

CALCUL NUMERIC

Dansând cu numere și jucându-ne cu computerul

RAFAEL PLA-LÓPEZ

Traducere de Elena-Anca Coman

CUPRINS

Capitolul 0

Aproximând (Stanislaw Ulam și metoda Monte Carlo) 7

Generare de numere pseudoaleatorii și operatori Monte Carlo 12

Capitolul 1

Ghicind curba 17

Dilatarea relativistă a timpului 46

Domeniul de aplicare al tehnicii relativiste 48

Capitolul 2

Estimând totalul 51

Capitolul 3

Urmând drumul 73

Capitolul 4

Buscando el punto 111

Catastrofă 115

Explorator newtonian 119

Un sport riscant 120

Călătorind în timp? 124

Rezolvarea newtoniană a unei probleme relativiste 130

Într-un univers alternativ 133

Anexă 135

Bibliografie recomandată 143

Capitolul 0

Aproximând (Stanislaw Ulam și metoda Monte Carlo)

Este foarte posibil ca tinerii de azi să nu-și amintească, dar a existat o vreme când matematica se făcea doar cu creionul pe hârtie. De fapt, au fost dezvoltate metode puternice de calcul diferențial și integral pentru a rezolva probleme complexe care necesitau calcule laborioase. Majorității i se părea cu adevărat plictisitor. Fără îndoială, unora dintre noi ne plăcea acea matematică datorită rigorii ei deductive și a capacității de a obține rezultate, dar mulți simțeau aversiune față de ea. Această matematică permitea un acces la adevăruri absolute, la care se ajungea prin raționamente pur formale, independente de experiență. Și matematicienii se lăudau cu aceasta și-și numeau disciplina „știință exactă”.

Acest lucru s-a accentuat odată cu apariția a ceea ce s-a numit „matematică modernă”, bazată pe teoria mulțimilor, care l-a făcut pe **David Hilbert** (1845-1918) să declare că „Nimeni nu ne va putea izgoni vreodată din paradisul pe care **Georg Cantor** (1845-1918) l-a creat pentru noi”. Dar tocmai acel caracter „din altă lume” i-a făcut pe mulți să vadă matematica drept ceva foarte îndepărtat și străin de viețile lor. Și, în plus, exista un obstacol: deși metodele sale erau puternice, existau multe probleme care nu puteau fi rezolvate. Mai bine spus, dacă scriam la întâmplare o funcție complexă și încercam

să-i calculăm integrala, cel mai probabil ar fi fost imposibil să-i obținem rezultatul exact. Cu alte cuvinte, chinurile pe care le-au suferit mulți studenți pentru a rezolva integrale erau împărtășite și de matematicieni experimentați. Și nu trebuia mers prea departe. De exemplu, dacă studiem mișcarea a două corpuri care sunt atrase de forța gravitațională conform legii lui Newton, este relativ ușor să obținem traiectoriile lor, care pot fi eliptice, parabolice sau hiperbolice. Totuși, dacă introducem un al treilea corp, nu este posibil să obținem exact traiectoriile lor în mod general.

Matematicienii din secolele trecute erau conștienți de acest lucru și au dezvoltat metode pentru a obține soluții aproximative. Unele dintre aceste metode i le datorăm lui însuși **Isaac Newton** (1642-1717). Cu toate acestea, problema cu astfel de metode era că necesitau un număr mare de calcule, care, deși erau simple, luate unul câte unul, în ansamblu erau costisitoare și obositoare. În plus, nu erau potrivite pentru a-i seduce pe cei care deja simțeau aversiune față de formule complexe. Să ne gândim, de exemplu, la calculul ariei suprafeței delimitate de o curbă și axa absciselor. Știm că integrala funcției care definește această curbă ne oferă acea arie. Dar, dacă nu știm să rezolvăm integrala, putem oricând să împărțim suprafața respectivă în pătrate mici (ceea ce se numește „mozaicare”) și să le numărăm. Totuși, pentru a obține o aproximare bună ar trebui să avem pătrate foarte mici, prin urmare, ar fi foarte multe și ne-am plictisi numărându-le.

În anii '40, oamenii de știință care încercau să dezvolte o bombă atomică în Laboratorul Național din Los Álamos, Statele Unite, au întâmpinat una dintre acele probleme aparent de nerezolvat pentru a calcula difuzia electronilor în materialul din fisiune, la fel cum era rezolvarea complexelor ecuații integro-diferențiale care guvernau dispersia și absorbția neutronilor. Conform anecdotei narate de unul dintre matematicienii care făceau parte din echipă, lui **Stanislaw Ulam** (1909-1984), în 1946, i-a trecut prin minte în timp ce juca Solitaire, în loc să facă o analiză combinatorică complicată pentru a deduce

mișcările câștigătoare, putea face mai multe teste cu cărțile și să numere proporția rezultatelor obținute. Și a crezut că poate folosi aceeași metodă pentru a aborda problema la care erau blocați. Această metodă era ceea ce se numea „eșantionare statistică“, ce poate fi folosită, de exemplu, pentru a calcula aria menționată anterior: plasând respectiva suprafață într-un dreptunghi, am putea să generăm aleatoriu puncte (de exemplu, folosind zaruri sau o ruletă) și să numărăm câte rămân înăuntru și câte în afara acestuia, astfel încât proporția dintre ele ne-ar oferi o estimare a dimensiunii sale. Dar, pentru ca estimarea să fie fiabilă, am avea nevoie de un eșantion cu un număr mare de puncte, astfel că ne-am plictisi aruncând zarurile sau învârtind roata. Totuși, la acea vreme aveau deja un calculator primitiv, ENIAC, o mașină uriașă care era programată prin conectare de cabluri, dar care permitea efectuarea relativ rapidă a unui număr mare de calcule, folosind, printre altele, un generator de numere aleatorii.

Să presupunem, de exemplu, că se dorește utilizarea acestei metode pentru a calcula aria din figura 1, sub o curbă $y = f(x)$. Primul lucru ar fi să putem genera aleatoriu puncte în pătratul de dimensiunea 20×20 în interiorul căruia se află suprafața.

Există mai multe moduri de a face acest lucru. De exemplu, am putea avea un limbaj de programare care ar putea genera aleatoriu un număr între 0 și 1. Astfel încât, dacă funcția care-l generează s-ar numi „random ()“, ar fi suficient să luăm $x = 20 \cdot \text{random}()$ și $y = 20 \cdot \text{random}()$. Desigur, de fiecare dată când s-ar executa random () ar da, în principiu, un număr diferit, deoarece aceasta este frumusețea generării de numere aleatorii. În felul acesta, am putea obține N puncte și am putea folosi un contor M care să crească într-o unitate de fiecare dată când ar întâlni un punct (x, y) care să îndeplinească $y = f(x)$. Evaluând ca M/N probabilitatea ca punctul să se găsească în interiorul figurii, am putea estima valoarea ariei sale ca:

$$20 \cdot 20 \cdot \frac{M}{N} = \frac{400 \cdot M}{N}.$$

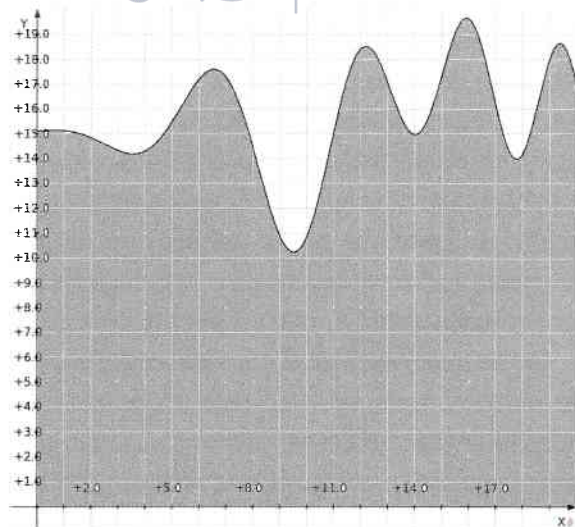


Figura 1.

Totuși, eroarea folosind N puncte este estimată ca:

$$\frac{1}{\sqrt{N}}$$

conform așa-numitei legi a numerelor mari. Astfel că, de exemplu, dacă 2 din 4 puncte ar fi găsite în interiorul figurii și 2 în afara ei, așa cum se arată în grafic, am avea ca estimare a ariei:

$$400 \cdot \left(\frac{2}{4} \pm \frac{1}{\sqrt{4}} \right) = 400 \cdot (0,5 \pm 0,5) = 200 \pm 200.$$

Și presupun că vă veți întreba: la ce folosește aceasta? Răspunsul este: la nimic. Pentru a obține o estimare adecvată ar trebui să generăm multe puncte. Dar, dacă am vrea să o facem cu creionul pe hârtie, de exemplu, aruncând un zar sau învârtind o ruletă, nu numai că ne-am plictisi, dar ne-ar obosi și mâna. Din acest motiv, eșantionarea statistică este utilă doar folosind un computer.

În felul acesta, generând un milion de puncte am estima eroarea în:

$$\frac{400}{\sqrt{1000000}} = \frac{400}{1000} = 0,4$$

și am putea obține o estimare pentru aria suprafeței după, să zicem, $312,8 \pm 0,4$. Și, de fiecare dată când am înmulți cu 100 numărul de puncte, am putea crește precizia cu o zecimală.

Metoda de eșantionare statistică este foarte flexibilă. Ar putea fi folosită, de exemplu, pentru a estima un volum prin aproximarea unei integrale multiple. Sau pentru a se apropia de soluția ecuațiilor integro-diferențiale care le furau somnul lui Stanislaw Ulam și colegilor săi, încăpățânați să controleze dispersia și absorbția neutronilor pentru a obține o bombă atomică.

Se spune că drumul spre iad este plin de intenții bune. Totuși, și intențiile rele (în acest caz, construcția unei arme de distrugere în masă) pot avea, uneori, ca rezultat colateral progresul cunoașterii umane. Cu toate acestea, **Albert Einstein** (1879-1955) și-a exprimat reticența față de mecanica cuantică spunând că „Dumnezeu nu joacă zaruri cu universul”, așa că ne putem imagina ce ar crede despre zarurile jucate pentru a construi o armă de distrugere în masă, inspirată de teoriile sale.

Stanislaw Ulam părea să aibă și simțul umorului, ceea ce l-a determinat să-și numească metoda „Monte Carlo”, cu referire la celebrul cazinou din Monaco. Și, având în vedere că orice computer mic de astăzi are mult mai multă putere de calcul decât computerele enorme primitive de la mijlocul secolului trecut, vă puteți imagina ce se poate face: folosind tehnici de analiză numerică, bazate pe realizarea masivă a unor calcule simple, putem lăsa repetarea plictisitoare a acestora pe seama acestor mașini, rezervând ființelor umane sarcina creativă și potențial distractivă de a le programa.

Continuare la p. 14

GENERARE DE NUMERE PSEUDOALEATORII ȘI OPERATORI MONTE CARLO

Deși am tot vorbit despre generarea de numere „aleatorii”, pentru a evita și mai mult confuzia, adevărul este că, dacă nu folosim calculul cuantic, numerele generate nu vor fi cu adevărat aleatorii, ci doar așa vor părea. Din acest motiv, dacă suntem riguroși, denumirea lor corectă ar fi „numere pseudoaleatorii”.

O modalitate simplă de a le obține sunt așa-numitele *generatoare congruențiale liniare*, care constau în a porni de la un număr inițial x_0 (valoare inițială), a-l înmulți cu un număr întreg pozitiv a (multiplicator), a-i aduna un alt număr întreg pozitiv b (increment), a-l împărți la un al treilea număr întreg pozitiv m (față de care calculăm restul) și a lua restul împărțirii întregi ca următorul număr din succesiune.

Putem exprima acest lucru ca: $x_n = (a \cdot x_{n-1} + b) \text{ modulo } m$.

De exemplu, dacă luăm $x_0 = 7, a = 5, b = 3, m = 6$, vom obține următoarea succesiune:

$$\begin{aligned}
 &7 \\
 5 \cdot 7 + 3 \text{ modulo } 16 &= 38 \text{ modulo } 16 = 6 \\
 5 \cdot 7 + 3 \text{ modulo } 16 &= 38 \text{ modulo } 16 = 6 \\
 5 \cdot 6 + 3 \text{ modulo } 16 &= 33 \text{ modulo } 16 = 1 \\
 5 \cdot 1 + 3 \text{ modulo } 16 &= 8 \text{ modulo } 16 = 8 \\
 5 \cdot 8 + 3 \text{ modulo } 16 &= 43 \text{ modulo } 16 = 11 \\
 5 \cdot 11 + 3 \text{ modulo } 16 &= 58 \text{ modulo } 16 = 10 \\
 5 \cdot 10 + 3 \text{ modulo } 16 &= 53 \text{ modulo } 16 = 5 \\
 5 \cdot 5 + 3 \text{ modulo } 16 &= 28 \text{ modulo } 16 = 12 \\
 5 \cdot 12 + 3 \text{ modulo } 16 &= 63 \text{ modulo } 16 = 15 \\
 5 \cdot 15 + 3 \text{ modulo } 16 &= 78 \text{ modulo } 16 = 14 \\
 5 \cdot 14 + 3 \text{ modulo } 16 &= 73 \text{ modulo } 16 = 9 \\
 5 \cdot 9 + 3 \text{ modulo } 16 &= 48 \text{ modulo } 16 = 0 \\
 5 \cdot 0 + 3 \text{ modulo } 16 &= 3 \text{ modulo } 16 = 3 \\
 5 \cdot 3 + 3 \text{ modulo } 16 &= 18 \text{ modulo } 16 = 2 \\
 5 \cdot 2 + 3 \text{ modulo } 16 &= 13 \text{ modulo } 16 = 13 \\
 5 \cdot 13 + 3 \text{ modulo } 16 &= 68 \text{ modulo } 16 = 4,
 \end{aligned}$$

apoi numerele s-ar repeta, după ce s-au obținut toate cele 16 resturi posibile, de la 0 la 15, rezultate din împărțirea la 16, motiv pentru care acest generator se numește *complet*.

Desigur, dacă dorim să obținem șiruri mult mai lungi de numere „pseudoaleatorii“, care să pară aleatorii, fără să se repete ciclic, va trebui să luăm o valoare a împărțitorului m mult mai mare, alegând multiplicatorul a și incrementul b în mod adecvat pentru ca numerele să nu se repete prematur. Iar dacă dorim să obținem numere între 0 și 1, va fi suficient să le împărțim ca numere raționale la m .

Pe de altă parte, și având în vedere că farmecul metodelor de analiză numerică este de a începe cu operații simple efectuate de mai multe ori, ar fi interesant să pornim de la un singur operator numit Monte Carlo, **M** → care în 50% dintre încercări să dea numărul 0 și în celelalte 50% să dea numărul 1.

Așadar, pentru a obține numere aleatorii, de exemplu, între 0 și $255=2^8-1$ va fi suficient să se combine 8 operatori Monte Carlo. De exemplu, $233=2^7+2^6+2^5+2^3+2^0$, care se exprimă în sistem binar ca 11101001, s-ar obține atunci când cei 8 operatori Monte Carlo vor genera respectiva succesiune de cifre de zero și de unu (să nu uităm: computerele funcționează generând tocmai succesiuni de zero și de unu).

Firește, folosind mai mulți operatori Monte Carlo, am putea genera aleatoriu numere mai mari.

Și invers, folosind un generator de numere pseudoaleatorii, am putea simula un operator Monte Carlo, de exemplu, luând 0 când ar fi par și 1 când ar fi impar.

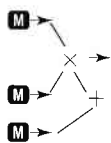
Pe de altă parte, combinând operatori Monte Carlo cu operatori logici, cum ar fi disjuncția (+) sau conjuncția ($\times$), putem obține orice probabilitate.

De exemplu:



Continuare la p. 14

Iar dacă acum luăm:



va genera numărul 1 cu o probabilitate de $25\% + 50\% = 75\%$.

Și pentru a obține, de exemplu, o probabilitate de $0,34375 = (\frac{1}{2})^2 + (\frac{1}{2})^4 + (\frac{1}{2})^5$, care s-ar exprima în sistem binar ca 0,01011, ar fi suficientă „înmulțirea” prin conjuncții, respectiv, 2, 4 și 5 operatori Monte Carlo și, în continuare, „adunarea” rezultatelor sale prin disjuncții.

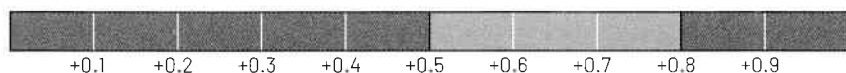
Însă, dacă lucrăm cu un generator de numere pseudoaleatorii, procedura obișnuită pentru a simula o anumită probabilitate va fi diferită.

Să ne imaginăm, de exemplu, că s-a estimat că, atunci când se joacă un meci de fotbal, probabilitatea de a câștiga (1) este de 50%, cea de a pierde (2) este de

Astfel, voi, din noile generații de nativi digital, ce sunteți obișnuiți să utilizați tot timpul computere sau smartphone-uri, puteți aborda folosirea metodelor de analiză numerică pentru a vă juca pe aceste dispozitive: astfel, frica de matematică poate deveni de domeniul trecutului, specifică oamenilor dintr-o altă epocă.

Problema este că matematica nu mai poate fi numită „știință exactă” dacă se are în vedere această variantă de calcul care, în mod normal, permite doar obținerea de rezultate aproximative. Cu toate acestea, tocmai acest lucru o poate face să pară mai aproape de viața reală, unde incertitudinea este la ordinea zilei. Dar aceasta nu este o scuză pentru a renunța la rigoarea metodelor folosite, care trebuie justificate în mod convingător. Chestiunea este că o asemenea rigoare necesită tocmai posibilitatea de a estima gradul de aproximare atins și, prin urmare, amplitudinea erorilor care pot fi comise.

30 % și cea de egalitate (x) este de 20 %. Atunci putem prezenta aceste probabilități într-un segment:



Și, numind R variabila care ar exprima rezultatul meciului, vom genera un număr x pseudoaleatoriu între 0 și 1, și atunci:

- dacă $0 \leq x < 0,5$, atribuim lui R valoare „1”, care reprezintă victoria;
- dacă $0,5 \leq x < 0,5 + 0,3$, atribuim lui R valoarea „2”, care reprezintă înfrângerea;
- dacă $0,5 + 0,3 \leq x < 1$, atribuim lui R valoarea „X”, care reprezintă egalitatea.

Și folosim rezultatul pentru a completa pariul.

Astfel că, în acest caz, rigoarea matematică impune să fim deosebit de critici cu privire la rezultatele obținute, refuzând să le considerăm adevăruri absolute.

Trebuie subliniat că, într-un fel, gândirea critică a fost întotdeauna esențială pentru matematică, fiindcă aceasta necesită demonstrarea tuturor afirmațiilor sale.

În această carte, care nu este un manual, multe lucruri vor trebui prezentate fără a fi demonstrate, ilustrându-le, cel mult, cu exemple. Dar cum nu voi pretinde să le acceptați ca dogme de credință, în majoritatea cazurilor le veți putea demonstra urmând raționamentele din anexă sau consultând bibliografia, în special ghidurile mele didactice. Totuși, în unele cazuri va trebui să pornim de la rezultate obținute la alte materii (la urma urmei, matematica formează un sistem interconectat), a căror demonstrație v-ar îndepărta de subiectul acestei cărți.

Va trebui să avem în vedere că orice demonstrație matematică riguroasă, pentru a fi așa, trebuie să fie banală sau să poată fi descompusă într-o serie de pași banali, care să nu necesite acte de credință pentru a-i asuma. Tocmai aceasta se urmărea cu seria de activități pe care elevii mei din anul I le rezolvau la vremea lor în concordanță cu un principiu pe care-l afirmam în mișcările de reînnoire pedagogică din anii '70, „se învață doar ceea ce se face”, și care în prezent a devenit teoretic la modă în cadrul așa-numitului „proces Bologna”, deși restricția mijloacelor disponibile universităților publice îngreunează implementarea acestuia.

În această carte vă propunem o serie de obiective specifice, care corespund fiecăruia dintre capitolele sale.

1. Prin *interpolare polinomială* vom încerca să ghicim o curbă care trece printr-o mulțime de puncte.
2. Prin *integrarea numerică* a unei funcții vom estima totalul valorilor dintr-o regiune.
3. Prin *aproximarea numerică a soluției unei ecuații diferențiale* vom căuta drumul urmat de o curbă cunoscând direcția acesteia în fiecare punct.
4. Și prin *aproximări succesive* vom căuta punctul care ne duce către *soluția unei ecuații*.