



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI



UNIwersytet
EKONOMICZNY
W KRAKOWIE



EDUKACJA
DLA
PRZEDSIĘBIORCZOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt „Uruchomienie unikatowego kierunku studiów Informatyka Stosowana odpowiedzią na zapotrzebowanie rynku pracy”
jest współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego.

Metody numeryczne

materiały do wykładu
dla studentów

1. Teoria błędów, notacja O

- 1.1. Błąd bezwzględny, błąd względny
- 1.2. Ogólna postać błędu
- 1.3. Problem odwrotny teorii błędów
 - zasada równego wpływu
 - metoda równych kresów górnych błędów bezwzględnych
 - metoda jednakowego pomiaru
- 1.4. Notacja O

1. Elementy teorii błędów

1.1. Błąd bezwzględny i względny

a – liczba przybliżona,

A – liczba dokładna

Liczba przybliżona a to liczba różniąca się nieznacznie od liczby dokładnej A i zastępująca ją w obliczeniach.

Definicja 1.1.

Błędem bezwzględnym Δa liczby przybliżonej a nazywamy wartość bezwzględną różnicy pomiędzy liczbą dokładną A i liczbą przybliżoną a :

$$\Delta a = |A - a| \quad (1.1)$$

Definicja 1.2.

Kres górny błędu bezwzględnego Δ_a to każda liczba nie mniejsza od błędu bezwzględnego, czyli:

$$\Delta a \leq \Delta_a \quad (1.2)$$

W praktyce za Δ_a wybiera się możliwie najmniejszą w danych warunkach liczbę spełniającą warunek (1.2).

Stosuje się zapis: $A = a \pm \Delta_a$.

Błąd bezwzględny (a nawet kres górny błędu bezwzględnego) nie charakteryzuje w pełni dokładności pomiarów czy obliczeń. Jeśli np. przy pomiarze dwóch prętów otrzymano wynik $l_1 = 100,87 \pm 0,1$ cm, $l_2 = 5,2 \pm 0,1$ cm, to chociaż $\Delta_a = 0,1$ cm, to w obu przypadkach pomiary nie są jednakowo dokładne.

Błąd bezwzględny przypadający na jednostkę pomiaru nazywamy błędem względnym i często podajemy go w procentach. Błąd względny charakteryzuje w sposób istotny dokładność pomiaru.

Definicja 1.3.

Błędem względnym δa liczby przybliżonej a nazywamy stosunek błędu bezwzględnego tej liczby (Δa) do wartości bezwzględnej liczby dokładnej A ($A \neq 0$), czyli:

$$\delta a = \frac{\Delta a}{|A|} \quad (1.3)$$

Równość (1.3) pozwala na zapis: $\Delta a = |A|\delta a$.

Kres górny błędu względnego (δ_a) spełnia zależność:

$$\delta a \leq \delta_a \quad (1.4)$$

czyli:

$$\frac{\Delta a}{|A|} \leq \delta_a,$$

a więc za kres górny błędu bezwzględnego danej wartości a można przyjąć wielkość:

$$\Delta_a = |A|\delta_a \quad (1.5)$$

Ponieważ nie znamy na ogół wartości dokładnej A , w praktyce $A \approx a$, można dla celów obliczeniowych posługiwać się zależnościami:

$$\delta_a \approx \frac{\Delta a}{|a|} \quad \text{oraz} \quad \Delta_a \approx |a|\delta_a \quad (1.6)$$

1.2. Ogólna postać błędu

Podstawowe zadanie teorii błędów można sformułować następująco:

Wyznaczyć błąd wartości funkcji, gdy znane są błędy wszystkich jej argumentów.

Niech będzie dana funkcja różniczkowalna $y = f(x_1, x_2, \dots, x_n)$ i niech $|\Delta x_i|$ dla $i = 1, 2, \dots, n$ będą błędami bezwzględnymi argumentów tej funkcji. Błąd bezwzględny funkcji można zapisać w postaci:

$$\begin{aligned}\Delta y &= |f(x_1 + \Delta x_1, x_2 + \Delta x_2, \dots, x_n + \Delta x_n) - f(x_1, x_2, \dots, x_n)| \approx \\ &\approx |df(x_1, x_2, \dots, x_n)| = \left| \sum_{i=1}^n f'_{x_i}(x_1, x_2, \dots, x_n) \cdot \Delta x_i \right| \leq \sum_{i=1}^n |f'_{x_i}(x_1, x_2, \dots, x_n)| \cdot |\Delta x_i|\end{aligned}$$

a więc:

$$\Delta y \leq \sum_{i=1}^n |f'_{x_i}| \cdot |\Delta x_i| \quad (1.7)$$

Przechodząc na kresy górne błędów otrzymujemy:

$$\Delta_y = \sum_{i=1}^n |f'_{x_i}| \cdot \Delta_{x_i} \quad (1.8)$$

Dzieląc obie strony nierówności (1.7) przez $|y|$ otrzymujemy oszacowanie błędu względnego funkcji f :

$$\delta y \leq \sum_{i=1}^n \left| \frac{f'_{x_i}}{f} \right| \cdot |\Delta x_i| = \sum_{i=1}^n \left| \frac{\partial}{\partial x_i} (\ln |f(x_1, x_2, \dots, x_n)|) \right| \cdot |\Delta x_i| \quad (1.9)$$

I znów dla kresów górnych mamy:

$$\delta_y = \sum_{i=1}^n \left| \frac{\partial}{\partial x_i} (\ln |y|) \right| \cdot \Delta_{x_i} \quad (1.10)$$

Przykład 1.1.

W pracach nad nowym smartfonem firma Gruszka wykorzystuje funkcję opisującą zadowolenie konsumentów z parametrów telefonu $f(x, y, z) = \frac{2x^2 + 3y^2}{z}$, gdzie x oznacza przekątną ekranu telefonu, y – szybkość procesora, a z – odsetek płatnych aplikacji w sklepie, przy czym przyjęto, że nowy smartfon będzie charakteryzował się parametrami $x = 5$ cali, $y = 2$ GHz, $z = 0,5$. W procesie produkcyjnym część modeli telefonu powstała z błędnymi wartościami parametrów, przy czym wartość każdego z parametrów zmieniła się o 2%. Jakiego można spodziewać się błędu bezwzględnego i względnego przy obliczaniu funkcji zadowolenia przy tak zmienionych parametrach?

1.3. Problem odwrotny teorii błędów

Problem odwrotny teorii błędów można sformułować następująco:

Jakie mogą być błędy bezwzględne argumentów funkcji, aby błąd bezwzględny wartości funkcji nie przekraczał zadanej wielkości?

Zarówno na podstawowy problem teorii błędów jak i na problem odwrotny można spojrzeć jak na równanie. Równanie to podaje nam wzór (1.8); przepisujemy je w tym miejscu (w miejsce symbolu Δ_y będziemy używać symbolu Δ_f):

$$\Delta_f = |f'_{x_1}| \cdot \Delta_{x_1} + |f'_{x_2}| \cdot \Delta_{x_2} + \dots + |f'_{x_n}| \cdot \Delta_{x_n}$$

W równaniu tym wartości $|f'_{x_i}|$ są zadane z góry dla wszystkich $i = 1, \dots, n$. Różnica pomiędzy podstawowym problemem teorii błędów i problemem odwrotnym polega na tym które wartości są dane, a które szukane:

- Problem podstawowy:
 - Dane: $\Delta_{x_1}, \dots, \Delta_{x_n}$
 - Szukane: Δ_f
- Problem odwrotny:
 - Dane: Δ_f
 - Szukane: $\Delta_{x_1}, \dots, \Delta_{x_n}$

Łatwo widać, że problem podstawowy to tak naprawdę jedno równanie z jedną niewiadomą (posiada jednoznaczne rozwiązanie), natomiast problem odwrotny to jedno równanie z n niewiadomymi (posiada zazwyczaj nieskończenie wiele rozwiązań). Co więcej, jeśli w problemie odwrotnym na przykład $f'_{x_n} \neq 0$, to wartości błędów $\Delta_{x_1}, \dots, \Delta_{x_{n-1}}$ można wybrać arbitralnie, a błąd Δ_{x_n} wyznaczyć względem pozostałych.

Z tych względów, aby zapewnić jednoznaczność rozwiązania problemu odwrotnego, potrzeba poczynić dodatkowe założenia. Poniżej przedstawiamy postaci błędów argumentów funkcji dla trzech różnych założeń (oczywiście liczba założeń, które można przyjąć, jest nieskończona).

1) Zasada równego wpływu.

Zakładamy, że różniczki cząstkowe $f'_{x_i} \cdot \Delta x_i$ dla $i = 1, 2, \dots, n$ wpływają jednakowo na powstanie błędu bezwzględnego funkcji $y = f(x_1, x_2, \dots, x_n)$, czyli:

$$|f'_{x_1}| \cdot \Delta_{x_1} = |f'_{x_2}| \cdot \Delta_{x_2} = \dots = |f'_{x_n}| \cdot \Delta_{x_n} = \frac{\Delta_f}{n} \quad (1.11)$$

Wtedy przekształcając wzór (1.8) otrzymamy:

$$\Delta_f = n \cdot |f'_{x_i}| \cdot \Delta_{x_i} \quad \text{dla } i = 1, 2, \dots, n$$

a stąd:

$$\Delta_{x_i} = \frac{\Delta_f}{n \cdot |f'_{x_i}|} \quad \text{dla } i = 1, 2, \dots, n \quad (1.12)$$

2) Metoda równych kresów górnych błędów bezwzględnych.

Zakładamy równość kresów górnych błędów bezwzględnych wszystkich argumentów funkcji, czyli:

$$\Delta_{x_1} = \Delta_{x_2} = \dots = \Delta_{x_n} \quad (1.13)$$

I wtedy z równości (1.8) mamy:

$$\Delta_{x_i} = \frac{\Delta_f}{\sum_{k=1}^n |f'_{x_k}|} \quad \text{dla } i = 1, 2, \dots, n \quad (1.14)$$

3) Pomiar jednakowo dokładny (metoda równych kresów górnych błędów względnych).

Zakładamy, że pomiar jest jednakowo dokładny dla wszystkich argumentów funkcji, czyli, że zachodzi równość kresów górnych błędów względnych δ_{x_i} ($i = 1, 2, \dots, n$) argumentów, czyli:

$$\delta_{x_1} = \delta_{x_2} = \dots = \delta_{x_n} = \delta \quad (1.15)$$

Zatem :

$$\frac{\Delta_{x_1}}{|x_1|} = \frac{\Delta_{x_2}}{|x_2|} = \dots = \frac{\Delta_{x_n}}{|x_n|} = \delta$$

a stąd:

$$\Delta_{x_i} = \delta |x_i| \quad \text{dla } i = 1, 2, \dots, n \quad (1.16)$$

Podstawiając (1.16) do równości (1.8) mamy:

$$\Delta_f = \sum_{i=1}^n |f'_{x_i}| \cdot \Delta_{x_i} = \sum_{i=1}^n |f'_{x_i}| \cdot \delta \cdot |x_i| = \delta \cdot \sum_{i=1}^n |f'_{x_i}| \cdot |x_i|$$

i stąd

$$\delta = \frac{\Delta_f}{\sum_{i=1}^n |f'_{x_i}| \cdot |x_i|}$$

Ostatecznie z równości (1.16) otrzymujemy:

$$\Delta_{x_i} = \frac{|x_i| \Delta_f}{\sum_{k=1}^n |f'_{x_k}| \cdot |x_k|} \quad \text{dla } i = 1, 2, \dots, n \quad (1.17)$$

Przykład 1.2.

Podczas igrzysk olimpijskich pewien łyżwiarz postanowił przed zawodami dosypać swojemu najgroźniejszemu konkurentowi nielegalną substancję do kawy, by spowodować jego dyskwalifikację. Musiał on zmieszać dwa składniki tak, by stężenie tego środka nie było ani

za małe (bo byłby niewykrywalny w testach po zawodach), ani za duże (żeby nikt się nie zorientował w podstępie). Zakładając, że stężenie substancji wyraża się wzorem $S(x, y, z) = \frac{xy+x^2}{z}$, gdzie x jest masą pierwszego składnika, y - masą drugiego składnika, a z - masą konkurenta oraz $x \approx 2$, $y \approx 5$, $z \approx 8$, oblicz z jaką dokładnością musi on znać wartości x , y , z , by stężenie było znane z dokładnością do 0,01?

Podsumowanie:

Problemy teorii błędów.

Problem	Przedmiot problemu
Podstawowy problem teorii błędów	Dane: $\Delta_{x_1}, \dots, \Delta_{x_n}$ Szukane: $\Delta_f = \sum_{i=1}^n f'_{x_i} \cdot \Delta_{x_i}$
Problem odwrotny teorii błędów	Dane: Δ_f Szukane: $\Delta_{x_1}, \dots, \Delta_{x_n}$

Problem odwrotny teorii błędów.

Metoda	Rozwiązanie
Zasada równego wpływu	Założenie: $ f'_{x_1} \cdot \Delta_{x_1} = f'_{x_2} \cdot \Delta_{x_2} = \dots = f'_{x_n} \cdot \Delta_{x_n}$ Postać błędu: $\Delta_{x_i} = \frac{\Delta_f}{n \cdot f'_{x_i} }$ dla $i = 1, 2, \dots, n$
Metoda równych kresów górnych błędów bezwzględnych	Założenie: $\Delta_{x_1} = \Delta_{x_2} = \dots = \Delta_{x_n}$ Postać błędu: $\Delta_{x_i} = \frac{\Delta_f}{\sum_{k=1}^n f'_{x_k} }$ dla $i = 1, 2, \dots, n$
Pomiar jednakowo dokładny	Założenie: $\delta_{x_1} = \delta_{x_2} = \dots = \delta_{x_n}$ Postać błędu: $\Delta_{x_i} = \frac{ x_i \Delta_f}{\sum_{k=1}^n f'_{x_k} \cdot x_k}$ dla $i = 1, 2, \dots, n$

1.4. Notacja O

Jednym z najistotniejszych czynników, spośród branych pod uwagę przy projektowaniu algorytmu, jest jego złożoność obliczeniowa. Najogólniej mówiąc wielkość ta informuje o ilości zasobów (np. czasu, pamięci) potrzebnych do wykonania algorytmu dla danych wejściowych o zadanej wielkości (np. długość ciągu, rozmiar macierzy). Oczywiście, spośród algorytmów rozwiązujących poprawnie pewien problem najlepszy będzie ten, którego wykonanie wymaga zaangażowania najmniejszych zasobów, zatem o najmniejszej złożoności obliczeniowej.

Z punktu widzenia czasu potrzebnego do wykonania algorytmu, za przydatne uważane są tylko takie algorytmy, dla których liczba operacji wymagana do ich wykonania rośnie co najwyżej wielomianowo wraz ze wzrostem wielkości danych wejściowych. Wykonanie algorytmów, dla których liczba operacji rośnie jak ciąg $n!$ bądź eksponencjalnie, trwałoby zbyt długo przy dużej liczbie danych wejściowych.

Przykład 1.3.

Dla zilustrowania powyższego stwierdzenia rozważmy hipotetyczny algorytm, który do rozwiązania pewnego problemu, przy danych wejściowych o rozmiarze n , potrzebuje wykonać 2^n operacji. Zrealizowanie tego algorytmu na procesorze wykonującym 2 miliony operacji na sekundę dla danych wejściowych o rozmiarze 80 zajęłoby około 19 miliardów lat (to więcej niż wynosi wiek wszechświata). Dla porównania, algorytm rozwiązujący ten sam problem za pomocą n^3 operacji, zaimplementowany na takim samym procesorze i przy takim samym rozmiarze danych wejściowych, pracowałby 0,256 sekundy.

Jedną z metod opisu złożoności obliczeniowej algorytmów jest notacja O (czytamy: o duże).

Definicja 1.4.

O dwóch ciągach o wartościach rzeczywistych $f(n)$ i $g(n)$ mówimy, że $f(n) = O(g(n))$ (czytamy: **$f(n)$ jest „o duże” od $g(n)$**), gdy istnieją $c > 0$ i $n_0 \in \mathbb{N}$ takie, że

$$|f(n)| \leq c \cdot |g(n)| \text{ dla wszystkich } n \geq n_0. \quad (1.18)$$

Mówiąc nieściśle symbol $f(n) = O(g(n))$ oznacza, że ciąg $f(n)$ rośnie nie szybciej niż pewna wielokrotność ciągu $g(n)$.

Twierdzenie 1.1 (Złożoność wielomianowa)

Wielomiany k -tego stopnia są $O(n^k)$.

Aby wykazać to stwierdzenie rozważmy wielomian stopnia k w najogólniejszej postaci $W_k(n) = a_k n^k + a_{k-1} n^{k-1} + \dots + a_1 n + a_0$. Wtedy

$$|W_k(n)| \leq |a_k|n^k + |a_{k-1}|n^k + \dots + |a_1|n^k + |a_0|n^k = \left(\sum_{i=0}^k |a_i| \right) \cdot n^k.$$

Przyjmując $c = (\sum_{i=0}^k |a_i|)$ oraz $n_0 = 1$ otrzymujemy nierówność (1.18), co oznacza, że $W_k(n) = O(n^k)$.

Twierdzenie 1.2 (Hierarchia znanych ciągów)

W poniższym zestawieniu każdy ciąg jest O od wszystkich ciągów na prawo od niego:

$$1, \log n, n^{\alpha_1}, n^{\alpha_2}, n, n \log n, n^{\beta_1}, n^{\beta_2}, 2^n, n!, n^n,$$

gdzie $\alpha_1 < \alpha_2 < 1 < \beta_1 < \beta_2$.

Z punktu widzenia algorytmów stwierdzenie, że algorytm ma złożoność obliczeniową równą $O(g(n))$ oznacza, że istnieje ciąg $f(n)$ spełniający nierówność (1.18) taki, że do zrealizowania algorytmu na danych wejściowych o wielkości n potrzeba wykonać $f(n)$ operacji elementarnych. Dodatkową zaletą stosowania notacji O do opisu złożoności obliczeniowej jest grupowanie algorytmów.

Przykład 1.4.

Spośród metod rozwiązywania w sposób dokładny układu równań liniowych najbardziej znane to metoda macierzy odwrotnej i metoda wzorów Cramera. Łatwo można się przekonać, że metody te są co najmniej $O(n!)$ (obliczenie wyznacznika metodą rozwinięcia Laplace'a jest $O(n!)$), zatem w przypadku układów równań o wielu zmiennych metody te są mało przydatne.

Algorytmy rozwiązujące układy równań liniowych zaprezentowane w kolejnym rozdziale są wszystkie $O(n^3)$, tzn. do rozwiązania w sposób dokładny układu równań zadanego przez macierz kwadratową o rozmiarach $n \times n$ każdy z tych algorytmów potrzebuje wykonać

pewną liczbę operacji elementarnych będącą wielomianem trzeciego stopnia zmiennej n (wielomiany te mogą być różne dla różnych algorytmów). Algorytmy te znajdują zastosowanie przy rozwiązywaniu układów równań o wielu zmiennych, gdyż ich złożoność obliczeniowa jest wielomianowa.

Przykład 1.5.

Przypuśćmy, że z grupy 2^m osobników chcemy wyizolować jednego zainfekowanego pewnym patogenem. Przyjmijmy dodatkowo, że patogen ten można zlokalizować za pomocą badania krwi, my zaś posiadamy pewną ilość krwi każdego z osobników. Rozważmy następujące metody przeprowadzenia badań:

- 1) Metoda naiwna: badamy jedną po drugiej próbkę krwi każdego z osobników.
W metodzie tej w najgorszym przypadku trzeba będzie wykonać 2^m badań. To oznacza, że algorytm ten jest $O(n)$ (ponieważ dla danych wejściowych rozmiaru $n = 2^m$ należy wykonać n operacji na tych danych).
- 2) Metoda ulepszona: metoda ta składa się z m kroków. W k -tym kroku ($1 \leq k \leq m$) bierzemy niewielką próbkę krwi każdego z osobników, próbki dzielimy na dwa zbiory, a następnie próbki w każdym ze zbiorów mieszamy ze sobą. Teraz wystarczy zbadać krew w pierwszym zbiorze: jeśli patogen został wykryty, to do kolejnego kroku przechodzą osobnicy o próbkach z tego zbioru, jeśli nie, to z drugiego. W ten sposób w m -tym kroku każdy ze zbiorów będzie składał się z jednej próbki. Po przeprowadzeniu ostatniego badania (np. próbki w pierwszym zbiorze) otrzymamy odpowiedź na pytanie o zainfekowanego osobnika.
W metodzie tej zawsze trzeba wykonać m kroków, a w każdym kroku przeprowadzić jedno badanie. To oznacza, że algorytm ten jest $O(\log_2 n)$ (ponieważ dla danych wejściowych rozmiaru $n = 2^m$ należy wykonać $m = \log_2 n$ operacji na tych danych).

Złożoność obliczeniowa metody ulepszonej jest mniejsza niż metody naiwnej, zatem to metoda ulepszona stanowi lepszy algorytm wyszukiwania zainfekowanego osobnika.

Dla zilustrowania tego stwierdzenia rozważmy grupę $65536 = 2^{16}$ osobników. Zakładając, że jesteśmy w stanie przeprowadzić jedno badanie w czasie jednej godziny, zrealizowanie algorytmu naiwnego zajmie nam prawie 7,5 roku (przy założeniu, że każdego dnia badania są przeprowadzane bez przerwy przez całą dobę). Wykorzystanie metody ulepszonej pozwoli na przeprowadzenie wszystkich badań w czasie 16 godzin.