

Seriál: Numerické metody a počítačové simulace

V minulém díle jste se seznámili s reprezentací čísel v počítači a s chybami, které mohou plynout z nepřesnosti datového typu s plovoucí desetinnou čárkou, a naučili jste se, jak tyto chyby minimalizovat či jim zcela předcházet. Také již umíte určovat časovou složitost počítačových algoritmů. Tyto znalosti využijete při numerickém výpočtu derivací, což je téma tohoto dílu seriálu.

V simulační části seriálu po předchozím náročném dílu trochu zvolníme a podíváme se na již slíbenou stochastickou metodu Monte Carlo, přičemž si zopakujeme a prohloubíme znalosti týkající se statistického zpracování výsledků simulací. Ke slovu zde opět přijdou generátory pseudonáhodných čísel.

Numerická derivace

Protože nepředpokládáme čtenářovu znalost pojmu derivace, pokusme se jej nejprve stručně nastínit. Derivace je operace, která zapůsobí na funkci f a vrátí jinou funkci – derivaci původní funkce, značíme f' . Pokud derivaci provedeme pro konkrétní hodnotu x , dostaneme tzv. *derivaci v bodě* $f'(x)$, tedy číslo. Při numerických výpočtech máme k dispozici pouze čísla, nikoliv spojitě funkce, numerickou derivací tedy budeme chápat právě jako derivaci v bodě.

Derivace funkce f v bodě x je definována vztahem

$$f'(x) = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}.$$

Odtud je hned patrný význam derivace, tedy že derivace je podíl změny funkční hodnoty f a příslušné změny x za předpokladu, že je tato změna velmi malá (ve skutečnosti nekonečně malá). To má řadu praktických analogií, například průměrná rychlost je podíl změny polohy a příslušného přírůstku času. Pokud bude přírůstek času velmi malý, přejde průměrná rychlost v rychlost okamžitou. Můžeme tedy říci, že okamžitá rychlost je derivace polohy podle času.

Další, matematictější význam derivace je, že derivace v bodě je směrnice tečny¹ ke grafu funkce f v bodě dotyku x .

Můžeme také provést derivaci derivace funkce, což bude opět funkce. Tu nazýváme *druhou derivací* funkce f a značíme f'' . Tak můžeme (pro hezké funkce) pokračovat dále. Obecně n tou derivací značíme $f^{(n)}$.

Čtenáři zběhlí v počítání derivací nejspíš ví, že výpočet derivace je pro hezké funkce přímočarý a derivaci v bodě pak už získáme snadno dosazením konkrétní hodnoty x , numerický výpočet derivace je v tomto případě zbytečný. Přesto existují situace, kdy se numerická derivace může hodit. Prvním příkladem jsou složité funkce, u kterých je náročné počítat derivaci analyticky. Druhým, častějším příkladem je situace, kdy neznáme původní funkci, ale pouze její funkční hodnoty v diskretních bodech. Příkladem může být situace, kdy máme nafilmovaný pohybující se objekt, z každého snímku odečteme jeho polohu a chceme zjistit časový průběh rychlosti, s jakou se objekt pohyboval. Pak se nějaké formě numerické derivace nevyhneme.

¹ Pokud tečnu zapíšeme rovnicí přímky $y = ax + b$, pak směrnice je koeficient a . Směrnice tedy udává sklon dané přímky.

Podívejme se nyní opět na definiční vztah derivace. Může nás napadnout, že numerickou derivaci implementujeme prostě tak, že Δx nebude nekonečně malé, ale bude to nějaký (konečně) malý krok $h > 0$. To lze chápat také tak, že místo tečny v bodě x konstruujeme sečnu protínající graf funkce v bodech x a $x + h$. Pak dostáváme vztah

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

Tomuto druhu numerické derivace se říká *dopředná diference*², ovšem, jak si za chvíli ukážeme, jde zpravidla o jeden z nejhorších a nejméně přesných způsobů výpočtu numerické derivace.

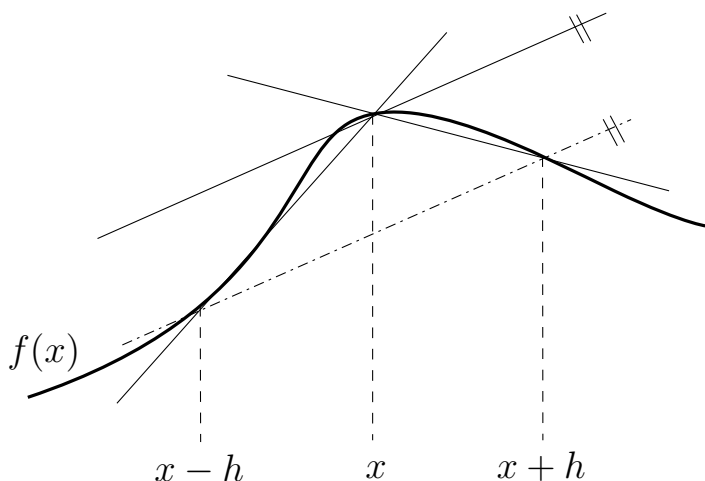
Obdobou výše zmíněného vzorce je *zpětná diference*

$$f'(x) = \frac{f(x) - f(x-h)}{h},$$

kdy derivaci aproximujeme „nahlédnutím o krok zpět“, namísto dopředu, jako u dopředné diference. Pro $h \rightarrow 0$ zpětná a dopředná diference splývají, pro konečné h to ale neplatí, jak je vidět z obrázku 1. Navíc by dle intuice pro co nejlepší aproximaci mělo platit, že neděláme „krok zpět“, ani „krok dopředu“, ale symetricky na obě strany. Takový typ numerické derivace se nazývá *centrovaná diference* a platí pro ni vztah

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

Z obrázku 1 lze nahlédnout, že sečna příslušející centrované diferenci aproximuje tečnu v bodě x daleko lépe než sečny příslušející dopředné a zpětné diferenci. Tuto skutečnost si později dokážeme a podrobně rozebereme.



Obr. 1: Sečny dopředné, zpětné a centrované diference v bodě x .

²Název pochází z toho, že členem $f(x+h)$ „nahlédneme o krok dopředu“.

Taylorův rozvoj

Odbočme nyní a povězme si o mocném matematickém nástroji, který využijeme nejen v tomto díle seriálu. Existuje matematická věta, která říká, že každou hezkou³ funkci můžeme nahradit nekonečným rozvojem

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \dots = \sum_{n=0}^{+\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n,$$

kde a je nějaká hodnota, kolem které chceme rozvoj provést. Ta je volena tak, aby funkční hodnoty f a jejich derivací šly dobře vyčíslit, často se tedy volí $a = 0$. Pak rozvoj nazýváme rozvojem Maclaurinovým.

Taylorův rozvoj má hezkou vlastnost, že pro přibližnou aproximaci nemusíme použít rozvoj celý, ale můžeme jej někde useknout. Pak platí, že čím více členů použijeme, tím je rozvoj přesnější. Také platí, že čím blíže je x ke zvolené konstantě a , tím je obvykle rozvoj přesnější. Konkrétně pro chybu konečného rozvoje

$$f(x) \approx T_N(x) = \sum_{n=0}^N \frac{f^{(n)}(a)}{n!}(x-a)^n$$

platí, že existuje číslo ξ mezi a, x takové, že

$$f(x) - T_N(x) = \frac{f^{(N+1)}(\xi)}{(N+1)!}(x-a)^{N+1}.$$

Pokud si nyní označíme krok $h \stackrel{\text{def}}{=} x - a$ a uvědomíme si, že chyba závisí pouze na mocnině h (faktoriál je konečná konstanta, ξ závisí na h , ale díky spojitosti můžeme $|f^{(N+1)}(\xi)|$ pro dostatečně malé h shora odhadnout konstantou), můžeme použít notaci velkého O , se kterou jsme se seznámili v minulém díle. Konečný rozvoj pak můžeme přepsat do tvaru

$$f(x+h) = \sum_{n=0}^N \frac{f^{(n)}(x)}{n!}h^n + O(h^{N+1}),$$

kde nyní provádíme rozvoj kolem pevného x .

Provedme nyní rozvoj funkce f do druhého řádu a upravme

$$\begin{aligned} f(x+h) &= f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + O(h^3), \\ f'(x) &= \frac{f(x+h) - f(x)}{h} - \frac{1}{2}f''(x)h + O(h^2), \\ f''(x) &= \frac{f(x+h) - f(x)}{h} + O(h), \end{aligned} \tag{1}$$

³Konkrétně funkci, která má všechny derivace spojitě. U konečného rozvoje do řádu N postačuje spojitost derivací do řádu $N+1$.

čímž jsme odvodili vztah pro dopřednou diferenci. Výhodou tohoto postupu je, že vidíme i chybu aproximace. Ta je řádu $O(h)$. Obdobně odvodíme i vztah pro zpětnou diferenci

$$f(x-h) = f(x) - f'(x)h + \frac{1}{2}f''(x)h^2 + O(h^3), \quad (2)$$

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

Vidíme, že chyba metody je opět řádu $O(h)$.

Pokud nyní odečteme rovnici (2) od rovnice (1) dostaneme

$$f(x+h) - f(x-h) = 2f'(x)h + O(h^3),$$

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

což je vzorec pro centrovanou diferenci. Vidíme ale, že chyba je v tomto případě až řádu $O(h^2)$, neboť se členy s nižším řádem odečetly.

Richardsonova extrapolace

Metody vyšších řádů lze odvodit sečtením a odečtením Taylorových rozvoju $f(x \pm nh)$ pro $n = 1, 2, 3, \dots$ tak, aby se všechny členy nízkých řádů (kromě členu s $f'(x)$) odečetly. Ukážeme si zde ale postup známý jako *Richardsonova extrapolace*. Ze vztahu pro centrovanou diferenci máme

$$A(h) = \frac{f(x+h) - f(x-h)}{2h} = f'(x) + Ch^2 + O(h^4),$$

kde $C = f'''(x)/3!$ a uvědomili jsme si, že člen s h^3 je roven nule. Zapišme ten samý vztah pro krok $2h$ (obecně bychom mohli nh)

$$A(2h) = \frac{f(x+2h) - f(x-2h)}{4h} = f'(x) + C(2h)^2 + O(h^4).$$

Nyní chceme odečíst tyto dvě rovnice tak, aby vypadl člen s h^2 . To se nám podaří kombinací

$$A(2h) - 4A(h) = -3f'(x) + O(h^4),$$

$$f'(x) = \frac{-f(x+2h) + f(x-2h) + 8f(x+h) - 8f(x-h)}{12h} + O(h^4).$$

Povedlo se nám tedy z původní centrované difference odstranit chybový člen nejnižšího řádu, musíme za to ale zaplatit znalostí funkční hodnoty ve čtyřech bodech namísto dvou.

Pro obecnou Richardsonovu extrapolaci platí, že pokud máme metodu

$$A(h) = F + Ch^k + O(h^{k+l}),$$

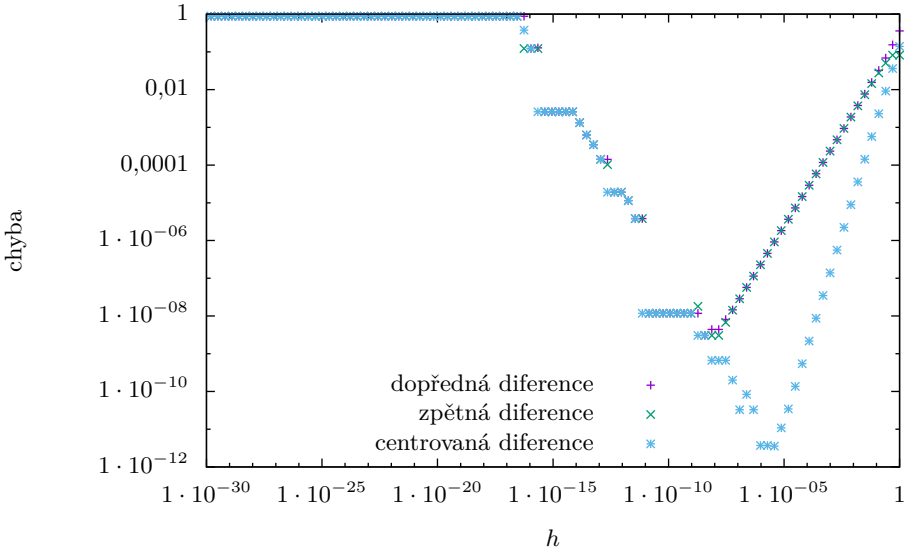
kde F je to, co chceme aproximovat (v našem případě $F = f'(x)$), pak platí vztah

$$F = \frac{n^k A(h) - A(nh)}{n^k - 1} + O(h^{k+l}). \quad (3)$$

Všimněme si, že jsme vzorec (3) nenapsali speciálně pro numerickou derivaci, ale obecněji. Richardsonova extrapolace je totiž obecný nástroj použitelný v různých partiích numerických metod. Například při aplikaci v numerické integraci vede na tzv. Rombergovu integraci nebo při řešení obyčejných diferenciálních rovnic je na ní založena Bulirsch-Stoerova metoda. Nejspíš se s Richardsonovou extrapolací v průběhu seriálu ještě setkáme.

Optimální volba kroku

V některých situacích máme krok pevně daný, například pokud máme už naměřená data z experimentu. Ale v situacích kdy máme možnost volby je vhodné vědět, jaký krok volit, abychom měli co nejmenší (nebo alespoň rozumně velkou) chybu. Z definice derivace, kdy používáme nekonečně malý krok, bychom očekávali, že bude vhodné vždy volit krok co nejmenší. Zde se ale projeví zrádnost numerických výpočtů, především zaokrouhlovací chyby známé z minulého dílu, takže vše bude nakonec úplně jinak.



Obr. 2: Závislost chyby určení derivace na kroku h . Jako testovací funkce byla zvolena $f(x) = \sin x$ v bodě $x = 0,5$.

Vykresleme si nyní závislost chyby určení derivace na velikosti kroku h . Jako testovací funkci si zvolme třeba $f(x) = \sin x$, derivace (přesný výsledek) pak je $f'(x) = \cos x$. Závislost pro dopřednou, zpětnou a centrovanou diferenci je v log-log⁴ grafu 2. Vidíme, že chyba s klesajícím h nejprve skutečně klesá, pro dopřednou a zpětnou diferenci lineárně, pro centrovanou diferenci rychleji – kvadraticky. Poté ale pro určitou hodnotu nastane zlom a chyba opět začne růst, případně se občas zastaví a vytvoří „schod“. Proč se to děje? Zatímco samotná chyba metody s klesajícím h stále klesá, zaokrouhlovací chyba, konkrétně typu cancellation, s klesajícím h roste. Pro určité optimální $h = h_{\text{opt}}$ je chyba metody rovna zaokrouhlovací chybě a pro $h < h_{\text{opt}}$ dominuje zaokrouhlovací chyba.

Dále vidíme, že pro centrovanou diferenci je optimální krok větší, než pro zpětnou a dopřednou diferenci. Je to náznak obecnějšího pravidla: „Čím větší řád metody, tím je optimální krok větší.“ Speciálně platí, že kombinace metody vysokého řádu a velmi malého kroku je cesta do pekla.

⁴Pokud nevíte, proč použít log-log graf pro mocnné závislosti, přečtěte si vzorové řešení úloh z 1. dílu.

Odvodme si nyní teoretickou předpověď pro h_{opt} pro dopřednou diferenci. Hledáme tedy h , pro které bude chyba metody rovna zaokrouhlovací chybě. Předpokládejme, že zaokrouhlovací chyba při vyčíslení funkce je $\Delta f(x) \approx f(x)\varepsilon$, kde ε je strojová přesnost. Pak pro derivaci a její chybu platí

$$f'(x) + \Delta f'(x) = \frac{f(x+h)(1+\varepsilon_1) - f(x)(1+\varepsilon_2)}{h} = \frac{f(x+h) - f(x)}{h} + \frac{f(x+h)\varepsilon_1 - f(x)\varepsilon_2}{h},$$

kde ε_1 a ε_2 jsou skutečné relativní chyby vyčíslení funkce, přičemž dle výše uvedeného platí, že jsou řádově rovny strojové přesnosti. Také poznamenejme, že nám jde o horní odhad chyby, proto v následujícím kroku sečteme všechny dílčí chyby v absolutní hodnotě. Pak pro zaokrouhlovací chybu dopředné diference platí

$$\Delta f'(x) \approx \frac{2\varepsilon|f(x)|}{h}.$$

Všimněme si, že $\Delta f'(x) \propto h^{-1}$. Právě tato vlastnost je důvodem, že pro malá h převáží.

Chyba metody dopředné diference δ je z Taylorova rozvoje rovna

$$\delta = \frac{|f''(\xi)|}{2!} h \approx \frac{|f''(x)|}{2!} h,$$

kde jsme provedli odhad chyby dosazením x místo ξ , protože hodnotu ξ neznáme. Pro $h = h_{\text{opt}}$ pak platí $\delta \approx \Delta f'(x)$, tedy (za předpokladu $f''(x) \neq 0$)

$$\frac{|f''(x)|}{2} h_{\text{opt}} \approx \frac{2\varepsilon|f(x)|}{h_{\text{opt}}},$$

$$h_{\text{opt}} \approx 2\sqrt{\frac{|f(x)|}{|f''(x)|}} \sqrt{\varepsilon}.$$

Protože nás zajímá pouze řádový odhad, platí pro dopřednou diferenci $h_{\text{opt}} \sim \sqrt{\varepsilon} \sim 10^{-8}$, což odpovídá optimální hodnotě v grafu 2. Pro zpětnou diferenci bychom obdrželi stejný výsledek, pro centrovanou diferenci bychom pak obdobným postupem dostali $h_{\text{opt}} \sim \varepsilon^{1/3} \sim 10^{-5}$.

Monte Carlo

Monte Carlo (zkráceně MC) je označení pro jakoukoli metodu, která k vyřešení úlohy používá náhodné vzorkování pravděpodobnostního rozdělení sledované veličiny pomocí generátoru pseudonáhodných čísel. Dle aplikace a použitých algoritmů se Monte Carlo dělí na kinetické, termodynamické (statistické), kvantové a mnoho dalších. V tomto seriálu se budeme postupně věnovat třem oblastem: MC integraci, geometrickému MC a zlehka se dotkneme i termodynamického MC. A aby nebyl výklad problematiky zbytečně komplikovaný, omezíme se zde na rovnoměrné rozdělení a složitější distribuce odsuneme do dalšího dílu.

Principy metody

Princip metody Monte Carlo si ukážeme na konkrétní úloze, která se nazývá *Buffonova jehla*. Původní znění pochází z 18. století od hraběte de Buffon: *Nechť máme vodorovnou plochu rozdělenou rovnoběžnými čarami na pásy šířky a . Na tuto plochu hodíme jehlu délky b . Jaká je pravděpodobnost, že jehla protne čáry?* Vzhledem k době, kdy byla úloha zadána, bylo možné vyřešit ji buď analyticky nebo experimentem, počítačová simulace nebyla dostupná. Naše zadání

bude znít ovšem trochu jinak: Jak na základě opakovaného provedení experimentu uvedeného v Buffonově úloze určíme hodnotu čísla π ?

Možná jste nyní zmatení z toho, odkud se vzalo číslo π . To se ihned ozřejmí, jakmile vyřešíme původní Buffonovu úlohu. Její podstatou hledání tzv. *geometrické pravděpodobnosti* ve dvou dimenzích. Tento typ úloh po nás obecně požaduje porovnat obsah plochy, kterou představují pozitivní výsledky, s obsahem celkové plochy, na které se pohybujeme. Jednoduchým příkladem je nalezení pravděpodobnosti, že trefíme střed terče (bull's eye) za předpokladu, že se vždy trefíme do terče a rozdělení bodů dopadu šipky je rovnoměrné. Hledaná pravděpodobnost je pak dána poměrem obsahu červeného středu a obsahu celého terče.

V Buffonově úloze máme tyto dvě souřadnice: souřadnice x středu jehly ve směru kolmém na rovnoběžné čáry a úhel ϑ natočení jehly v rovině. (Nenechte se zmást, souřadnice y v rovnoběžném směru zde nehraje žádnou roli.) Nyní přidáme dodatečnou podmínku⁵ $b < a$, tedy délka jehly musí být menší než šířka pásů. Potom pravděpodobnost, že poloha středu x bude ve vzdálenosti menší než $b/2$ od čáry (a tedy čára bude v dosahu jehly), udává vztah

$$P_1 = \frac{1}{a} \left(\frac{b}{2} + \frac{b}{2} \right) = \frac{b}{a}. \quad (4)$$

Nyní nalezneme pravděpodobnost P_2 , že jehla protne rovnoběžku, ke které je nejbližší, pokud je tato rovnoběžka v dosahu jehly – všimněte si slova „pokud“, jde o podmíněnou pravděpodobnost. Pravděpodobnost, že jehla protne rovnoběžku, bude díky nezávislosti souřadnic dána součinem $P_1 \cdot P_2$. Aby jehla ve vzdálenosti $x \leq b/2$ od rovnoběžky tuto rovnoběžku prořezala, musí platit $x \leq \frac{b}{2} \cos|\vartheta|$. Toto si rozmyslíme z pravoúhlého trojúhelníku s přeponou $b/2$ a přilehlou odvěsnou x . Inverzně, úhel ϑ (v intervalu od $-\pi/2$ do $\pi/2$, měřený od kolmice na rovnoběžku) leží v intervalu $\langle \vartheta_{\min}, \vartheta_{\max} \rangle = \langle -\arccos(2x/b), \arccos(2x/b) \rangle$. Pro dané x bychom nyní snadno určili hledanou pravděpodobnost jako poměr velikostí uvedeného intervalu s úhlem π (v radiánech). Jelikož jsou však ϑ_{\min} a ϑ_{\max} závislé na x , musíme provést integraci přes x od 0 do $b/2$:

$$P_2 = \frac{2}{b} \int_0^{b/2} \frac{2 \arccos(2x/b)}{\pi} dx = \frac{2}{\pi}. \quad (5)$$

Při výpočtu jsme využili tabulkový integrál

$$\int \arccos x dx = x \arccos x - \sqrt{1-x^2} + C. \quad (6)$$

Řešení Buffonovy úlohy tedy zní: Jehla protne rovnoběžku s pravděpodobností $P = P_1 P_2 = \frac{2b}{a\pi}$.

Přímější, ale méně názorný postup řešení by vypadal následovně: pravděpodobnostní funkce souřadnic x (vzdálenost od nejbližší čáry) a ϑ jsou

$$f_x(x) = \begin{cases} \frac{2}{a} & \text{pro } x \in \langle 0, a/2 \rangle \\ 0 & \text{jinak} \end{cases}$$

a

$$f_\vartheta(\vartheta) = \begin{cases} \frac{1}{\pi} & \text{pro } \vartheta \in \langle -\pi/2, \pi/2 \rangle \\ 0 & \text{jinak.} \end{cases}$$

⁵Řešení úlohy bez této podmínky je také možné, ale obsahuje trochu komplikovanější integrál a metoda výpočtu π pak již není tak zřejmá.

Souřadnice x a ϑ jsou nezávislé náhodné proměnné. V takovém případě můžeme sdruženou pravděpodobností funkci $P(x, \vartheta)$ vyjádřit jako součin jednorozměrných pravděpodobnostních funkcí

$$f_{x,\vartheta}(x, \vartheta) = f_x(x)f_\vartheta(\vartheta) = \begin{cases} \frac{2}{a\pi} & \text{pro } x \in \langle 0, a/2 \rangle \wedge \vartheta \in \langle -\pi/2, \pi/2 \rangle \\ 0 & \text{jinak .} \end{cases} \quad (7)$$

Výše jsme si rozmysleli, že k protnutí rovnoběžky dojde pro $x \leq \frac{b}{2} \cos |\vartheta|$. Hledanou pravděpodobnost tedy získáme jako dvojný integrál

$$P = \int_{-\pi/2}^{\pi/2} \int_0^{\frac{b}{2} \cos |\vartheta|} \frac{2}{a\pi} dx d\vartheta = \int_{-\pi/2}^{\pi/2} \frac{b \cos |\vartheta|}{a\pi} d\vartheta = \frac{2b}{a\pi}. \quad (8)$$

Nabízí se tedy následující metoda, jak určit hodnotu čísla π : Obstaráme si velké množství jehel (pro uskutečnění experimentu jsou vhodnější špejle) a postupně⁶ je budeme házet na pružovanou plochu a počítat, kolikrát došlo k protnutí některé z rovnoběžek. Při házení musíme dbát na to, aby byl pohyb jehly vzduchem dostatečně chaotický a výslednou konfiguraci jehly bylo možné považovat za náhodnou. Nakonec vydělíme počet protnutí počtem jehel a dostaneme určité číslo c . Podle zákona velkých čísel pro n hodů platí

$$\lim_{n \rightarrow \infty} c = P, \quad (9)$$

tj. v limitě velkého počtu hodů vyjádříme číslo π vztahem

$$\pi = \frac{2b}{ac}. \quad (10)$$

Tato metoda k nalezení čísla π spadá do rodiny Monte Carlo. Před rozmachem MC simulací se opakované pokusy používaly pouze k odhadu nejistoty⁷ výsledku získaného deterministickou simulací, například numerickou integrací diferenciálních rovnic. Metoda Monte Carlo přímo řeší danou úlohu na základě pravděpodobnostního přístupu.

Shrňme si nyní v obecnosti *základní kroky metody Monte Carlo*:

- Stanovíme množinu všech možných vstupních hodnot. V případě Buffonovy jehly jde o všechny dvojice hodnot, kterých můžou nabývat proměnné x a ϑ .
- Zjistíme, jakými pravděpodobnostními funkcemi se řídí náhodné veličiny vstupující do simulace. V probrané úloze šlo o dvě rovnoměrná rozdělení pro veličiny x a ϑ .
- Pomocí generátoru (pseudo)náhodných čísel vybíráme realizace zmíněných náhodných veličin. V experimentální formě Buffonovy jehly je náhodnost zaručena chaotickým letem jehly vzduchem.
- Na vybrané realizace aplikujeme deterministický vzorec a získáme výstupní hodnoty simulace.

Poslední bod zní velmi obecně, neboť se podstatně liší problém od problému – při zkoumání pohybu částic v plynu (termodynamické MC) uplatníme zcela jiné vztahy než při výpočtu integrálů. V Buffonově úloze je výstupní hodnotou odhad čísla π . Tato hodnota však nikdy nebude přesná, protože počet hodů jehlou/špejlí je konečný. Rádi bychom proto kvantifikovali chybu, které se při výpočtu dopouštíme. Bohužel, ne každá aplikace MC umožňuje snadno definovat, jak určíme nejistotu výsledku. Jak je tomu v případě Buffonovy úlohy?

⁶Nebo všechny naráz, musíme však zaručit, že srážky jehel nenaruší náhodnost.

⁷Připomeňme, že nejistotou myslíme veličinu charakterizující distribuci výsledku okolo očekávané hodnoty. Více viz následující kapitola.

Monte Carlo integrace

Než zodpovíme otázku položenou na konci minulé kapitoly, podívejme se na Buffonovu úlohu obecněji. Při aplikaci Monte Carlo metody vlastně řešíme za pomoci simulace integrál (5), respektive (8). Řešení Buffonovy úlohy tedy spadá do specifické skupiny metod zvaných *Monte Carlo integrace*. Obecně tedy hledáme hodnotu integrálu

$$I = \int_{\Omega} f(x) dx, \quad (11)$$

kde $f(x)$ je známá funkce a Ω je oblast, přes kterou integrujeme. Jelikož nepředpokládáme, že každý čtenář je dobře obeznámen s integrálním počtem, krátce si zde vysvětlíme, jak pojem integrálu chápat.

Smyslem integrálního počtu je zobecnit pojmy jako je délka, plocha či objem. Pokud ve výrazu (11) položíme $f(x) = 1$, pak I bude udávat objem oblasti Ω . Slovo „objem“ může být poněkud zavádějící, protože pokud je oblastí například dvourozměrná plocha vytyčená body $(0, 0)$, $(2, 0)$, $(2, 3)$ a $(0, 3)$, bude integrál představovat obsah dané plochy (zde obdélník o obsahu 6) – často však používáme termín objem pro oblasti s libovolnou dimenzí (obsah je tedy „2D objem“). Pokud je tvar oblasti komplikovaný, může být vyčíslení integrálu také komplikované. Vždy je však možné aplikovat numerický přístup: rozřežeme oblast na velký počet krychliček a sečteme jejich objem. Čím jemnější dělení použijeme, tím přesnějšího výsledku dosáhneme.

Z numerického pohledu se problém integrace příliš nezmění ani když budeme pracovat s libovolnou funkcí $f(x)$. Pouze budeme muset při sčítání krychliček každý elementární objem vynásobit vahou danou hodnotou funkce $f(x)$ ve středu krychličky.⁸

Teď když máme hrubou představu, jak může takový numerický výpočet integrálu vypadat, budeme schopni lépe ocenit přínos metody Monte Carlo. Prvním krokem je uzavření obecně komplikované množiny Ω do boxu. Tento box představuje definiční obor pravděpodobnostní funkce $p(x)$ od náhodné veličiny X . Stále pro jednoduchost uvažujeme, že X má rovnoměrné rozdělení, $p(x)$ je tedy konstantní funkce, která všude nabývá hodnoty $1/V$, kde V je objem boxu (vzpomeňte, že pravděpodobnostní funkce je normovaná, takže integrál funkce $p(x)$ přes celý box musí být roven 1). Navíc předpokládejme, že mimo oblast Ω je funkce $f(x)$ nulová. Pomocí generátoru náhodných čísel vybereme jednu realizaci x veličiny X a zkonstruujeme odhad integrálu

$$I_1(X = x) = \frac{f(x)}{p(x)}. \quad (12)$$

Očekáváme, že jednobodový odhad bude velmi špatný. Uvažujme na chvíli případ, kdy počítáme objem oblasti Ω (označme ho O), tj. $f(x) = 1$ všude na Ω . Pokud $x \in \Omega$, pak $I_1 = V$. Pokud $x \notin \Omega$, pak $I_1 = 0$. Hledaná hodnota zřejmě leží někde mezi 0 a V , ale nevíme kde. Dobrou vlastností tohoto odhadu je tzv. nestrannost. To znamená, že jeho střední hodnota je rovna přesnému výsledku. To si snadno potvrdíme aplikací definice střední hodnoty spojité náhodné veličiny

$$E[I_1] = \int_{\Omega} I_1(X = x)p(x)dx = \int_{\Omega} \frac{f(x)}{p(x)}p(x)dx = \int_{\Omega} f(x)dx = O; \quad (13)$$

⁸Nemusí jít o střed a nemusí jít o krychličky, ale zatím s touto představou vystačíme. Fundovanější vysvětlení numerické integrace těmito metodami naleznete v dalších dílech seriálu.

tento výsledek platí pro libovolné $f(x)$. Pro výpočet rozptylu použijeme vztah⁹

$$\text{Var} [I_1] = \text{E} [I_1^2] - (\text{E} [I_1])^2, \quad (14)$$

tedy pro $f(x) = 1$ a $p(x) = 1/V$ dostaneme

$$\text{Var} [I_1] = \int_{\Omega} \left(\frac{f(x)}{p(x)} \right)^2 p(x) dx - O^2 = \int_{\Omega} V dx - O^2 = VO - O^2 = O(V - O). \quad (15)$$

Vidíme tedy, že rozptyl našeho odhadu je přímo úměrný rozdílu mezi objemem obklopujícího boxu a oblasti zájmu Ω . Kdybychom volili box shodný s Ω , takže by platilo $O = V$, dostali bychom exaktní výsledek, ale pak celá úloha postrádá smysl, protože hledaný objem již od začátku známe. Jak tedy docílíme nižšího rozptylu (a v důsledku nižší směrodatné odchylky), aniž bychom museli omezit naši benevolenci při volbě boxu?

Odhad I_1 je špatný, nicméně z geometrického pohledu víme, že pravděpodobnost zvolení $x \in \Omega$ lze vyjádřit jako O/V . Tedy způsobem, jak odhad zlepšit, je vzít velký počet nezávislých vzorků, označme ho N , a provést průměr přes hodnoty $I_1(X = x_i)$, $i = 1, \dots, N$. Novým odhadem je tedy veličina

$$I_N = \frac{1}{N} \sum_{i=1}^N \frac{f(x_i)}{p(x_i)}. \quad (16)$$

Rozptyl této veličiny je¹⁰

$$\begin{aligned} \text{Var} [I_N] &= \text{Var} \left[\frac{1}{N} \sum_{i=1}^N \frac{f(x_i)}{p(x_i)} \right] = \frac{1}{N^2} \left(\text{Var} \left[\frac{f(x_1)}{p(x_1)} \right] + \text{Var} \left[\frac{f(x_2)}{p(x_2)} \right] + \dots \right) = \\ &= \frac{1}{N} \text{Var} [I_1] = \frac{1}{N} O(V - O), \end{aligned} \quad (17)$$

přičemž poslední úprava je opět pouze pro $f(x) = 1$. Pro směrodatnou odchylku tedy platí

$$\sigma [I_N] = \frac{1}{\sqrt{N}} \sqrt{O(V - O)}. \quad (18)$$

Výraz $\sqrt{O(V - O)}$ je nějaký konstantní faktor daný volbou boxu a není příliš podstatný. Zásadní význam má ovšem faktor $1/\sqrt{N}$. Říká nám, že s rostoucím počtem vzorků klesá směrodatná odchylka s odmocninou počtu vzorků. Tedy zvýšíme-li počet vzorků stokrát, sníží se chyba desetkrát. To možná nezní jako příliš rychlá konvergence, ale při výpočtu mnohorozměrných integrálů se ukazuje být velice dobrou. Lze ukázat, že v dimenzi d konverguje výpočet integrálu pomocí dělení objemu na krychlové elementární buňky rychlostí $N^{-2/d}$, zatímco integrace Monte Carlo není na dimenzi závislá. Pro dimenze $d > 4$ je tedy MC integrace preferovanou volbou.

Krátká poznámka k vyšším dimenzím – výpočet objemů geometrických těles v mnohorozměrných Eukleidových prostorech nemá mimo oblast čisté matematiky velký význam. Pokud však budeme každý

⁹V předchozím dílu seriálu nebyl zmíněn, ale lze ho odvodit z definice střední hodnoty a rozptylu.

¹⁰To, že rozptyl součtu můžeme rozdělit na součet rozptylů, plyne z nezávislosti veličin $I_1(X = x_i)$. Můžete si to taky odvodit.

volný parametr fyzikálního systému považovat za jednu dimenzi, nabízí se široká škála aplikací. Jako příklad uveďme kanonický soubor částic ideálního plynu ve statistické mechanice, kde každá z N částic plynu je popsána třemi složkami hybnosti a třemi složkami polohy. Integrací přes těchto $6N$ proměnných získáme tzv. partiční sumu. Ta v sobě obsahuje termodynamické vlastnosti systému jako jsou vnitřní energie, entropie či tlak.

Odvozený vztah pro výpočet nejistoty odhadu integrálu za použití metody Monte Carlo můžeme aplikovat na Buffonovu jehlu. Pomocí generátoru náhodných čísel vybíráme N dvojic vzorků z rovnoměrných rozdělení $x \in \langle 0, a/2 \rangle$, $\vartheta \in \langle -\pi/2, \pi/2 \rangle$ a k výsledku přičteme $1/N$ pokaždé, když je splněna podmínka $x \leq \frac{b}{2} \cos |\vartheta|$. Objemem boxu je zde

$$V = (a/2 - 0)(\pi/2 - (-\pi/2)) = \frac{a\pi}{2}, \quad (19)$$

přičemž hledaným výsledkem je $O/V = \frac{2b}{a\pi}$, a tedy $O = b$. Směrodatná odchylka výsledku se tedy chová jako

$$\sigma = \frac{1}{\sqrt{N}} \sqrt{O(V - O)} = \frac{1}{\sqrt{N}} \sqrt{\frac{ab\pi}{2} - b^2}. \quad (20)$$

Zvolíme-li konkrétní hodnoty parametrů, např. $b = a = 1$, dostaneme směrodatnou odchylku

$$\sigma = 0,76 \frac{1}{\sqrt{N}}. \quad (21)$$

Poznamenejme ještě, že využít počítačovou simulaci Buffonovy úlohy k nalezení hodnoty čísla π není nejpraktičtější, protože při generování náhodných čísel potřebujeme číslo π již předem znát.

Existují metody, jak urychlit snižování rozptylu, ale jejich rozbor ponecháme do příštího dílu. Jednu speciální metodu přeci jen ve stručnosti uvedeme – nazývá se kvazi-Monte Carlo (qMC) a od běžné MC metody se liší tím, že nepoužívá pseudonáhodná čísla, ale tzv. kvazináhodná čísla (odborněji se mluví o posloupnostech s nízkou diskrepancí). Výhodou qMC je, že zatímco při generování náhodných čísel dochází k vytváření shluků, kvazináhodná čísla pokryjí zvolenou oblast rovnoměrněji. Nemůže se tedy stát, že bychom několikrát započítali stejný bod (nebo velmi blízké body). Zároveň však kvazináhodná čísla nejsou rozmístěna v pravidelné mříži, při změně oblasti tedy není potřeba polohy bodů přepočítat. Řád konvergence směrodatné odchylky se díky rovnoměrnému pokrytí může blížit až $1/N$. Za tento bonus však musíme zaplatit – kvazináhodná čísla nejsou nezávislá, jejich využití je proto omezeno (není možné je použít například v kryptografii) a statistické zpracování se kvůli korelacím komplikuje.

Ukázková úloha: Integrace kruhu

Zadání: Vypočtete obsah čtvrtkruhu o poloměru $r = 1$ pomocí metody Monte Carlo.

Řešení: Čtvrtkruh vložíme do čtverce o straně délky r (tj. poloměry čtvrtkruhu jsou shodné s dvěma přilehlými stranami čtverce) a budeme generovat dvojice rovnoměrně rozdělených náhodných čísel na intervalu $\langle 0, 1 \rangle$. Dvojice čísel představují souřadnice x, y bodů uvnitř čtverce, celkem jich vygenerujeme N . Poté spočteme všechny body, které splňují podmínku $x^2 + y^2 \leq 1$, tj. leží uvnitř čtvrtkruhu; tento počet označme M . Obsah je poté dán vztahem

$$S = \frac{M}{N} d^2 = \frac{M}{N}.$$

Jelikož známe vzoreček pro obsah kruhu, očekáváme výsledek

$$M/N = \pi/4 \pm 0,41 \frac{1}{\sqrt{N}},$$

kde jsme faktor 0,41 získali ze vztahu (18) pro $V = r^2 = 1$ a $O = \pi r^2/4 = \pi/4$.

K řešení byl použit následující kód, který můžete s mírnými úpravami využít v seriálové úloze. Uvnitř kódu je pomocí komentářů vysvětlena potřebná syntax jazyka Python.¹¹

```
# nacteme matematickou a numerickou knihovnu
import math
import numpy as np

# kolik bodu budeme generovat
pocet = 1000000
# pole (matice) nahodnych cisel od 0 do 1,
# ktera ma "pocet" radku a 2 sloupce, pricemz kazdy
# radek predstavuje souradnice x a y jednoho bodu
nahodna = np.random.rand(pocet,2)
# inicializace promenne, do niz budeme ukladat pocet zasahu dovnitr kruhu
pocet_uvnitr = 0

# cyklus, ktery prochazi cisla od 0 do pocet-1
for i in range(0,pocet):
    # v hranate zavorce piseme radkovy a sloupcovy index oddeleny carkou,
    # pomoci Pythagorovy vety kontrolujeme, ze bod lezi v kruhu
    if nahodna[i,0]**2 + nahodna[i,1]**2 < 1:
        # pokud je bod v kruhu, zvysime pocet zasahu o jedna
        pocet_uvnitr += 1

# vypocet prumeru; ve jmenovateli je prictena nula, aby byl vysledek float
prumer = pocet_uvnitr/(pocet+0.)
# odchylka od ocekavaneho vysledku a smerodatna odchylka dle textu serialu
# math.pi vola hodnota cisla pi
odchylka = np.abs(prumer - math.pi/4.)
smodch = np.sqrt(math.pi/4.*(1. - math.pi/4.)/pocet)

# nechame vypsats vysledek na obrazovku
print("Prumer: {}, smodch: {}, odchylka: {}".format(prumer, smodch, odchylka))
```

Výsledek prvního běhu programu vypadal následovně:

Prumer: 0.785928, smodch: 0.000410545841934, odchylka: 0.000529836602552.

Jelikož jsme použili 1 000 000 bodů, pohybuje se relativní směrodatná odchylka okolo 1/1000 (odmocnina). Na obrázku 3 vidíte, jak se vyvíjí odhad průměru s rostoucím N .

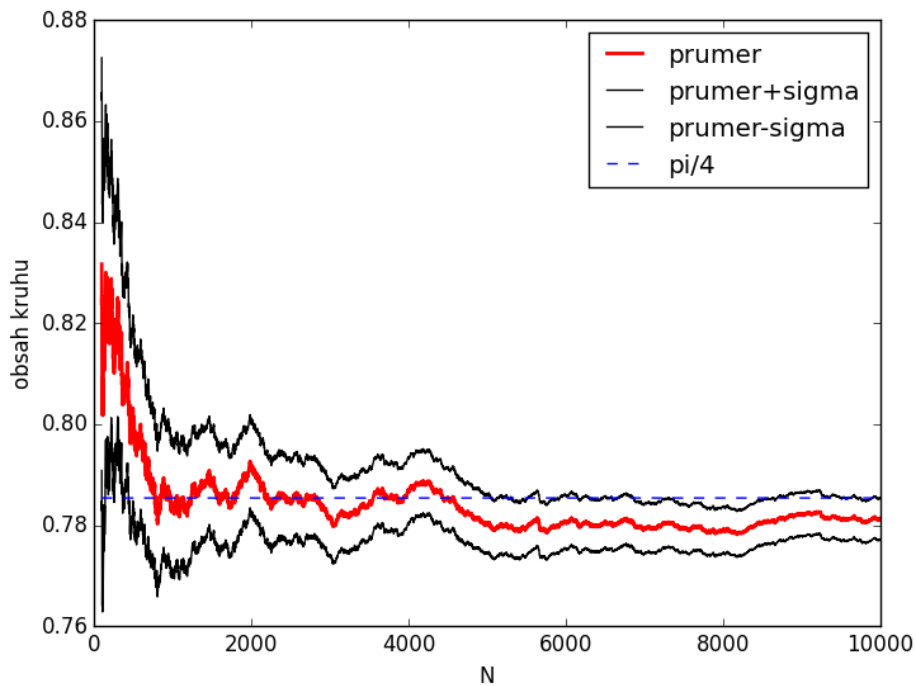
Úlohu lze jednoduše vyřešit i v programu Microsoft Excel. Pomocí funkce RAND() naplníme dva sloupce do délky N . Volme třeba $N = 1\,000$ a naplněny necht' jsou sloupce A a B. Do buňky C1 vepíšeme vzorec =A12+B12 a vytáhneme do délky N . Poté do prázdné buňky vepíšeme =COUNTIF(C1:C1000;"<1")/COUNT(C1:C1000)

První funkce spočte všechny buňky obsahující číslo větší než 1, druhá funkce započte všechny neprázdné buňky. Tak získáme odhad obsahu kruhu s průměrem $d = 1$.

Fyzikální korespondenční seminář je organizován studenty MFF UK. Je zastřešen Oddělením pro vnější vztahy a propagaci MFF UK a podporován Ústavem teoretické fyziky MFF UK, jeho zaměstnanci a Jednotou českých matematiků a fyziků.

Toto dílo je šířeno pod licencí Creative Commons Attribution-Share Alike 3.0 Unported.
Pro zobrazení kopie této licence navštivte <http://creativecommons.org/licenses/by-sa/3.0/>.

¹¹Kód je psán tak, aby byl srozumitelný, nikoli efektivní.



Obr. 3: Vývoj obsahu kruhu odhadnutého metodou Monte Carlo v závislosti na počtu bodů N . Prvních sto kroků je v grafu vynecháno kvůli velkým fluktuacím.