -\hat{c}(x_k) \\ Z p_2 = 0 \end{cases}$$

Décomposition en sous-systèmes

p_1 fixé par les contraintes

$$P_1 L_1 p_1 = -\hat{c}(x_k)$$

d'où $L_1 p_1 = -P_1^T \hat{c}(x_k) (= b)$

p_2 dans l'espace nul

$$\min_{p_2} \frac{1}{2} \|J_2 p_2 + (J_1 p_1 + r(x_k))\|^2$$

$$(J_k Q_1 = (J_1 \quad J_2))$$

Rang de \hat{A} et stabilisation

$$\bar{t} = \max_{1 \leq j \leq t} \{j \mid |l_{jj}| \geq \varepsilon_{relative}\}$$

$\bar{t} < t \Rightarrow$ stabilisation

- $L_1 = Q_2 \begin{pmatrix} R_2 \\ 0 \end{pmatrix} P_2^T$
- on résout alors $R_2 P_2^T p_1 = -Q_2^T P_1^T \hat{c}(x) (= b_1)$

Calcul de p_1

- Soit $\omega_1 \leq t$, $R_2^{(\omega_1)}$ bloc supérieur gauche $\omega_1 \times \omega_1$

Calcul de p_1

- Soit $\omega_1 \leq t$, $R_2^{(\omega_1)}$ bloc supérieur gauche $\omega_1 \times \omega_1$
- $R_2^{(\omega_1)} \delta p_1^{(\omega_1)} = b_1^{(\omega_1)}$

Calcul de p_1

- Soit $\omega_1 \leq t$, $R_2^{(\omega_1)}$ bloc supérieur gauche $\omega_1 \times \omega_1$
- $R_2^{(\omega_1)} \delta p_1^{(\omega_1)} = b_1^{(\omega_1)}$
- $p_1 = P_2 \begin{pmatrix} \delta p_1^{(\omega_1)} \\ 0 \end{pmatrix}$

Calcul de p_2 (1/2)

p_1 fixé, on minimise dans l'espace nul :

$$\min_{p_2} \frac{1}{2} \|J_2 p_2 + J_1 P_2 p_1 + r(x_k)\|^2$$

$$J_2 = Q_3 \begin{pmatrix} R_3 \\ 0 \end{pmatrix} P_3^T$$

Le problème devient $\Rightarrow \min_{p_2} \frac{1}{2} \|R_3 P_3 p_2 + Q_3^T [J_1 P_2 p_1 + r(x_k)]\|^2$

Calcul de p_2 (2/2)

Soient $\omega_2 \leq n - t$ et d le second membre

- $R_3^{(\omega_2)} \delta p_2^{(\omega_2)} = d^{(\omega_2)}$

- $p_2 = P_3^T \begin{pmatrix} \delta p_2^{(\omega_2)} \\ 0 \end{pmatrix}$

Finalement :

$$p^{(\omega_1, \omega_2)} = Q_1 \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}$$

Choix des dimensions ω_1 et ω_2

Deux approches différentes

GN rang plein

- $\omega_1 = \text{rang}(\hat{A}_k)$
- $\omega_2 = \text{rang}(J_2)$

Minimisation de sous-espace

- ω_1 et ω_2 calculés spécifiquement

Utilisation de la factorisation QR (1/2)

Avantages

- Adapté aux moindres carrés
- Facilite la résolution de systèmes et calculs de rang
- Permet de repérer les linéarités dans les contraintes
- Facilement calculable en Julia (fonction `qr`)

Utilisation de la factorisation QR (2/2)

Inconvénients

- Matrices trop encombrantes à stocker pour des grandes valeurs de m , n et l (et indirectement t)
- Possibilité d'échantillonnage adaptatif pour réduire m mais le problème persiste pour les contraintes

Bilan sur la méthode

- Tire parti de la structure de moindres carrés
- Intègre des contraintes d'égalité de d'inégalité non linéaires
- Possibilité d'ajustement

1 Introduction

- Présentation du problème
- Idée générale de la résolution

2 Calcul de la direction de descente

- Méthode de Gauss Newton
- Méthode de Newton

3 Mise à jour de l'ensemble de travail

4 Calcul du pas

- Fonction de mérite et pénalités
- Méthode de calcul du pas
- Commentaires sur la méthode

5 Critères de convergence

6 Résultats avec des contraintes d'égalité

Principe

Si GN pas calculable ou proximité de la solution, on résout :

$$\left\{ \begin{array}{l} \min_{p \in \mathbb{R}^n} \left[\frac{1}{2} p^T \nabla_{xx}^2 \mathcal{L}(x, \lambda) p + \nabla f(x)^T p \right] \\ \text{s.c} \\ \hat{A}p = -\hat{c}(x) \end{array} \right.$$

Avec

$$\nabla_{xx}^2 \mathcal{L}(x, \lambda) = J(x)^T J(x) + \sum_{i=1}^m r_i(x) \nabla^2 r_i(x) - \sum_{i=1}^t \lambda_i \nabla^2 \hat{c}_i(x)$$

Utilisation

Avantages

- Efficace proche de la solution
- Calculs de hessiennes facilités en Julia (package ForwardDiff)

Utilisation

Avantages

- Efficace proche de la solution
- Calculs de hessiennes facilités en Julia (package ForwardDiff)

Inconvénients

- Calculs quand même coûteux
- Solution pas toujours calculable

1 Introduction

- Présentation du problème
- Idée générale de la résolution

2 Calcul de la direction de descente

- Méthode de Gauss Newton
- Méthode de Newton

3 Mise à jour de l'ensemble de travail

4 Calcul du pas

- Fonction de mérite et pénalités
- Méthode de calcul du pas
- Commentaires sur la méthode

5 Critères de convergence

6 Résultats avec des contraintes d'égalité

Présentation

A l'optimum x^* :

$$\hat{A}^T \lambda^* = \nabla f(x^*)$$

$\forall i \in \text{contraintes d'inégalité}, \lambda_i \geq 0$

Une approche primal

- p toujours réalisable i.e $A_k p + c(x_k) \geq 0$
- λ pas nécessairement réalisable i.e $\exists i$ tq $\lambda_i < 0$
- λ calculé en dehors de l'itération et n'est pas pris en compte dans le calcul du pas

⇒ Travailler dans une prédiction de l'ensemble actif à chaque itération

Stratégie de suppression de contrainte

- Estimer λ en résolvant :
 - ▶ $\hat{A}_k \lambda = \nabla f(x_k)$
 - ▶ ou $\hat{A}_k \lambda = \nabla f(x_k) + J_k^T J_k p_{GN}$
- On retire de l'ensemble actif la contrainte au plus petit multiplicateur < 0
- On calcule la direction de descente avec le nouvel ensemble de travail

Piste d'amélioration

Passer à une approche primal-dual

- Meilleure prise en compte de l'espace dual
- Calcul du pas pourrait concerner les directions des problèmes primal et dual

1 Introduction

- Présentation du problème
- Idée générale de la résolution

2 Calcul de la direction de descente

- Méthode de Gauss Newton
- Méthode de Newton

3 Mise à jour de l'ensemble de travail

4 Calcul du pas

- Fonction de mérite et pénalités
- Méthode de calcul du pas
- Commentaires sur la méthode

5 Critères de convergence

6 Résultats avec des contraintes d'égalité

Fonction de mérite et calcul du pas

$$\Psi : (x, w) \mapsto f(x) + \sum_{i \in \mathcal{W}} w_i c_i(x)^2 + \sum_{i \notin \mathcal{W}} w_i \min(0, c_i(x))^2$$

$w \in \mathbb{R}^I$ vecteur des pénalités

Une fois p_k calculée :

$$\alpha_k = \min_{\alpha} \phi(\alpha) = \Psi(x_k + \alpha p_k, w_k)$$

On prend $\alpha = 1$ si p_k calculée avec méthode de Newton

Pénalités : Intérêt

- Diriger la recherche du pas optimal
- Maintien dans le domaine où les linéarisations sont valables

Calcul des pénalités à l'itération k

Objectif : Résoudre

$$\left\{ \begin{array}{l} \min_{w \in \mathbb{R}^I} \|w\| \\ \text{s.c} \\ y^T w \geq \tau \\ \forall i = 1, \dots, I, \quad w_i \geq w_i^{(old)} \end{array} \right.$$

Avec y et τ définis de différentes façons à partir des contraintes.
 $w^{(old)}$ contient les plus petites valeurs de poids sur les 4 dernières itérations.

- 1 Introduction
 - Présentation du problème
 - Idée générale de la résolution
- 2 Calcul de la direction de descente
 - Méthode de Gauss Newton
 - Méthode de Newton
- 3 Mise à jour de l'ensemble de travail
- 4 Calcul du pas
 - Fonction de mérite et pénalités
 - Méthode de calcul du pas
 - Commentaires sur la méthode
- 5 Critères de convergence
- 6 Résultats avec des contraintes d'égalité

Principe de la méthode

Rappel : $\alpha_k = \min_{\alpha} \phi(\alpha) = \Psi(x_k + \alpha p_k, w_k)$

Idée : On calcule α_k par minimisations successives d'approximations polynomiales de ϕ dans un intervalle $[\alpha_{min}, \alpha_{max}]$

α_{min} et α_{max} calculés pour éviter de violer les contraintes inactives

Déroulement général

- On part de α_0 proche de 1

Déroulement général

- On part de α_0 proche de 1
- On crée $P : \alpha \mapsto \frac{1}{2} \|v_0 + v_1\alpha + v_2\alpha^2\|^2$ qui interpole ϕ en 0 et α_0 .

Déroulement général

- On part de α_0 proche de 1
- On crée $P : \alpha \mapsto \frac{1}{2} \|v_0 + v_1\alpha + v_2\alpha^2\|^2$ qui interpole ϕ en 0 et α_0 .
- On calcule α^* minimum de P sur $[\alpha_{min}, \alpha_{max}]$

Déroulement général

- On part de α_0 proche de 1
- On crée $P : \alpha \mapsto \frac{1}{2} \|v_0 + v_1\alpha + v_2\alpha^2\|^2$ qui interpole ϕ en 0 et α_0 .
- On calcule α^* minimum de P sur $[\alpha_{min}, \alpha_{max}]$
- On réitère avec $\alpha_0 = \alpha^*$ jusqu'à valeur satisfaisante ou trop petite

Déroulement général

- On part de α_0 proche de 1
- On crée $P : \alpha \mapsto \frac{1}{2} \|v_0 + v_1\alpha + v_2\alpha^2\|^2$ qui interpole ϕ en 0 et α_0 .
- On calcule α^* minimum de P sur $[\alpha_{min}, \alpha_{max}]$
- On réitère avec $\alpha_0 = \alpha^*$ jusqu'à valeur satisfaisante ou trop petite
- Si trop petite, on utilise une méthode d'Armijo-Goldstein

- 1 Introduction
 - Présentation du problème
 - Idée générale de la résolution
- 2 Calcul de la direction de descente
 - Méthode de Gauss Newton
 - Méthode de Newton
- 3 Mise à jour de l'ensemble de travail
- 4 Calcul du pas
 - Fonction de mérite et pénalités
 - Méthode de calcul du pas
 - Commentaires sur la méthode
- 5 Critères de convergence
- 6 Résultats avec des contraintes d'égalité

Commenaitres

- Ψ non différentiable à cause des termes en $\min \rightarrow$ approximations
- Pénalités utilisées uniquement dans le calcul du pas et non dans la direction de descente
- Ressemblance avec méthodes de régions de confiance

Pistes d'amélioration

- Intégration de pénalités dans le calcul de direction de descente
- Utilisation du Lagrangien augmenté
- Implémentation d'une méthode de régions de confiance

- 1 Introduction
 - Présentation du problème
 - Idée générale de la résolution
- 2 Calcul de la direction de descente
 - Méthode de Gauss Newton
 - Méthode de Newton
- 3 Mise à jour de l'ensemble de travail
- 4 Calcul du pas
 - Fonction de mérite et pénalités
 - Méthode de calcul du pas
 - Commentaires sur la méthode
- 5 Critères de convergence
- 6 Résultats avec des contraintes d'égalité

Conditions nécessaires

- $\|\hat{c}(x_k)\| < \varepsilon_{rel}$ et les contraintes inactives doivent être strictement positives
- $\|\hat{A}_k^T \lambda - \nabla f(x_k)\| < \sqrt{\varepsilon_{rel}} (1 + \|\nabla f(x_k)\|)$
- $\min_{i \in \mathcal{I}} \{\lambda_i \mid \lambda_i > 0\} \geq \varepsilon_{rel} \max_{1 \leq j \leq t} |\lambda_j|$
- ou $\min_{i \in \mathcal{I}} \{\lambda_i \mid \lambda_i > 0\} \geq \epsilon_{rel} (1 + \|r(x_k)\|^2)$ s'il n'y a qu'une seule inégalité

Conditions suffisantes

- $\|d\|^2 \leq \varepsilon_{rel}^2 \|r(x_k)\|^2$
- $\|r(x_k)\|^2 \leq \varepsilon_{rel}^2$
- $\|x_{k-1} - x_k\| < \varepsilon_{rel} \|x_k\|$
- $\frac{\sqrt{\varepsilon_{rel}}}{\|p_k\|} > 0.25$

1 Introduction

- Présentation du problème
- Idée générale de la résolution

2 Calcul de la direction de descente

- Méthode de Gauss Newton
- Méthode de Newton

3 Mise à jour de l'ensemble de travail

4 Calcul du pas

- Fonction de mérite et pénalités
- Méthode de calcul du pas
- Commentaires sur la méthode

5 Critères de convergence

6 Résultats avec des contraintes d'égalité

Réalisation des tests

- Données générées à partir d'un modèle avec paramètres fixés
- On perturbe les données avec un bruit gaussien
- On utilise l'algorithme pour retrouver le modèle originel

Premier exemple (1/2)

Vraie fonction

$$t \mapsto (t - 2)(t - 6)(t - 10)$$

Modèle à ajuster

$$g : (t, x_1, x_2, x_3) \mapsto (t - x_1)(t - x_2)(t - x_3)$$

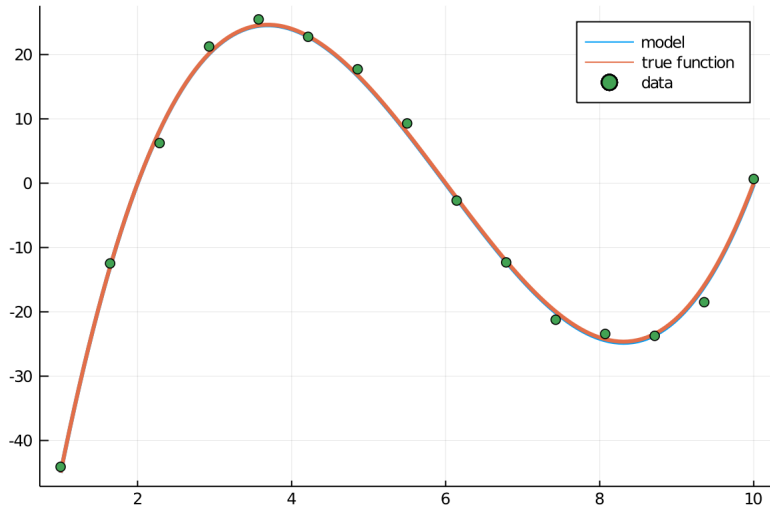
s.c.

$$x_1 + x_2 + x_3 = 18$$

$$x_1 x_2 x_3 = 120$$

Premier exemple (2/2)

- $x_0 = [1, 0, 0]$
- 13 itérations
- $x^* = \begin{pmatrix} 5,99202 \\ 10,00665 \\ 2,00133 \end{pmatrix}$



Second exemple (1/2)

Vraie fonction

$$t \mapsto 1 - \frac{t^2}{2} + \frac{t^4}{24}$$

Modèle à ajuster

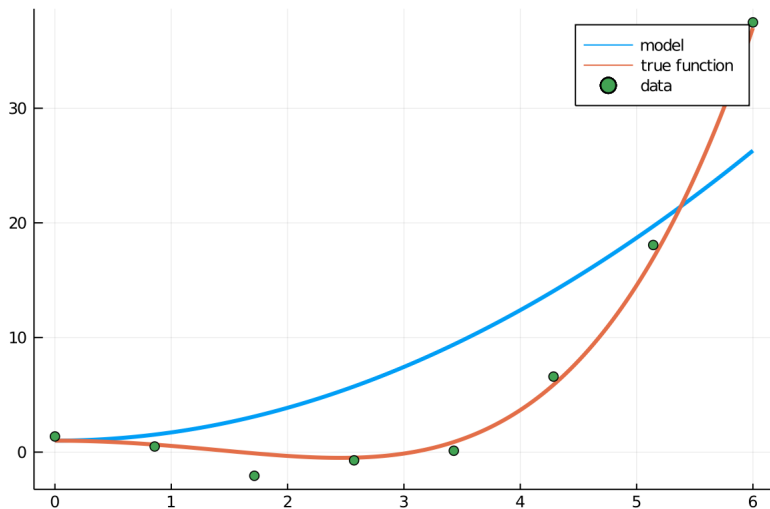
$$h : (t, x_1, x_2) \mapsto 1 + x_1 t^2 + x_2^3 \frac{t^4}{3}$$

s.c.

$$x_1 + 2x_2 = \frac{1}{2}$$

Second exemple (2/2)

- $x_0 = [1, 0]$
- itérations
- $x^* = \begin{pmatrix} 0.71843 \\ -0.10921 \end{pmatrix}$
 \Rightarrow minimum local



Conclusion

- Résultats assez satisfaisants
- Que des itérations en Gauss-Newton sur ces exemples sans inégalités