

Chapitre 5

DÉRIVATION, INTÉGRATION

1 Introduction

Dans ce chapitre, je m'intéresse au calcul numérique de la dérivée en un point d'une fonction et au calcul (numérique) de l'intégrale d'une fonction sur un intervalle donné. Pour l'une ou l'autre tâche, il est bon de traiter séparément le cas d'une fonction définie analytiquement (par une formule) et le cas d'une fonction définie numériquement (par une table de valeurs).

2 Dérivée d'une fonction analytique

2.1 Théorème de Taylor

Je considère une fonction f définie par une formule ou un algorithme, dérivable dans un intervalle. Cette fonction est trop compliquée pour que le calcul analytique de sa dérivée soit pratique, mais je souhaite pourtant connaître la valeur numérique de $f'(x)$. Le procédé le plus simple s'inspire de la définition même de la dérivée. Soit $f(x)$ la fonction à dériver, je sais que :

$$f' \cong \frac{f(x+h) - f(x)}{h}.$$

Comment faut-il choisir h ? Ce paramètre doit être plus grand que le plus petit réel représentable (environ 10^{-38} en simple précision), sinon le calcul s'arrête et la machine affiche «Division par Zéro!». h doit aussi être assez grand pour que $f(x)$ et $f(x+h)$ soit reconnus comme différents (10^{-7} en simple précision). Je suppose que h est assez grand pour être à l'abri des erreurs d'arrondi. Il faut maintenant éviter de tomber de Charybde en Scylla, et garder h assez petit pour que les erreurs de troncation soient négligeables. Ces dernières sont faciles à estimer. Je peux écrire une version exacte de la formule précédente, d'après le théorème de Taylor :

$$f' = \frac{f(x+h) - f(x)}{h} + \frac{h}{2} f''(x + \theta h); 0 \leq \theta \leq 1.$$

Vous voyez que l'erreur est $O(h)$, ou encore que la formule est du premier ordre : elle est exacte pour une fonction linéaire.

En faisant un petit effort d'imagination, je peux construire une formule plus précise, mais qui ne me coûtera aucun travail supplémentaire (même nombre d'évaluations de la fonction f). Appliquons deux fois le théorème de Taylor, pour les points $x-h$ et $x+h$:

$$\begin{aligned} f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2} f''(x) + O(h^3) \\ f(x-h) &= f(x) - hf'(x) + \frac{h^2}{2} f''(x) + O(h^3) \end{aligned}$$

Grâce à la parité des termes en h^2 , ceux-ci disparaissent de la différence terme à terme des deux lignes précédentes :

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + O(h^2).$$

Les erreurs d'arrondi sont les mêmes que précédemment, mais l'erreur de troncation est maintenant en h^2 , la formule est exacte pour les fonctions quadratiques.

Exemple. On a extrait d'une table les valeurs suivantes :

| x | $\ln x$ |
|-------|----------|
| 0.159 | -1.83885 |
| 0.160 | -1.83258 |
| 0.161 | -1.82635 |

et on demande de calculer la dérivée de $\ln x$ en 0.160. J'ai trouvé

$$\frac{f(0.161) - f(0.160)}{0.001} = 6.23 \quad ; \quad \frac{f(0.161) - f(0.159)}{0.002} = 6.25.$$

Quelle est la valeur exacte ?

2.2 Méthode des coefficients indéterminés

La méthode des coefficients indéterminés permet de retrouver très facilement les formules d'approximation des dérivées, sans toutefois donner d'indications sur le terme d'erreur. Je cherche trois nombres a_- , a_0 et a_+ tels que l'expression suivante soit «aussi vraie que possible» :

$$f'(x) = a_+ f(x+h) + a_0 f(x) + a_- f(x-h).$$

Je vais imposer trois conditions : la relation devra être vérifiée pour $f = x^0, x^1$, et x^2 (les trois premières puissances de x). Ceci s'écrit :

$$0 = a_+ + a_0 + a_-, \quad 1 = a_+(x+h) + a_0x + a_-(x-h), \quad 2x = a_+(x+h)^2 + a_0x^2 + a_-(x-h)^2.$$

Ce système de trois équations linéaires à trois inconnues se résout facilement. Compte tenu de la première équation, la deuxième s'écrit :

$$1/h = a_+ + a_0 - a_-.$$

Tenant compte de deux premières, la dernière équation devient :

$$0 = a_+ + a_-.$$

D'où je tire facilement

$$a_0 = 0, \quad a_+ = -a_- = 1/2h.$$

L'heureuse disparition de certains termes dans les équations précédentes ne doit rien au hasard. Ces relations doivent être vérifiées quel que soit x : il faut donc que x ne figure pas dans le système qui détermine les coefficients a . J'aurais donc pu partir d'une valeur particulièrement commode de x , comme $x = 0$. Avec l'habitude, on peut gagner encore un peu de temps en posant $h = 1$ et en restaurant h à la fin : f' a les dimensions de f/h .

Vous pourrez établir d'autres formules, en nombre limité uniquement par votre patience. Certaines impliqueront un grand nombre de valeurs de f , pour une erreur de troncation moindre, d'autres pourront être dissymétriques, pour s'approcher sans danger d'une discontinuité : on peut,

par exemple, calculer f'_0 en fonction de f_0, f_1 et f_2 . Elles s'obtiennent par la méthode des coefficients indéterminés, par le théorème de Taylor ou à partir des différences latérales (voir plus loin).

De façon analogue, j'obtiens une approximation de la dérivée seconde en appliquant deux fois les règles précédentes. Ainsi :

$$hf''(x) \cong f'(x) - f'(x-h) = \frac{f(x+h) - 2f(x) + f(x-h)}{h}.$$

Le terme d'erreur s'obtient aisément à l'aide du théorème de McLaurin : il dépend de $h^2 f'''$.

Exemple. La dérivée seconde de $\ln x$, au point 0.160, vaut, d'après les données précédentes, -40.

2.3 Polynôme d'interpolation

Il existe une autre méthode systématique pour construire des formules d'approximation de f' : il suffit de dériver un polynôme d'interpolation de f . Je vous rappelle la formule de Newton impliquant les différences latérales :

$$P(x_0 + hm) = f_0 + m\Delta f_0 + \frac{m(m-1)}{2}\Delta^2 f_0 + \dots$$

dont la dérivée (par rapport à $x = x_0 + hm$) est :

$$P'(x_0 + hm) = \frac{1}{h} \left[\Delta f_0 + \frac{1}{2}(2m-1)\Delta^2 f_0 + \dots \right].$$

Au point x_0 ($m = 0$), je trouve :

$$P'_0 = \frac{1}{h} \left[\Delta f_0 - \frac{1}{2}\Delta^2 f_0 + (1/3)\Delta^3 f_0 - \dots \right].$$

On peut obtenir les dérivées d'ordre supérieur par cette méthode et pas mal d'algèbre.

2.4 Accélération de la convergence

L'extrapolation de Richardson permet d'améliorer la précision d'une approximation en n'utilisant qu'une connaissance qualitative du terme d'erreur. Je commence par estimer la dérivée de f avec un pas h :

$$f'(x) \cong \frac{f(x+h) - f(x-h)}{2h} + \frac{h^2}{3} f'''(\xi_1) \equiv f'_1 + C_1 h^2,$$

puis avec un pas $2h$:

$$f'(x) \cong \frac{f(x+2h) - f(x-2h)}{4h} + 4\frac{h^2}{3} f'''(\xi_2) \equiv f'_2 + 4C_2 h^2.$$

À une certaine approximation, je peux considérer que $C_1 = C_2$, si bien qu'en ajoutant 4 fois la première équation à l'opposée de la seconde, j'obtiens :

$$f' \cong (4f'_1 - f'_2)/3 = f'_1 + (f'_1 - f'_2)/3$$

apparemment sans terme d'erreur. En réalité, l'erreur de troncation n'ôta pas disparue, mais elle est maintenant d'un ordre plus élevé (lequel ?). L'extrapolation de Richardson peut s'appliquer chaque fois que l'on utilise une approximation dépendant d'un paramètre lequel tend de façon continue vers zéro et à condition de connaître l'expression du terme principal de l'erreur.

3 Dérivée d'une fonction empirique

Je peux avoir à calculer la dérivée d'une fonction définie par une table de valeurs, soit qu'elle ait été calculée par quelqu'un d'autre, soit qu'elle résulte de mesures expérimentales. Les méthodes précédentes se révèlent être malcommodes dans ce cas. D'une part, le pas h est maintenant imposé et n'est pas forcément adapté au but que je poursuis. De plus, les entrées dans la table ne sont pas forcément équidistantes. D'autre part (et là, je pense surtout à des résultats expérimentaux), les inévitables erreurs de mesures vont être amplifiées par la division par h (qui est «petit»). Considérez en effet deux entrées successives prises dans une table de résultats expérimentaux :

| variable indépendante | variable dépendante |
|-----------------------|---------------------|
| x_k | $y_k + b_k$ |
| x_{k+1} | $y_{k+1} + b_{k+1}$ |

J'ai supposé que les valeurs de x étaient connues sans erreur, alors que les valeurs de y étaient entachées d'une erreur aléatoire (un bruit) b . Une valeur approchée de la dérivée est alors :

$$y' \cong \frac{y_{k+1} - y_k}{x_{k+1} - x_k} + \frac{b_{k+1} - b_k}{x_{k+1} - x_k}.$$

La première fraction est la dérivée cherchée ; sa valeur dépend peu de la différence $x_{k+1} - x_k$ (tant que l'on respecte les contraintes exposées au début de ce chapitre). La deuxième fraction est une fluctuation aléatoire d'autant plus grande que l'intervalle tabulaire est petit.

Une première manière d'éviter ces difficultés consiste à construire un polynôme d'interpolation s'appuyant sur les points $x_{-m}, x_{-m+1}, \dots, x_0, \dots, x_m$. En supposant que c'est f'_0 qui m'intéresse, je n'ai plus qu'à dériver ce polynôme. Le polynôme de Lagrange ou la fonction spline, avec 2 à 4 pivots, donnent de bons résultats. Une autre méthode consiste à construire un polynôme d'approximation au sens des moindres carrés, puis à dériver cette expression. Pour des abscisses équidistantes, le polynôme et ses dérivées ont des expressions simples (on parle souvent de polynômes de Golay). C'est la méthode que je recommande.

4 Intégration numérique : généralités

Je regarde maintenant vers le calcul numérique d'intégrales (on dit aussi quadrature numérique). Il importe de distinguer le cas des intégrales définies :

$$I = \int_a^b f(x) dx$$

où je cherche un nombre I connaissant un intervalle $[a, b]$ et la fonction f , des intégrales indéfinies :

$$J(x) = \int_a^x f(x) dx$$

où je calcule en fait une suite de nombres, représentant la fonction $J(x)$. Ce dernier cas est généralement abordé comme un problème différentiel :

$$J' = f(x); J(a) = 0.$$

et ceux-ci seront traités au chapitre suivant.

Je vais donc chercher à approcher numériquement une intégrale définie. Toutes les méthodes que j'envisagerai peuvent s'écrire sous la forme la forme :

$$I = \int_a^b f(x)dx = \sum_1^n H_j f(x_j) + E$$

où E est l'erreur de troncation, les x_j des abscisses (pivots ou noeuds) et les H_j des « poids ». Je décrirai deux sortes d'algorithmes. Pour une première classe (Newton-Cotes), les pivots sont équidistants et les poids sont les seuls paramètres ajustables ; comme il y en a n , je pourrai satisfaire n contraintes et rendre la méthode exacte pour un polynôme de degré $n-1$. Dans une deuxième classe (Gauss), les pivots et les poids seront considérés comme ajustables ; le choix de ces $2n$ paramètres nous permettra de rendre la méthode exacte pour un polynôme de degré $2n-1$. Évidemment, les méthodes de Gauss sont plus performantes que celles de Newton-Cotes, puisqu'elles permettent une précision plus grande pour un même temps de calcul, au pris d'une théorie plus compliquée et d'une programmation un peu plus lourde.

5 Méthodes de Newton-Cotes

Comme je ne sais pas intégrer analytiquement la fonction f , je la remplace par un polynôme d'interpolation dont l'intégration est banale. J'appelle h l'intervalle entre pivots. Les diverses méthodes de ce groupe se distinguent par le choix des pivots dans l'intervalle d'intégration, supposé fini.

5.1 Intervalle fermé

Je vais calculer l'intégrale de $f(x)$ sur le segment $[a, b]$, divisé en n intervalles de taille h ; je pose donc

$$h = (b - a)/n, \quad x_0 = a, \quad x_n = b, \quad x_j = a + jh.$$

Le polynôme d'interpolation de Lagrange construit sur les pivots $x_j, 0 \leq j \leq n$, s'écrit :

$$p(x) = \sum_0^n L_j(x)f(x_j)$$

où $L_j(x)$ représente un polynôme élémentaire de Lagrange. L'intégrale de $p(x)$ est une approximation de I :

$$I = \int_a^b f(x)dx \cong \sum_0^n H_j f_j + E$$

où j'ai posé $f_j = f(x_j)$, $H_j = \int_a^b L_j(x)dx$. Une démonstration trop longue pour être exposée ici permet d'aboutir aux formules ci-dessous pour l'erreur de troncation. Elles proviennent de l'intégrale du terme d'erreur de la formule de Lagrange. Celle-ci contient un facteur $f^{(n+1)}(\xi)$ ou $f^{(n+2)}(\xi)$, où ξ est une fonction inconnue de x .

$$E = \frac{f^{(n+1)}(\xi)}{(n+1)!} \int_0^n u(u-1)(u-2) \cdots (u-n)du, \text{ n impair,}$$

$$E = \frac{f^{(n+2)}(\xi)}{(n+2)!} \int_0^n u^2(u-1)(u-2) \cdots (u-n)du, \text{ n pair.}$$

Les «nombres de Cotes» H_j ont été calculés une fois pour toutes pour $n \leq 10$. Les formules de Newton-Cotes les plus courantes sont résumées ci-dessous, avec la notation : $H_j = hAW_j$. C est le coefficient de $h^{k+1}f^{(k)}$ dans le terme d'erreur, où $k = n+1$ si n est impair et $k = n+2$ si n est pair.

| n | A | W_0 | W_1 | W_2 | W_3 | W_4 | C | |
|-----|------|-------|-------|-------|-------|-------|--------|------------|
| 1 | 1/2 | 1 | 1 | | | | -1/12 | trapèzes |
| 2 | 1/3 | 1 | 4 | 1 | | | -1/90 | Simpson |
| 3 | 3/8 | 1 | 3 | 3 | 1 | | -3/80 | Simpson |
| 4 | 2/45 | 7 | 32 | 12 | 32 | 7 | -8/945 | Villarceau |

Ces formules se démontrent très facilement par la méthode des coefficients indéterminés, à condition de ne pas demander l'expression de l'erreur. Je prends l'exemple de la première formule de Simpson. Je cherche trois nombres W_0, W_1 et W_2 tels que :

$$\int_a^b f(x)dx = W_0f(a) + W_1f(a+h) + W_2f(a+2h)$$

où $b = a + 2h$. Cette relation doit être exacte pour $f = x^0, x^1$ et x^2 , ce qui conduit au système :

$$\begin{cases} 2h = W_0 + W_1 + W_2, \\ 2ah + 2h^2 = aW_0 + (a+h)W_1 + (a+2h)W_2, \\ (1/3)[6a^2h + 12ah^2 + 8h^3] = W_0a^2 + W_1(a+h)^2 + W_2(a+2h)^2. \end{cases}$$

En substituant la première équation dans la seconde, puis les deux premières dans la troisième, je trouve :

$$\begin{cases} 2h = W_0 + W_1 + W_2, \\ 2h = W_1 + 2W_2, \\ (8/3)h = W_1 + 4W_2. \end{cases}$$

système dont la solution est : $W_0 = W_2 = h/3, W_1 = 4h/3$. Ici encore, j'aurais pu poser $a = 0$ dès le début, puisque les coefficients doivent être indépendants de l'abscisse initiale.

5.2 Intervalles ouverts

Il peut arriver que la fonction à intégrer ne soit pas définie pour l'une des bornes de l'intervalle d'intégration (ou pour les deux) ou qu'elle y présente une singularité quelconque. Il est alors souhaitable de ne pas s'approcher «trop» de cette limite, ce qui est impossible avec le découpage de l'intervalle d'intégration que j'ai choisi ($x_0 = a, x_n = b$). Il existe des formules de quadrature dites «ouvertes» (parce qu'elles impliquent des intervalles ouverts) qui n'utilisent pas les valeurs de la fonction aux bornes. Elles s'écrivent en général :

$$I = \int_a^b f(x)dx = \sum_1^{n-1} H_j f(x_j) + E$$

où les bornes x_0 et x_n **ne figurent pas**. La formule «ouvertes» la plus courante, que j'appellerai «formule du point milieu», par analogie avec le terme anglais, est définie (avec les notations précédentes) par $n = 2, A = 2$ et $W_1 = 1$, soit encore :

$$I = \int_a^{a+2h} f(x)dx = 2hf[(a+b)/2] + (h^3/24)f''(\xi).$$

5.3 Formules composées

Que faire si la précision d'une intégrale numérique me paraît insuffisante ? Une première possibilité consiste à choisir une formule d'intégration d'ordre plus élevé. Comme dans le cas de l'interpolation polynômiale, cette solution est à rejeter, et pour les mêmes raisons : lorsque l'ordre n devient grand (quand le degré du polynôme que l'on intègre en lieu et place de la fonction devient grand), le terme d'erreur peut avoir un comportement anarchique en fonction de ξ . La bonne méthode est semblable à celle utilisée pour l'interpolation spline : découper l'intervalle d'intégration en plusieurs «sous-intervalles» et utiliser une méthode d'ordre peu élevé dans chaque sous-intervalle. Comme la borne droite du sous-intervalle de rang p coïncide avec la limite gauche du sous-intervalle de rang $p + 1$, je vais économiser un certain nombre d'évaluations de la fonction.

Je divise par exemple l'intervalle global $[a, b]$ en m sous intervalles et j'applique la méthode de Newton «fermée» d'ordre un dans chaque sous-intervalle. Il vient :

$$I = \int_a^b f(x)dx = (h/2)[f_0 + f_1 + f_1 + f_2 + \cdots + f_{m-2} + f_{m-1} + f_{m-1} + f_m].$$

J'aboutis ainsi à la formule composée (ou composite ou encore étendue) de Newton-Cotes d'ordre 1 (souvent appelée encore méthode des trapèzes) :

$$I = \int_a^b f(x) = h[f_0/2 + f_1 + f_2 + \cdots + f_{m-2} + f_{m-1} + f_m/2] + E$$

où tous les termes «intérieurs» ont doublé. L'application du théorème de la moyenne montrerait que

$$E = -(h^2/12)(b - a)f''(\xi).$$

Dans la pratique, on programme ce calcul de façon itérative. Supposez que je connaisse $I(m)$, calculée avec m sous-intervalles, mais que la précision me paraisse encore insuffisante. Je calcule alors $I(2m)$ à partir de $I(m)$ et des seules nouvelles valeurs de f , aux points intermédiaires, d'abscisses $(i + 1/2)h$. Je recommence tant que le critère de convergence n'est pas atteint. Quel critère d'arrêt vais-je choisir ? Ce peut être la condition que la variation relative de l'intégrale d'une étape à l'autre est inférieure à un seuil choisi à l'avance : $|[I(m+1) - I(m)]/I(m)| \leq \varepsilon$.

On opère de la même manière avec les formules de Newton-Cotes «ouvertes». Cependant, doubler le nombre d'intervalles n'apporte aucune économie, puisque ceux-ci ne comportent aucun pivot commun. Je vous suggère de vérifier qu'il faut tripler le nombre d'intervalles à chaque itération pour économiser des calculs. Il est possible de combiner une formule de Newton-Cotes ouverte avec $m - 1$ formules fermées : ceci pourrait servir à éviter une région «dangereuse» autour de $x = a$.

Comme les termes d'erreur de toutes les méthode de Newton-Cotes sont connus, il est tentant d'améliorer la précision d'une quadrature, sans trop d'effort, par l'extrapolation de Richardson. En réalité, cela se fait rarement, car on connaît une généralisation puissante de l'extrapolation de Richardson : c'est la méthode de Romberg que je vais décrire dans le paragraphe suivant.

Exemple. On demande de calculer l'intégrale de $1/x$ de 1 à 3 avec deux puis quatre sous-intervalles, par la méthode des trapèzes. Je trouve :

$$m = 2 : \quad I_2 \cong (1) \left[\frac{1}{2}(1) + 1/2 + \frac{1}{2}(1/3) \right] = 7/6 = 1,166666\dots;$$

$$m = 4 : \quad I_4 \cong (1/2) \left[\frac{1}{2}(1) + 2/3 + 1/2 + 2/5 + \frac{1}{2}(1/3) \right] = 67/60 = 1,116666\dots$$

La formule de Richardson donne : $I \cong I_4 + (1/3)(I_4 - I_2) = 11/10 = 1,1$, soit une erreur relative un peu supérieure à $1/1000$.

6 Méthode de Romberg

J'utilise la notation $I_{k,0}$ = valeur approchée de I , calculée par la méthode des trapèzes en utilisant $m = 2^k$ sous-intervalles. En utilisant une expression du terme d'erreur plus précise que celle mentionnée (et non démontrée) au paragraphe précédent, je pourrais écrire

$$I_{k,0} = I - \sum_{j=1}^{\infty} \alpha_j \left[\frac{b-a}{2^k} \right]^{2j}.$$

Si je répète le même calcul avec des intervalles deux fois plus nombreux et deux fois plus petits, j'ai

$$I_{k+1,0} = I - \sum_{j=1}^{\infty} \alpha_j \left[\frac{b-a}{2^{k+1}} \right]^{2j}.$$

Comme pour une extrapolation de Richardson, je forme :

$$I_{k,1} = 4I_{k+1,0} - I_{k,0} = 3I - \sum_{j=2}^{\infty} \alpha_j \left[\frac{b-a}{2^k} \right]^{2j} [-1 + 4/2^{2j}].$$

Je viens de gagner un ordre dans le développement de l'erreur en fonction des puissances de $(b-a)$. Le procédé se généralise aisément. Je construis un tableau triangulaire dont la première colonne est formée des $I_{k,0}$, la deuxième des $I_{k,1}$ et dont le terme général s'écrit :

$$I_{k,m+1} = \frac{4^m I_{k+1,m} - I_{k,m}}{4^m - 1}.$$

Le gros du travail est effectué lors du calcul des $I_{k,0}$, puisque c'est à ce moment-là que je calcule la fonction compliquée f ; la suite n'est qu'une série de combinaison linéaires. Si $k \leq K$, alors le dernier élément de la diagonale principale, $I_{K,K}$, est la meilleure approximation de I que l'on puisse obtenir en évaluant 2^K fois f .

7 Intégration de Gauss

J'abandonne maintenant l'hypothèse que les pivots sont régulièrement répartis sur l'axe; je cherche au contraire à les répartir «au mieux» pour avoir un algorithme aussi exact que possible. J'aurais toujours

$$I = \int_a^b f(x)dx = \sum_1^n p_j f(a_j) + E.$$

Ici, la méthode des coefficients indéterminés est assez malcommode; lorsque j'impose que la formule ci-dessus soit exacte pour $x^0, x^1, \dots, x^k, \dots$, j'obtiens un système d'équations non-linéaires (en a_j) dès que $k > 1$, système dont la solution est en général inaccessible.

Je vais plutôt utiliser un raisonnement indirect, à partir du polynôme d'interpolation de Hermite. La formule d'interpolation de Hermite s'écrit :

$$f(x) = \sum_1^n h_j(x)f(a_j) + \sum_1^n h_j^*(x)f'(a_j) + [\pi(x)]^2 \frac{f^{2n}(\xi)}{(2n)!}.$$

Le terme d'erreur disparaît si f est un polynôme de degré $2n - 1$ au plus. J'intègre terme à terme cette relation entre les abscisses a et b :

$$I = \int_a^b f(x)dx = \sum_1^n H_j f(a_j) + \sum_1^n H_j^* f'(a_j) + E$$

où H_j et H_j^* sont respectivement les intégrales de $h_j(x)$ et de $h_j^*(x)$ entre les mêmes bornes. Cette formule d'intégration est exacte pour tout polynôme de degré inférieur à $2n$: c'est le mieux que je puisse faire avec les $2n$ paramètres H_j, a_j .

Pour obtenir la forme annoncée, il faut que les H_j^* soient tous nuls. Comment parvenir à ce résultat ? Les seuls paramètres dont je dispose sont les a_j . Je dois donc choisir ces nombres de telle manière que l'intégrale de h_j^* soit nulle. Je vous rappelle la forme de ce polynôme (établie dans le chapitre 2) : $h_j^*(x) = (x - a_j)[L_j(x)]^2$, si L_j est le polynôme élémentaire de Lagrange construit sur les $\{a_j\}$. En utilisant une fois la relation $L_j = \pi(x)/((x - a_j)\pi'(a_j))$, j'obtiens

$$H_j^* = \int_a^b (x - a_j)[L_j(x)]^2 = \int_a^b \pi(x) \frac{L_j(x)}{\pi'(a_j)} dx.$$

Je veux que cette intégrale soit nulle ; autrement dit, je veux que les polynômes $\pi(x)$ et $L_j(x)$ soient orthogonaux sur le segment $[a, b]$ par rapport à la fonction de poids $w \equiv 1$.

Vous savez que $\pi(x)$ (défini en 2.5) est le produit de tous les termes de la forme $x - a_j$; $\pi(x)$ est défini (implicitement) par l'ensemble de ses zéros. Il doit être orthogonal aux L_j , polynômes élémentaires de Lagrange pour n pivots, donc de degré $n - 1$. Plutôt que d'imposer cette condition particulière, je vais imposer une condition plus générale : π devra être orthogonal à tout polynôme de degré $n - 1$ ou inférieur. Je vous laisse le soin de démontrer la proposition réciproque et j'admets que la condition nécessaire et suffisante pour que la formule de Gauss soit valable est que $\pi(x)$ soit orthogonal à tout polynôme de degré inférieur, pour une fonction de poids égale à l'unité, sur $[a, b]$.

À quelques détails près, vous connaissez des polynômes répondant à cette définition : ce sont les polynômes de Legendre. Pour que l'identification soit parfaite, il faut ramener l'intervalle d'intégration à $[-1, 1]$ et, si l'on veut être très soigneux, normaliser correctement $\pi(x)$. Le changement de variable $x = \frac{1}{2}(a + b) - \frac{1}{2}(a - b)t$ permet d'intégrer sur $[-1, 1]$:

$$I = \int_a^b f(x)dx = \frac{1}{2}(b - a) \int_{-1}^1 f(t)dt.$$

Si maintenant je choisis comme pivots a_j les zéros du polynôme de Legendre de degré n , $\pi(x)$ coïncidera, à un facteur constant près, avec ce polynôme ; sur le segment $[-1, 1]$ il sera donc orthogonal à tout polynôme de degré inférieur, et en particulier aux $L_j(x)$.

Il reste à calculer les poids, à partir de l'expression des $h_j(x)$. On montre facilement que

$$H_k = \int_{-1}^1 [L_k(x)]^2.$$

d'où l'on déduit que

$$H_k = \int_{-1}^1 L_k(x) dx.$$

Il existe des tables des arguments et des poids pour l'intégration de Gauss, pour n jusqu'à 128. Pour éviter les erreurs de transcription, il vaut mieux écrire un sous-programme pour les recalculer.

Exemple. Je calcule la même intégrale qu'au paragraphe précédent, avec trois pivots. Les tables donnent :

$$\begin{array}{rcl} a_j & 0 & \pm 0.774597 \\ H_j & 8/9 & 5/9 \end{array}$$

Le changement de variable $y = x - 2$ transforme l'intervalle $[1, 3]$ en $[-1, 1]$ et l'intégrant en $1/(y+2)$. Un calcul élémentaire donne alors $I \cong 1.098039$, pour une erreur relative un peu supérieure à $5 \cdot 10^{-4}$.

8 Généralisations

En choisissant un autre intervalle d'intégration et une autre fonction de poids, on peut construire de nouvelles formules d'intégration de type gaussien. Je sais par exemple que les polynômes de Laguerre sont orthogonaux sur $[0, \infty[$ par rapport à la fonction de poids e^{-x} . Je peux donc écrire :

$$\int_0^{\infty} e^{-x} f(x) dx = \sum_1^n W_j f(a_j) + E,$$

où les a_j sont les zéros du polynôme de Laguerre d'ordre n et les W_j des poids calculables facilement. Chaque famille de polynômes orthogonaux donne ainsi naissance à une formule d'intégration.

Il ne faut pas s'empresse de conclure que l'on peut intégrer n'importe quelle fonction entre zéro et l'infini, au prix du calcul des valeurs de f en quelques points. Par ailleurs, aucun algorithme numérique ne pourra transformer une intégrale divergente en intégrale convergente. J'écris quelques formules pour préciser la question.

$$\int_0^{\infty} g(x) dx = \int_0^{\infty} e^{-x} [e^x g(x)] dx = \int_0^{\infty} e^{-x} f(x) dx$$

où $f = e^x g$; ces relations n'ont rien d'anormal, mais c'est lorsque je tenterai d'appliquer la formule de Gauss-Laguerre à f que les problèmes apparaîtront. Si g ne décroît pas assez vite à l'infini, le terme d'erreur, en $f^{(2n)}$, deviendra rapidement intolérable.

Que faire si la précision sur l'intégrale obtenue par la méthode de Gauss est insuffisante? Il existe ici encore deux possibilités : utiliser une formule d'ordre supérieure ou employer une méthode composée. Comme d'habitude, la première solution est à rejeter. La seconde est assez simple à mettre en oeuvre. Je divise l'intervalle $[a, b]$ en m sous-intervalles, $[x_m, x_{m+1}]$. Pour chaque sous-intervalle, je fait un changement de variable qui ramène les bornes à $-1, 1$ et j'emploie une formule de Gauss d'ordre peu élevé. Comme aucun pivot ne coïncide avec les bords des intervalles, on n'économise aucun calcul, à la différence des méthodes de Newton fermées. Cependant, la précision est largement supérieure à ce que l'on obtient avec Newton-Cotes ; de plus, l'existence de discontinuités en a ou b est peu gênante, puisque ces points ne sont pas des pivots.

9 Intégrales généralisées

Une intégrale généralisée est mathématiquement bien définie ; l'ennui, c'est que l'ordinateur ne le sait pas. Par exemple, le calcul analytique de

$$J = \int_0^{\infty} x^n e^{-x} dx$$

est facile, mais, par définition, un algorithme comporte un nombre fini d'opérations. Comment alors intégrer numériquement jusqu'à $+\infty$? Vous connaissez une réponse dans ce cas particulier : on peut employer la méthode de Gauss-Laguerre.

Pour un intégrand quelconque, je peux calculer certaines intégrales généralisées en faisant un changement de variable astucieux. Si $a > 0$, je peux écrire :

$$\int_a^{\infty} f(x) dx = \int_0^{1/a} \frac{1}{t^2} f\left(\frac{1}{t}\right) dt.$$

Un changement de variable analogue est valable pour $a < 0$.

Il peut aussi arriver qu'un même changement de variable soit malcommode sur l'ensemble de l'intervalle d'intégration : il faut alors couper celui-ci en deux ou plusieurs morceaux.

Je suppose maintenant que $f(x)$ est équivalente à $(x-a)^{-1/2}$ pour x proche de a ($a < b$). Un nouveau changement de variable fera disparaître cette singularité intégrable :

$$\int_a^b f(x) dx = \int_0^{\sqrt{b-a}} 2t f(a+t^2) dt.$$

10 Intégrales multiples

J'examine, pour fixer les idées, le cas d'une intégrale double ; je cherche l'intégrale de $f(x, y)$ dans un domaine \mathcal{D} . Les valeurs extrêmes de x dans \mathcal{D} sont x_1 et x_2 ; pour une valeur donnée de x , les valeurs extrêmes de y sont $y_1(x)$ et $y_2(x)$. J'intègre d'abord par rapport à y , puis par rapport à x :

$$J = \int \int f(x, y) dx dy = \int_{x_1}^{x_2} dx \int_{y_1(x)}^{y_2(x)} f(x, y) dy$$

Ayant fait le choix d'une méthode numérique pour calculer l'intégrale en y , je peux programmer une fonction dont la valeur sera :

$$I(x) = \int_{y_1}^{y_2} f(x, y) dy$$

où les bornes y_1 et y_2 dépendent de x . Je suis maintenant capable de calculer l'intégrale de $I(x)$:

$$J = \int_{x_1}^{x_2} I(x) dx$$

On remarque que la suite logique des opérations dans l'ordinateur est l'inverse de celle qui vient d'être décrite. Le programme d'intégration en x «choisit» les pivots dont il a besoin ; pour chaque pivot, il appelle $I(x)$, qui «décide» pour son propre compte des valeurs de y à utiliser. L'ensemble est en général plus précis que le calcul *a priori* de $f(x, y)$ sur un quadrillage régulier de points, suivi d'intégrations. Dernière remarque : si l'on utilise un sous-programme d'intégration («trapèze» par exemple), qui sera appelé aussi bien pour calculer $I(x)$ que J , il faut prendre garde à ne pas laisser croire au programme que l'on utilise une procédure récursive, s'appelant elle-même.