Wpisz pełny tytuł skryptu

Imię Drugie-imię Nazwisko-PIERWSZEGO-autora
E-mail: email1@mimuw.edu.pl
WWW: http://adres.strony1.www

Imię Drugie-imię Nazwisko-DRUGIEGO-autora
E-mail: email2@mimuw.edu.pl
WWW: http://adres.strony2.www

2 stycznia 2012

Streszczenie. Wpisz krótką informację o tematyce skryptu (kilka zdań)

Sk?ad w systemie LaTeX, z wykorzystaniem m.in. pakiet?w beamer oraz listings. Szablony podr?cznika i prezentacji: Piotr Krzy?anowski, projekt: Robert D?browski.

Spis treści

1 Zagadnienie własne I **4**

1.1 Podstawy teoretyczne 4

1.2 Teoria zaburzeń dla macierzy niesymetrycznych 5

1.2.1 Wstępny przykład 5

1.2.2 Wrażliwość wartości własnych 6

1.2.3 Wrażliwość wektorów własnych 8

1.3 Teoria zaburzeń dla macierzy symetrycznych 9

1.3.1 Wrażliwość wartości własnych 9

1.3.2 Wrażliwość wektorów własnych 10

2 Zagadnienie własne II **13**

2.1 Uwagi wstępne 13

2.1.1 Sprowadzanie macierzy do postaci Hessenberga 14

2.2 Metoda Hymana 14

2.3 Metoda potęgowa 16

2.3.1 Definicja metody 16

2.3.2 Analiza zbieżności 16

2.4 Modyfikacje metody potęgowej 19

3 Zagadnienie własne III **21**

3.1 Iteracje podprzestrzeni 21

3.1.1 Algorytm ogólny 21

3.1.2 Iteracje ortogonalne 22

3.2 Metoda QR 24

3.2.1 Wyprowadzenie metody 24

3.2.2 QR a iteracje ortogonalne 24

3.2.3 Analiza zbieżności 25

3.3 QR z przesunięciami 27

4 Kwadratury interpolacyjne w wielu wymiarach **28**

4.1 Sformułowanie zadania 28

4.2 Interpolacja na siatkach regularnych 29

4.2.1 Postać wielomianu interpolacyjnego . . . 29
4.2.2 Błąd interpolacji . . . 31
4.3 Kwadratury interpolacyjne . . . 32
4.3.1 Kwadratury proste . . . 32
4.3.2 Kwadratury złożone . . . 33
4.4 Przekleństwo wymiaru . . . 34

5 Metody Monte Carlo **36**
5.1 Wstęp, metody niedeterministyczne . . . 36
5.2 Klasyczna metoda Monte Carlo . . . 36
5.2.1 Definicja i błąd . . . 36
5.2.2 Całkowanie z wagą . . . 38
5.3 Redukcja wariancji . . . 39
5.3.1 Losowanie warstwowe . . . 39
5.3.2 Funkcje kontrolne . . . 41
5.4 Generowanie liczb (pseudo-)losowych . . . 42
5.4.1 Liniowy generator kongruencyjny . . . 42
5.4.2 Odwracanie dystrybuanty i ‘akceptuj albo odrzuć’ . . . 43
5.4.3 Metoda Box-Muller dla rozkładu gaussowskiego . . . 44

6 Metody quasi-Monte Carlo **45**
6.1 Co to są metody quasi-Monte Carlo? . . . 45
6.2 Dyskrepancja . . . 46
6.3 Błąd quasi-Monte Carlo . . . 47
6.3.1 Formuła Zaremby . . . 47
6.3.2 Nierówność Koksmy-Hlawki . . . 49
6.4 Ciągi o niskiej dyskrepancji . . . 50
6.4.1 Ciąg Van der Corputa . . . 50
6.4.2 Konstrukcje Haltona i Sobol’a . . . 51
6.4.3 Sieci (t, m, d) i ciągi (t, d) . . . 52

Rozdział 1

Zagadnienie własne I

Trzy początkowe wykłady poświęcimy *zagadnieniu własnemu*. Naszym zadaniem będzie obliczenie, a raczej numeryczna aproksymacja wartości własnych danej macierzy kwadratowej (jednej, kilku, lub wszystkich). W ogólniejszym sformułowaniu możemy chcieć obliczyć również odpowiednie wektory własne.

Pierwszy wykład będzie dotyczyć przede wszystkim wrażliwości wartości i wektorów własnych na zaburzenia macierzy. Jak wiemy z podstawowego wykładu analizy numerycznej, jest to istotne z punktu widzenia numerycznej jakości algorytmów.

1.1 Podstawy teoretyczne

Zaczniemy od przypomnienia podstawowych pojęć i faktów z algebry liniowej dotyczących zagadnienia własnego, z których skorzystamy w dalszej części wykładu.

Definicja 1.1. Liczbę $\lambda \in \mathbb{C}$ (zespoloną) nazywamy *wartością własną* macierzy kwadratowej $A \in \mathbb{C}^{n,n}$ jeśli istnieje niezerowy wektor $\vec{x} \in \mathbb{C}^n$ taki, że

$$A * \vec{x} = \lambda * \vec{x}.$$

Wektor $\vec{x}$ nazywamy *wektorem własnym* odpowiadającym wartości własnej λ.

Równoważnie, λ jest wartością własną macierzy A gdy jest zerem jej wielomianu charakterystycznego, tzn. gdy wyznacznik

$$\det(A - \lambda * I) = 0,$$

gdzie I jest macierzą identycznościową $n \times n$. *Krotnością algebraiczną* wartości własnej λ nazywamy jej krotność jako zera wielomianu charakterystycznego.

Zbiór wszystkich wektorów własnych odpowiadających danej wartości własnej λ, uzupełniony o wektor zerowy, czyli zbiór $\{\vec{x} \in \mathbb{C}^n : A * \vec{x} = \lambda * \vec{x}\}$, jest podprzestrzenią liniową $\mathbb{C}^n$ zwaną *podprzestrzenią własną* odpowiadającą λ. Wymiar tej podprzestrzeni to *krotność geometryczna* wartości własnej λ.

Wektory własne odpowiadające różnym wartościom własnym są liniowo niezależne.

Macierze $A, B \in \mathbb{C}^{n,n}$ są *podobne* gdy istnieje nieosobliwa macierz $C \in \mathbb{C}^{n,n}$ taka, że $B = C^{-1} * A * C$. Oczywiście, relacja 'bycia macierzami podobnymi' jest symetryczna.

Macierze podobne mają te same wartości własne. Jeśli bowiem $A * \vec{x} = \lambda * \vec{x}$ to $B * (C * \vec{x}) = \lambda * (C * \vec{x})$, tzn. λ jest wartością własną B, a odpowiadający jej wektor własny to $\vec{y} = C * \vec{x}$.

Macierz $A \in \mathbb{C}^{n,n}$ jest *diagonalizowalna* gdy jest podobna do macierzy diagonalnej, czyli gdy istnieje nieosobliwa $V = [\vec{v}_1, \ldots, \vec{v}_n] \in \mathbb{C}^{n,n}$ taka, że

$$V^{-1} * A * V \;=\; \Lambda \;=\; \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n).$$

Zauważmy, że wtedy możemy równoważnie zapisać $A * V = V * \Lambda$, albo $A * \vec{v}_j = \lambda_j * \vec{v}_j$ dla $1 \leqslant j \leqslant n$. Stąd elementy diagonalne macierzy Λ są wartościami własnymi macierzy A (gdzie ta sama wartość własna powtarza się tyle razy ile wynosi jej krotność), a kolumny macierzy V są odpowiednimi wektorami własnymi.

Szczególne własności mają macierze *hermitowskie* albo, w przypadku rzeczywistym, macierze *symetryczne*, bowiem dla nich istnieją bazy ortonormalne (odpowiednio w $\mathbb{C}^n$ lub $\mathbb{R}^n$) wektorów własnych. Odpowiednie twierdzenie przypominamy jedynie w przypadku rzeczywistym, który ma zasadnicze znaczenie w obliczeniach praktycznych.

Twierdzenie 1.1. *Niech A będzie macierzą symetryczną o współczynnikach rzeczywistych,*

$$A \;=\; A^T \;\in\; \mathbb{R}^{n,n}.$$

Wtedy istnieje w $\mathbb{R}^n$ baza ortonormalna wektorów własnych macierzy A, tzn. istnieje układ wektorów własnych $\vec{q}_1, \vec{q}_2, \ldots, \vec{q}_n \in \mathbb{R}^n$ taki, że

$$(\vec{q}_i, \vec{q}_j)_2 \;=\; \vec{q}_j^{\,T} * \vec{q}_i \;=\; \begin{cases} 0, & i \neq j, \\ 1, & i = j. \end{cases}$$

Oczywiście, odpowiednie wartości własne λ_j, $1 \leqslant j \leqslant n$, są też rzeczywiste.

Powyższa własność macierzy symetrycznych oznacza tyle, że są one diagonalizowalne przy pomocy macierzy *ortogonalnych*. Rzeczywiście, oznaczając

$$Q \;:=\; [\vec{q}_1, \vec{q}_2, \ldots, \vec{q}_n], \qquad \Lambda \;=\; \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$$

mamy $Q^T * Q = I = Q * Q^T$ oraz

$$Q^T * A * Q \;=\; \Lambda, \qquad A \;=\; Q * \Lambda * Q^T.$$

1.2 Teoria zaburzeń dla macierzy niesymetrycznych

Zobaczmy najpierw, czy mamy w ogóle szansę numerycznego rozwiązania zadania znalezienia wartości własnych macierzy. W tym celu zbadamy jego uwarunkowanie, czyli wrażliwość na małe względne zaburzenia współczynników macierzy.

1.2.1 Wstępny przykład

W ogólności, uwarunkowanie naszego zadania może być niestety dowolnie duże.

Przyk?ad 1.1. Niech macierz

$$J_\varepsilon \;:=\; \begin{bmatrix} \mu & 1 & 0 & \cdots & 0 \\ 0 & \mu & 1 & & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & & & \mu & 1 \\ \varepsilon & 0 & \cdots & 0 & \mu \end{bmatrix} \;\in\; \mathbb{C}^{n,n}.$$

Dla $\varepsilon = 0$, macierz $J = J_0$ jest pojedynczą klatką Jordana, której jedyną wartością własną jest μ (o krotności algebraicznej n i krotności geometrycznej 1). Jeśli teraz zaburzymy wyraz w lewym dolnym rogu o $\varepsilon > 0$ to otrzymana macierz J_ε ma już n różnych wartości własnych. Rzeczywiście, jej wielomian charakterystyczny

$$\det(J_\varepsilon - \lambda I) = (\mu - \lambda)^n + (-1)^{n+1}\varepsilon$$

ma pierwiastki

$$\mu_k = \mu + \varepsilon^{1/n} \cdot e^{2\pi\imath k/n}, \qquad 0 \leqslant k \leqslant n-1 \qquad (\imath = \sqrt{-1}).$$

Względne zaburzenie macierzy na poziomie ε powoduje więc względne zaburzenie wartości własnych na poziomie $\varepsilon^{1/n}$. Stąd, dla $n \geqslant 2$, uwarunkowanie rośnie do $+\infty$ gdy $\varepsilon \to 0^+$.

Oczywiście, otrzymane oszacowanie nie oznacza, że w powyższym przykładzie w ogóle nie potrafimy aproksymować wartości własnej. Tracimy jednak na liczbie poprawnie obliczonych cyfr znaczących wyniku. Niech $n = 2$. Wtedy przy dokładności arytmetyki 10^{-8} możemy spodziewać się jedynie wyniku z dokładnością do czterech cyfr znaczących, a przy dokładności 10^{-16} z dokładnością do ośmiu cyfr znaczących. Co więcej, im większy format n klatki Jordana tym gorzej.

Powyższy przykład pokazuje również, że praktycznie niemożliwe jest numeryczne wyznaczenie struktury macierzy. Nawet drobne zaburzenie macierzy powoduje bowiem, że pojedyncza klatka Jordana staje się macierzą diagonalizowalną.

1.2.2 Wrażliwość wartości własnych

Dalej będziemy już zakładać, że macierz A jest diagonalizowalna, czyli że jej postać Jordana jest macierzą diagonalną. Wtedy mamy następujące oszacowanie *Bauera-Fike'a*.

Twierdzenie 1.2. *Niech $A \in \mathbb{C}^{n,n}$ będzie macierzą diagonalizowalną o wartościach własnych λ_k, $1 \leqslant k \leqslant n$,*

$$A = V * \Lambda * V^{-1}.$$

Niech dalej μ będzie dowolną wartością własną macierzy 'sąsiedniej' $A + E$. Wtedy

$$\min_{1 \leqslant k \leqslant n} |\lambda_k - \mu| \leqslant \operatorname{cond}(V) \cdot \|E\|,$$

gdzie $\operatorname{cond}(V) = \|V\| \cdot \|V^{-1}\|$ oraz $\|\cdot\|$ jest normą macierzy indukowaną przez dowolną normę p-tą wektora.

Dowód. Załóżmy bez zmniejszenia ogólności, że μ jest różne od każdej wartości własnej λ_k. Niech $\vec{x}$ będzie wektorem własnym odpowiadającym wartości własnej μ macierzy $A+E$. Wtedy

$$V * (\Lambda + V^{-1} * E * V) * V^{-1} * \vec{x} = \mu * \vec{x}. \tag{1.1}$$

Oznaczając $\vec{y} = V^{-1} * \vec{x} \neq \vec{0}$ mamy dalej

$$(\Lambda - \mu * I) * \vec{y} = -V^{-1} * E * V * \vec{y}, \tag{1.2}$$

czyli

$$\|\vec{y}\| \leqslant \|\Lambda^{-1}\| \, \|V^{-1}\| \, \|V\| \, \|E\| \, \|\vec{y}\|.$$

Ponieważ $\Lambda^{-1} = \operatorname{diag}((\lambda_1 - \mu)^{-1}, \ldots, (\lambda_n - \mu)^{-1})$ to ostatecznie

$$\min_{1 \leqslant k \leqslant n} |\lambda_k - \mu| \leqslant \operatorname{cond}(V) * \|E\|.$$

□

Dla macierzy diagonalizowalnych zaburzenia wartości własnych są więc lipschitzowską funkcją zaburzeń macierzy, przy czym stała Lipschitza wynosi $\mathrm{cond}(V)$.

Podane oszacowanie jest globalne dla wszystkich wartości własnych. Zaburzenia macierzy A mogą jednak w różny sposób przenosić się na zaburzenia różnych wartości własnych. Od czego to zależy? Tak jak w dowodzie twierdzenia Bauera-Fike'a, niech

$$(A+E)*\vec{x} \;=\; \mu*\vec{x}, \qquad \|\vec{x}\|_2^2 = \vec{x}^{\,H}*\vec{x} = 1.$$

Biorąc jedynie i-tą współrzędną po obu stronach równania (1.2) dostajemy

$$(\lambda_i-\mu)*\vec{z}_i^{\,H}*\vec{x} \;=\; -\vec{z}_i^{\,H}*E*\vec{x}, \tag{1.3}$$

gdzie $\vec{z}_i$ jest znormalizowaną i-tą kolumną macierzy

$$(V^{-1})^H \;=\; [\vec{w}_1,\vec{w}_2,\ldots,\vec{w}_n],$$

tzn. $\vec{z}_i = \vec{w}_i/\|\vec{w}_i\|_2$. W szczególności, $\|\vec{z}_i\|_2 = 1$ oraz

$$\vec{z}_i \;\perp\; \mathrm{span}(\vec{v}_1,\ldots,\vec{v}_{i-1},\vec{v}_{i+1},\ldots,\vec{v}_n)$$

(ortogonalność ze względu na zwykły iloczyn skalarny). Stąd, jeśli $\vec{z}_i$ i $\vec{x}$ nie są ortogonalne to

$$|\lambda_i-\mu| \;\leqslant\; \frac{\|E\|}{|\vec{z}_i^{\,H}*\vec{x}|}.$$

Dla ustalenia uwagi załóżmy teraz, że minimum w twierdzeniu 1.2 jest osiągane dla $k=1$, a podprzestrzenią własną wartości własnej λ_1 jest

$$\mathcal{V} \;=\; \mathrm{span}\{\vec{v}_1,\vec{v}_2,\ldots,\vec{v}_s\},$$

gdzie wektory własne $\vec{v}_1,\ldots,\vec{v}_s$ tworzą bazę ortonormalną w $\mathcal{V}$. Jeśli wektor $\vec{x}$ jest 'dostatecznie blisko' podprzestrzeni własnej $\mathcal{V}$ (co, jak pokażemy w następnym podrozdziale, jest prawdą dla dostatecznie małego zaburzenia E) to $|\lambda_1-\mu|$ można w przybliżeniu oszacować z góry przez maksymalną wartość wyrażenia

$$\frac{\|E\|}{\max_{1\leqslant i\leqslant s}|\vec{z}_i^{\,H}*\vec{y}|}$$

po wszystkich $\vec{y}\in\mathcal{V}$ takich, że $\|\vec{y}\|_2=1$. Niech więc $\vec{y}=\sum_{i=1}^s a_i*\vec{v}_i\in\mathcal{V}$, przy czym $\sum_{i=1}^s|a_i|^2 = 1$. Wobec tego, że dla każdego i mamy

$$\vec{z}_i^{\,H}*\vec{y} \;=\; a_i*\vec{z}_i^{\,H}*\vec{v}_i \;=\; a_i/\|w_i\|_2,$$

interesujące nas 'max' jest najmniejsze gdy $a_i = \|w_i\|_2\left(\sum_{j=1}^s\|w_j\|^2\right)^{-1/2}$. Stąd dostajemy

$$|\lambda_1-\mu| \;\lessapprox\; \left(\sum_{i=1}^s\|w_i\|_2^2\right)^{1/2}\cdot\|E\| \qquad (\|E\|\to 0^+).$$

Ponieważ $\|w_i\| = 1/(\vec{z}_i^{\,H}*\vec{v}_i)$ oraz $\vec{z}_i$ jest ortogonalny do $\mathrm{span}\{\vec{v}_j : j\neq i\}$, otrzymana nierówność sugeruje, że wartość własna λ_1 może być bardzo wrażliwa na zaburzenia macierzy gdy odpowiadająca jej podprzestrzeń własna $\mathrm{span}\{\vec{v}_1,\ldots,\vec{v}_s\}$ jest 'prawie ortogonalna' do $\mathrm{span}\{\vec{v}_{s+1},\ldots,\vec{v}_n\}^\perp$. Dobrze ilustruje to następujący przykład.

Przyk?ad 1.2. Rozpatrzmy macierz

$$A = \begin{bmatrix} 2 & -1/\delta \\ 0 & 1 \end{bmatrix},$$

gdzie $\delta > 0$ jest 'małe'. Wartości własne A wynoszą 1 i 2, natomiast wektory własne $[1,0]^T$ i $[1,\delta]^T$ są prawie liniowo zależne. Zaburzając macierz przez dodanie $\varepsilon = -2\delta$ do wyrazu w lewym dolnym rogu dostajemy macierz o wartościach własnych 0 i 3, a odpowiadające im wektory własne wynoszą $[1,2\delta]^T$ i $[1,-2\delta]^T$. Dodajmy, że w tym przypadku $\mathrm{cond}(V)$ jest proporcjonalne do $1/\delta$.

1.2.3 Wrażliwość wektorów własnych

Przez zaburzenie wektora własnego będziemy rozumieć odległość wektora $\vec{x}$ od podprzestrzeni własnej $\mathcal{V}$, tzn.

$$\mathrm{dist}(\vec{x},\mathcal{V}) := \min\{\, \|\vec{x}-\vec{v}\|_2 : \ \vec{v}\in\mathcal{V}\,\},$$

albo, równoważnie, długość rzutu ortogonalnego $P\vec{x}$ wektora $\vec{x}$ na podprzestrzeń

$$\mathcal{V}^\perp = \mathrm{span}(\vec{z}_{s+1},\ldots,\vec{z}_n).$$

Pokażemy, że podobnie jak przypadku wartości własnych, zaburzenie wektorów własnych jest lipszitzowską funkcją $\|E\|$. W tym celu, wybierzmy w $\mathcal{V}^\perp$ dowolną bazę ortonormalną

$$Y = [\vec{y}_{s+1},\ldots,\vec{y}_n] \in \mathbb{C}^{n,n-s}.$$

Niech $B^H \in \mathbb{C}^{n-s,n-s}$ będzie macierzą przejścia z bazy $Z = [\vec{z}_{s+1},\ldots,\vec{z}_n]$ do Y,

$$Y = Z * B^H.$$

Wtedy

$$\|P\vec{x}\|_2 = \|Y^H * P\vec{x}\|_2 = \|B * Z^H * P\vec{x}\|_2.$$

Wobec równości (1.3) mamy

$$Z^H * P\vec{x} = -\mathrm{diag}\left((\lambda_{s+1}-\mu)^{-1},\ldots,(\lambda_n-\mu)^{-1}\right) * B^{-1} * Y^H * E * \vec{x}.$$

Ponieważ na podstawie twierdzenia Bauera-Fike'a mamy

$$|\lambda_1-\mu| \geqslant |\lambda_i-\lambda_1| - |\lambda_1-\mu| \geqslant |\lambda_i-\lambda_1| - \mathrm{cond}(V)\cdot\|E\|,$$

otrzymujemy następującą przybliżoną nierówność

$$\mathrm{dist}(\vec{x},\mathcal{V}) = \|P\vec{x}\|_2 \lessapprox \frac{\|B\|_2\|B^{-1}\|_2}{\min_{s+1\leqslant i\leqslant n}|\lambda_i-\lambda_1|}\,\|E\| \qquad (\|E\|\to 0^+). \tag{1.4}$$

Uwaga. W przykładzie 1.2 istotnemu zaburzeniu uległy wartości własne, natomiast wektory własne nie. Jest to zrozumiałe w świetle ostatniego wyniku. W przypadku macierzy 2×2 mamy bowiem $B = 1 \in \mathbb{C}^{1,1}$ i w konsekwencji wrażliwość wektorów własnych zależy tylko od różnicy $|\lambda_1-\lambda_2|$.

Poniższy przykład dla macierzy symetrycznej (!) pokazuje jak ważne dla wrażliwości wektorów własnych jest odseparowanie różnych wartości własnych.

Przyk?ad 1.3. Wartościami własnymi macierzy symetrycznej

$$A = \begin{bmatrix} 1+\varepsilon & 0 \\ 0 & 1 \end{bmatrix}$$

są $(1+\varepsilon)$ i 1, a odpowiadająca baza wektorów własnych to $\vec{e}_1$ i $\vec{e}_2$. Dla macierzy zaburzonej na poziomie ε (czyli różnicy wartości własnych),

$$A + E = \begin{bmatrix} 1+\varepsilon & 0 \\ 0 & 1 \end{bmatrix} + \varepsilon * \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix},$$

wartościami własnymi są $(1+\varepsilon)$ i $(1-\varepsilon)$, a bazę wektorów własnych tworzą $\vec{e}_1 + \vec{e}_2$ i $\vec{e}_1 - \vec{e}_2$. Baza wektorów własnych została obrócona o możliwie maksymalny kąt $\pi/4$.

1.3 Teoria zaburzeń dla macierzy symetrycznych

W tej części będziemy zakładać, że macierz jest rzeczywista i symetryczna,

$$A = A^T \in \mathbb{R}^{n,n}.$$

Ponieważ elementy $a_{i,j}$ i $a_{j,i}$ macierzy A są w praktycznych obliczeniach reprezentowane przez tą samą zmienną, zasadne jest założenie, że macierz zaburzona $A+E$ jest również symetryczna, $E = E^T \in \mathbb{R}^{n,n}$.

1.3.1 Wrażliwość wartości własnych

Przypomnijmy twierdzenie 1.1, które mówi, że dla macierzy symetrycznej istnieje baza ortonormalna Q jej wektorów własnych. Ponieważ dla macierzy ortogonanych mamy $\|Q\|_2 = 1$,

$$\mathrm{cond}_2(Q) = \|Q\|_2 \|Q^T\|_2 = 1.$$

To zaś, razem z twierdzeniem 1.2 implikuje, że każda watość własna μ macierzy sąsiedniej $A+E$ spełnia nierówność

$$\min_{1 \leqslant i \leqslant n} |\lambda_i - \mu| \leqslant \|E\|_2.$$

Możemy jednak pokazać dużo więcej. Zachodzi bowiem następujące twierdzenie Weyla.

Twierdzenie 1.3. *Niech $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n$ będą wartościami własnymi macierzy $A = A^T \in \mathbb{R}^{n,n}$, a $\mu_1 \geqslant \mu_2 \geqslant \cdots \geqslant \mu_n$ wartościami własnymi macierzy sąsiedniej $A+E$, gdzie $E = E^T \in \mathbb{R}^{n,n}$. Wtedy dla wszystkich $1 \leqslant k \leqslant n$ mamy*

$$|\lambda_k - \mu_k| \leqslant \|E\|_2.$$

Dowód twierdzenia opiera się na następującej pożytecznej nierówności.

Lemat 1.1. *Dla dowolnej rzeczywistej macierzy symetrycznej A mamy*

$$\max_{\dim(\mathcal{V})=k} \min_{\vec{0} \neq \vec{x} \in \mathcal{V}} \frac{\vec{x}^T * A * \vec{x}}{\vec{x}^T * \vec{x}} = \lambda_k = \min_{\dim(\mathcal{W})=n-k+1} \max_{\vec{0} \neq \vec{y} \in \mathcal{W}} \frac{\vec{x}^T * A * \vec{x}}{\vec{x}^T * \vec{x}}, \tag{1.5}$$

przy czym odpowiednie maksimum i minimum osiągane są dla $\mathcal{V}^* = \mathrm{span}(\vec{v}_k, \vec{v}_{k+1}, \ldots, \vec{v}_n)$ oraz $\mathcal{W}^* = \mathrm{span}(\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_k)$.

Dowód. Niech $\mathcal{V}$ i $\mathcal{W}$ będą dowolnymi podprzestrzeniami o wskazanych wymiarach. Ponieważ suma wymiarów wynosi $n+1$ to istnieje wektor niezerowy $\vec{z} \in \mathcal{V} \cap \mathcal{W}$. W konsekwencji

$$\min_{\vec{0}\neq\vec{x}\in\mathcal{V}} \frac{\vec{x}^T * A * \vec{x}}{\vec{x}^T * \vec{x}} \leqslant \frac{\vec{z}^T * A * \vec{z}}{\vec{z}^T * \vec{z}} \leqslant \max_{\vec{0}\neq\vec{y}\in\mathcal{W}} \frac{\vec{x}^T * A * \vec{x}}{\vec{x}^T * \vec{x}}.$$

Biorąc maximum po $\mathcal{V}$ i minimum po $\mathcal{W}$ dostajemy, że w (1.5) lewa strona nie jest większa od prawej. Aby pokazać odwrotną nierówność, zauważmy, że dla $\mathcal{V}^*$ i $\mathcal{W}^*$ mamy odpowiednio

$$\min_{\vec{0}\neq\vec{x}\in\mathcal{V}^*} \frac{\vec{x}^T * A * \vec{x}}{\vec{x}^T * \vec{x}} \geqslant \min_{\vec{0}\neq\sum_{j\geqslant k} a_j * \vec{v}_j} \frac{\sum_{j\geqslant k} a_j^2 \lambda_j}{\sum_{j\geqslant k} a_j^2} = \lambda_k,$$

$$\max_{\vec{0}\neq\vec{y}\in\mathcal{W}^*} \frac{\vec{y}^T * A * \vec{y}}{\vec{y}^T * \vec{y}} \leqslant \max_{\vec{0}\neq\sum_{j\leqslant k} b_j * \vec{v}_j} \frac{\sum_{j\leqslant k} b_j^2 \lambda_j}{\sum_{j\leqslant k} b_j^2} = \lambda_k.$$

□

Dla dowodu twierdzenia 1.3 zastosujemy lemat 1.5 najpierw do macierzy $A+E$, a potem do macierzy $A = (A+E) - E$. Otrzymujemy odpowiednio

$$\begin{aligned} \mu_k &= \min_{\dim(\mathcal{W})=n-k+1} \max_{\vec{0}\neq\vec{y}\in\mathcal{W}} \frac{\vec{x}^T * (A+E) * \vec{x}}{\vec{x}^T * \vec{x}} \\ &\leqslant \min_{\dim(\mathcal{W})=n-k+1} \max_{\vec{0}\neq\vec{y}\in\mathcal{W}} \frac{\vec{x}^T * A * \vec{x}}{\vec{x}^T * \vec{x}} + \|E\|_2 \\ &= \lambda_k + \|E\|_2, \end{aligned}$$

oraz podobnie nierówność odwrotną $\lambda_k \leqslant \mu_k + \|E\|_2$, czyli ostatecznie

$$|\lambda_k - \mu_k| \leqslant \|E\|_2.$$

Zanotujmy jeszcze, że wielkość

$$\frac{\vec{x}^T * A * \vec{x}}{\vec{x}^T * \vec{x}}$$

znana jest pod nazwą *ilorazu Rayleigh'a*.

1.3.2 Wrażliwość wektorów własnych

Niech $\mathcal{V}_k \subset \mathbb{R}^n$ będzie podprzestrzenią własną odpowiadającą wartości własnej λ_k macierzy A, a $\vec{x}_k$ wektorem własnym odpowiadającym wartości własnej μ_k macierzy sąsiedniej $A+E$. Niech dalej θ_k będzie kątem pomiędzy $\vec{x}_k$ i podprzestrzenią $\mathcal{V}_k$. Wtedy

$$\sin\theta_k \lessapprox \frac{\|E\|_2}{\min_{\lambda_j\neq\lambda_k} |\lambda_k - \lambda_j|} \qquad (\|E\|_2 \to 0^+).$$

Rzeczywiście, to wynika bezpośrednio z twierdzenia 1.3, nierówności (1.4) oraz faktu, że macierz B w (1.4) jest identycznością. Pokażemy teraz nierówność dokładną.

Twierdzenie 1.4.

$$\frac{1}{2}\sin 2\theta_k \leqslant \frac{\|E\|_2}{\min_{\lambda_j\neq\lambda_k} |\lambda_k - \lambda_j|}.$$

Dowód. Dla ustalenia uwagi załóżmy, że $k = 1$ oraz odpowiednia podprzestrzeń własna $\mathcal{V}_1 = \mathrm{span}(\vec{v}_1, \ldots, \vec{v}_s)$. Przedstawmy wektor własny $\vec{x}_1$, $\|\vec{x}\|_2 = 1$, w jednoznaczny sposób w postaci

$$\vec{x}_1 \;=\; \vec{v} + \vec{d},$$

gdzie $\vec{v} \in \mathcal{V}_1$ i $\vec{d} \in \mathcal{V}_1^{\perp}$,

$$\vec{d} \;=\; \sum_{j=s+1}^{n} b_j * \vec{v}_j.$$

Możemy też założyć, bez straty ogólności, że $\vec{d} \neq \vec{0}$ i $\vec{x}_1 \notin \mathcal{V}_1^{\perp}$, tzn. $0 < \theta < \pi/2$.
Przekształcając równanie $(A + E) * (\vec{v} + \vec{d}) = \mu_1 * (\vec{v} + \vec{d})$ dostajemy

$$\vec{z} \;:=\; (A - \lambda_1 * I) * \vec{d} \;=\; ((\mu_1 - \lambda_1) * I - E) * (\vec{v} + \vec{d}).$$

Ponieważ każdy z wektorów $\vec{v}_i$ dla $1 \leqslant i \leqslant s$ należy do jądra macierzy symetrycznej $A - \lambda_1 * I$ to wektor $(A - \lambda_1 * I) * \vec{d}$ jest ortogonalny do $\mathcal{V}_1$ i tym samym możemy zapisać

$$\vec{z} \;=\; \sum_{j=s+1}^{n} a_j * \vec{v}_j.$$

Mamy dalej

$$(A - \lambda_1 * I) * \vec{d} \;=\; (A - \lambda_1 * I) * \left(\sum_{j=s+1}^{n} b_j * \vec{v}_j \right) \;=\; \sum_{j=s+1}^{n} (\lambda_j - \lambda_1) b_j * \vec{v}_j,$$

a stąd $a_j = (\lambda_j - \lambda_1) b_j$ i

$$\|\vec{d}\|_2 \;=\; \left(\sum_{j=s+1}^{n} \frac{a_j^2}{(\lambda_j - \lambda_1)^2} \right)^{1/2} \leqslant \frac{\|\vec{z}\|_2}{\min_{s+1 \leqslant j \leqslant n} |\lambda_j - \lambda_1|}.$$

Z kolei z równości $\vec{v}^T * ((\mu_1 - \lambda_1) * I - E) * (\vec{v} + \vec{d}) = 0$ dostajemy

$$(\mu_1 - \lambda_1) \, \|\vec{v}\|_2^2 \;=\; \vec{v}^T * E * (\vec{v} + \vec{d}),$$

skąd

$$\vec{z} \;=\; \frac{1}{\|\vec{v}\|_2^2} (\vec{v} + \vec{d}) * \vec{v}^T * E * (\vec{v} + \vec{d}) - E * (\vec{v} + \vec{d}) \;=\; \left(\frac{1}{\|\vec{v}\|_2^2} (\vec{v} + \vec{d}) * \vec{v}^T - I \right) * E * (\vec{v} + \vec{d}).$$

Gdybyśmy teraz wiedzieli, że

$$\left\| \frac{1}{\|\vec{v}\|_2^2} (\vec{v} + \vec{d}) * \vec{v}^T - I \right\|_2 \;=\; \frac{1}{\|\vec{v}\|_2} \tag{1.6}$$

to moglibyśmy napisać

$$\|\vec{z}\|_2 \;\leqslant\; \frac{\|\vec{v} + \vec{d}\|_2^2}{\|\vec{v}\|_2} \|E\|_2 \;=\; \frac{\|E\|_2}{\|\vec{v}\|_2}$$

i ostatecznie

$$\frac{1}{2} \sin 2\theta_1 \;=\; \sin\theta_1 \cos\theta_1 \;=\; \|\vec{d}\|_2 \|\vec{v}\|_2 \;\leqslant\; \frac{\|\vec{z}\|_2 \|\vec{v}\|_2}{\min_{s+1 \leqslant j \leqslant n} |\lambda_j - \lambda_1|} \;\leqslant\; \frac{\|E\|_2}{\min_{s+1 \leqslant j \leqslant n} |\lambda_j - \lambda_1|}.$$

Pozostaje więc do pokazania równość (1.6). W tym celu, weźmy $\vec{y} = \alpha * \vec{v}/\|\vec{v}\|_2 + \beta * \vec{h}$, gdzie $\|h\|_2 = 1$, $\vec{h} \perp \vec{v}$ i $\|\vec{y}\|_2 = (\alpha^2 + \beta^2)^{1/2} = 1$. Wtedy

$$\frac{(\vec{v} + \vec{d}) * \vec{v}^T}{\|\vec{v}\|_2^2} * \vec{y} - \vec{y} = \frac{\alpha * \vec{d}}{\|\vec{v}\|_2} - \beta * \vec{h}.$$

Dla danych α i β, wektor po prawej stronie ma największą normę gdy $\vec{h} = \pm\vec{d}/\|\vec{d}\|_2$, przy czym bierzemy plus wtedy i tylko wtedy gdy α i β mają różne znaki. Dlatego poszukiwana norma wynosi

$$\max_{\alpha^2+\beta^2=1} \alpha \cdot \frac{\|\vec{d}\|_2}{\|\vec{v}\|_2} + \beta.$$

Zamieniając zmienne na $\alpha = \cos\phi$, $\beta = \sin\phi$ łatwo dostajemy, że optymalne $\phi \in (0, \pi/2)$ spełnia $\sin\phi = \|\vec{d}\|_2$, $\cos\phi = \|\vec{v}\|_2$, a stąd dostajemy wynik

$$\frac{\|\vec{d}\|_2}{\|\vec{v}\|_2}\|\vec{d}\|_2 + \|\vec{v}\|_2 = \frac{\|\vec{d}\|_2^2 + \|\vec{v}\|_2^2}{\vec{v}\|_2} = \frac{1}{\|\vec{v}\|_2}.$$

□

Pokazaliśmy, że wektory własne są mało wrażliwe na zaburzenia macierzy o ile wartości własne są odseparowane od siebie na poziomie dużo większym niż $\|E\|_2$. W szczególności, jeśli dodatkowo $\lambda_i \neq \lambda_j$, $i \neq j$, to mamy jednolite oszacowanie

$$\frac{1}{2}\sin 2\theta_k \leqslant \frac{\|E\|_2}{\min_{i\neq j}|\lambda_i - \lambda_j|}.$$

Przypomnijmy jeszcze przykład 1.3 pokazujący, że warunek odseparowania wartości własnych jest konieczny.

Rozdział 2

Zagadnienie własne II

Kolejne dwa wykłady poświęcimy konkretnym metodom numerycznym rozwiązywania zagadnienia własnego.

2.1 Uwagi wstępne

Ponieważ wartości własne macierzy A są zerami jej wielomianu charakterystycznego, narzucająca się metoda poszukiwania wartości własnych polegałaby na wyliczeniu współczynników tego wielomianu, na przykład w bazie potęgowej, a następnie na zastosowaniu jednej ze znanych metod znajdowania zer wielomianu. Dla wielomianów o współczynnikach rzeczywistych mogłaby to być np. metoda bisekcji, metoda Newtona (siecznych), albo pewna kombinacja obu method. Należy jednak przestrzec przed dokładnie takim postępowaniem. Rzecz w tym, że zera wielomianu mogą być bardzo wrażliwe na zaburzenia jego współczynnikówi. Dobrze ilustruje to następujący przykład.

Przyk?ad 2.1. Załóżmy, że A jest macierzą symetryczną formatu 20×20 o wartościach własnych $1, 2, \ldots, 20$. Zapisując wielomian charakterystyczny w postaci potęgowej mamy

$$\det(A - \lambda * I) \;=\; \prod_{k=1}^{20} (\lambda_k - k) \;=\; \sum_{k=0}^{20} w_k \lambda^k.$$

Okazuje się, że zaburzenie współczynnika w_{19} mogą powodować 10^{10} razy większe zaburzenie wartości własnej $\lambda_{16} = 16$.

Nie znaczy to oczywiście, że metody bazujące na obliczaniu wielomianu charakterystycznego, czy jego pochodnych trzeba z gruntu odrzucić. Trzeba tylko pamiętać, aby przy obliczeniach korzystać bezpośrednio ze współczynników $a_{i,j}$ macierzy A, ponieważ, jak wiemy z poprzedniego rozdziału, zadanie jest ze względu na te współczynniki dobrze uwarunkowane.

Z drugiej strony zauważmy, że policzenie wyznacznika macierzy pełnej wprost z definicji jest raczej kosztowne. Dlatego w praktyce metody wyznacznikowe są zwykle poprzedzone prekomputingiem polegającym na sprowadzeniu macierzy przez podobieństwa ortogonalne do prostszej postaci, z której wyznacznik można już obliczyć dużo tańszym kosztem niż dla macierzy pełnej. Tego typu precomputing ma również zastosowanie w innych metodach, w którym np. trzeba wykonywać mnożenie macierzy przez wektor.

Jedną z takich wygodnych postaci macierzy jest postać Hessenberga, a dla macierzy hermitowskich postać trójdiagonalna.

2.1.1 Sprowadzanie macierzy do postaci Hessenberga

Macierz $A = (a_{i,j}) \in \mathbb{C}^{n,n}$ jest postaci *Hessenberga* ("prawie" trójkątną górną) jeśli wszystkie wyrazy $a_{i,j}$ dla $i \geqslant j+2$ są zerami, tzn.

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n-1} & a_{2,n} \\ 0 & a_{3,2} & \cdots & a_{3,n-1} & a_{3,n} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & a_{n,n-1} & a_{n,n} \end{bmatrix}.$$

Oczywiście, jeśli macierz jest hermitowska, $A = A^H$, to postać Hessenberga jest równoważna postaci trójdiagonalnej

$$A = \begin{bmatrix} c_1 & b_2 & 0 & \cdots & 0 \\ \overline{b}_2 & c_2 & b_3 & \cdots & 0 \\ 0 & \overline{b}_3 & c_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \cdots & c_n \end{bmatrix}. \tag{2.1}$$

Aby daną macierz A sprowadzić do postaci Hessenberga przez podobieństwa ortogonalne (a więc nie zmieniając wartości własnych), możemy zastosować znane nam odbicia Householdera. Przypomnijmy, że dla dowolnego niezerowego wektora $\vec{a} \in \mathbb{C}^n$ istnieje macierz ortogonalna (odbicie Householdera) $P = P^H = P^{-1} \in \mathbb{C}^{n,n}$ postaci $P = I - 2 * \vec{u} * \vec{u}^H$, gdzie $\|\vec{u}\|_2 = 1$, która przekształca $\vec{a}$ na kierunek pierwszego wersora, tzn. $P * \vec{a} = \ell * \vec{e}_1$, $|\ell| = \|\vec{a}\|_2$. (Odbicia Householdera zostały dokładnie przedstawione na wykładzie *Analiza Numeryczna I.*)

Niech $A = (a_{i,j}) \in \mathbb{C}^{n,n}$ będzie dowolną macierzą. Algorytm konstruuje najpierw odbicie $P_1 \in \mathbb{C}^{n-1,n-1}$ dla wektora $[a_{2,1}, a_{3,1}, \ldots, a_{n,1}]^T$, a następnie biorąc macierz

$$\widehat{P}_1 = \begin{bmatrix} 1 & \vec{0} \\ \vec{0}^T & P_1 \end{bmatrix} \in \mathbb{C}^{n,n}$$

oblicza $A^{(1)} = \widehat{P}_1 * A * \widehat{P}_1^H$. Łatwo zobaczyć, że wtedy elementy $(i,1)$ dla $i = 3, 4, \ldots, n$ w macierzy $A^{(1)}$ są równe zeru.

Postępując indukcyjnie załóżmy, że dostaliśmy już macierz $A^{(k)} = (a_{i,j}^{(k)})$, $1 \leqslant k \leqslant n-3$, w której elementy $a_{i,j}^{(k)}$ dla $i \geqslant j+2$, $1 \leqslant j \leqslant k$, są wyzerowane. W kroku $(k+1)$-szym algorytm konstruuje odbicie $P_{k+1} \in \mathbb{C}^{n-k,n-k}$ dla wektora $[a_{k+2,k+1}^{(k)}, \ldots, a_{n,k+1}^{(k)}]^T$, a następnie biorąc macierz

$$\widehat{P}_{k+1} = \begin{bmatrix} I_k & 0 \\ 0^T & P_{k+1} \end{bmatrix} \in \mathbb{C}^{n,n}$$

oblicza $A^{(k+1)} = \widehat{P}_{k+1} * A^{(k)} * \widehat{P}_{k+1}^H$. Wtedy wszystkie elementy $a_{i,j}^{(k+1)}$ dla $i \geqslant j+2$, $1 \leqslant j \leqslant k+1$, są zerami. Po $(n-2)$ krokach otrzymana macierz $\widetilde{A} = A^{(n-1)}$ jest postaci Hessenberga.

Jasne jest, że jeśli $A = A^H$ to opisany algorytm prowadzi do macierzy trójdiagonalnej, $\widetilde{A} = \widetilde{A}^H$. (Obliczenia można wtedy w każdym kroku wykonywać jedynie na głównej diagonali i pod nią.)

2.2 Metoda Hymana

Przy pomocy metody *Hymana* możemy w zręczny sposób obliczyć wartości i pochodne wielomianu charakterystycznego $\det(A - \lambda * I)$ dla macierzy Hessenberga $A = (a_{i,j}) \in \mathbb{C}^{n,n}$. Bez

zmniejszenia ogólności będziemy zakładać, że wszystkie elementy $a_{i+1,i} \neq 0$. W przeciwnym przypadku wyznacznik można obliczyć jako iloczyn wyznaczników macierzy Hessenberga niższych rzędów.

Metoda Hymana polega na dodaniu do pierwszego wiersza macierzy $A - \lambda * I$ wiersza i-tego pomnożonego przez pewien współczynnik $q_i = q_i(\lambda)$, dla $i = 2, 3, \ldots, n$ tak, aby wyzerować elementy $(1, i)$ dla $i = 1, 2, \ldots, n-1$. Ponieważ

$$A - \lambda * I = \begin{bmatrix} a_{1,1} - \lambda & a_{1,2} & \cdots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} - \lambda & \cdots & a_{2,n-1} & a_{2,n} \\ 0 & a_{3,2} & \cdots & a_{3,n-1} & a_{3,n} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & a_{n,n-1} & a_{n,n} - \lambda \end{bmatrix},$$

mamy następujące równania na q_i:

$$\begin{aligned} (a_{1,1} - \lambda) + a_{1,2}q_2 &= 0, \\ a_{1,1} + a_{2,1}q_2 + \cdots + a_{i-1,i}q_{i-1} + (a_{i,i} - \lambda)q_i + a_{i+1,i}q_{i+1} &= 0, \end{aligned} \tag{2.2}$$

dla $i = 2, 3, \ldots, n-1$. Stąd, definiując dodatkowo $q_1 = 1$, dostajemy następujące równania rekurencyjne:

$$\begin{aligned} q_2(\lambda) &= \frac{-(a_{1,1} - \lambda)}{a_{2,1}}, \\ q_{i+1}(\lambda) &= \frac{-\left(a_{1,i}q_1 + \cdots + a_{i-1,i}q_{i-1} + (a_{i,i} - \lambda)q_i\right)}{a_{i+1,i}}. \end{aligned}$$

Po opisanym przekształceniu macierzy $A - \lambda * I$ zmieni się jedynie jej pierwszy wiersz; wyrazy $a_{1,i}$, $1 \leqslant i \leqslant n-1$, zostaną wyzerowane, a $a_{1,n}$ zostanie przekształcony do

$$\widehat{a}_{1,n} = a_{1,n}q_1 + \cdots + a_{n-1,n}q_{n-1} + (a_{n,n} - \lambda)q_n.$$

Rozwijając wyznacznik względem (przekształconego) pierwszego wiersza otrzymujemy

$$\det(A - \lambda * I) = (-1)^{n+1} a_{2,1}a_{3,2}\cdots a_{n,n-1}\left(a_{1,n}q_1 + \cdots + a_{n-1,n}q_{n-1} + (a_{n,n} - \lambda)q_n\right),$$

a stąd zera wielomianu charakterystycznego są równe zerom wielomianu

$$q_{n+1}(\lambda) = -\left(a_{1,n}q_1 + \cdots + a_{n-1,n}q_{n-1} + (a_{n,n} - \lambda)q_n\right).$$

Oczywiście, wartości $q_{n+1}(\lambda)$ można obliczać stosując wzory rekurencyjne (2.2) przedłużając je o $i = n$, przy dodatkowym formalnym podstawieniu $a_{n+1,n} = 1$.

Aby móc zastowoać netodę Newtona (stycznych) do znalezienia zer wielomianu q_{n+1} potrzebujemy również wiedzieć jak obliczać jego pochodne. To też nie jest problem, bo rekurencyjne wzory na pochodne można uzyskać po prostu różniczkując wzory (2.2). Dodajmy, że koszt obliczenia wartości $q_{n+1}(\lambda)$ i $q'_{n+1}(\lambda)$ jest proporjonalny do n^2, co jest istotnym zyskiem w porównaniu z kosztem n^3 obliczania wyznacznika pełnej macierzy za pomocą faktoryzacji metodą eliminacji Gaussa.

Formuły na obliczanie wyznacznika $\det(A - \lambda * I)$ i jego pochodnych uproszczają się jeszcze bardziej gdy macierz A jest dodatkowo hermitowska, czyli jest macierzą trójdiagonalną postaci

(2.1). Wtedy, stosując zwykłe rozumowanie rekurencyjne, łatwo się przekonać, że kolejne minory główne (czyli wyznaczniki macierzy kątowych) spełniają zależności

$$\begin{aligned} p_0(\lambda) &= 1, \qquad p_1(\lambda) = c_1 - \lambda, \\ p_i(\lambda) &= (c_i - \lambda)p_{i-1}(\lambda) - |b_i|^2 p_{i-2}(\lambda) \qquad \text{dla} \quad i = 2, 3, \ldots, n. \end{aligned}$$

Różniczkując otrzymane wzory otrzymujemy formuły na pochodne kolejnych minorów po λ. Wartości wielomianu $\det(A - \lambda * I) = p_n(\lambda)$ oraz jego pochodnych można więc obliczać kosztem liniowym w n.

Sprawę konkretnej implementacji metod iteracyjnych bisekcji i/lub Newtona do wyznaczenia zer wielomianu $\det(A - \lambda * I)$ tutaj pomijamy. Ograniczymy się jedynie do stwierdzenia, że nie jest to rzecz całkiem trywialna.

2.3 Metoda potęgowa

2.3.1 Definicja metody

Metoda potęgowa zdefiniowana jest w następujący prosty sposób. Rozpoczynając od dowolnego wektora $\vec{x}_0 \in \mathbb{C}^n$ o normie $\|\vec{x}_0\|_2 = 1$ obliczamy kolejno dla $k = 0, 1, 2, \ldots$

$$\vec{y}_k := A * \vec{x}_k, \qquad \vec{x}_{k+1} := \frac{\vec{y}_k}{\|\vec{y}_k\|_2}.$$

Równoważnie możemy napisać

$$\vec{x}_k = \frac{A^k * \vec{x}_0}{\|A^k * \vec{x}_0\|_2}.$$

Wektory $\vec{x}_k$ stanowią kolejne przybliżenia wektora własnego. Odpowiadającą mu wartość własną przybliżamy wzorem

$$\eta_k := \vec{x}_k^H * A * \vec{x}_k = \vec{x}_k^H * \vec{y}_k.$$

2.3.2 Analiza zbieżności

Analizę metody potęgowej przeprowadzimy przy założeniu, że macierz A jest diagonalizowalna. Oznaczmy przez μ_i, $1 \leqslant i \leqslant s$, różne wartości własne macierzy A, a przez $\mathcal{V}_i$ odpowiadające im podprzestrzenie własne,

$$\mathcal{V}_1 \oplus \mathcal{V}_2 \oplus \cdots \oplus \mathcal{V}_s = \mathbb{C}^n. \tag{2.3}$$

Założymy, że μ_1 jest dominującą wartością własną oraz

$$|\mu_1| > |\mu_2| > \cdots > |\mu_s|.$$

Przedstawmy $\vec{x}_0$ jednoznacznie w postaci

$$\vec{x}_0 = \sum_{j=1}^{s} \vec{v}_j$$

gdzie $v_j \in \mathcal{V}_j$. Dla uproszczenia, będziemy zakładać, że każda ze składowych

$$\vec{v}_j \neq \vec{0}.$$

Podkreślmy, że nie jest to założenie ograniczające, bo w przeciwnym przypadku składowe zerowe po prostu ignorujemy w poniższej analizie zbieżności. Poza tym, jeśli wektor początkowy jest wybrany losowo, to teoretyczne prawdopodobieństwo takiego zdarzenia jest zerowe. Co więcej, nawet jeśli jedna ze składowych znika to wskutek błędów zaokrągleń w procesie obliczeniowym składowa ta w wektorze $\vec{x}_1$ będzie z pewnością niezerowa. (Mamy tu do czynienia z ciekawym zjawiskiem, kiedy błędy zaokrągleń pomagają!)

Prosty rachunek pokazuje, że

$$A^k * \vec{x}_0 \;=\; \sum_{j=1}^{s} A^k * \vec{v}_j \;=\; \sum_{j=1}^{s} \mu_j^k * \vec{v}_j \;=\; \mu_1^k \left(\vec{v}_1 + \sum_{j=2}^{s} \left(\frac{\mu_j}{\mu_1} \right)^k * \vec{v}_j \right).$$

Ponieważ każdy z ilorazów $\mu_j/\mu_1 < 1$ to $\vec{x}_k$ zbiega do wektora własnego $\vec{v}_1/\|\vec{v}_1\|_2$, a η_k do dominującej wartości własnej μ_1. Przyjrzyjmy się od czego zależy szybkość zbieżności.

Odległość $\mathrm{dist}(\vec{x}_k, \mathcal{V}_1)$ wektora $\vec{x}_k$ od podprzestrzeni $\mathcal{V}_1$ można oszacować z góry przez odległość $\vec{x}_k$ od $\mathrm{span}(\vec{v}_1)$, która z kolei jest równa długości rzutu $P_1\vec{x}_k$ tego wektora na podprzestrzeń ortogonalną do $\vec{v}_1$,

$$P_1\vec{x}_k \;=\; \vec{x}_k \;-\; \frac{\vec{v}_1^{\,H} * \vec{x}_k}{\|\vec{v}_1\|_2^2} * \vec{v}_1.$$

Oznaczając

$$\beta_k \;:=\; \left\| \vec{v}_1 + \sum_{j=2}^{s} \left(\frac{\mu_j}{\mu_1} \right)^k * \vec{v}_j \right\|_2$$

mamy $\lim_{k\to\infty} \beta_k = \|\vec{v}_1\|_2$ oraz

$$\begin{aligned} P_1\vec{x}_k \;&=\; \frac{1}{\beta_k} \left(\left(\vec{v}_1 + \sum_{j=2}^{s} \left(\frac{\mu_j}{\mu_1} \right)^k * \vec{v}_j \right) - \left(\|\vec{v}_1\|_2^2 + \sum_{j=2}^{s} \left(\frac{\mu_j}{\mu_1} \right)^k * \vec{v}_1^{\,H} * \vec{v}_j \right) * \frac{\vec{v}_1}{\|\vec{v}_1\|_2^2} \right) \\ &=\; \frac{1}{\beta_k} \left(\sum_{j=2}^{s} \left(\frac{\mu_j}{\mu_1} \right)^k * \left(\vec{v}_j - \frac{\vec{v}_1^{\,H} * \vec{v}_j}{\|\vec{v}_1\|_2^2} * \vec{v}_1 \right) \right) \\ &\approx\; \left(\frac{\mu_j}{\mu_1} \right)^k \frac{1}{\|\vec{v}_1\|_2} \left(\vec{v}_2 - \frac{\vec{v}_1^{\,H} * \vec{v}_2}{\|\vec{v}_1\|_2^2} * \vec{v}_1 \right) \qquad (k \to \infty). \end{aligned}$$

Biorąc normę i stosując nierówność trójkąta dostajemy

$$\mathrm{dist}(\vec{x}_k, \mathcal{V}_1) \;=\; \|P_1\vec{x}_k\|_2 \;\lessapprox\; 2 \cdot \rho_1^k \cdot \frac{\|\vec{v}_2\|_2}{\|\vec{v}_1\|_2} \qquad (k \to \infty),$$

gdzie

$$\rho_1 \;:=\; \frac{|\mu_2|}{|\mu_1|} \;<\; 1.$$

Zbieżność jest więc liniowa z ilorazem ρ_1.

Zobaczmy teraz jak bardzo η_k różni się od μ_1. Mamy

$$\begin{aligned} \eta_k \;&=\; \vec{x}_k^{\,H} * A * \vec{x}_k \;=\; \frac{\mu_1}{\beta^2} \left(\vec{v}_1^{\,H} + \sum_{j=2}^{s} \left(\frac{\mu_j}{\mu_1} \right)^k * \vec{v}_j^{\,H} \right) * \left(\vec{v}_1 + \sum_{j=2}^{s} \left(\frac{\mu_j}{\mu_1} \right)^{k+1} * \vec{v}_1^{\,H} * \vec{v}_j \right) \\ &=\; \frac{\mu_1}{\beta^2} \left(\|\vec{v}_1\|_2^2 + \sum_{j=2}^{2} \left(\frac{\mu_j}{\mu_1} \right)^k \left(\vec{v}_j^{\,H} * \vec{v}_1 + \frac{\mu_j}{\mu_1} * \vec{v}_1^{\,H} * \vec{v}_j \right) + \sum_{i,j=2}^{s} \left(\frac{\mu_i^k \mu_j^{k+1}}{\mu_1^{2k+1}} \right) * \vec{v}_i^{\,H} * \vec{v}_j \right). \end{aligned}$$

Stąd wynika, że błąd $|\mu_1 - \eta_k|$ zależy od iloczynów skalarnych $\vec{v}_j^H * \vec{v}_1$ oraz od stosunków

$$\rho_j := \frac{|\mu_{j+1}|}{|\mu_1|}, \qquad 1 \leqslant j \leqslant s-1.$$

Dokładniej, niech

$$\ell := \min\left\{2 \leqslant j \leqslant s : \ \vec{v}_j^H * \vec{v}_1 \neq 0\right\}$$

albo $\ell = s+1$ jeśli powyższy zbiór jest pusty. Jeśli teraz $\ell = s+1$ albo mamy jednocześnie $2 \leqslant \ell \leqslant s$ i $\rho_\ell < \rho_1^2$ to

$$|\mu_1 - \eta_k| \lessapprox |\mu_1| \cdot \rho_1^{2k+1} \cdot \frac{\|\vec{v}_2\|_2}{\|\vec{v}_1\|_2}$$

i zbieżność jest liniowa z ilorazem ρ_1^2. Jeśli zaś $2 \leqslant \ell \leqslant s$ i $\rho_\ell \geqslant \rho_1^2$ to

$$|\mu_1 - \eta_k| \lessapprox |\mu_1| \cdot \rho_\ell^k \cdot \frac{\|\vec{v}_2\|_2}{\|\vec{v}_1\|_2}$$

i zbieżność jest liniowa z ilorazem ρ_ℓ. W szczególności, jeśli $\vec{v}_2^H * \vec{v}_1 \neq 0$ to zbieżość jest z ilorazem ρ_1.

Dla macierzy rzeczywistych i symetrycznych możemy stosunkowo łatwo pokazać dokładne nierówności na błąd metody potęgowej.

Twierdzenie 2.1. *Załóżmy, że macierz $A = A^T \in \mathbb{R}^{n,n}$. Niech γ_k będzie kątem pomiędzy wektorem k-tego przybliżenia $\vec{x}_k$ otrzymanego metodą potęgową i podprzestrzenią własną $\mathcal{V}_1$ odpowiadającą dominującej wartości własnej μ_1. Wtedy*

$$\operatorname{tg}\gamma_k \leqslant \rho_1 \cdot \operatorname{tg}\gamma_{k-1}$$

oraz

$$|\mu_1 - \eta_k| \leqslant \max_{2\leqslant j\leqslant s} |\mu_1 - \mu_j| \cdot \sin^2\gamma_k.$$

Dowód. Ponieważ macierz jest symetryczna, podprzestrzenie własne $\mathcal{V}_j$ zdefiniowane w (2.3) są parami ortogonalne. Dlatego

$$\vec{x}_k = \frac{A^k * \vec{x}_0}{\|A^k * \vec{x}_0\|_2} = \frac{\vec{v}_1 + \sum_{j=2}^s \left|\frac{\mu_j}{\mu_1}\right|^k * \vec{v}_j}{\sqrt{\|\vec{v}_1\|_2^2 + \sum_{j=2}^s \left|\frac{\mu_j}{\mu_1}\right|^{2k} \|\vec{v}_j\|_2^2}}.$$

Tangens kąta γ_k jest równy stosunkowi długości składowej wektora $\vec{x}_k$ w podprzestrzeni $\mathcal{V}_2 \oplus \cdots \oplus \mathcal{V}_s$ do długości składowej tego wektora w podprzestrzeni $\mathcal{V}_1$. Mamy więc

$$\operatorname{tg}\gamma_k = \left(\sum_{j=2}^s \left|\frac{\mu_j}{\mu_1}\right|^{2k} \frac{\|\vec{v}_j\|_2^2}{\|\vec{v}_1\|_2^2}\right)^{1/2} \leqslant \max_{2\leqslant j\leqslant s}\left|\frac{\mu_j}{\mu_1}\right| \cdot \operatorname{tg}\gamma_{k-1}.$$

Aby pokazać pozostałą część twierdzenia, przedstawmy $\vec{x}_k$ w postaci $\vec{x}_k = \sum_{j=1}^s \vec{z}_j$, gdzie $\vec{z}_j \in \mathcal{V}_j$. Wtedy $\sum_{j=1}^s \|\vec{z}_j\|_2^2 = \|\vec{x}_k\|_2^2 = 1$ oraz

$$\vec{x}_k^T * A * \vec{x}_k = \sum_{j=1}^s \mu_j \|\vec{z}_j\|_2^2 = \mu_1 + \sum_{j=2}^s (\mu_j - \mu_1)\|\vec{z}_j\|_2^2,$$

a stąd

$$|\mu_1 - \eta_k| \leqslant \max_{2\leqslant j\leqslant s} |\mu_1 - \mu_j| \cdot \sum_{j=2}^s \|\vec{z}_j\|_2^2 = \max_{2\leqslant j\leqslant s} |\mu_1 - \mu_j| \cdot \sin^2\gamma_k.$$

□

Przypomnijmy, że $\sin\gamma_k \leqslant \operatorname{tg}\gamma_k$ przy czym mamy asymptotyczną równość gdy $\gamma_k \to 0$. Z twierdzenia 2.1 wynika więc, że

$$\operatorname{tg}\gamma_k \leqslant \rho_1^k \cdot \operatorname{tg}\gamma_0, \qquad \text{oraz} \qquad |\mu_1 - \eta_k| \leqslant \rho_1^{2k} \cdot \max_{2\leqslant i\leqslant s} |\mu_1 - \mu_i| \cdot \operatorname{tg}\gamma_0.$$

Metoda potęgowa nie opłaca się gdy wymiar n nie jest duży, albo macierz A jest pełna. Inaczej jest, gdy n jest "wielkie", np. rzędu co najmniej kilkuset, a macierz A jest rozrzedzona, tzn. ma jedynie proporcjonalnie do n elementów niezerowych. Z taką sytuacją mamy do czynienia, gdy np. macierz jest pięciodiagonalna. Wtedy istotne dla metody mnożenie $A * \vec{x}$ można wykonać kosztem proporcjonalnym do n i poświęcając tyle samo pamięci (wyrazy zerowe można zignorować).

2.4 Modyfikacje metody potęgowej

Metoda potęgowa pozwala wyznaczyć największą co do modułu wartość własną i odpowiadający jej wektor własny. Naturalne jest teraz pytanie o to jak postępować, aby znaleźć inne pary własne.

Jeśli macierz jest hermitowska, $A = A^H$, oraz znaleźliśmy wektor własny $\vec{v}_1$ odpowiadający wartości własnej μ_1, to możemy ponowić proces metody potęgowej startując z wektora $\vec{x}_0$ prostopadłego do $\vec{v}_1$, tzn.

$$\vec{x}_0 := \frac{\vec{v}}{\|\vec{v}\|_2}, \qquad \vec{v} := \vec{w} - \vec{v}_1 * (\vec{v}_1^H * \vec{w}),$$

gdzie $\vec{w} \neq \vec{0}$ jest wybrany losowo. Przy idealnej realizacji procesu konstruowany ciąg $\{\vec{x}_k\}$ należałby do podprzestrzeni prostopadłej do $\vec{v}_1$. W obecności błędów zaokrągleń należałoby to wymusić poprzez reortogonalizację,

$$\vec{y}_k := \vec{y}_k - \vec{v}_1 * (\vec{v}_1^H * \vec{y}_k),$$

wykonywaną np. co kilka kroków. Jeśli $\dim(\mathcal{V}_1) > 1$ to proces zbiega do wektora w $\mathcal{V}_1$. W ten sposób otrzymujemy dwa wektory ortogonalne w $\mathcal{V}_1$, a postępując podobnie dalej, bazę ortogonalną podprzestrzeni własnej $\mathcal{V}_1$. Jeśli zaś $\dim(\mathcal{V}_1) = 1$ to proces zbiega do wektora własnego w $\mathcal{V}_2$.

Inna metoda znajdowania pozostałych par własnych, zwana *odwrotną metodą potęgową* albo *metodą Wielandta*, polega na zastosowaniu (prostej) metody potęgowej do macierzy $(A-\sigma*I)^{-1}$. Dokładniej, startując z przybliżenia początkowego $\vec{x}_0$ o jednostkowej normie drugiej obliczamy kolejno dla $k = 0, 1, 2, \ldots$

$$\vec{y}_k := (A - \sigma * I)^{-1} * \vec{x}_k, \qquad \vec{x}_{k+1} := \frac{\vec{y}_k}{\|\vec{y}_k\|_2},$$

oraz $\eta_k := \vec{x}_k^H * A * \vec{x}_k$.

Od razu nasuwają się dwie uwagi. Po pierwsze, iteracja odwrotna ma sens tylko wtedy gdy parametr σ jest tak dobrany, że macierz $A-\sigma*I$ jest nieosobliwa, co jest równoważne warunkowi $\sigma \neq \mu_i$ dla wszystkich i. Po drugie, w realizacji numerycznej, dla znalezienia wektora $\vec{y}_k$ nie odwracamy macierzy $(A - \sigma * I)$, ale rozwiązujemy układ równań $(A - \sigma * I) * \vec{y} = \vec{x}_k$, co czyni metodę tak samo kosztowną co iteracja prosta.

Analiza iteracji odwrotnych przebiega podobnie do analizy iteracji prostych dla macierzy $(A - \sigma * I)^{-1}$. Nietrudno zauważyć, że macierz ta ma te same wektory własne co A, a jej wartości własne wynoszą

$$\mu_i^{(\sigma)} = \frac{1}{\mu_i - \sigma}, \qquad 1 \leqslant i \leqslant s.$$

(To wynika bezpośrednio z równości $(A + \sigma * I) * \vec{v}_i = (\mu_i + \sigma) * \vec{v}_i$.) Dlatego iteracje odwrotne zbiegają do wektora własnego odpowiadającego wartości własnej μ_{i^*} takiej, że

$$|\mu_{i^*} - \sigma| = \min_{1 \leqslant i \leqslant s} |\mu_i - \sigma|,$$

przy czym szybkość zbieżności zależy teraz od

$$\rho_j^{(\sigma)} := \frac{|\mu_{i^*} - \sigma|}{\min_{i \neq i^*} |\mu_i - \sigma|}$$

(a nie od $\rho_j = |\mu_{j+1}|/|\mu_1|$, jak w iteracji prostej).

Niewątpliwą zalety odwrotnej metody potęgowej jest to, że zbiega do wartości własnej najbliższej σ, przy czym im σ bliższe μ_{i^*} tym lepiej. Metoda jest więc szczególnie efektywna w przypadku gdy znamy dobre przybliżenia wartości własnych macierzy A. Niestety, taka informacja nie zawsze jest dana wprost.

Rozdział 3

Zagadnienie własne III

Metodę potęgową można traktować jako iteracje podprzestrzeni jednowymiarowej startujące ze span($\vec{x}_0$). Naturalnym uogólnieniem tego procesu są iteracje podprzestrzeni o wyższych wymiarach. Właśnie iteracje podprzestrzeni są punktem wyjścia konstrukcji obecnie najpopularniejszego algorytmu QR, którego zadaniem jest obliczenie jednocześnie wszystkich wartości własnych.

Chociaż metody prezentowane w tym rozdziale mogą być stosowane w większej ogólności, dla uproszczenia będziemy zakładać, że macierz A jest nieosobliwa, rzeczywista i symetryczna, tzn. $A = A^T \in \mathbb{R}^{n,n}$ i $\det(A) \neq 0$. Przypomnijmy, że wtedy istnieje baza ortonormalna $\{\vec{\xi}_i\}_{i=1}^n$ wektorów własnych macierzy,

$$A * \vec{\xi}_i = \lambda_i * \vec{\xi}_i, \quad 1 \leqslant i \leqslant n.$$

Będziemy również zakładać, że wartości własne są uporządkowane tak, że

$$|\lambda_1| \geqslant |\lambda_2| \geqslant \cdots \geqslant |\lambda_n| > 0.$$

3.1 Iteracje podprzestrzeni

3.1.1 Algorytm ogólny

Niech $1 \leqslant p \leqslant n-1$ oraz $\mathcal{Y}_0 \subseteq \mathbb{R}^n$ będzie podprzestrzenią o wymiarze p. Dla $k = 1, 2, 3, \ldots$ rozpatrzmy następujące iteracje:

$$\mathcal{Y}_k := A(\mathcal{Y}_{k-1}) = \{\, A * \vec{x} : \vec{x} \in \mathcal{Y}_{k-1} \,\}.$$

Oczywiście, wobec nieosobliwości A wszystkie podprzestrzenie $\mathcal{Y}_k$ mają też wymiar p. Pokażemy, że przy pewnych niekrępujących założeniach ciąg podprzestrzeni $\mathcal{Y}_k$ zbiega do podprzestrzeni własnej

$$\mathcal{P}^{(p)} := \operatorname{span}(\vec{\xi}_1, \vec{\xi}_2, \ldots, \vec{\xi}_p),$$

przy czym przez "zbieżność" rozumiemy tu zbieżność do zera kąta pomiędzy tymi podprzestrzeniami.

Przypomnijmy, że kąt pomiędzy dwiema podprzestrzeniami $\mathcal{Y}, \mathcal{Z} \subseteq \mathbb{R}^n$ definiujemy jako

$$\angle(\mathcal{Y}_k, \mathcal{P}^{(p)}) := \max_{\vec{y} \in \mathcal{Y}_k} \min_{\vec{z} \in \mathcal{P}^{(p)}} \angle(\vec{y}, \vec{z}).$$

(Jako ćwiczenie można wykazać, że jeśli $\dim(\mathcal{Y}) = \dim(\mathcal{Z})$ to $\angle(\mathcal{Y}, \mathcal{Z}) = \angle(\mathcal{Z}, \mathcal{Y})$.)

Twierdzenie 3.1. *Niech*

$$\rho_p := \frac{|\lambda_{p+1}|}{|\lambda_p|} < 1 \quad \textit{oraz} \quad \gamma_k := \angle(\mathcal{Y}_k, \mathcal{P}^{(p)}), \quad k = 0, 1, 2, \ldots.$$

Jeśli $\gamma_0 < \pi/2$ *to*

$$\operatorname{tg}\gamma_k \leqslant \rho_p \cdot \operatorname{tg}\gamma_{k-1} \leqslant \rho_p^k \cdot \operatorname{tg}\gamma_0.$$

Dowód. Niech $\vec{y}$ będzie wektorem spełniającym $\angle(\vec{y}, \mathcal{P}^{(p)}) < \pi/2$. Wtedy

$$\vec{y} = \sum_{i=1}^{n} \vec{v}_i, \qquad \vec{v}_i \in \operatorname{span}(\vec{\xi}_i), \quad 1 \leqslant i \leqslant n,$$

gdzie nie wszystkie $\vec{v}_i$ dla $1 \leqslant i \leqslant p$, są są zerowe, oraz $A * \vec{y} = \sum_{i=1}^{n} \lambda_i \vec{v}_i$. Stąd

$$\operatorname{tg}\angle(A * \vec{y}, \mathcal{P}^{(p)}) = \sqrt{\frac{\sum_{i=p+1}^{n} |\lambda_i|^2 \|\vec{v}_i\|_2^2}{\sum_{i=1}^{p} |\lambda_i|^2 \|\vec{v}_i\|_2^2}} \leqslant \frac{|\lambda_{p+1}|}{|\lambda_p|} \cdot \sqrt{\frac{\sum_{i=p+1}^{n} \|\vec{v}_i\|_2^2}{\sum_{i=1}^{p} \|\vec{v}_i\|_2^2}} = \rho_p \cdot \operatorname{tg}\angle(\vec{y}, \mathcal{P}^{(p)}p). \quad (3.1)$$

Jeśli teraz $\gamma_{k+1} = \angle(\vec{y}_{k+1}, \mathcal{P}^{(p)})$ dla $\vec{y}_{k+1} \in \mathcal{Y}_{k+1}$ oraz $\vec{y}_{k+1} = A * \vec{y}_k$ to z (3.1) wynika, że

$$\operatorname{tg}\gamma_{k+1} \leqslant \rho_p \cdot \operatorname{tg}\angle(\vec{y}_k, \mathcal{P}^{(p)}) \leqslant \rho_p \cdot \operatorname{tg}\gamma_k,$$

co kończy dowód. □

3.1.2 Iteracje ortogonalne

W praktyce, iteracje podprzestrzeni realizujemy reprezentując kolejne $\mathcal{Y}_k$ przez ich bazy ortonormalne. Wtedy algorytm, zwany w tym przypadku również iteracjami ortogonalnymi, przybiera następującą postać. Wybieramy macierz kolumnowo-ortogonalną Z_0 formatu $n \times p$ jako macierz startową, a następnie dla $k = 1, 2, 3, \ldots$ iterujemy:

$$\text{(IO)} \quad \begin{cases} Y_k := A * Z_{k-1}, \\ Y_k := Z_k * R_k, \quad \text{(ortonormalizacja)} \end{cases}$$

przy czym w drugiej linii następuje ortonormalizacja macierzy Y_k realizowana poprzez rozkład tej macierzy na iloczyn macierzy kolumnowo-ortonormalnej Z_k formatu $n \times p$ i macierzy trójkątnej górnej R_k formatu $p \times p$. Dokonuje się tego znanym nam już algorytmem wykorzystującym odbicia Householdera.

Łatwo zauważyć, że jeśli przyjmiemy, iż wektory-kolumny macierzy Z_0 tworzą bazę ortonormalną podprzestrzeni $\mathcal{Y}_0$ to w kolejnych krokach wektory-kolumny Z_k tworzą bazę ortonormalną podprzestrzeni $\mathcal{Y}_k$. Rzeczywiście, ponieważ $Y_k = A * Z_{k-1}$ to kolumny Y_k stanowią bazę $\mathcal{Y}_k$. Kolumny te są z kolei liniową kombinacją kolumn Z_k, czyli rozpinają tą samą podprzestrzeń co kolumny Z_k.

Poczynimy teraz kluczową obserwację. Przypuśćmy, że wszystkie wartości własne λ_i są parami różne. Gdybyśmy rozpoczynali iteracje od macierzy $\overline{Y}_0$ składającej się jedynie z $\overline{p}$ początkowych kolumn macierzy Y_0, przy czym $1 \leqslant \overline{p} < p$, to otrzymane w procesie iteracyjnym macierze $\overline{Y}_k$ i $\overline{Z}_k$ (formatu $n \times \overline{p}$ i $\overline{p} \times \overline{p}$) byłyby odpowiednio "okrojonymi" macierzami Y_k i Z_k. Stąd, jeśli wszystkie wartości własne macierzy A mają różne moduły, to dla każdego takiego $\overline{p}$ podprzestrzeń $\mathcal{Y}_k^{(\overline{p})}$ rozpięta na początkowych $\overline{p}$ wektorach macierzy Z_k "zbiega" do podprzestrzeni własnej $\mathcal{P}^{(\overline{p})}$, przy czym

$$\operatorname{tg}\angle\left(\mathcal{Y}_k^{(\overline{p})}, \mathcal{P}^{(\overline{p})}\right) \leqslant \rho_{\overline{p}}^k \cdot \operatorname{tg}\angle\left(\mathcal{Y}_0^{(\overline{p})}, \mathcal{P}^{\overline{p}}\right).$$

Z powyższej obserwacji wynika, że gdybyśmy realizowali iteracje podprzestrzeni przyjmując $p = n$ i startując z macierzy identycznościowej $Z_0 := I$ formatu $n \times n$ to mielibyśmy zbieżność $\mathcal{Y}_k^{(p)}$ do podprzestrzeni własnych $\mathcal{P}^{(p)}$ dla wszystkich $p = 1, 2, \ldots, n$.
W przypadku gdy pracujemy na bazach ortogonalnych i startujemy z $Z_0 = I$, zbieżność tą można wyrazić również w następujący sposób. Oznaczmy

$$V = [\vec{\xi}_1, \vec{\xi}_2, \ldots, \vec{\xi}_n], \qquad Z_k = [\vec{z}_1^{(k)}, \vec{z}_2^{(k)}, \ldots, \vec{z}_n^{(k)}].$$

Lemat 3.1. *Niech B_k będzie macierzą przejścia od bazy $(\vec{\xi}_1, \ldots, \vec{\xi}_n)$ do $(\vec{z}_1^{(k)}, \ldots, \vec{z}_n^{(k)})$,*

$$Z_k = V * B_k.$$

Dla $1 \leqslant p \leqslant n-1$, niech

$$B_k = \begin{bmatrix} B_{1,1}^{(k)} & B_{1,2}^{(k)} \\ B_{2,1}^{(k)} & B_{2,2}^{(k)} \end{bmatrix},$$

gdzie $B_{1,1}^{(k)}$ jest podmacierzą formatu $p \times p$, a pozostałe podmacierze odpowiednio formatów $p \times (n-p)$, $(n-p) \times p$, $(n-p) \times (n-p)$. Jeśli $\rho_p = |\lambda_{p+1}|/|\lambda_p| < 1$ i $\gamma_0 < \pi/2$ to

$$\|B_{1,2}^{(k)}\|_2, \|B_{2,1}^{(k)}\|_2 \leqslant \rho_p^k \cdot \operatorname{tg} \gamma_0.$$

Dowód. Niech $Z_k = [Z_1^{(k)}, Z_2^{(k)}]$, $V = [V_1, V_2]$, gdzie $Z_1^{(k)}$ i V_1 są formatu $n \times p$. Wtedy

$$Z_1^{(k)} = V_1 * B_{1,2}^{(k)} + V_2 * B_{2,1}^{(k)}.$$

Mnożąc to równanie z lewej strony przez V_2^T dostajemy $B_{2,1}^{(k)} = V_2^T * Z_1^{(k)}$. Stąd

$$\|B_{2,1}^{(k)}\|_2 = \|V_2^T * Z_1^{(k)}\|_2 = \sup_{\|\vec{x}\|_2 = 1} \|V_2^T * (Z_1^{(k)} * \vec{x})\|_2.$$

Oznaczając $\vec{y} = Z_1^{(k)} * \vec{x}$ zauważamy, że $\vec{y} \in \mathcal{Y}_k := \operatorname{span}(\vec{z}_1^{(k)}, \ldots, \vec{z}_p^{(k)})$ i $\|\vec{y}\|_2 = 1$. Stąd wektor $V_2^T * \vec{y}$ zawiera współczynniki rzutu ortogonalnego $\vec{y}$ na $(\mathcal{P}^{(p)})^\perp := \operatorname{span}(\vec{\xi}_{p+1}, \ldots, \vec{\xi}_n)$ w bazie $(\vec{\xi}_{p+1}, \ldots, \vec{\xi}_n)$. Z definicji normy drugiej i twierdzenia 3.1 dostajemy więc

$$\|B_{2,1}^{(k)}\|_2 = \sup_{\|\vec{x}\|_2 = 1} \sin \angle \left(Z_1^{(k)} * \vec{x}, (\mathcal{P}^{(p)})^\perp \right) \leqslant \rho_p^k \cdot \operatorname{tg} \gamma_0.$$

Rozumując analogicznie i wykorzystując fakt, że

$$\sin \angle(\mathcal{Y}_k, \mathcal{P}^{(p)}) = \sin \angle \left(\mathcal{Y}_k^\perp, (\mathcal{P}^{(p)})^\perp \right)$$

dostajemy taką samą nierówność dla $\|B_{1,2}^{(k)}\|_2$. □

Z lematu 3.1 wynika natychmiast, że jeśli moduły wartości własnych macierzy A są parami różne to wyrazy pozadiagonalne $b_{i,j}^{(k)}$ macierzy B_k zbiegają do zera, a ponieważ B_k jest ortogonalna, to $|b_{i,i}^{(k)}|$ zbiegają do jedynki. Algorytm (IO) można więc uzupełnić o obliczanie w każdym kroku macierzy

$$A_k := Z_k^T * A * Z_k,$$

która powinna zbiegać do $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$. Fakt ten, łącznie z analizą szybkości zbieżności, pokażemy formalnie w następnym podrozdziale.

3.2 Metoda QR

3.2.1 Wyprowadzenie metody

Załóżmy, że realizujemy algorytm iteracji ortogonalnych (IO) z macierzą początkową $Z_0 = I$. Wtedy $A * Z_{k-1} = Z_k * R_k$, skąd

$$A_{k-1} = Z_{k-1}^T * A * Z_{k-1} = (Z_{k-1}^T * Z_k) * R_k.$$

Oznaczając $Q_k := Z_{k-1}^T * Z_k$ widzimy, że ostatnia równość to nic innego jak rozkład ortogonalno-trójkątny macierzy A_{k-1},

$$A_{k-1} = Q_k * R_k.$$

Prawdziwe są więc związki rekurencyjne

$$\begin{aligned} A_k &= Z_k^T * A * Z_k = (Z_k^T * Z_{k-1}) * (Z_{k-1}^T * A * Z_{k-1}) * (Z_{k-1}^T * Z_k) \\ &= Q_k^T * A * Q_k, \\ Z_k &= Z_{k-1} * (Z_{k-1}^T * Z_k) = Z_{k-1} * Q_k. \end{aligned}$$

Zależności te prowadzą do *ALGORYTMU QR*, który oblicza kolejne macierze A_k i Z_k w następujący sposób.

$$\text{(QR)} \quad \begin{array}{l} A_0 := A; \quad Z_0 := I \\ \text{dla} \quad k = 1, 2, 3, \ldots \\ \left\{ \begin{array}{l} A_{k-1} := Q_k * R_k \quad \text{(ortonormalizacja)}, \\ A_k := R_k * Q_k; \quad Z_k := Z_{k-1} * Q_k, \end{array} \right. \end{array}$$

3.2.2 QR a iteracje ortogonalne

Przypomnijmy, że w rozkładzie ortogonalno-trójkątnym macierz ortogonalna jest wyznaczona jednoznacznie z dokładnością do znaków jej wektorów-kolumn. Dlatego macierze Z_k (a tym samym także A_k) powstałe w wyniku realizacji algorytmów (IO) i (QR) nie muszą być takie same. Algorytmy te są jednak równoważne w następującym sensie.

Lemat 3.2. *Niech $\widetilde{Z}_k$, $\widetilde{A}_k$ i Z_k, A_k będą macierzami powstałymi w wyniku realizacji odpowiednio algorytmów (IO) i (QR). Wtedy*

$$\widetilde{Z}_k = Z_k * D_k \qquad \textit{oraz} \qquad \widetilde{A}_k = D_k * A_k * D_k,$$

gdzie D_k są pewnymi macierzami diagonalnymi z elementami ± 1 na głównej przekątnej.

Dowód. Zastosujemy indukcję względem k. Dla $k = 1$ mamy $\widetilde{Z}_0 = I = Z_0$ oraz $\widetilde{A}_0 = A = A_0$. Załóżmy, że lemat jest prawdziwy dla $k-1$, tzn. $\widetilde{Z}_{k-1} = Z_k * D_{k-1}$. Wtedy, z jednej strony

$$\widetilde{A}_{k-1} = D_{k-1} * A_{k-1} * D_{k-1} = D_{k-1} * Q_k * R_k * D_{k-1},$$

a z drugiej strony

$$\widetilde{A}_{k-1} = \widetilde{Z}_{k-1}^T * \widetilde{Z}_k * \widetilde{R}_k = D_{k-1} * Z_{k-1}^T * \widetilde{Z}_k * \widetilde{R}_k.$$

Mamy więc

$$D_{k-1} * \widetilde{A}_{k-1} = (Z_{k-1}^T * \widetilde{Z}_k) * \widetilde{R}_k = Q_k * (R_k * D_{k-1}),$$

przy czym równości te przedstawiają dwa rozkłady ortogonalno-trójkątne tej samej macierzy $D_{k-1} * \widetilde{A}_{k-1}$. A jeśli tak, to czynniki ortogonalne tych rozkładów różnią się jedynie znakami kolumn. Równoważnie, istnieje macierz diagonalna D_k z elementami ± 1 na głównej przekątnej taka, że

$$Z_{k-1}^T * \widetilde{Z}_k \;=\; Q * D_k \qquad (\text{oraz} \quad \widetilde{R}_k = D_k * R_k * D_{k-1}).$$

Stąd

$$\widetilde{Z}_k \;=\; Z_{k-1} * Q_k * D_k \;=\; Z_k * D_k.$$

Dowód kończy następujący rachunek:

$$\widetilde{A}_k \;=\; \widetilde{Z}_k^T * A * \widetilde{Z}_k \;=\; D_k^T * (Z_k^T * A * Z_k) * D_k \;=\; D_k * A_k * D_k.$$

□

Zauważmy jeszcze, że jeśli przez $\widetilde{a}_{i,j}^{(k)}$ i $a_{i,j}^{(k)}$ oznaczymy odpowiednio elementy macierzy $\widetilde{A}_k$ i A_k oraz $D_k = \operatorname{diag}(d_1^{(k)}, \ldots, d_n^{(k)})$ z $d_i^{(k)} = \pm 1$ to

$$\widetilde{a}_{i,j}^{(k)} = d_i^{(k)} d_j^{(k)} a_{i,j}^{(k)}.$$

To oznacza, że dla $i \neq j$ elementy te różnią się jedynie znakiem, a dla $i = j$ są sobie równe.

3.2.3 Analiza zbieżności

Jesteśmy już gotowi do przedstawienia twierdzeń o zbieżności metody QR.

Twierdzenie 3.2. *Dla $1 \leqslant p \leqslant n$, niech $\rho_p := |\lambda_{p+1}/\lambda_p| < 1$ i $\gamma_k := \angle\left(\mathcal{Y}_k, \mathcal{P}^{(p)}\right)$. Niech dalej*

$$A_k = \begin{bmatrix} A_{1,1}^{(k)} & A_{1,2}^{(k)} \\ A_{2,1}^{(k)} & A_{2,2}^{(k)} \end{bmatrix}, \quad Q_k = \begin{bmatrix} Q_{1,1}^{(k)} & Q_{1,2}^{(k)} \\ Q_{2,1}^{(k)} & Q_{2,2}^{(k)} \end{bmatrix}, \quad R_k = \begin{bmatrix} R_{1,1}^{(k)} & R_{1,2}^{(k)} \\ 0 & R_{2,2}^{(k)} \end{bmatrix},$$

gdzie podmacierze $A_{1,1}^{(k)}, Q_{1,1}^{(k)}, R_{1,1}^{(k)}$ są formatu $p \times p$, a pozostałe podmacierze formatów odpowiednio $p \times (n-p)$, $(n-p) \times p$ i $(n-p) \times (n-p)$. Wtedy, przyjmując

$$\varepsilon_p^{(k)} = \rho_p^k \cdot \operatorname{tg} \gamma_0$$

mamy $\operatorname{tg} \gamma_k \leqslant \varepsilon_p^{(k)}$ oraz

$$\|A_{1,2}^{(k)}\|_2 = \|A_{2,1}^{(k)}\|_2 \;\leqslant\; 2\,\varepsilon_p^{(k)} \|A\|_2, \tag{3.2}$$

$$\|Q_{1,2}^{(k)}\|_2,\, \|Q_{2,1}^{(k)}\|_2 \;\leqslant\; 2\,\varepsilon_p^{(k)}, \tag{3.3}$$

$$\|R_{1,2}^{(k)}\|_2 \;\leqslant\; 4\,\varepsilon_p^{(k)} \|A\|_2. \tag{3.4}$$

Dowód. Wobec wykazanej w lemacie 3.2 (teoretycznej) równoważności metody QR i iteracji (IO), nierówność $\operatorname{tg} \gamma_k \leqslant \varepsilon_p^{(k)}$ została już udowodniona w twierdzeniu 3.1. Aby pokazać (3.3), zapiszemy Z_k w postaci $Z_k = [Z_1^{(k)}, Z_2^{(k)}]$, gdzie $Z_1^{(k)}$ i $Z_2^{(k)}$ składają się odpowiednio z pierwszych p i z ostatnich $(n-p)$ kolumn macierzy Z_k. Wtedy

$$Q_{2,1} = Z_2^{(k-1)} * Z_1^{(k)} = \left(V^T * Z_2^{(k-1)}\right)^T * \left(V^T * Z_1^{(k)}\right),$$

gdzie $V = [\vec{\xi}_1, \vec{\xi}_2, \ldots, \vec{\xi}_n]$. Pisząc $V = [V_1, V_2]$, podobnie jak dla Z_k, mamy teraz

$$V^T * Z_1^{(k)} = \begin{bmatrix} V_1^T * Z_1^{(k)} \\ V_2^T * Z_1^{(k)} \end{bmatrix} =: \begin{bmatrix} F \\ G \end{bmatrix}.$$

Ponieważ $\|V_1\|_2 = \|Z_1^{(k)}\|_2 = 1$ to $\|F\|_2 \leqslant 1$, natomiast $\|G\|_2 \leqslant \varepsilon_p^{(k)}$, jak pokazaliśmy w dowodzie lematu 3.1.

Pisząc z kolei

$$V^T * Z_2^{(k-1)} =: \begin{bmatrix} F' \\ G' \end{bmatrix},$$

w analogiczny sposób pokazujemy, że $\|F'\|_2 \leqslant 1$ i $\|G'\|_2 \leqslant \varepsilon_p^{(k)}$. Stąd mamy

$$\|Q_{2,1}^{(k)}\|_2 = \|G^T * F + F^T * G\|_2 \leqslant \|G'\|_2\|F\| + \|F'\|_2\|G\|_2 \leqslant \varepsilon_{k-1} + \varepsilon_p^{(k)} \leqslant 2 \cdot \varepsilon_p^{(k)}.$$

Dokładnie tak samo pokazujemy oszacowanie dla $\|Q_{1,2}\|_2$.

Aby pokazać (3.2) zauważmy, że wobec $A_k = Q_k * R_k$ mamy też $A_{2,1}^{(k)} = Q_{2,1}^{(k)} * R_{1,1}^{(k)}$. Ale

$$\|R_{1,1}\|_2 \leqslant \|R_k\|_2 = \|A_2\|_2 = \|A\|_2,$$

co w połączeniu z (3.3) dowodzi (3.2).

W końcu, wobec $R_k = A_k * Q_k^T$ mamy

$$R_{1,2}^{(k)} = A_{1,1}^{(k)} * {Q_{2,1}^{(k)}}^T + A_{1,2}^{(k)} * {Q_{2,2}^{(k)}}^T.$$

Nierówność (3.4) wynika teraz z nierówności $\|A_{1,1}^{(k)}\|_2 \leqslant \|A_k\|_2 = \|A\|_2$, (3.2), (3.3), oraz $\|Q_{2,2}^{(k)}\|_2 \leqslant 1$. □

Z twierdzenia 3.2 wynika, że elementy pozadiagonalne macierzy A_k zbiegają do zera liniowo, przy czym iloraz zbieżności zależy od wartości $\rho_p = |\lambda_{p+1}|/|\lambda_p|$. Dokładniej, dla danego elementu $a_{i,j}^{(k)}$ macierzy A_k, gdzie $1 \leqslant i < j \leqslant n$, prawdziwe jest oszacowanie

$$|a_{i,j}^{(k)}| \leqslant \min\left(\varepsilon_i^{(k)}, \varepsilon_{i+1}^{(k)}, \ldots \varepsilon_{j-1}^{(k)}\right)$$

(gdzie skorzystaliśmy także z faktu, że moduł pojedynczego elementu macierzy jest nie większy od normy drugiej tej macierzy).

A jaka jest zbieżność elementów diagonalnych A_k do wartości własnych macierzy A? Oznaczmy

$$\overline{\varepsilon}_p^{(k)} = \begin{cases} \varepsilon_1^{(k)} & p = 1, \\ \max\left(\varepsilon_{p-1}^{(k)}, \varepsilon_p^{(k)}\right) & 2 \leqslant p \leqslant n-1, \\ \varepsilon_{n-1}^{(k)} & p = n. \end{cases}$$

Theorem 1. *Dla wszystkich* $1 \leqslant p \leqslant n$ *mamy*

$$|a_{p,p}^{(k)} - \lambda_p| \leqslant 4 \cdot \left(\overline{\varepsilon}_p^{(k)}\right)^2 \cdot \|A\|_2.$$

Dowód. Wobec $Z_k = V * B_k$ mamy

$$\begin{aligned} a_{p,p}^{(k)} &= (\vec{z}_p^{(k)})^T * A * \vec{z}_p^{(k)} = \left(V * \vec{b}_p^{(k)}\right)^T * A * \left(V * \vec{b}_p^{(k)}\right) \\ &= \vec{b}_p^{(k)} * V^T * A * V * \vec{b}_p^{(k)} = \vec{b}_p^{(k)} * \Lambda * \vec{b}_p^{(k)} \\ &= \lambda_p \left(b_{p,p}^{(k)}\right)^2 + \sum_{i \neq p} \lambda_i \left(b_{i,j}^{(k)}\right)^2. \end{aligned}$$

Stąd i z lematu 3.1 otrzymujemy

$$\begin{aligned} |a_{p,p}^{(k)} - \lambda_p| &= \left| -\lambda_p \left(1 - \left(b_{p,p}^{(k)}\right)^2\right) + \sum_{i \neq p} \lambda_i \left(b_{i,p}^{(k)}\right)^2 \right| = \left| \sum_{i \neq p} \left(b_{i,p}^{(k)}\right)^2 (\lambda_i - \lambda_p) \right| \\ &\leqslant 2 \cdot \|A\|_2 \cdot \left(\sum_{i<p} \left(b_{i,p}^{(k)}\right)^2 + \sum_{p<i} \left(b_{i,p}^{(k)}\right)^2 \right) \leqslant 4 \cdot \left(\overline{\varepsilon}_p^{(k)}\right)^2 \cdot \|A\|_2. \end{aligned}$$

□

Elementy diagonalne $a_{p,p}^{(k)}$ macierzy A_k zbiegają więc do wartości własnych λ_p macierzy A co najmniej tak szybko jak $|\lambda_2/\lambda_1|^{2k}$ dla $p = 1$, $|\lambda_n/\lambda_{n-1}|^{2k}$ dla $p = n$, oraz $\min\left\{|\lambda_p/\lambda_{p-1}|^{2k}, |\lambda_{p+1}/\lambda_p|^{2k}\right\}$ dla $2 \leqslant p \leqslant n-1$.

3.3 QR z przesunięciami

Rozdział 4

Kwadratury interpolacyjne w wielu wymiarach

4.1 Sformułowanie zadania

Ostatnie trzy wykłady poświęcimy numerycznemu całkowaniu funkcji wielu zmiennych. Dokładniej, dla danej funkcji $f : [0,1]^d \to \mathbb{R}$ chcemy obliczyć (przybliżyć) wartość

$$I_d(f) \;=\; \int_{[0,1]^d} f(\vec{x})\, d\vec{x} \;=\; \underbrace{\int_0^1 \int_0^1 \cdots \int_0^1}_{d} f(x_1, x_2, \ldots, x_d)\, dx_1 dx_2 \cdots dx_d.$$

Zakładamy, że powyższa całka istnieje. W ogólniejszym sformułowaniu, chcielibyśmy obliczyć całkę z wagą ω funkcji $f : \mathbb{R}^d \to \mathbb{R}$, która jest postaci

$$I_{d,\omega}(f) \;=\; \int_{\mathbb{R}^d} f(\vec{x})\, \omega(\vec{x})\, d\vec{x}.$$

Waga ω jest tutaj nieujemna i całkowalna.

Zauważmy, że ograniczenie się w ostatnim przypadku do $\mathbb{R}^d$ nie zmniejsza ogólności, gdyż całkę po dowolnym mierzalnym obszarze $D \subseteq \mathbb{R}^d$ można wymodelować przyjmując, że waga $\omega(\vec{x}) = 0$ dla wszystkich $\vec{x} \notin D$.

Zadanie całkowania funkcji wielu zmiennych ma ogromne znaczenie praktyczne i dlatego warto znać skuteczne metody numeryczne jego rozwiązywania.

Przyk?ad 4.1. Wycena obecnej wartości wielu instrumentów finansowych, w tym tzw. opcji, opiera się na założeniu, że przyszłe ceny podlegają losowym zmianom kolejnych odcinkach czasowych. Obecna wartość opcji obliczana jest jako wartość oczekiwana funkcji wypłaty. Odpowiada to obliczaniu całki oznaczonej funkcji d zmiennych, gdzie d jest liczbą odcinków czasowych. Jest to często całka ze standardową d wymiarową wagą gaussowską postaci

$$(2\pi)^{-d/2} \int_{\mathbb{R}^d} f(\xi_1, \ldots, \xi_d) \exp\Big(-\frac{1}{2}(\xi_1^2 + \cdots, \xi_d^2)\Big)\, d\xi_1 \cdots d\xi_d,$$

przy czym f jest (zwykle skomplikowaną) funkcją wypłaty na końcu okresu, a ξ_j reprezentują czynniki losowe w kolejnych odcinkach czasu. Wymiar d może wynosić nawet kilka tysięcy.

Z podstawowego wykładu z metod numerycznych każdy z nas wie jak numerycznie całkować funkcje jednej zmiennej. Stosowane metody w znakomitej większości przypadków sprowadzają

się do scałkowania funkcji, która jest kawałkami wielomianem określonego stopnia interpolującym funkcję podcałkową. Pomysł ten może być uogólniony na przypadek funkcji wielu zmiennych. Aby jednak mówić o kwadraturach interpolacyjnych w wielu wymiarach, musimy najpierw zastanowić się nad rozwiązywalnością odpowiedniego zadania interpolacyjnego.

4.2 Interpolacja na siatkach regularnych

4.2.1 Postać wielomianu interpolacyjnego

Niech

$$a \leqslant t_1 < t_2 < \cdots < t_r \leqslant b.$$

Jeśli f jest funkcją jednej zmiennej, $f : [a, b] \to \mathbb{R}$, to wielomian

$$p_f(x) = \sum_{j=1}^{r} f(t_j)\, l_j(x),$$

gdzie l_j jest j-tym wielomianem Lagrange'a,

$$l_j(x) = \prod_{j \neq i = 1}^{r} \frac{x - t_j}{t_i - t_j}, \qquad 1 \leqslant j \leqslant r,$$

(przy czym $l_1 \equiv 1$ dla $r = 1$) jest stopnia co najwyżej $(r-1)$ i interpoluje f w punktach t_j, tzn. przyjmuje w tych punktach te same wartości co f. W przypadku $d \geqslant 2$ możemy podobnie zdefiniować 'wielowymiarowe' wielomiany Lagrange'a.

W tym celu zakładamy, że na każdej współrzędnej dany jest przedział, a w nim układ r punktów

$$a^{(k)} \leqslant t_1^{(k)} < t_2^{(k)} < \ldots < t_r^{(k)} \leqslant b^{(k)}, \qquad 1 \leqslant k \leqslant d.$$

Oznaczając przez $l_j^{(k)}$ odpowiednie wielomiany Lagrange'a jednej zmiennej dla k-tego podziału, definiujemy wielomiany Lagrange'a d zmiennych jako

$$l_{j_1,\ldots,j_d}(x_1, \ldots, x_d) = l_{j_1}^{(1)}(x_1)\, l_{j_2}^{(2)}(x_2) \cdots l_{j_d}^{(d)}(x_d)$$

dla wszystkich $1 \leqslant j_k \leqslant r$, $1 \leqslant k \leqslant d$. Dla skrócenia zapisu, będziemy dalej używać zapisu wektorowego $\vec{j} = (j_1, \ldots, j_d)$, a $1 \leqslant \vec{j} \leqslant d$ będzie oznaczać, że nierówności zachodzą dla każdej współrzędnej j_k, $1 \leqslant k \leqslant d$. Podobnie, $t_{\vec{j}} = (t_{j_1}^{(1)}, t_{j_2}^{(2)}, \ldots, t_{j_d}^{(d)})$.

Wielomiany $l_{\vec{j}}$ należą do przestrzeni $\mathcal{P}_d^r$ wielomianów d zmiennych postaci

$$p(\vec{x}) = p(x_1, \ldots, x_d) = \sum_{0 \leqslant \vec{i} \leqslant r-1} a_{\vec{i}} \cdot x_1^{i_1} x_2^{i_2} \cdots x_d^{i_d},$$

gdzie $a_{\vec{i}}$ są dowolnymi wsoółczynnikami rzeczywistymi. Zauważmy, że $p \in \mathcal{P}_d^r$ wtedy i tylko wtedy gdy p jest wielomianem stopnia co najwyżej $(r-1)$ ze względu na każdą zmienną x_k.

Lemat 4.1. *Jeśli wielomian $p \in \mathcal{P}_d^r$ zeruje się we wszystkich r^d punktach $t_{\vec{j}}$, $1 \leqslant \vec{j} \leqslant r$, to p jest wielomianem zerowym.*

Dowód. Dowód przeprowadzimy przez indukcję ze względu na wymiar d. Dla $d = 1$ lemat jest oczywiście prawdziwy, bo na podstawie zasadniczego twierdzenia algebry niezerowy wielomian stopnia co najwyżej $(r-1)$ nie może mieć r różnych zer.

Niech $d \geqslant 2$. Niech $a_{\vec{j}}$ będą współczynnikami wielomianu p. Dla ustalonej k zdefiniujmy wielomian $p_k \in \mathcal{P}^r_{d-1}$ jako

$$p_k(x_1, \ldots, x_{d-1}) \;=\; p(x_1, \ldots, x_{d-1}, t^{(d)}_k).$$

Wielomian ten zeruje się w $r(d-1)$ punktach $t_{i_1,\ldots,i_{d-1}}$. Zapisując go w postaci

$$p_k(x_1, \ldots, x_{d-1}) \;=\; \sum_{0 \leqslant i_1, \ldots, i_{d-1} \leqslant d-1} b^{(k)}_{i_1,\ldots,i_{d-1}} \cdot x_1^{i_1} \cdots x_{d-1}^{i_{d-1}},$$

gdzie współczynniki

$$b^{(k)}_{i_1,\ldots,i_{d-1}} \;=\; \sum_{0 \leqslant i_d \leqslant d-1} a_{\vec{i}} \cdot (t^{(d)}_k)^{i_d},$$

oraz stosując założenie indukcyjne mamy, że $b^{(k)}_{i_1,\ldots,i_{d-1}} = 0$. A więc dla wszystkich wyborów indeksów $i_1, \ldots, i_{d-1}$ wielomian jednej zmiennej $\sum_{i_d=0}^{d-1} a_{\vec{i}} \cdot t^{i_d}$ zeruje się w r punktach $t = t^{(d)}_s$. To zaś wymusza $a_{\vec{i}} = 0$ dla wszystkich $0 \leqslant \vec{i} \leqslant d-1$ i w konsekwencji $p \equiv 0$. □

Lemat 4.1 wykorzystamy do pokazania następującego twierdzenia.

Twierdzenie 4.1. *Wielomiany $l_{\vec{j}}$, $1 \leqslant \vec{j} \leqslant r$, tworzą bazę przestrzeni $\mathcal{P}^r_d$. W szczególności,* $\dim(\mathcal{P}^r_d) = r^d$.

Dowód. Zauważmy, że podobnie jak w przypadku $d = 1$,

$$l_{\vec{j}}(t_{\vec{i}}) \;=\; \begin{cases} 1, & \text{jeśli } \vec{i} = \vec{j}, \\ 0, & \text{w przeciwnym przypadku.} \end{cases}$$

Stąd, jeśli kombinacja liniowa $p = \sum_{\vec{j}} \alpha_{\vec{j}} l_{\vec{j}}$ jest wielomianem zerowym to dla wszystkich $\vec{i}$

$$0 = p(t_{\vec{i}}) = \sum_{\vec{j}} \alpha_{\vec{j}} l_{\vec{j}}\,(t_{\vec{i}}) = \alpha_{\vec{i}},$$

czyli układ $\{l_{\vec{i}} : 1 \leqslant \vec{i} \leqslant r\}$ jest liniowo niezależny. Z drugiej strony, układ ten rozpina $\mathcal{P}^r_d$, bo dla dowolnego wielomianu p z tej przestrzeni mamy

$$p = \sum_{1 \leqslant \vec{j} \leqslant r} p(t_{\vec{j}})\, l_{\vec{j}}. \tag{4.1}$$

Rzeczywiście, w przeciwnym przypadku różnica wielomianu p i prawej strony (4.1) byłaby niezerowym wielomianem w $\mathcal{P}^r_d$, który zeruje się we wszystkich r^d punktach $t_{\vec{i}}$. To zaś przczyłoby lematowi 4.1. □

Stąd już jeden krok do następującego wniosku podsumowującego nasze dotychczasowe rozważania. Niech

$$D \;=\; [a^{(1)}, b^{(1)}] \times [a^{(2)}, b^{(2)}] \times \cdots \times [a^{(d)}, b^{(d)}]$$

będzie d wymiarowym prostokątem.

Wniosek 4.1. *Dla dowolnej funkcji $f : D \to \mathbb{R}$ wielomian*

$$p_f(\vec{x}) \;=\; \sum_{0 \leqslant \vec{j} \leqslant r} f(t_{\vec{j}})\, l_{\vec{j}}(\vec{j})$$

jest jedynym wielomianem w $\mathcal{P}^r_d$ interpolującym f w punktach $t_{\vec{j}}$ tzn. takim, że

$$p_f(t_{\vec{j}}) = f(t_{\vec{j}})$$

dla wszystkich $1 \leqslant \vec{j} \leqslant r$.

4.2.2 Błąd interpolacji

Zastanówmy się teraz jaki jest błąd otrzymanej interpolacji. Dla uproszczenia będziemy od teraz zakładać, że D jest kostką d wymiarową, tzn. wszystkie krawędzie mają tą samą długość, którą oznaczymy przez H, a węzły na każdej współrzędnej

$$t_j^{(k)} = a^{(k)} + u_j H, \qquad 1 \leqslant j \leqslant r,$$

gdzie

$$0 \leqslant u_1 < u_2 < \cdots < u_r \leqslant 1$$

jest pewną ustaloną siatką na odcinku jednostkowym.

W przypadku skalarnym, o ile funkcja f jest r-krotnie różniczkowalna w sposób ciągły, to

$$f(x) - p_f(x) \;=\; (x-t_1)(x-t_2)\cdots(x-t_r)\frac{f^{(r)}(\xi)}{r!},$$

przy czym $\xi \in [a,b]$ zależy od x. Stąd w szczególności mamy

$$|f(x) - p_f(x)| \;\leqslant\; \frac{(b-a)^r}{r!}\|f^{(r)}\|_\infty, \tag{4.2}$$

gdzie $\|f^{(r)}\|_\infty = \max_{a\leqslant t\leqslant b}|f^{(r)}(t)|$. Aby wyprowadzić formułę na błąd interpolacji w przypadku wielowymiarowym, będziemy potrzebować pewnego prostego uogólnienia ostatniego wzoru.

Załóżmy, że zamiast dokładnych wartości $f(t_i)$ mamy jedynie wartości przybliżone y_i takie, że błąd

$$|y_i - f(t_i)| \;\leqslant\; \delta, \qquad 1 \leqslant i \leqslant r. \tag{4.3}$$

Niech dalej $\tilde{p}_f$ będzie wielomianem stopnia co najwyżej $(r-1)$ interpolującym dane przybliżone y_i w punktach t_i. Ponieważ $(p_f - \tilde{p}_f)$ jest wielomianem interpolującym dane $f(t_j) - y_j$, na podstawie wzoru (4.1) mamy

$$|p_f(x) - \tilde{p}_f(x)| \;\leqslant\; \delta \cdot \sum_{i=1}^{r} |l_i(x)| \;\leqslant\; \delta \cdot S_r,$$

gdzie $S_1 = 1$, a dla $r \geqslant 2$

$$S_r = \max_{0\leqslant z\leqslant 1} \sum_{i=1}^{r} \prod_{i\neq j=1}^{r} \left|\frac{z-u_j}{u_i-u_j}\right|.$$

Stąd i z formuły na błąd interpolacji dla dokładnych danych otrzymujemy

$$|f(x) - \tilde{p}_f(x)| \;\leqslant\; |f(x) - p_f(x)| + |p_f(x) - \tilde{p}_f(x)| \;\leqslant\; \frac{(b-a)^r}{r!}\|f^{(r)}\|_\infty \;+\; \delta \cdot S_r. \tag{4.4}$$

Wprowadzimy jeszcze klasę $\mathcal{F}_r(D)$ funkcji $f : D \to \mathbb{R}$, które w całej swojej dziedzinie są r-krotnie różniczkowalne w sposób ciągły ze względu na każdą zmienną. Dla $f \in \mathcal{F}_r(D)$ definiujemy

$$B_r(f) \;=\; \max_{1\leqslant i\leqslant d} \left\{ \left\|\frac{\partial^r f}{\partial x_1^r}\right\|_\infty, \ldots, \left\|\frac{\partial^r f}{\partial x_d^r}\right\|_\infty \right\}.$$

Twierdzenie 4.2. *Niech $D = [a^{(1)}, a^{(1)} + H] \times \cdots \times [a^{(d)}, a^{(d)} + H]$. Jeśli $f \in \mathcal{F}_r(D)$ to dla każdego $\vec{x} \in D$ błąd interpolacji*

$$|f(\vec{x}) - p_f(\vec{x})| \;\leqslant\; \frac{H^r}{r!}\, C_{r,d}\, B_r(f),$$

gdzie $C_{1,d} = d$, a dla $r \geqslant 2$

$$C_{r,d} \;=\; \frac{S_r^d - 1}{S_r - 1}.$$

Dowód. Rozpatrzymy tylko $r \geqslant 2$ pozostawiając przypadek $r = 1$ jako proste ćwiczenie.
Dla $d = 1$ nierówność w tezie jest równoważna (4.2). Załóżmy więc, że $d \geqslant 2$. Ponieważ dla każdego ustalonego $t_k^{(d)}$ wielomian $(d-1)$ zmiennych $p_f(x_1, \ldots, x_{d-1}, t_k^{(d)})$ jest wielomianem interpolacyjnym dla funkcji $(d-1)$ zmiennych $f(x_1, \ldots, x_{d-1}, t_k^{(d)})$, na podstawie założenia indukcyjnego mamy

$$|f(x_1, \ldots, x_{d-1}, t_k^{(d)}) - p_f(x_1, \ldots, x_{d-1}, t_k^{(d)})| \leqslant \frac{H^r}{r!} B_r(f) \left(\frac{S_r^{d-1} - 1}{S_r - 1} \right). \tag{4.5}$$

Zauważmy, że dla ustalonych z kolei pierwszych $(d-1)$ współrzędnych $x_1, \ldots, x_{d-1}$ wielomian $p_f(x_1, \ldots, x_{d-1}, t)$ jest wielomianem jednej zmiennej t interpolującym funkcję jednej zmiennej $f(x_1, \ldots, x_{d-1}, t)$ w punktach $t_k^{(d)}$ na podstawie danych zaburzonych na poziomie δ równym prawej stronie (4.5). Stąd i z (4.4) ostatecznie otrzymujemy

$$\begin{aligned} |f(\vec{x}) - p_f(\vec{x})| &\leqslant \frac{H^r}{r!} B_r(f) + \delta \cdot S_r \\ &= \frac{H^r}{r!} B_r(f) \left(1 + \frac{S_r^{d-1} - 1}{S_r - 1} S_r \right) \\ &= \frac{H^r}{r!} B_r(f) \left(\frac{S_r^d - 1}{S_r - 1} \right). \end{aligned}$$

□

4.3 Kwadratury interpolacyjne

4.3.1 Kwadratury proste

Jesteśmy już dobrze uzbrojeni w mechanizm interpolacyjny i możemy zdefiniować wielowymiarowe kwadratury interpolacyjne dla całkowania funkcji $f : D \to \mathbb{R}$ zdefiniowanych na kostce

$$D = [a^{(1)}, a^{(1)} + H] \times \cdots \times [a^{(d)}, a^{(d)} + H].$$

Kwadratury te dane są równością

$$Q_{r,d}(f) = \int_D p_f(\vec{x})\, d\vec{x}, \tag{4.6}$$

gdzie $p_f \in \mathcal{P}_d^r$ jest wielomianem interpolującym f w punktach $t_{\vec{j}}$, $1 \leqslant \vec{j} \leqslant r$.
Chociaż postać (4.6) kwadratury znakomicie nadaje się do rozważań teoretycznych, nie jest jednak praktyczna ze względu na obliczenia. Zauważmy, że

$$\begin{aligned} Q_{r,d}(f) &= \int_D \sum_{\vec{j}} f(t_{\vec{j}}) l_{\vec{j}}(\vec{x})\, d\vec{x} = \sum_{\vec{j}} f(t_{\vec{j}}) \int_D l_{\vec{j}}(\vec{x})\, d\vec{x} \\ &= H^d \cdot \sum_{\vec{j}} f(t_{\vec{j}}) \prod_{k=1}^{d} \left(\int_0^1 l_{j_k}(u)\, du \right), \end{aligned}$$

gdzie l_j jest j-tym wielomianem Lagrange'a dla punktów $u_1, u_2, \ldots, u_d$. Stąd, wprowadzając oznaczenie

$$a_k = \int_0^1 l_k(u)\, du,$$

kwadraturę interpolacyjną można zapisać w postaci

$$Q_{r,d}(f) \;=\; H^d \cdot \sum_{1\leqslant j_1,\ldots,j_d\leqslant r} a_{j_1} a_{j_2} \cdots a_{j_d} \cdot f(t^{(1)}_{j_1}, t^{(2)}_{j_2}, \ldots, t^{(d)}_{j_d}).$$

Zauważmy, że a_k są współczynnikami jednowymiarowej kwadratury interpolacyjnej $Q_r(f) = \sum_{k=1}^{r} a_k f(t_k)$ opartej na punktach u_k, przybliżającej całkę $\int_0^1 f(u)\,du$. Mówiąc inaczej, zdefiniowana przez nas wielowymiarowa kwadratura interpolacyjna jest d-produktem tensorowym wybranej kwadratury jednowymiarowej.

Na koniec tego podrozdziału podamy oszacowanie błędu kwadratury $Q_{r,d}$. Ponieważ

$$\int_D f(\vec{x})\,d\vec{x} \;-\; Q_{r,d}(f) \;=\; \int_D \left(f(\vec{x}) - p_f(\vec{x})\right) d\vec{x},$$

z twierdzenia 4.2 natychmiast otrzymujemy następujący wniosek.

Wniosek 4.2. *Jeśli $f \in \mathcal{F}_r(D)$ to błąd kwadratury interpolacyjnej $Q_{r,d}$ jest ograniczony przez*

$$\left|\int_D f(\vec{x})\,d\vec{x} \;-\; Q_{r,d}(f)\right| \;\leqslant\; \frac{H^{r+d}}{r!}\, C_{r,d}\, B_r(f).$$

4.3.2 Kwadratury złożone

Podobnie jak w przypadku funkcji jednej zmiennej, definiujemy kwadratury złożone dla funkcji wielu zmiennych. Dla uproszczenia zakładamy, że całkujemy po kostce jednostkowej $[0,1]^d$.

Dla danego n wprowadzamy podział kostki na n^d podkostek

$$\left[\frac{i_1-1}{n}, \frac{i_1}{n}\right] \times \left[\frac{i_2-1}{n}, \frac{i_2}{n}\right] \times \cdots \left[\frac{i_d-1}{n}, \frac{i_d}{n}\right], \qquad 1 \leqslant i_k \leqslant n, \quad 1 \leqslant k \leqslant d.$$

Następnie na każdej podkostce stosujemy prostą kwadraturę interpolacyjną opartą na siatce regularnej składającej się z r^d punktów. Skonstruowaną w ten sposób kwadraturę złożoną oznaczymy przez $Q^{(n)}_{r,d}$.

Przyk?ad 4.2. Jeśli bazową kwadraturą jednowymiarową jest reguła punktu środkowego,

$$Q_1(f) \;=\; (b-a) \cdot f\left(\frac{a+b}{2}\right),$$

to wynikową kwadraturą złożoną na $[0,1]^d$ jest po prostu reguła prostokątów

$$Q^{(n)}_{r,d}(f) \;=\; \left(\frac{1}{n}\right)^d \cdot \sum_{1\leqslant i_1,\ldots,i_d\leqslant n} f\left(\frac{i_1 - 1/2}{n}, \cdots, \frac{i_d - 1/2}{n}\right).$$

Nasze rozważania wieńczy twierdzenie o błędzie kwadratury złożonej, które natychmiast wynika z wniosku 4.2 oraz sposobu konstrukcji kwadratury.

Twierdzenie 4.3. *Kwadratura złożona $Q^{(n)}_{r,d}$ korzysta z co najwyżej*

$$N \;=\; (r\,n)^d$$

wartości funkcji f. Jeśli $f \in \mathcal{F}_r([0,1]^d)$ to jej błąd

$$\left|\int_{[0,1]^d} f(\vec{x})\,d\vec{x} \;-\; Q^{(n)}_{r,d}(f)\right| \;\leqslant\; \left(\frac{1}{N}\right)^{r/d} \left(\frac{r^r}{r!}\right) C_{r,d}\, B_r(f).$$

4.4 Przekleństwo wymiaru

Złożone kwadratury interpolacyjne mogą być z powodzeniem stosowane dla niskich wymiarów, powiedzmy $d = 2, 3$. Dla dużych wymiarów d mają one bowiem tą niepożądaną własność, że liczba węzłów rośnie wykładniczo szybko wraz z zagęszczaniem siatki. Nawet jeśli weźmiemy po 2 punkty na każdej współrzędnej to całkowita liczba punktów siatki regularnej wyniesie 2^d. Pamiętamy, że w wielu praktycznych zastosowaniach d może sięgać nawet kilku tysięcy. W takich przypadkach obliczenie wartości kwadratury jest zadaniem praktycznie niewykonalnym.

To jednak nie koniec złych wiadomości. Przyjrzyjmy się jeszcze błędowi złożonej kwadratury interpolacyjnej. Twierdzenie 4.3 mówi, że błąd ten jest ograniczony z góry proporcjonalnie do $N^{-r/d}$, gdzie N jest liczbą wszystkich użytych punktów. To drugi powód do niepokoju, uzasadniony poniższym przykładem.

Przyk?ad 4.3. Załóżmy, że chcemy całkować funkcję 360 zmiennych i jako kwadraturę bazową stosujemy kwadraturę Simpsona, dla której $r = 4$. Górne ograniczenie błędu sugeruje, że aby być pewnym wyniku z dokładnością 10^{-2} to musimy obliczyć wartości funkcji w aż 10^{180} punktach. Czy naprawdę jest aż tak źle?

Rzeczywiście jest tak źle, a nawet gorzej. Okazuje się, że rzędu zbieżości $N^{-r/d}$ nie da się poprawić w klasie funkcji $\mathcal{F}_r([0,1]^d)$ jeśli bierzemy pod uwagę błąd najgorszy (pesymistyczny). Mówi o tym następujące twierdzenie.

Twierdzenie 4.4. *Istnieje $c > 0$ o następującej własności: dla dowolnej aproksymacji całki wykorzystującej*

$$N \geqslant \left(2^{\frac{1}{r+d}} - 1\right)^{-d}$$

wartości funkcji istnieje $f \in \mathcal{F}_r([0,1]^d)$ dla której $B_r(f) = 1$, a błąd aproksymacji całki wynosi co najmniej $c\,N^{-r/d}$.

Dowód. Załóżmy, że dana aproksymacja całki oblicza wartości funkcji w punktach $\vec{t}_j$, $1 \leqslant j \leqslant N$. Dowód twierdzenia polega na konstrukcji dwóch funkcji, f_+ i f_-, które zerują się we wszystkich $\vec{t}_j$ (a tym samym ich całki są aproksymowane tą samą liczbą), dla których $B_r(f_+) = 1 = B_r(f_-)$, ale różnica całek

$$\int_{[0,1]^d} (f_+ - f_-)(\vec{t})\, d\vec{t} \geqslant 2c\, N^{-r/d},$$

dla pewnej c niezależnej od f i d. Wtedy, przynajmniej dla jednej z tych funkcji błąd aproksymacji całki wynosi co najmniej $cN^{-r/d}$.

W tym celu, oznaczmy przez n najmniejszą liczbę naturalną spłniającą $N \leqslant n^d$ i skonstruujmy na $[0,1]^d$ regularną siatkę składającą się z $(2n)^d$ kostek, każda o krawędzi długości $h = 1/(2n)$.

Niech dalej $\phi : \mathbb{R} \to \mathbb{R}$ będzie dowolną funkcją r-krotnie różniczkowalną w sposób ciągły spełniającą następujące warunki:

1. $\phi(x) = 0$ dla $x \notin (0,1)$,
2. $\phi^{(j)}(0) = 0 = \phi^{(j)}(1)$ dla $0 \leqslant j \leqslant r$,
3. $\int_0^1 \phi(t)\, dt =: a > 0$.

Każdej kostce

$$K_{\vec{i}} := [(i_1 - 1)h, i_1 h] \times \cdots \times [(i_d - 1)h, i_d h]$$

naszej regularnej siatki przyporządkujemy funkcję

$$\phi_{\vec{i}}(x_1,\ldots,x_d) := h^r \phi(x_1/h - i_1,\ldots,x_d/h - i_d).$$

Zauważmy, że $B_r(\phi_{\vec{i}}) = 1$ oraz

$$\int_{[0,1]^d} \phi_{\vec{i}}(\vec{t})\, d\vec{t} = \int_{K_{\vec{i}}} \phi_{\vec{i}}(\vec{t})\, d\vec{t} = a\, h^{r+d}.$$

Jasne jest, że istnieje co najmniej

$$(2n)^d - N \geqslant (2^d - 1)N$$

multi-indeksów $\vec{i}$ (kostek) takich, że żaden z punktów $\vec{t}_j$ nie należy do wnętsza $K_{\vec{i}}$. Oznaczmy zbiór takich indeksów przez S i zdefiniujmy funkcje

$$f_+ := \sum_{\vec{i}\in S} \phi_{\vec{i}}, \qquad f_- := -f_+.$$

Wtedy obie funkcje zerują się w $\vec{t}_j$, $B_r(f_+) = 1 = B_r(f_-)$, oraz

$$\int_{[0,1]^d} f_+(\vec{t})\, d\vec{t} = -\int_{[0,1]^d} f_-(\vec{t})\, d\vec{t} \geqslant \left(2^d - 1\right) N\, a\, h^{r+d}.$$

Warunek na N oraz nierówność $(n-1)^d < N$ implikują, że

$$h^{r+d} = (2n)^{-(r+d)} \geqslant (2(N^{1/d}+1))^{-(r+d)} \geqslant 2^{-(r+d+1)} N^{-(r/d+1)}.$$

Stąd

$$\int_{[0,1]^d} f_+(\vec{t})\, d\vec{t} \geqslant \frac{a}{2^{r+2}} N^{-r/d},$$

a to oznacza, że teza twierdzenia zachodzi z $c = a2^{-(r+2)}$. □

Opisane zjawisko nosi nazwę *przekleństwa wymiaru*.

Rozdział 5

Metody Monte Carlo

5.1 Wstęp, metody niedeterministyczne

Poprzedni wykład zakończyliśmy pesymistycznym twierdzeniem 4.4, że nie istnieją efektywne metody numerycznego całkowania funkcji wielu zmiennych, ponieważ ma miejsce zjawisko przekleństwa wymiaru. Zwróćmy jednak uwagę na to, że fakt istnienia przekleństwa wymiaru stwierdziliśmy przy założeniach, że:

(i) model obliczeniowy jest *deterministyczny*,

(ii) funkcje podcałkowe są r-krotnie różniczkowalne po każdej zmiennej.

Można mieć nadzieję, że przekleństwo wymiaru zniknie, albo zostanie złagodzone, gdy przynajmniej jedno z tych założeń nie będzie spełnione.

Ten wykład poświęcimy (klasycznej) metodzie Monte Carlo numerycznego całkowania, która jest przykładem metody niedeterministycznej, tzn. takiej, która oblicza wynik wykorzystując zjawiska losowe. Chociaż może to brzmieć dziwnie, to właśnie niedeterministyczne zachowanie metody pozwala pokonać przekleństwo wymiaru.

Opisana dalej klasyczna metoda Monte Carlo związana jest ściśle ze Stanisławem Ulamem, uczniem Stefana Banacha i reprezentantem Lwowskiej Szkoły Matematycznej. Ulam zastosował metodę Monte Carlo do obliczania skomplikowanych całek w ramach "Manhattan Project"w Los Alamos (USA), w czasie II Wojny Światowej.

5.2 Klasyczna metoda Monte Carlo

5.2.1 Definicja i błąd

Tak jak w poprzednim rozdziale chcemy obliczyć całkę

$$I_d(f) := \int_{[0,1]^d} f(\vec{x})\, d\vec{x} = \underbrace{\int_0^1 \int_0^1 \cdots \int_0^1}_{d} f(x_1, x_2, \ldots, x_d)\, dx_1 dx_2 \cdots dx_d.$$

Zakładamy przy tym, że $f : [0,1]^d \to \mathbb{R}$ jest funkcją, której kwadrat jest całkowalny,

$$\int_{[0,1]^d} |f(\vec{x})|^2\, d\vec{x} < \infty.$$

Definicja 5.1. *Klasyczna metoda Monte Carlo* polega na przybliżeniu $I_d(f)$ średnią arytmetyczną wartości funkcji f w losowo wybranych punktach, tzn.

$$MC_{d,N}(f) \;=\; MC_{d,N}(f;\vec{t}_1,\vec{t}_2,\ldots,\vec{t}_N) \;:=\; \frac{1}{N}\cdot\sum_{j=1}^{N} f(\vec{t}_j),$$

gdzie $\vec{t}_1,\vec{t}_2,\ldots,\vec{t}_N$ są punktami wylosowanymi niezależnie od siebie, każdy zgodnie z rozkładem jednostajnym na $[0,1]^d$.

Konsekwencją zastosowania losowości jest to, że przy różnych realizacjach metody otrzymujemy różne wyniki, w zależności od wyboru punktów $\vec{t}_j$. Wynik $MC_{d,N}(f)$ jest więc zmienną losową, której wartość oczekiwana wynosi

$$\begin{aligned} E\left(MC_{d,N}(f)\right) &= \int_{[0,1]^{d\cdot N}} MC_{d,N}(f;\vec{t}_1,\ldots,\vec{t}_N)\,d\vec{t}_1\cdots d\vec{t}_N \\ &= \frac{1}{N}\sum_{j=1}^{N}\int_{[0,1]^d} f(\vec{t})\,d\vec{t} \;=\; I_d(f). \end{aligned}$$

Ponieważ różnica $I_d(f)-MC_{d,N}(f)$ jest też zmienną losową, za błąd metody Monte Carlo dla danej funkcji f przyjmiemy odchylenie standardowe,

$$e(f;MC_{d,N}) \;:=\; \sqrt{E\left(I_d(f)-MC_{d,N}(f)\right)^2}.$$

Twierdzenie 5.1. *Dla danej funkcji f błąd metody Monte Carlo wynosi*

$$e(f;MC_{d,N}) \;=\; \frac{\sigma(f)}{\sqrt{N}},$$

gdzie

$$\sigma(f) \;:=\; \sqrt{I_d(f^2)-I_d^2(f)}$$

jest wariancją funkcji f.

Zanim przystąpimy do dowodu zauważmy, że $\sigma(f)$ jest dobrze zdefiniowaną wielkością, bowiem nierówność

$$|I_d(f)| \leqslant \sqrt{I_d(f^2)}$$

jest szczególnym przypadkiem znanej *nierówności Schwarza* dla całek.

Dowód. Oznaczmy, dla uproszczenia, zmienną losową $X = MC_{d,N}(f)$. Wtedy

$$E(X-E(X))^2 \;=\; E\left(X(X-E(X))-E(X)(X-E(X))\right) \;=\; E(X^2)-E^2(X). \tag{5.1}$$

Ponadto

$$\begin{aligned} E(X^2) &= E\left(\frac{1}{N}\left(\sum_{j=1}^{N} f(\vec{t}_j)\right)^2\right) = \frac{1}{N^2}E\left(\sum_{j=1}^{N} f^2(\vec{t}_j)+\sum_{i\neq j} f(\vec{t}_i)f(\vec{t}_j)\right) \\ &= \frac{1}{N^2}\left(NI_d(f^2)+(N^2-N)I_d^2(f)\right) \;=\; \frac{1}{N}I_d(f^2)+\left(1-\frac{1}{N}\right)I_d^2(f), \end{aligned}$$

gdzie skorzystaliśmy z niezależności zmiennych losowych $f(t_i)$ i $f(t_j)$ dla $i\neq j$. Stąd i z (5.1) dostajemy

$$e^2(f;MC_{d,N}) \;=\; E(X-E(X))^2 \;=\; \frac{1}{N}I_d(f^2)+\left(1-\frac{1}{N}\right)I_d^2(f)-I_d^2(f) \;=\; \frac{1}{N}\left(I_d(f^2)-I_d^2(f)\right),$$

co kończy dowód. □

Uwaga 5.1. Zauważmy, że w dowodzie pokazaliśmy przy okazji nierówność Schwarza posługując się narzędziami rachunku prawdopodobieństwa.

Twierdzenie (5.1) mówi, że błąd metody Monte Carlo jest proporcjonalny do $N^{-1/2}$ przy bardzo słabych wstępnych założeniach na funkcję (jedynie całkowalność kwadratu funkcji). Jest to istotna poprawa w porównaniu do błędu $N^{-r/d}$ dla metod deterministycznych. W szczególności ważne jest, że wykładnik $1/2$ przy N^{-1} jest niezależny od wymiaru d, a konsekwencją tego pokonanie przekleństwa wymiaru.

Dziwnym może wydawać się, że przekleństwo wymiaru można zlikwidować używając metod niedeterministycznych (losowych). Jednak niczego nie ma za darmo. Należy pamiętać, że jest to możliwe za cenę niepewności wyniku. O ile bowiem metoda deterministyczna produkuje zawsze ten sam wynik, metoda niedeterministyczna (taka jak Monte Carlo) produkuje różne wyniki zależnie od konkretnych realizacji zmiennych losowych. Dlatego, mimo iż błąd oczekiwany jest proporcjonalny do $N^{-1/2}$ to nie mamy całkowitej pewności, że przy konkretnej realizacji otrzymany wynik jest tego samego rzędu. Z tego punktu widzenia warto przytoczyć następującą równość, która wynika z *centralnego twierdzenia granicznego*; mianowicie, dla dowolnych $c_1 < c_2$ mamy

$$\lim_{N\to\infty} \mathrm{Prob}\left(\frac{c_1\sigma(f)}{\sqrt{N}} \leqslant I_d(f) - MC_{d,N}(f) \leqslant \frac{c_2\sigma(f)}{\sqrt{N}}\right) = \frac{1}{\sqrt{2\pi}}\int_{c_1}^{c_2} e^{-t^2/2}dt,$$

gdzie Prob oznacza prawdopodobieństwo względem rozkładu jednostajnego na $[0,1]^{dN}$.

5.2.2 Całkowanie z wagą

Deterministyczne metody interpolacyjne z poprzedniego rozdziału można stosować jedynie do całkowania na d-wymiarowych prostokątach. Metoda Monte Carlo ma oprócz wymienionych również i tą zaletę, że łatwo ją uogólnić na przypadek całkowania z wagą. Dla przybliżenia wartości

$$I_{d,\omega}(f) := \int_{\mathbb{R}^d} f(\vec{x})\omega(\vec{x})\,d\vec{x}, \qquad \text{gdzie} \qquad \int_{\mathbb{R}^d} \omega(\vec{x})\,d\vec{x} =: W < \infty,$$

możemy bowiem zastosować wzór

$$MC^{\omega}_{d,N}(f) := \frac{W}{N}\cdot\sum_{j=1}^{N} f(\vec{t}_j),$$

przy czym $\vec{t}_1,\ldots,\vec{t}_N$ są tym razem punktami wybranymi losowo i niezależnie od siebie, zgodnie z rozkładem na $\mathbb{R}^d$ o gęstości ω/W.

Adaptując odpowiednio dowód twierdzenia 5.1 otrzymujemy następujące wyrażenie na błąd uogólnionej metody Monte Carlo.

Twierdzenie 5.2. *Niech* $I_{d,\omega}(f^2) = \int_{\mathbb{R}^d} f^2(\vec{x})\omega(\vec{x})\,d\vec{x} < \infty$. *Wtedy*

$$e(f; MC^{\omega}_{d,N}) = \frac{\sigma_\omega(f)}{\sqrt{N}},$$

gdzie

$$\sigma_\omega(f) = \sqrt{W\, I_{d,\omega}(f^2) - I^2_{d,\omega}(f)}.$$

5.3 Redukcja wariancji

Zauważyliśmy, że zaletą metody Monte Carlo jest nie tylko jej prostota, ale również to, że błąd średni wynosi $\sigma_\omega(f)N^{-1/2}$. Naturalnym jest teraz pytanie, czy błędu tego nie można poprawić. Temu celowi służą metody *redukcji wariancji*, które polegają w ogólności na redukcji czynnika $\sigma_\omega(f)$. Spośród wielu technik redukcji wariancji skupimy uwagę na dwóch: losowaniu warstwowemu oraz funkcjach kontrolnych. Dla uproszczenia będziemy zakładać, że całkujemy z wagą jednostkową na kostce

$$D = [0,1]^d.$$

5.3.1 Losowanie warstwowe

Podzielmy obszar całkowania D na K rozłącznych podzbiorów D_i tak, że

$$D = \bigcup_{i=1}^{K} D_i$$

i zastosujmy Monte Carlo do całkowania po każdym D_i, tzn. całkę $\int_D f(\vec{x})\,d\vec{x}$ przybliżymy wielkością

$$\overline{MC}_{d,N}(f) := \sum_{i=1}^{K} MC^{(i)}_{d,N_i}(f),$$

gdzie $MC^{(i)}_{d,N_i}$ jest metodą Monte Carlo zastosowaną do całki

$$I^{(i)}_d(f) := \int_{D_i} f(\vec{x})\,d\vec{x}, \qquad 1 \leqslant i \leqslant K,$$

oraz $N = \sum_{i=1}^{K} N_i$.

Oznaczmy przez $|D_i|$ objętość d-wymiarową podzbioru D_i. Ponieważ zmienne losowe $I^{(i)}_d(f) - MC^{(i)}_{d,N_i}(f)$ są parami niezależne dla $1 \leqslant i \leqslant K$, na podstawie twierdzenia 5.2 mamy

$$\begin{aligned} E\left((I_d(f) - \overline{MC}_{d,N}(f))^2\right) &= E\left(\left(\sum_{i=1}^{K} I^{(i)}_d(f) - MC_{d,N_i}(f)\right)^2\right) \\ &= \sum_{i=1}^{K} E\left((I^{(i)}_d(f) - MC^{(i)}_{d,N_i}(f))^2\right) \\ &= \sum_{i=1}^{K} \frac{1}{N_i}\left(|D_i|\, I^{(i)}_d(f^2) - (I^{(i)}_d(f))^2\right). \end{aligned}$$

Przyjmijmy teraz, że

$$N_i = |D_i| \cdot N, \qquad 1 \leqslant i \leqslant K,$$

przy czym dla uproszczenia (ale bez utraty ogólności) zakładamy, że wielkości te są całkowite. Wtedy otrzymujemy

$$e(f, \overline{MC}_{d,N}) = \frac{1}{\sqrt{N}} \cdot \sqrt{I_d(f^2) - \sum_{i=1}^{K} \frac{1}{|D_i|}(I^{(i)}_d(f))^2}. \tag{5.2}$$

Błąd tak zdefiniowanej metody $\overline{MC}_{d,N}$ nie jest większy od błędu klasycznej metody $MC_{d,N}$ z Twierdzenia 5.1.

Twierdzenie 5.3. *Dla dowolnej funkcji f takiej, że $I_d(f^2) < \infty$ mamy*

$$e(f, \overline{MC}_{d,N}) \leqslant e(f, MC_{d,N}),$$

przy czym równość zachodzi tylko wtedy gdy iloraz $I_d^{(i)}(f)/|D_i|$ jest stały, niezależnie od i.

Dowód. Rzeczywiście, oznaczając

$$a_i = \sqrt{|D_i|}, \qquad b_i = \frac{I_d^{(i)}(f)}{\sqrt{|D_i|}},$$

oraz wykorzystując nierówność Schwarza dla ciągów mamy

$$I_d^2(f) = \left(\sum_{i=1}^K a_i b_i\right)^2 \leqslant \left(\sum_{i=1}^K a_i^2\right)\left(\sum_{i=1}^K b_i^2\right) = \sum_{i=1}^K b_i^2 = \sum_{i=1}^K \frac{1}{|D_i|}(I_d^{(i)}(f))^2,$$

przy czym równość zachodzi tylko wtedy gdy wektory $(a_1, \ldots, a_K)^T$ i $(b_1, \ldots, b_K)^T$ są liniowo zależne, co jest równoważne warunkowi w treści twierdzenia.

Prawdziwość tezy pokazuje teraz porównanie wzorów na błędy obu metod. □

Widzimy, że stosując losowanie warstwowe z ustalonym podziałem na K podzbiory D_i możemy co prawda zmiejszyć błąd, ale szybkość zbieżności $N^{-1/2}$ pozostaje ta sama. A czy można poprawić zbieżność stosując różne podziały dla różnych wartości N? Okazuje się, że tak, o ile założymy pewną regularność funkcji f.

Aby to uzyskać, najpierw przekształcimy wzór (5.2) na błąd metody $\overline{MC}_{d,N}$ do postaci

$$e(f, \overline{MC}_{d,N}) = \frac{1}{\sqrt{N}}\sqrt{\sum_{i=1}^K I_d^{(i)}((f - c_i)^2)}. \tag{5.3}$$

gdzie

$$c_i := \frac{I_d^{(i)}(f)}{|D_i|}, \qquad 1 \leqslant i \leqslant K.$$

Załóżmy teraz, że f spełnia warunek Lipschitza ze stałą L,

$$|f(\vec{x}) - f(\vec{y})| \leqslant L \cdot \|\vec{x} - \vec{y}\|_\infty, \qquad \vec{x}, \vec{y} \in D.$$

Wtedy istnieją $\vec{x}_i \in \overline{D}_i$ takie, że $f(\vec{x}_i) = c_i$, a stąd i z lipschitzowskości f mamy, że dla dowolnego $\vec{x} \in D_i$

$$|f(\vec{x}) - c_i| \leqslant L \cdot \|\vec{x} - \vec{x}_i\|_\infty \leqslant L \cdot \operatorname{diam}_\infty(D_i),$$

gdzie

$$\operatorname{diam}_\infty(D_i) := \sup\{\|\vec{x} - \vec{y}\|_\infty : \vec{x}, \vec{y} \in D_i\}$$

jest średnicą zbioru D_i w normie max. W konsekwencji, ze wzoru (5.3) dostajemy następujące oszacowanie błędu:

$$e(f, \overline{MC}_{d,N}) \leqslant \frac{L}{\sqrt{N}}\sqrt{\sum_{i=1}^K |D_i| \operatorname{diam}^2(D_i)}.$$

Ustalmy teraz równomierny podział kostki D na $K = N$ podkostek D_i, każda o krawędzi długości $N^{-1/d}$ (zakładamy, bez zmniejszenia ogólności, że $N^{1/d}$ jest całkowita) tak, że nasza

metoda aproksymuje całkę na D_i używając tylko jednej wartości. Wtedy $\mathrm{diam}(D_i) = N^{-1/d}$ oraz

$$e(f, \overline{MC}_{d,N}) \leqslant L \cdot N^{-(1/2+1/d)}.$$

Ostatecznie, otrzymany w ten sposób wariant losowania warstwowego jest zbieżny z wykładnikiem większym niż $1/2$. Oczywiście, może to mieć praktyczne znaczenie jedynie dla małych wymiarów d, bowiem dla dużych d wykładnik $1/2 + 1/d$ jest właściwie równy $1/2$.

5.3.2 Funkcje kontrolne

Podobny efekt zwiększenia szybkości zbieżności można uzyskać stosując klasyczną Monte Carlo bezpośrednio do funkcji $f - g$, gdzie g jest pewną specjalnie dobraną funkcją, zwaną *funkcją kontrolną*.

Rzeczywiście, przedstawmy f w postaci $f = g + (f - g)$. Wtedy

$$I_d(f) = I_d(g) + I_d(f - g),$$

co sugeruje zastosowanie następującej metody:

$$\widetilde{MC}_{d,N}(f) := I_d(g) + MC_{d,N}(f - g).$$

Ponieważ

$$\begin{aligned} I_d(f) - \widetilde{MC}_{d,N}(f) &= (I_d(g) + I_d(f - g)) - (I_d(g) + MC_{d,N}(f - g)) \\ &= I_d(f - g) - MC_{d,N}(f - g), \end{aligned}$$

z twierdzenia 5.1 natychmiast wynika, że błąd

$$e(f; \widetilde{MC}_{d,N}) = e(f - g; MC_{d,N}) = \frac{\sigma(f - g)}{\sqrt{N}}.$$

Pozostaje kwestia doboru funkcji g tak, aby istotnie zmniejszyć wariancję $\sigma(f - g)$. Weźmy najpierw funkcję schodkową postaci

$$g(\vec{x}) = \sum_{i=1}^{K} c_i \mathbf{1}_{D_i}(\vec{x}),$$

gdzie

$$\mathbf{1}_{D_i}(\vec{x}) = \begin{cases} 1 & \vec{x} \in D_i, \\ 0 & \vec{x} \notin D_i, \end{cases}, \qquad 1 \leqslant i \leqslant K,$$

jest funkcją charakterystyczną zbioru D_i, a $c_i = f(\vec{x}_i)$ dla dowolnych $\vec{x}_i \in D_i$. Oczywiście, najlepiej byłoby wziąć $\vec{x}_i$ tak, aby $c_i = I_d^{(i)}(f)/|D_i|$, ale jest to niemożliwe, bo nie znamy całek $I_d^{(i)}(f)$. (Znajomość tych całek nie była konieczna w przypadku losowania warstwowego!) Wtedy dostajemy ten sam efekt jak dla $\overline{MC}_{d,N}$, tzn. dla lipschitzowskiej f

$$\sigma^2(f - g) \leqslant I_d((f - g)^2) \leqslant C^2 \cdot \sum_{i=1}^{K} |D_i| \,\mathrm{diam}^2(D_i) \tag{5.4}$$

i dla g odpowiadającej równomiernemu podziałowi na N podkostek otrzymujemy błąd proporcjonalny do $N^{-(1/2+1/d)}$.

W ogólności, funkcję g należy w praktyce wybierać tak, aby dobrze aproksymowała funkcję f. Oczywiście, wybór ten musi bazować na informacji jaką posiadamy o f.

Na przykład, dla funkcji r-gładkich, tzn. dla $f \in \mathcal{F}_r(D)$ można w ten sposób dostać jeszcze lepszy wykładnik. Rzeczywiście, niech $g = g_f$ będzie kawałkami wielomianem stopnia najwyżej $r-1$ po każdej zmiennej interpolującym f, dla równomiernego podziału kostki jednostkowej na $N^{1/d}$ podkostek. Z rozdziału 4 wiemy, że wtedy dla funkcji $f \in \mathcal{F}_r(D)$ błąd interpolacji jest postaci (zob. twierdzenie 4.2)

$$|f(\vec{x}) - g_f(\vec{x})| \leqslant \frac{C_{r,d} B_r(f)}{r!} \cdot N^{-r/d}$$

W konsekwencji, dla tak skonstruowanej metody dostajemy następujący błąd oczekiwany.

Twierdzenie 5.4. *Dla $f \in \mathcal{F}_r(D)$ mamy*

$$e(f; \widetilde{MC}_{d,N}) \leqslant \frac{C_{r,d} B_r(f)}{r!} \cdot N^{-(1/2+r/d)}.$$

Dodajmy, że $\widetilde{MC}_{d,N}$ jest w istocie metodą mieszaną, gdyż używa wartości funkcji f obliczanych w N punktach wybranych deterministycznie oraz w N punktach wybranych losowo.

Zbieżności $N^{-(1/2+r/d)}$ nie da się już poprawić w klasie $\mathcal{F}_r(D)$. Dokładniej, można pokazać następujące twierdzenie, które jest odpowiednikiem twierdzenia 4.4 dla algorytmów niedeterministycznych.

Twierdzenie 5.5. *Istnieje $c > 0$ o następującej własności: dla dowolnej (deterministycznej lub niedeterministycznej) aproksymacji całki wykorzystującej N wartości funkcji istnieje $f \in \mathcal{F}_r([0,1]^d)$ dla której $B_r(f) = 1$, a błąd oczekiwany aproksymacji całki wynosi co najmniej $c\, N^{-(1/2+r/d)}$.*

5.4 Generowanie liczb (pseudo-)losowych

Dotychczas milcząco przyjmowaliśmy, że umiemy generować ciągi

$$X_1, X_2, X_3, \ldots, X_n, \ldots$$

niezależnych liczb losowych zgodnie z danym rozkładem prawdopodobieństwa. Nie jest to jednak zadanie trywialne. W praktyce obliczeniowej liczby losowe uzyskujemy przez zastosowanie specjalnych programów. Ponieważ komputer jest urządzeniem deterministycznym, tak uzyskane ciągi nie są idealnie losowe już choćby dlatego, że są okresowe. Fakt ten wpływa na pogorszenie jakości wyniku i w szczególności powoduje, że ich użycie pozwala uzyskać jedynie kilka liczb znaczących, przy czym im większy wymiar d zadania tym gorsza graniczna dokładność. Z tych względów mówimy raczej o *generatorach liczb pseudo-losowych.*

Generowanie liczb pseudolosowych jest bardzo obszernym tematem, my tylko zwrócimy uwagę na podstawowe metody.

5.4.1 Liniowy generator kongruencyjny

Liniowe generatory kongruencyjne służą generowaniu ciągów losowych U_i o rozkładzie jednostajnym na odcinku $[0,1]$ i zdefiniowane są w następujący prosty sposób. Startujemy z $U_0 = x_0$ i kolejno obliczamy dla $i = 1, 2, 3, \ldots$

$$\begin{cases} x_i & := & (a\, x_{i-1} + c) \bmod m, \\ U_i & := & x_i / m, \end{cases}.$$

Jakość takiego generatora zależy istotnie o wyboru liczb całkowitych a, b i m. W szczgólności pożądane jest, aby generator miał maksymalny okres m. Jeśli $c \neq 0$ to warunkami dostatecznymi na to są:

(a) c i m są względnie pierwsze,

(b) jeśli p dzieli m to p dzieli $a-1$,

(c) jeśli 4 dzieli m to 4 dzieli $a-1$.

Dostęp do dobrego generatora liczb losowych o rozkładnie jednostajnym $U \sim \mathrm{Unif}([0,1])$ jest ważny również z tego względu, że jest on zwykle podstawą dla konstrukcji generatorów ciągów o bardziej skomplikowanych rozkładach prawdopodobieństwa.

5.4.2 Odwracanie dystrybuanty i 'akceptuj albo odrzuć'

Jeśli znana jest dystrybuanta żądanego rozkładu, czyli funkcja

$$F(x) := \mathrm{Prob}(X \leqslant x),$$

oraz łatwo obliczyć jej odwrotność zdefiniowaną jako

$$F^{-1}(u) := \inf\{x : F(x) \geqslant u\},$$

to potrzebne ciągi losowe mogą być wygenerowane według wzoru

$$X := F^{-1}(U), \qquad U \sim \mathrm{Unif}([0,1]).$$

Rzeczywiście, mamy

$$\mathrm{Prob}(X \leqslant x) = \mathrm{Prob}(F^{-1}(U) \leqslant x) = \mathrm{Prob}(U \leqslant F(x)) = F(x).$$

Na przykład, jeśli $F(x) = 1 - e^{-x^2/2}$ to można zastosować wzór $X = \sqrt{-2\ln(U)}$.

Niestety, dla wielu rozkładów dystrybuanta nie może być dokładnie obliczona. Wtedy jakość metody zależy od jakości zastosowanej numerycznej aproksymacji funkcji $F^{-1}(x)$.

Inna uniwersalna metoda generowania liczb losowych o dowolnym rozkładzie na $\mathbb{R}$, zwana *akceptuj albo odrzuć*, polega na wykorzystaniu istniejącego 'dobrego' generatora liczb innego rozkładu na $\mathbb{R}$. Dokładniej, załóżmy, że dysponujemy generatorem liczb losowych Y zgodnie z rozkładem o gęstości g, a interesują nas liczby X pochodzące z rozkładu o gęstości f. Załóżmy ponadto, że

$$f(x) \leqslant c \cdot g(x)$$

dla pewnej stałej $c > 0$. Wtedy możemy użyć następującego algorytmu:

```
{   repeat
        generuj Y zgodnie z g;
        generuj U ~ Unif([0,1])
    until U ⩽ f(Y)/(cg(Y));
    return X := Y
}
```

Aby pokazać poprawność takiego generatora zauważmy, że

$$\begin{aligned}\mathrm{Prob}(X \in A) &= \mathrm{Prob}(\, Y \in A \,:\, U \leqslant f(Y)/(cg(Y))\,) \\ &= \frac{\mathrm{Prob}(\, Y \in A,\, U \leqslant f(Y)/(cg(Y))\,)}{P(\, U \leqslant f(Y)/(cg(Y))\,)}.\end{aligned}$$

Ponieważ dla $a \in [0,1]$ prawdopodobieństwo, że $U \leqslant a$ wynosi a to

$$\mathrm{Prob}(U \leqslant f(Y)/(cg(Y))\,) = \int_{\mathbb{R}} \frac{f(x)}{cg(x)} g(x)\,dx = \frac{1}{c}$$

i w konsekwencji dostajemy

$$\begin{aligned}\mathrm{Prob}(X \in A) &= c \cdot \mathrm{Prob}(Y \in A,\, U \leqslant f(Y)/(cg(Y))\,) \\ &= c \cdot \int_A \frac{f(x)}{cg(x)} g(x)\,dx = \int_A f(x)\,dx.\end{aligned}$$

5.4.3 Metoda Box-Muller dla rozkładu gaussowskiego

Normalny rozkład gaussowski na $\mathbb{R}$ o funkcji gęstości

$$g(t) = \frac{1}{\sqrt{2\pi}} e^{-t^2/2}$$

jest najczęściej stosowanym rozkładem niejednostajnym. Dla rozkładu gaussowskiego bardzo efektywne okazują się algorytmy bazujące na odwracaniu dystrybuanty. Używają one dość skomplikowanych aproksymacji funkcji F^{-1}.

Prostszą i najbardziej popularną jest metoda *Box-Muller*. Generuje ona od razu dwie niezależne liczby Z_1 i Z_2 (albo, równoważnie, punkt $(Z_1, Z_2) \in \mathbb{R}^2$ zgodnie z rozkładem normalnym w $\mathbb{R}^2$), na podstawie dwóch liczb losowych o rozkładzie jednostajnym.

Przedstawmy $(x, y) \in \mathbb{R}^2$ we współrzędnych biegunowych,

$$\begin{cases} x &= r \cdot \cos\varphi, \\ y &= r \cdot \sin\varphi, \end{cases}$$

gdzie $r \in [0, \infty)$ i $\varphi \in [0, 2\pi)$. Metoda polega na wygenerowaniu zmiennych φ i r, a następnie zastosowaniu powyższego wzoru. Generowanie φ jest proste, bo ma rozkład jednostajny. Policzmy dystrybuantę rokładu zmiennej r. Mamy

$$\begin{aligned}\mathrm{Prob}(r \leqslant R) &= \frac{1}{2\pi} \int_{x^2+y^2 \leqslant R^2} e^{-(x^2+y^2)/2}\, d(x,y) = \frac{1}{2\pi} \int_0^{2\pi} \int_0^R r e^{-r^2/2} dr d\varphi \\ &= \int_0^R r e^{-r^2/2} dr = -e^{-r^2/2}\Big|_0^R = 1 - e^{-R^2/2}.\end{aligned}$$

Stąd, stosując metodę odwracania dystrybuanty mamy $R = \sqrt{-2\ln U}$, $U \sim \mathrm{Unif}([0,1])$. Rachunki te prowadzą do następującego generatora.

{ generuj $U_1, U_2 \sim \mathrm{Unif}([0,1])$;
$R := -2\ln U_1$; $V := 2\pi U_2$;
$Z_1 := \sqrt{R}\cos(V)$; $Z_2 := \sqrt{R}\sin(V)$;
return Z_1, Z_2
}

Rozdział 6

Metody quasi-Monte Carlo

6.1 Co to są metody quasi-Monte Carlo?

Metody Monte Carlo potrafią pokonać przekleństwo wymiaru, mają jednak również swoje wady. Najważniejsze z nich to:

(i) niepewność wyniku (który ma charakter probablistyczny),

(ii) stosunkowo wolna zbieżność (praktycznie $N^{-1/2}$),

(iii) konieczność stosowania (niekiedy skomplikowanych) generatorów liczb losowych.

Można powiedzieć, że skoro używamy z powodzeniem generatorów liczb pseudo-losowych, to implementacje Monte Carlo są w istocie deterministyczne. Dlatego, wybierając punkty deterministycznie, ale tak, aby w jakiś sposób 'udawały' i 'uśredniały' wybór losowy, powinniśmy dostać metodę deterministyczną o zbieżności co najmniej $N^{-1/2}$. A przy okazji usunęlibyśmy dwie z wymienionych wad Monte Carlo.

Takie intuicyjne myślenie wydaje się nie mieć racjonalnych podstaw, bo przecież metody deterministyczne podlegają przekleństwu wymiaru. Pamiętajmy jednak, że twierdzenie 4.4 o przekleństwie zachodzi w klasie $\mathcal{F}_r(D)$ funkcji różniczkowalnych r razy po każdej zmiennej. Zwróciliśmy na to uwagę już na początku rozdziału 5. Dlatego zasadne jest poszukiwanie pozytywnych rozwiązań dla funkcji o innej regularności.

Quasi-Monte Carlo jest deterministycznym odpowiednikiem klasycznej metody Monte Carlo dla aproksymacji całek na kostkach,

$$I_d(f) := \int_D f(\vec{x})\, d\vec{x}, \qquad D = [0,1]^d.$$

Definicja 6.1. Metoda *quasi-Monte Carlo* polega na przybliżeniu $I_d(f)$ średnią arytmetyczną,

$$\mathrm{QMC}_{d,N}(f) = \mathrm{QMC}_{d,N}(\vec{t}_1, \ldots, \vec{t}_N) := \frac{1}{n} \cdot \sum_{j=1}^{N} f(\vec{t}_j),$$

gdzie $\vec{t}_1, \vec{t}_2, \ldots, \vec{t}_N$ są pewnymi szczególnymi punktami w D wybranymi w sposób deterministyczny.

Przez długi czas od swojego powstania uważano, że metody quasi-Monte Carlo są efektywne jedynie dla całek o niskich wymiarach. Dopiero pod koniec lat 90-tych ubiegłego wieku zauważono,

że dają istotnie lepsze wyniki niż Monte Carlo w obliczaniu wartości niektórych instrumentów finansowych. Obecnie quasi-Monte Carlo jest powszechnie uznaną i bardzo popularną metodą, której znaczenie trudno przecenić mimo, że dotychczas nie udało się znaleźć pełnego teoretycznego wytłumaczenia jej efektywności.

6.2 Dyskrepancja

Rozważania na temat quasi-Monte Carlo zaczniemy od zdefiniowania pojęcia *dyskrepancji*, które odgrywa fundamentalną rolę w analizie efektywności tych metod.
Dla $\vec{x} = [x_1, \ldots, x_d]^T \in D$ niech

$$[\vec{0}, \vec{x}) \;:=\; [0, x_1) \times \cdots \times [0, x_d)$$

oznacza d wymiarowy prostokąt w D, 'zakotwiczony' w zerze. Zakładamy przy tym, że $[\vec{0}, \vec{0}) = \emptyset$.

Definicja 6.2. Dyskrepancją (z gwiazdką) danego zbioru N punktów $\vec{t}_j \in [0,1)^d$, $1 \leqslant j \leqslant N$, nazywamy wielkość

$$\mathrm{DISC}_d^*(\vec{t}_1, \ldots, \vec{t}_N) \;=\; \sup_{\vec{x} \in D} \Big| \mathrm{DISC}_d(\vec{x}; \vec{t}_1, \ldots, \vec{t}_N) \Big|,$$

gdzie

$$\mathrm{DISC}_d(\vec{x}; \vec{t}_1, \ldots, \vec{t}_N) \;=\; x_1 \ldots x_d - \frac{\#\left\{ j \;:\; \vec{t}_j \in [\vec{0}, \vec{x}) \right\}}{N},$$

a $\#A$ oznacza liczbę elementów zbioru A.

Ponieważ $x_1 \cdots x_d$ jest d-wymiarową objętością prostokąta $[\vec{0}, \vec{x})$, dyskrepancja pokazuje jak dobrze danych N punktów aproksymuje objętości tych prostokątów. Równoważnie, oznaczając przez $\mathbf{1}_{[\vec{0},\vec{x})}$ funkcję charakterystyczną prostokąta $[\vec{0}, \vec{x})$ mamy

$$\int_D \mathbf{1}_{[\vec{0},\vec{x})}(t)\, dt = x_1 \cdots x_d \qquad \text{oraz} \qquad \frac{1}{N} \sum_{j=1}^{N} \mathbf{1}_{[\vec{0},\vec{x})}(\vec{t}_j) = \frac{\#\{j \;:\; \vec{t}_j \in [\vec{0}, \vec{x})\}}{N}.$$

Stąd dyskrepancję można również interpretować jako maksymalny błąd aproksymacji funkcji charakterystycznych prostokątów przez odpowiedni algorytm quasi-Monte Carlo.

Pojęcie dyskrepancji zilustrujemy najpierw na przykładzie jednowymiarowym $d = 1$. Nietrudno pokazać, że dla dowolnych t_j

$$\mathrm{DISC}_1^*(t_1, \ldots, t_N) \;\geqslant\; \frac{1}{2N}. \tag{6.1}$$

Rzeczywiście, $\mathrm{DISC}_1(x; t_1, \ldots, t_N)$ jako funkcja x ma na każdym z przedziałów $[0, t_1)$, $[t_{j-1}, t_j)$, $2 \leqslant j \leqslant N$, pochodną równą 1, oraz przyjmuje zero dla $x = 0$. Zatem, jeśli dyskrepancja byłaby mniejsza od $1/(2N)$ to mielibyśmy

$$t_1 < \frac{1}{2N}, \qquad t_j - t_{j-1} < \frac{1}{N}, \quad 2 \leqslant j \leqslant N,$$

a stąd

$$t_N \;=\; t_1 + \sum_{j=2}^{N} (t_j - t_{j-1}) \;<\; \frac{1}{2N} + \frac{N-1}{N} \;=\; 1 - \frac{1}{2N}.$$

Otrzymujemy sprzeczność, bo dla $t_N < x < 1 - 1/(2N)$

$$\mathrm{DISC}_1(x; t_1, \ldots, t_N) \;<\; \left(1 - \frac{1}{2N}\right) - 1 \;=\; -\frac{1}{2N}.$$

Z dowodu wynika, że równość w (6.1) zachodzi jedynie dla równomiernego rozmieszczenia punktów,

$$t_j^* \;=\; \frac{j - 1/2}{N}, \qquad 1 \leqslant j \leqslant N.$$

W tym przypadku algorytm $\mathrm{QMC}_{1,N}$ redukuje się do zasady punktu środkowego.

Z punktu widzenia praktycznych obliczeń dobrze byłoby mieć ciąg nieskończony

$$t_1, t_2, \ldots, t_n, \ldots \subset [0, 1)$$

i konstruować kolejne aproksymacje używając N początkowych wyrazów tego ciągu. Ciekawe, że wtedy zbieżność N^{-1} nie może być zachowana. Dokładniej, można pokazać istnienie $c > 0$ takiego, że dla każdego ciągu nierówność

$$\mathrm{DISC}_1^*(t_1, \ldots, t_N) \;\geqslant\; c\,\frac{\ln N}{N}$$

zachodzi dla nieskończenie wielu N.

Analiza dyskrepancji w wymiarach $d \geqslant 2$ just dużo bardziej skomplikowana. Na razie ograniczymy się do zauważenia, że siatka równomierna jest fatalnym wyborem. Dokładniej, niech

$$\vec{t}_{j_1,\ldots,j_d} \;=\; [t_{j_1}^*, \ldots, t_{j_d}^*]^T, \qquad 1 \leqslant j_i \leqslant n, 1 \leqslant i \leqslant d,$$

będzie równomiernym rozkładem $N = n^d$ punktów w D. Rozpatrzenie prostokąta

$$[0, \tfrac{1}{2n}) \times \underbrace{[0,1) \times \cdots \times [0,1)}_{d-1}$$

wystarcza, aby się przekonać, że wtedy dyskrepancja wynosi co najmniej $\frac{1}{2n} = \frac{1}{2}N^{-1/d}$.

6.3 Błąd quasi-Monte Carlo

6.3.1 Formuła Zaremby

Jeśli $f : [0,1] \to \mathbb{R}$ jest funkcją różniczkowalną to dla każdego x mamy

$$f(x) \;=\; f(1) - \int_x^1 f'(t)\,dt \;=\; f(1) - \int_0^1 \mathbf{1}_{(x,1]}(t)\,f'(t)\,dt. \tag{6.2}$$

Wzór ten uogólnimy na przypadek funkcji wielu zmiennych w następujący sposób.

Najpierw wprowadzimy kilka niezbędnych oznaczeń. Dla podzbioru indeksów U,

$$\emptyset \;\neq\; U \;\subseteq\; \{1, 2, \ldots, d\},$$

niech $D_U = [0,1]^{|U|}$, gdzie $|U|$ jest liczbą elementów w U. Niech dalej $\vec{x}_U \in D_U$ będzie wektorem powstałym z wektora $\vec{x} = [x_1, x_2, \ldots, x_d]^T \in D$ poprzez usunięcie współrzędnych x_j z $j \notin U$, a $(\vec{x}_U; 1) \in D$ wektorem, którego j-ta współrzędna wynosi x_j dla $j \in U$ oraz wynosi 1 dla $j \notin U$.

Na przykład, jeśli $d = 5$ i $U = \{1, 4\}$ to dla $\vec{x} = [x_1, x_2, x_3, x_4, x_5]^T$ mamy $\vec{x}_U = [x_1, x_4]^T$ i $(\vec{x}_U; 1) = [x_1, 1, 1, x_4, 1]^T$.

W końcu, niech

$$f'_U = \frac{\partial^{|U|} f}{\prod_{j \in U} \partial u_j}$$

będzie skrótowym zapisem odpowiedniej pochodnej mieszanej funkcji f.

Lemat 6.1. *Jeśli funkcja $f : D \to \mathbb{R}$ ma ciągłe pochodne cząstkowe mieszane f'_U dla wszystkich $\emptyset \neq U \subseteq \{1, 2, \ldots, d\}$ to*

$$f(\vec{x}) = f(\vec{1}) + \sum_U (-1)^{|U|} \int_{D_U} \mathbf{1}_{(\vec{x}_U, \vec{1}_U]}(\vec{z}_U)\, f'_U(\vec{z}_U; 1)\, d\vec{z}_U, \qquad \vec{x} \in D \tag{6.3}$$

($\vec{1} = [1, 1, \ldots, 1] \in \mathbb{R}^d$).

Dowód. Dowód przeprowadzimy przez indukcję względem d. Dla $d = 1$ równość (6.3) jest równoważna (6.2). Niech więc $d \geqslant 2$. Stosując założenie indukcyjne do f z ustaloną ostatnią współrzędną x_d mamy

$$f(\vec{x}) = f(\vec{x}_{\{d\}}; 1) + \sum_V (-1)^{|V|} \int_{D_V} \mathbf{1}_{(\vec{x}_V, \vec{1}_V]}(\vec{z}_V)\, f'_V(\vec{z}_V, x_d; 1)\, d\vec{z}_V,$$

gdzie sumowanie jest po wszystkich $\emptyset \neq V \subseteq \{1, 2, \ldots, d-1\}$. Stosując dalej wzór (6.2) ze względu na x_d odpowiednio do $f(\vec{x}_{\{d\}}; 1)$ i $f'_V(\vec{z}_V, x_d; 1)$ dostajemy

$$\begin{aligned} f(\vec{x}) = {} & f(\vec{1}) - \int_{D_{\{d\}}} \mathbf{1}_{(\vec{x}_{\{d\}}, \vec{1}_{\{d\}}]}\left(\vec{z}_{\{d\}}\right) f'_{\{d\}}\left(\vec{z}_{\{d\}}; 1\right) d\vec{z}_{\{d\}} \\ & + (-1)^{|V|} \int_{D_V} \mathbf{1}_{(\vec{x}_V, \vec{1}_V]}(\vec{z}_V)\, f'_V(\vec{z}_V; 1)\, d\vec{z}_V \\ & + \sum_{V \cup \{d\}} (-1)^{|V \cup \{d\}|} \int_{D_{V \cup \{d\}}} \mathbf{1}_{(\vec{x}_{V \cup \{d\}}, \vec{1}_{V \cup \{d\}}]}\left(\vec{z}_{V \cup \{d\}}\right) f'_{V \cup \{d\}}\left(\vec{z}_{V \cup \{d\}}; 1\right) d\vec{z}_{V \cup \{d\}}. \end{aligned}$$

Teraz wystarczy porównać otrzymaną formułę do prawej strony (6.3). Drugi składnik odpowiada $U = \{d\}$, trzeci składnik podzbiorom $U \neq \emptyset$ do których nie należy d, a czwarty podzbiorom U takim, że $d \in U$ i $|U| \geqslant 2$. □

Zauważmy, że rozwijając $f(\vec{x})$ i $f(\vec{t}_j)$ zgodnie ze wzorem (6.3) mamy

$$\int_D f(\vec{x})\, d\vec{x} = f(\vec{1}) + \sum_U (-1)^{|U|} \int_{D_U} \left(\int_D \mathbf{1}_{(\vec{x}_U, \vec{1}_U]}(\vec{z}_U)\, d\vec{x} \right) f'_U(\vec{z}_U; 1)\, d\vec{z}_U,$$

$$\frac{1}{N} \sum_{j=1}^N f(\vec{t}_j) = f(\vec{1}) + \sum_U (-1)^{|U|} \int_{D_U} \left(\frac{1}{N} \sum_{j=1}^N \mathbf{1}_{((\vec{t}_j)_U, \vec{1}_U]}(\vec{z}_U) \right) f'_U(\vec{z}_U; 1)\, d\vec{z}_U.$$

Ponieważ wartość funkcji charakterystycznej odcinka $(\vec{a}, \vec{b}]$ w punkcie $\vec{c}$ jest równa wartości funkcji charakterystycznej odcinka $[\vec{0}, \vec{c})$ w punkcie $\vec{a}$ to

$$\int_D \mathbf{1}_{(\vec{x}_U, \vec{1}_U]}(\vec{z}_U)\, d\vec{x} = \int_{D_U} \mathbf{1}_{[\vec{0}_U, \vec{z}_U)}(\vec{x}_U)\, d\vec{x}_U = \prod_{j \in U} z_j,$$

$$\frac{1}{N}\sum_{j=1}^{N}\mathbf{1}_{((\vec{t}_j)_U,\vec{1}_U]}(\vec{z}_U) \;=\; \frac{1}{N}\,\#\left\{j\;:\;(\vec{t}_j)_U\in[\vec{0}_U,\vec{z}_U)\right\} \;=\; \frac{1}{N}\,\#\left\{j\;:\;\vec{t}_j\in\left[\vec{0},(\vec{z}_U;1)\right)\right\}.$$

Stąd otrzymujemy następującą *formułę Zaremby* na błąd quasi-Monte Carlo:

$$I_d(f)-\mathrm{QMC}_{d,N}(f) \;=\; \sum_U(-1)^{|U|}\int_{D_U}\mathrm{DISC}_d\left((\vec{z}_U;1);\vec{t}_1,\ldots,\vec{t}_N\right)\cdot f'_U(\vec{z}_U;1)\,d\vec{z}_U. \tag{6.4}$$

6.3.2 Nierówność Koksmy-Hlawki

Oznaczmy przez $\mathcal{V}_d(D)$ klasę funkcji $f:D\to\mathbb{R}$, których pochodne mieszane f'_U istnieją i są ciągłe dla wszystkich $\emptyset\neq U\subseteq\{1,\ldots,d\}$.

Definicja 6.3. *Wahaniem (w sensie Hardy-Krause)* funkcji $f\in\mathcal{V}_d(D)$ nazywamy wielkość

$$V_d(f) \;:=\; \sum_U\int_{D_U}\left|f'_U(\vec{z}_U;1)\right|\,d\vec{z}_U.$$

Zauważmy, że dla $d=1$,

$$V_1(f) \;=\; \int_0^1|f'(x)|\,dx \;=\; \sup\left\{\sum_{j=1}^{k}|f(z_j)-f(z_{j-1})|\;:\;k\geqslant 1, 0=z_0<z_1<\ldots<z_k=1\right\}$$

jest wahaniem funkcji w klasycznym sensie. Następujące bardzo ważne oszacowanie błędu metody quasi-Monte Carlo nosi nazwę *nierówności Koksmy-Hlawki*.

Twierdzenie 6.1. *Jeśli $f\in\mathcal{V}_d(D)$ to błąd metody quasi-Monte Carlo*

$$\mathrm{QMC}_{d,N}(f)=\frac{1}{N}\sum_{j=1}^{N}f(\vec{t}_j)$$

dla aproksymacji całki $I_d=\int_D f(\vec{x})\,d\vec{x}$ szacuje się przez

$$\left|I_d(f)-\mathrm{QMC}_{d,N}(f)\right| \;\leqslant\; \mathrm{DISC}^*_d(\vec{t}_1,\ldots,\vec{t}_N)\cdot V_d(f).$$

Dowód. Nierówność wynika natychmiast z formuły Zaremby (6.4), bowiem

$$\begin{aligned}|I_d(f)-\mathrm{QMC}_{d,N}(f)| \;&\leqslant\; \sum_U\int_{D_U}\left|\mathrm{DISC}_d\left((\vec{z}_U;1);\vec{t}_1,\ldots,\vec{t}_N\right)\right|\cdot\left|f'_U(\vec{z}_U;1)\right|\,d\vec{z}_U\\ &\leqslant\; \mathrm{DISC}^*_d(\vec{t}_1,\ldots,\vec{t}_N)\cdot\sum_U\int_{D_U}\left|f'_U(\vec{z}_U;1)\right|\,d\vec{z}_U.\end{aligned}$$

□

W nierówności Koksmy-Hlawki czynnik błędu $V_d(f)$ zależny jedynie od funkcji jest oddzielony od czynnika błędu $\mathrm{DISC}^*_d(\vec{t}_1,\ldots,\vec{t}_N)$ zależnego od wyboru punktów. O ile nie mamy wpływu na wahanie funkcji, możemy starać się wybrać punkty $\vec{t}_j$ tak, aby zminimalizować ich dyskrepancję. Zasadnicze pytanie brzmi: jak mała może być dyskrepacja? W szczególności, czy można wybrać nieskończony ciąg punktów tak, że oparte na nim algorytmy quasi-Monte Carlo pokonują przekleństwo wymiaru w klasie $\mathcal{V}_d(D)$?

Na miejscu jest teraz uwaga, że dzięki obecności pochodnych mieszanych rozumowanie analogiczne do tego z dowodu twierdzenia 4.4 prowadzi w klasie funkcji $f \in \mathcal{V}_d(D)$ z $V(f) \leqslant 1$ jedynie do oszacowania z dołu błędu przez cN^{-1} (a nie $cN^{-r/d}$).

Okazuje się, że pełne odpowiedzi na zadane pytania nie są znane. Najlepsze ciągi nieskończone $\vec{t}_1^*, \vec{t}_2^*, \ldots, \vec{t}_n^*, \ldots$ spełniają nierówność

$$\mathrm{DISC}_d^*\left(\vec{t}_1^*, \ldots, \vec{t}_N^*\right) \leqslant C_d \frac{\ln^d N}{N}, \qquad N = 1, 2, 3, \ldots,$$

gdzie $C_d > 0$ nie zależy od N.

Wydaje się więc, że w klasie $\mathcal{V}_d(D)$ metody quasi-Monte Carlo dają istotnie lepsze wyniki od Monte Carlo, bo błąd nie tylko jest deterministyczny, ale też dla $f \in \mathcal{V}_d(D)$ zbiega do zera dużo szybciej, tzn. błąd jest rzędu $N^{-1+\epsilon}$ dla dowolnego $\epsilon > 0$ w przypadku quasi-Monte Carlo, oraz $N^{-1/2}$ w przypadku Monte Carlo. Nie do końca jest to prawdą. Zauważmy bowiem, że w praktycznych obliczeniach wnioski asymptotyczne nie mają zastosowania, gdy wymiar d jest bardzo duży. Wtedy czynnik $C_d \ln^d N$ może mieć istotne znaczenie i wręcz sprawiać, że nierówność Koksmy-Hlawki staje się bezużyteczna. Dodatkowo, dobre oszacowanie C_d jest wyjątkowe trudne.

A jednak metody quasi-Monte Carlo są z powodzeniem stosowane w praktyce obliczeniowej nawet dla dużych wymiarów d. Istnieje wiele hipotez tworzonych w celu wyjaśnienia tego pozornego paradoksu. Jedna z najbardziej popularnych i już dość dobrze uzasadnionych teoretycznie mówi, że w praktyce mamy co prawda do czynienia z funkcjami bardzo wielu zmiennych, ale istotnych jest jedynie kilka zmiennych albo grupy kilku zmiennych. Matematycznie oznacza to, że w odpowiadających tym założeniom przestrzeniach zachodzą dużo mocniejsze odpowiedniki nierówności Koksmy-Hlawki.

6.4 Ciągi o niskiej dyskrepancji

Definicja 6.4. Ciąg nieskończony $\vec{t}_1, \vec{t}_2, \vec{t}_3, \ldots$ nazywamy *ciągem o niskiej dyskrepancji* jeśli istnieje $C_d > 0$ taka, że dla wszystkich N

$$\mathrm{DISC}_d^*\left(\vec{t}_1, \ldots, \vec{t}_N\right) \leqslant C_d \frac{\ln^d N}{N}.$$

Istnieje bardzo dużo efektywnych konstrukcji ciągów punktów o niskiej dyskrepancji. Teraz poznamy jedynie kilka najbardziej popularnych z nich.

6.4.1 Ciąg Van der Corputa

Niech $b \geqslant 2$ będzie ustaloną liczbą naturalną. Wtedy dowolną liczbę naturalną n można jednoznacznie przedstawić w postaci

$$n = \sum_{j=0}^{\infty} a_j(n)\, b^j,$$

gdzie $a_j \in \{0, 1, \ldots, b-1\}$ i tylko dla skończonej liczby indeksów j mamy $a_j \neq 0$. Mówiąc inaczej, a_j są kolejnymi cyframi rozwinięcia liczby n w bazie b. Wykorzystując powyższe rozwinięcie funkcję *radykalnej odwrotności* $\psi_b : \{0, 1, 2, \ldots\} \to [0, 1)$ definujemy jako

$$\psi_b(n) := \sum_{j=1}^{\infty} a_j(n)\, b^{-(j+1)}.$$

Na przykład, dla bazy $b = 2$ mamy $13 = 2^0 + 2^2 + 2^3 = (1101)_2$, czyli $\psi_2(13) = \frac{1}{2} + \frac{1}{8} + \frac{1}{16} = \frac{11}{16}$. Kolejne wartości radykalnej odwrotności,

$$\psi_b(1), \psi_b(2), \ldots, \psi_b(n), \ldots,$$

tworzą jednowymiarowy ciąg *Van der Corputa*. Dla $b = 2$ kolejne punkty ciągu wynoszą więc

$$\frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{1}{8}, \frac{5}{8}, \frac{3}{8}, \frac{7}{8}, \frac{1}{16}, \frac{9}{16}, \frac{5}{16}, \frac{13}{16}, \frac{3}{16}, \frac{11}{16}, \frac{7}{16}, \frac{15}{16}, \frac{1}{32}, \frac{17}{32}, \ldots$$

Ciąg Van der Corputa jest ciągiem o niskiej dyskrepancji dla dowolnie dobranej bazy b, tzn. dyskrepancja N początkowych wyrazów szacuje się z góry przez $C_1 \ln N/N$. Zauważmy jednak, że dla $N = b^k - 1$, $k \geqslant 1$, dostajemy równomierne rozmieszczenie punktów, tzn. tworzą one zbiór $\{j/(N+1) : 1 \leqslant j \leqslant N\}$, którego dyskrepancja wynosi $(N+1)^{-1}$. Stąd, pożądane są raczej mniejsze bazy b; im większe b tym rzadziej ze względu na N osiągana jest dyskrepancja proporcjonalna do $1/N$.

6.4.2 Konstrukcje Haltona i Sobol'a

Jednowymiarowy ciąg Van der Corputa jest podstawą konstrukcji wielu ciągów w wyższych wymiarach d. Jedna z nich prowadzi do ciągu *Haltona* $\{\vec{h}_k\}_{k\geqslant 1}$, którego k-ty punkt wynosi

$$\vec{h}_k \;=\; [\psi_{b_1}(k), \psi_{b_2}(k), \ldots, \psi_{b_d}(k)]^T .$$

Liczby $b_1, \ldots, b_d$ są tu danymi bazami. Od razu zauważamy, że wybór $b_1 = \cdots = b_d$ nie jest dobry, bo prowadzi do siatki równomiernej w $[0,1)^d$, zob. rozdział 6.2. Jeśli jednak b_i są liczbami pierwszymi to $\{\vec{h}_k\}_{k\geqslant 1}$ jest już ciągiem o niskiej dyskrepancji.

Minusem konstrukcji Haltona jest to, że dla dużych wymiarów d bazy b_d też są duże co, jak wcześniej zauważyliśmy, nie jest korzystne. Problemu tego unika bardziej skomplikowana konstrukcja *Sobol'a*, gdzie na każdej zmiennej pracujemy z tą samą bazą $b = 2$.

Ideę konstrukcji ciągu Sobol'a $\{\vec{s}_k\}_{k\geqslant 1}$ przedstawimy najpierw zakładając $d = 1$. Niech

$$\vec{a}(k) = [a_0(k), \ldots, a_{r-1}(k)]^T$$

będzie wektorem kolejnych bitów w rozwinięciu dwójkowym liczby k,

$$k \;=\; a_0(k) + 2a_1(k) + \cdots + 2^{r-1}a_{r-1}(k).$$

Wtedy

$$s_k \;=\; \frac{y_1(k)}{2} + \frac{y_2(k)}{4} + \cdots + \frac{y_r(k)}{2^r},$$

gdzie

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ \vdots \\ y_r(k) \end{bmatrix} \;=\; V \cdot \begin{bmatrix} a_0(k) \\ a_1(k) \\ \vdots \\ a_{r-1}(k) \end{bmatrix} \mod 2,$$

a V jest specjalnie dobraną macierzą o elementach 0 i 1, zwaną macierzą generującą. (Zauważmy, że jeśli V jest identycznością to otrzymany ciąg jest ciągiem Van der Corputa.)

Oznaczając $V = [\vec{v}_1, \ldots, \vec{v}_r]$ możemy równoważnie zapisać

$$\vec{y}(k) \;=\; a_0(k) \cdot \vec{v}_1 \oplus \cdots \oplus a_{r-1}(k) \cdot \vec{v}_r,$$

gdzie $\oplus$ jest operacją XOR, czyli dodawaniem bitów modulo 2,

$$0 \oplus 0 = 0, \quad 0 \oplus 1 = 1, \quad 1 \oplus 0 = 1, \quad 1 \oplus 1 = 0.$$

Dla $d \geqslant 2$, kolejne współrzędne punktu $\vec{s}_k$ wyliczane są według powyższej recepty, ale używając różnych macierzy generujących V. I właśnie problem wyboru macierzy generujących tak, aby otrzymać ciąg o niskiej dyskrepancji jest istotą konstrukcji Sobol'a. Dodajmy, że jest to problem wysoce nietrywialny.

6.4.3 Sieci (t, m, d) i ciągi (t, d)

Dla $b \geqslant 2$ definiujemy b-prostokąt w $[0,1)^d$ jako

$$\left[\frac{a_1}{b^{j_1}}, \frac{a_1 - 1}{b^{j_1}}\right) \times \cdots \times \left[\frac{a_d}{b^{j_d}}, \frac{a_d - 1}{b^{j_d}}\right),$$

gdzie $j_i \in \{0, 1, 2, \ldots\}$, $a_i \in \{0, 1, \ldots, b^{j_i - 1}\}$, $1 \leqslant i \leqslant d$. Na przykład, jeśli $d = 1$ i $b = 2$ to przedziały $[3/4, 1)$ i $[3/4, 7/8)$ są b-prostokątami, ale nie jest nim $[5/8, 7/8)$. Zauważmy, że objętość b-prostokąta wynosi $b^{-(j_1 + \cdots + j_d)}$.

Definicja 6.5. Niech $b \geqslant 2$ i $0 \leqslant t \leqslant m$. Siecią (t, m, d) w bazie b nazywamy zbiór b^m punktów w $[0,1)^d$ o własności, że w każdym b-prostokącie o objętości b^{t-m} znajduje się dokładnie b^t punktów tego zbioru.

Mówiąc inaczej, sieć (t, m, d) dokładnie pokazuje objętości b-prostokątów poprzez iloraz b^t/b^m liczby punktów, które należą do prostokąta do liczby wszystkich punktów.

Definicja 6.6. Ciąg nieskończony $\vec{t}_1, \vec{t}_2, \ldots$ w $[0,1)^d$ nazywamy ciągiem (t, d) w bazie b jeśli dla wszystkich $m > t$ i $j = 0, 1, 2, \ldots$ segment

$$\left\{ \vec{t}_i \,:\, jb^m < i \leqslant (j+1)b^m \right\}$$

jest siecią (t, m, d) w bazie b.

Następujące twierdzenie jest podstawą wielu konstrukcji ciągów o niskiej dyskrepancji.

Twierdzenie 6.2. *Każdy ciąg (t, d) w bazie b jest ciągiem o niskiej dyskrepancji.*

Pokazanie konkretnych konstrukcji ciągów (t, d) wykracza poza ramy tego wykładu. Powiemy tylko, że szczególne wybory macierzy generujących w konstrukcji Sobol'a prowadzą do ciągów (t, d). Inne konstrukcje należą do Faure'a, Niederreitera, Tezuki i in.