

Méthode de Levenberg et Marquardt

Minimisation de fonction non linéaire

Pierre Aubert

7 juin 2015

1 Matrice Jacobienne

Soit une fonction F définie par ses m fonctions composantes à valeurs réelles :

$$F : \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \mapsto \begin{pmatrix} f_1(x_1, \dots, x_n) \\ \vdots \\ f_m(x_1, \dots, x_n) \end{pmatrix} \quad (1)$$

Où les fonctions f_i sont définies comme :

$$\begin{aligned} f_i & : \quad \mathbb{R}^n & \longrightarrow & \mathbb{R} \\ & (x_1, \dots, x_n) & \longmapsto & z_i \end{aligned} \quad (2)$$

La matrice jacobienne de la fonction F est définie comme suit :

$$J_F(M) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \dots & \frac{\partial f_m}{\partial x_n} \end{pmatrix} \quad (3)$$

2 Méthode de Gauss-Newton

En mathématiques, l'algorithme de Gauss-Newton est une méthode de résolution des problèmes de moindres carrés non linéaires. Elle peut être vue comme une modification de la méthode de Newton-Raphson dans le cas multidimensionnel afin de trouver le minimum d'une fonction (à plusieurs variables). Mais l'algorithme de Gauss-Newton est totalement spécifique à la minimisation d'une somme de fonctions au carré et présente le grand avantage de ne pas nécessiter les dérivées secondes, parfois complexes à calculer.

Les problèmes de moindres carrés non linéaires surviennent par exemple dans les problèmes de régressions non linéaires, où les paramètres du modèle sont recherchés afin de correspondre au mieux aux observations disponibles.

Mathématiquement parlant on peut définir l'algorithme comme ça :

Soient m fonction r_i ($i = 1, \dots, m$), de n variables $\beta = (\beta_1, \beta_2, \dots, \beta_n)$ qui vont donner les résidus par rapport à des données réelles pour un modèle par exemple. Avec $m \geq n$. L'algorithme de Gauss-Newton permet de trouver le minimum de la somme des carrés :

$$S(\beta) = \sum_{i=1}^m r_i^2(\beta) \quad (4)$$

Si on suppose une valeur β^0 du minimum des carrés on peut écrire l'itération :

$$\beta^{s+1} = \beta^s + \delta\beta \quad (5)$$

Où l'incrément $\delta\beta$ vérifie les équations normales :

$$(J_{\mathbf{r}}^T J_{\mathbf{r}}) \delta\beta = -J_{\mathbf{r}}^T \mathbf{r} \quad (6)$$

Où \mathbf{r} est le vecteur des fonctions $\mathbf{r} = (r_1, r_2, \dots, r_m)$, et $J_{\mathbf{r}}$ est la matrice Jacobienne $m \times n$ de \mathbf{r} par rapport à β , et $J_{\mathbf{r}}^T$ sa matrice transposée.

L'idée est de trouver les paramètres β d'un modèle $y = f(x, \beta)$ permettant un meilleur ajustement des données (x_i, y_i) .

Les résidus sont donnés par les fonction r_i :

$$r_i(\beta) = y_i - f(x_i, \beta) \quad (7)$$

Alors l'incrément $\delta\beta$ peut s'exprimer en fonction de la jacobienne de la fonction f :

$$(J_{\mathbf{f}}^T J_{\mathbf{f}}) \delta\beta = J_{\mathbf{f}}^T \mathbf{r} \quad (8)$$

Si on remplace 8 dans 5 on obtient :

$$\beta^{s+1} = \beta^s - (J_{\mathbf{r}}^T J_{\mathbf{r}})^{-1} J_{\mathbf{r}}^T \mathbf{r} \quad (9)$$

C'est donc du Newton-Raphson à N dimensions (c'est donc ce que j'ai fais avec Pascal en M1).
À une dimension on a bien :

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (10)$$

Cette méthode est efficace si l'erreur quadratique est faible ou si les fonctions traitées ne sont pas trop non linéaires. Elle est très sensible aux points aberrants (loin de la courbe modèle).

Il peut y avoir des problèmes de convergence si la matrice $J_{\mathbf{r}}$ est mal conditionnée (rapport entre la plus grande valeur et la plus petite de la matrice en valeur absolue), ou si le point de départ est trop loin du point de convergence.

3 Méthode de Levenberg et Marquardt

La méthode de Levenberg et Marquardt est une amélioration de la méthode de Gauss-Newton.

Dans le cas général on a :

$$\beta^{s+1} = \beta^s + \alpha \delta\beta \quad (11)$$

Mais il peut arriver que le α de l'équation 11 soit proche de 0, c'est là que la méthode de Levenberg et Marquardt intervient. Dans cette méthode on modifie l'équation 9 pour que l'incrément soit décalé dans la direction de la pente la plus forte :

$$(J^T J + \lambda D) \delta\beta = J^T r \quad (12)$$

Où D est une matrice diagonale positive, et λ , le paramètre de Marquardt (aussi appelé paramètre d'amortissement). Il est possible de déterminer ce paramètre par une méthode de recherche linéaire, mais ce n'est pas une bonne idée car cela rend l'algorithme très lent puisqu'il faut le recalculer à chaque itération.

Le plus simple est d'incrémenter λ quand la diminution des résidus s'accélère, et le diminuer si elle ralentit. On peut dire que de ce point de vu là, cette méthode se rapproche de la méthode du gradient.

Le gros avantage, est qu'il peut converger même si le point de départ est très éloigné du point de convergence. Mais il peut arrivé qu'il soit un peu plus lent à converger sur des fonctions très irrégulières.

4 Exemple

Faisons un exemple avec une fonction F avec 2 fonctions :

$$F : \begin{pmatrix} x \\ y \end{pmatrix} \mapsto \begin{pmatrix} f_1(x, y) = x^2 + y^2 \\ f_2(x, y) = x \cdot y \end{pmatrix} \quad (13)$$

La matrice jacobienne de la fonction F est définie comme suit :

$$J_F(x, y) = \begin{pmatrix} 2x & 2y \\ y & x \end{pmatrix} \quad (14)$$

On peut calculer la matrice transposée J_F^T :

$$J_F^T(x, y) = \begin{pmatrix} 2x & y \\ 2y & x \end{pmatrix} \quad (15)$$

Ensuite on calcule $J_F^T \times J_F$:

$$J_F^T(x, y) \times J_F(x, y) = \begin{pmatrix} 2x & y \\ 2y & x \end{pmatrix} \times \begin{pmatrix} 2x & 2y \\ y & x \end{pmatrix} \quad (16)$$

$$= \begin{pmatrix} 4x^2 + y^2 & 4xy + xy \\ 4xy + xy & 4x^2 + y^2 \end{pmatrix} \quad (17)$$

$$= \begin{pmatrix} 4x^2 + y^2 & 5xy \\ 5xy & 4x^2 + y^2 \end{pmatrix} \quad (18)$$