

# Monte Carlo metoda aproksimacije

---

Hankić, Benjamin

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, School of Applied Mathematics and Informatics / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet primijenjene matematike i informatike**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:126:503055>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom](#).

Download date / Datum preuzimanja: **2025-03-11**



**mathos**

Repository / Repozitorij:

[Repository of School of Applied Mathematics and Informatics](#)





SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU  
FAKULTET PRIMIJENJENE MATEMATIKE I INFORMATIKE

Sveučilišni diplomski studij matematike  
modul: Financijska matematika i statistika

# Monte Carlo metoda aproksimacije

DIPLOMSKI RAD

Mentor:

**doc. dr. sc. Ivan Papić**

Student:

**Benjamin Hankić**

Osijek, 2024.



# Sadržaj

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>Generatori slučajnih brojeva</b>	<b>3</b>
2.1	Kvaliteta generatora slučajnih brojeva . . . . .	3
2.1.1	Tehnički izrazi . . . . .	5
2.2	Primjeri generatora slučajnih brojeva . . . . .	7
2.2.1	Linearni kongruencijalni generatori . . . . .	7
2.2.2	Fibonaccijski generatori s lagovima . . . . .	11
2.2.3	$\mathbb{F}_2$ -linearni generatori . . . . .	11
<b>3</b>	<b>Monte Carlo metoda</b>	<b>13</b>
3.1	Grubi oblik Monte Carlo metode . . . . .	13
3.2	Primjeri korištenja Monte Carlo metode . . . . .	16
3.3	Poboljšanje brzine konvergencije . . . . .	21
3.3.1	Metode smanjivanja varijance . . . . .	21
3.3.2	Uzorkovanje po važnosti . . . . .	30
3.3.3	Primjena metoda smanjenja varijanci . . . . .	34
3.4	Važnost generatora slučajnih brojeva . . . . .	35
3.5	Primjena Monte Carlo metode . . . . .	36
	<b>Literatura</b>	<b>43</b>
	<b>Sažetak</b>	<b>45</b>
	<b>Summary</b>	<b>47</b>
	<b>Životopis</b>	<b>49</b>



# 1 | Uvod

Monte Carlo metoda osmišljena je tijekom 40-ih godina prošloga stoljeća za vrijeme i neposredno nakon Drugog svjetskog rata. Razvili su je matematičari *Stanislaw Ulam* i *John von Neumann* dok su radili na projektu Manhattan u okviru istraživanja vezanih uz razvoj atomske bombe. Ime "Monte Carlo" inspirirano je kockarnicama u Monte Carlu, jer metoda koristi nasumično uzorkovanje, slično kao u igrama na sreću.

Monte Carlo metoda integracije i aproksimacije je numerička tehnika koja koristi nasumično uzorkovanje za aproksimaciju vrijednosti integrala i složenih matematičkih problema. Ova metoda se često primjenjuje kada je teško ili nemoguće riješiti problem analitički, osobito u višedimenzionalnim prostorima. Zbog svoje fleksibilnosti i jednostavnosti Monte Carlo tehnika našla je široku primjenu u područjima poput fizike, financija i statistike.

Kako je metoda zasnovana na nasumičnom uzorkovanju slučajnih brojeva, na početku ćemo se upoznati s generatorima slučajnih brojeva i njihovim raznim oblicima koji sa sobom nose određene prednosti i nedostatke.

Nakon toga prelazimo na Monte Carlo metodu. Prvo se upoznajemo s grubim oblikom te metode, a zatim i na njene unaprjeđenije oblike.

Dotaknut ćemo se važnosti generatora slučajnih brojeva za Monte Carlo metodu, te na samom kraju demonstrirati metodu na nekoliko primjera.



## 2 | Generatori slučajnih brojeva

Mogućnost generiranja slučajnih brojeva unaprijed određenom distribucijom je nužna za Monte Carlo metodu. Glavni problem u ovoj situaciji je pronalazak brojeva koji su uistinu slučajni i nepredvidivi. Slučajno generiranje poput bacanje igraće kocke je presporo za veliku većinu primjena jer za ispravan rezultat pokusa najčešće je potreban veliki broj slučajnih brojeva. Zbog toga, alternativan način je korištenje fizikalnih fenomena poput radioaktivnog raspada koji se često smatra sinonimom za slučajnost, te nakon toga treba transformirati rezultate u slučajne brojeve. S pravom opremom ovo je puno brži način.

Uistinu, današnja istraživanja omogućavaju korištenje fizikalnih instrumenata za generiranje slučajnih brojeva pomoću transformacije mjera na način da rezultiraju dobrim aproksimacijama unaprijed zadanih distribucija. Nažalost, ti fizikalni instrumenti su spori za opsežne simulacije. Najčešći način na koji se slučajni brojevi odabiru za Monte Carlo metode jest uz pomoć numeričkih algoritama. Time nastaje deterministički niz brojeva, te iz tog razloga ovako generirani slučajni brojevi često nazivaju pseudoslučajni brojevi. No, ako ih promatramo bez da znamo koji algoritam je korišten djeluju kao sasvim slučajni brojevi.

Najprije, koncentriramo se na generiranje uniformno distribuiranih slučajnih brojeva. U sljedećem koraku uniformno distribuirane slučajne brojeve transformiramo u slučajne brojeve iz željene distribucije (npr. normalne ili Gamma(1)).

Danas postoji puno različitih metoda za generiranje slučajnih brojeva, te je i samo istraživanje u ovom području veoma aktivno i dalje. Iz tog razloga treba biti spreman na činjenicu da algoritmi za generiranje slučajnih brojeva koji su danas među najboljima i najkorisnijima već sutra postanu zastarjeli zbog pronalaska boljih.

### 2.1 Kvaliteta generatora slučajnih brojeva

Za pouzdani rezultat simulacije kvalitetni slučajni brojevi su nužnost. Loše izabrani slučajni brojevi mogu dovesti do besmislenih simulacija čiji nas rezultat može zavarati i navesti na krive zaključke. Prije same uporabe generatora slučajnih brojeva integriranih u softverske programe moramo ih prvo provjeriti. Navesti ćemo nekoliko kriterija koji ukazuju na kvalitetu generatora kojih korisnici trebaju biti svjesni:

- Naravno, uniformno distribuirani slučajni brojevi, tj slučajno generirani brojevi iz  $U(0,1)$  trebaju biti jednako distribuirani na intervalu  $[0,1]$ , ali sama struktura ne bi trebala biti previše pravilna (regularna) jer u protivnom neće



se smatrati slučajnima. No, u nekim simulacijama možemo iskoristiti pravilnost kao prednost, pogotovo kada se bavimo s glatkim funkcijama. U tom slučaju koristimo kvazislučajnu sekvencu iliti nisko-diskrepantnu sekvencu (quasirandom sequence) koje ne izgledaju sasvim slučajno, ali su jednako distribuirane.

- Kako Monte Carlo simulacije često zahtijevaju puno slučajnih brojeva treba postojati mogućnost njihovog brzog i efikasnog generiranja bez pretjerane potrošnje memorije. Dakle, brzina i memorijski kapaciteti su važni.
- Algoritmi slučajnih brojeva rade sa konačnim skupom podataka stoga će generator slučajnih brojeva naposljetku ponoviti sekvencu brojeva. Maksimalna duljina sekvence prije nego li se ponovi zove se **period** generatora slučajnih brojeva. Današnje simulacije zahtijevaju ogromne količine slučajnih brojeva stoga period mora biti dovoljno dugačak kako bi se izbjeglo učestalo korištenje ponavljajućih slučajnih brojeva. Pravilo kaže da duljina perioda generatora slučajnih brojeva treba biti barem jednako duga kao kvadrat količine potrebnih slučajnih brojeva. Inače, deterministički aspekt RNG-a (*engl. random number generator*) dolazi do izražaja i čini slučajne brojeve međusobno koreliranima. U prošlosti su RNG-ovi s periodom od oko  $10^9$  smatrani dobrima, ali danas se ti algoritmi smatraju nedostatnima. Nažalost, postoje generatori s periodom od samo  $10^5$  koji su još uvijek u upotrebi koji su često uzrok vrlo loših Monte Carlo simulacija.
- Unutar Monte Carlo metoda, sekvence slučajnih brojeva trebale bi imati svojstvo **reproduciranja**. Prvi razlog je zbog otklanjanja pogrešaka. Dobijemo li čudne rezultate simulacija moći ćemo provjeriti sekvence kako bi uvidjeli postoji li neka pogreška među njima. Također, u nekim sferama istraživanja je dosta učestalo da se iste sekvence ponovo koriste, a još jedna prednost leži u tome što je usporedba različitih izračunskih metoda ili sličnih financijskih proizvoda učinkovitija korištenjem istih slučajnih brojeva.
- Algoritam bi trebao biti programiran tako da izbacuje iste slučajne brojeve na svakom računalu. Ovo svojstvo se naziva **prenosivost**. Mora biti omogućeno da ponovi izračune u drugim računalima i na kraju uvijek dobije isti rezultat. Ista početna vrijednost za inicijalizaciju uvijek bi trebala vratiti isti niz slučajnih brojeva.
- Kako bi izračuni bili što brži poželjno ih je raditi na računalu s paralelnim procesorima, odnosno bitno je uzeti u obzir mogućnost korištenja **paralelizacije**.
- Struktura slučajnih točaka je vrlo važna. Tipična značajka nekih generatora slučajnih brojeva je da, ako se  $d$  – *dimenzionalni* vektori konstruiraju od uzastopnih slučajnih brojeva, te točke leže na hiperplohami. Loši generatori slučajnih brojeva imaju samo nekoliko hiperploha u nekim dimenzijama koje su međusobno udaljene, ostavljajući velike praznine bez ikakvih slučajnih vektora u prostoru. Nadalje, treba biti svjestan velikih korelacija između nekih

slučajnih brojeva. Na primjer, ako algoritam konstruira novi slučajni broj iz dva prethodna slučajna broja, grupe od tri slučajna broja mogu biti previše povezane. Ta korelacija može dovesti do lažnih rezultata simulacije ako model ima sličnu strukturu.

- Ponekad je potrebno odabrati generatore slučajnih brojeva koji su jednostavni za implementaciju, tj. za koje je kod jednostavan i kratak. Ovo je općenitija točka i odnosi se na sve dijelove simulacije. Vrlo često je korisno imati drugu simulacijsku rutinu za provjeru važnih rezultata simulacije. U tom slučaju, mogućnost pogreške je manja.

### 2.1.1 Tehnički izrazi

Svi generatori slučajnih brojeva bazirani na determinističkim rekurzijama mogu se opisati kao petorka  $(\mathcal{S}, \mu, f, \mathcal{U}, g)$  gdje:

- $\mathcal{S}$  je konačan skup stanja, takozvani **prostor stanja**.
- $s_n \in \mathcal{S}$  je određeno **stanje** na  $n$ -tom koraku iteracije.
- $\mu$  je vjerojatnosna mjera odabira  $s_0$ , početnog iliti nultog stanja, iz  $\mathcal{S}$ .  $s_0$  se naziva početna vrijednost (*engl. seed*) generatora slučajnih brojeva (*RNG*).
- Funkcija  $f$ , također poznata kao **funkcija prijelaza** opisuje algoritam,  $s_{n+1} = f(s_n)$
- $\mathcal{U}$  je **prostor izlaznih vrijednosti**. Uglavnom uzimamo u obzir  $[0, 1]$ ,  $[0, 1)$ ,  $(0, 1]$  ili  $(0, 1)$ .
- **Izlazna funkcija**  $g : \mathcal{S} \rightarrow \mathcal{U}$  preslikava stanje  $s_n \in \mathcal{S}$  u broj  $u_n \in \mathcal{U}$ , odnosno krajnji slučajni broj koji nas zanima.

Navedimo nekoliko napomena vezano za neke početne karakteristike i uvjete generatora koje nikako ne smijemo zanemariti:

- Početna vrijednost je vrlo važna. Inicijalni vektor brojeva ne bi smio imati pretežito nule iz razloga što to može dovesti do lošeg niza slučajnih brojeva. Točnije, kada su pretežito nule u inicijalnom vektoru slučajni brojevi se ne razlikuju toliko puno. Poželjno je spremati početnu vrijednost u slučaju potrebe otklanjanja pogrešaka.
- Kako se stvarni brojevi mogu prikazati samo s konačnom točnošću u računalima, skup stanja  $\mathcal{S}$  mora biti konačan, a tako je i raspon  $g(\mathcal{S}) \subset \mathcal{U}$  također konačan. Budući da smo zainteresirani za ravnomjernu distribuciju, želimo da praznine u intervalu  $[0, 1)$  ne budu prevelike. Stoga,  $\mathcal{S}$  mora biti veliki skup.

- Još jedna posljedica konačnosti skupa  $\mathcal{S}$  je  $s_{n+\rho} = s_n$  za neku konstantu  $\rho$ , odnosno niz će se naposljetku ponoviti. Najmanji broj  $\rho$  naziva se **period** generatora slučajnih brojeva. Ne smije biti veći od  $|\mathcal{S}|$  što je još jedan razlog zašto skup mora biti velik. Kod nekih generatora periodi se mogu razlikovati za različite inicijalne vrijednosti, stoga moramo paziti s kojom vrijednosti započinjemo.

## 2.2 Primjeri generatora slučajnih brojeva

### 2.2.1 Linearni kongruencijalni generatori

Linearni kongruencijalni generatori ("*Linear Congruential Generators*" - LCG) su jedni od najstarijih i najjednostavnijih metoda za generiranje pseudoslučajnih brojeva.

**Definicija 1.** LCG se definira na sljedeći način:

$$s_{n+1} = (as_n + c) \pmod{m}, n \in \mathbb{N}$$

gdje je

- $m \in \mathbb{N}$   $0$  tzv. **modul**,
- $a \in \mathbb{N}$  **multiplikator**  $a < m$ ,
- $c \in \mathbb{N}$  **aditivna konstanta**,  $c < m$ ,
- $s_0 \in \mathbb{N}$  **početna vrijednost** ili "sjeme" (engl. *seed*),  $s_0 < m$ .

Brojevi  $[0, 1)$  dobiveni su putem

$$u_n = \frac{s_n}{m}.$$

Ovi generatori se često označavaju uređenom trojkom  $LCG(m, a, c)$ . Prostor stanja može se opisati kao

$$S \subseteq \mathbb{N}, S = \{0, 1, \dots, m - 1\} \text{ ili } \{1, 2, \dots, m - 1\}.$$

Za aditivnu konstantnu  $c = 0$  generator ne može započeti s početnom vrijednošću 0. U tom slučaju 0 je isključena iz skupa stanja.

Ukoliko je period generatora  $m$  ili  $m - 1$  (u slučaju  $c = 0$ ) ovo svojstvo zovemo **puni period** pošto je period maksimalne moguće duljine. Kako bi dostigli dugački period preporučljivo je za modul izabrati vrlo veliki broj. Naprimjer ako za parametre izaberemo  $c = 0$ ,  $m$  prosti broj,  $a$  primitivni korijen modulo  $m$  tada dobivamo generator s punim periodom. U slučaju kada je  $c \neq 0$  kriterij za puni period je puno kompliciraniji.

Navedimo sad neke glavne karakteristike LCG-a:

- **Jednostavnost:** LCG je jednostavan za primjenu i lako razumljiv.
- **Deterministički izlaz:** Za istu početnu vrijednost (*seed*), LCG uvijek generira isti niz brojeva.
- **Periodičnost:** LCG ima konačan period, što znači da će niz generiranih brojeva nakon određenog broja koraka početi ponavljati.

Slijede sada prednosti i nedostatci. Prednosti:

- **Brzina:** LCG je vrlo brz zbog jednostavnih aritmetičkih operacija.
- **Jednostavnost implementacije:** Zbog jednostavne formule, lako ga je implementirati u većini programskih jezika.
- Uz jednostavnu implementaciju veoma su laki za razumijevanje.
- Jedna od velikih prednosti je i ta što ovi generatori ne zauzimaju previše memorije.

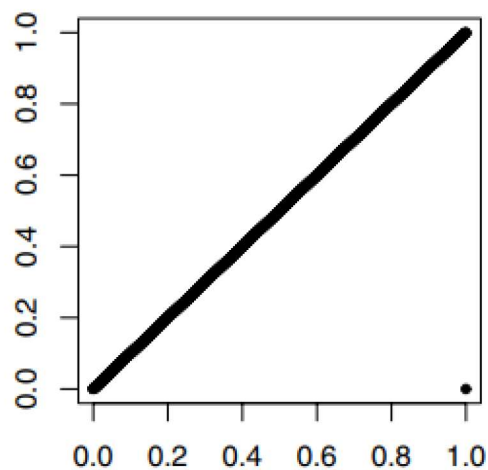
Nedostatci:

- **Kratak period:** Ako nije pažljivo dizajniran, LCG može imati kratak period, što znači da će se niz brojeva brzo ponavljati.
- **Niska kvaliteta slučajnosti:** LCG može generirati brojeve s lošom statističkom slučajnošću, što može rezultirati korelacijama među generiranim brojevima.
- **Loša distribucija u višim dimenzijama:** LCG može imati problema s ravnomjernom distribucijom brojeva u višedimenzionalnim prostorima.
- Pogledajmo pobliže skup svi mogućih  $t$ -dimenzionalnih vektora sastavljenih od uzastopnih slučajnih brojeva, odnosno konačni skup

$$\Psi_t := \{(u_1, u_2, \dots, u_t) | s_0 \in \mathcal{S}\} \subseteq [0, 1)^t.$$

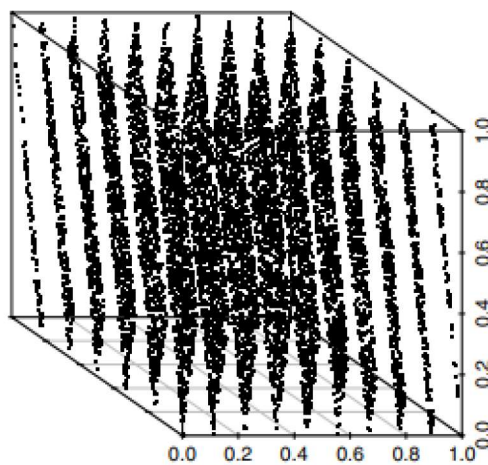
Ukoliko su pseudoslučajni brojevi uistinu slučajni i nezavisni,  $t$ -dimenzionalna hiperkocka će biti ravnomjerno ispunjena bez neke prepoznatljive strukture. No, struktura uzorka  $\Psi_t$  iz LCG-a je vrlo pravilna, to jest sve točke leže na ekvidistantnim paralelnim hiperravninama. Ovo se naziva rešetkasta struktura (*engl. lattice structure*) LCG-a.

**Primjer 1.** *Kao osnovni primjer pogledajmo LCG  $s_{n+1} = (1 \cdot s_n + 1) \bmod 3331$ . Ovaj generator ima puni period koji iznosi 3331. No,  $\Psi_2$  sastoji se od samo jedne hiperravnine i jedne točke kao što je prikazano na slici 2.1.*



Slika 2.1: Prikaz hiperravnina iz primjera 1 [1]

**Primjer 2.** Prikazan je jedan od najkorištenijih LCG-ova proteklih godina - LCG RANDU. Koeficijenti u tom generatoru su sljedeći:  $a = 65539$ ,  $c = 0$  i  $m = 2^{31}$ . No, pogledamo li trodimenzionalni vektor koji je izgrađen od uzastopnih slučajnih brojeva vidimo da sve točke  $\Psi_3$  leže na "samo" 15 hiperravnina (Slika 2.2).



Slika 2.2: Prikaz hiperravnina LCG RANDU-a [1]

Naziv	LCG(m,a,c)	Period
Park and Miller	$LCG(2^{31} - 1, 16807, 0)$	$2^{31} - 2$
Fishman and Moore	$LCG(2^{31} - 1, 950706376, 0)$	$2^{31} - 2$
Fishman	$LCG(2^{31} - 1, 428271, 0)$	$2^{31} - 2$
L'Ecuyer	$LCG(2^{31} - 249, 40692, 0)$	$2^{31} - 250$
ran2(kombinirani LCG)	$LCG_1(2147483563, 40014, 0)$ $LCG_2(2147483399, 40692, 0)$	$2.3 \cdot 10^{18}$

Tablica 2.1: Neki od preporučenih LCG-ova za korištenje [1]

**Definicija 2.** Skup  $\Psi_t := \{(u_1, u_2, \dots, u_t) | s_0 \in \mathcal{S}\} \subseteq [0, 1)^t$  je skup svih  $t$ -dimenzionalnih vektora proizvedenih uzastopnim slučajnim brojevima pomoću određenog generatora slučajnih brojeva.  $s_0$  simbolizira "sjeme" odnosno početnu vrijednost,  $u_1$  je prvi pravi slučajni broj dobiven pomoću početne vrijednosti  $s_0$ ,  $u_2$  drugi slučajni broj i tako dalje. Ovaj skup se promatra kao multiskup i stoga je  $|\Psi_t| = |\mathcal{S}|$ . Nazvat ćemo ga "skup svih  $t$ -dimenzionalnih izlaznih vektora".

### 2.2.2 Fibonaccijevi generatori s lagovima

Fibonaccijevi generatori slučajnih brojeva s lagovima (LFG) su generalizacija Fibonaccijevog niza

$$s_n = s_{n-1} + s_{n-2}.$$

Vidimo da ovo baš i nije najbolji generator slučajnih brojeva. No, neke reprezentacije generalizirane verzije LFG-a su vrlo korisne:

$$s_n = s_{n-q} \odot s_{n-p}.$$

Operator  $\odot$  može biti zbrajanje, oduzimanje ili množenje modulo  $m$ . Ove generatore ćemo označavati kao uređenu trojku  $LFG(p, q, \odot)$ , gdje su  $p$  i  $q$  takozvani **lagovi**,  $p, q \in \mathbb{N} \setminus \{0\}$ ,  $p > q$ . Množenje mora biti izvršeno na skupu neparnih cijelih brojeva. Modul  $m$  može biti bilo koji cijeli broj, uključujući i 1.

Ovi generatori se mogu generalizirati koristeći tri ili više laga. Navedimo prednosti i nedostatke ovih generatora. Prednosti su:

- **Jednostavna implementacija:** LFG-ovi su relativno jednostavni za implementaciju.
- **Brzina:** Mogu biti vrlo brzi zbog jednostavnih aritmetičkih operacija.
- **Dug period:** Ako se pravilno odaberu parametri, LFG-ovi mogu imati vrlo dug period, što je korisno za mnoge simulacijske primjene.

Dok su nedostaci:

- **Niska kvaliteta slučajnosti:** U nekim slučajevima, LFG-ovi mogu proizvesti nizove brojeva s lošom statističkom slučajnošću.
- **Korelacija:** Može postojati korelacija između generiranih brojeva, što može biti problematično u nekim primjenama.
- **Loša distribucija u višim dimenzijama:** Kao i kod LCG-ova, LFG-ovi mogu imati problema s ravnomjernom distribucijom brojeva u višedimenzionalnim prostorima.

### 2.2.3 $\mathbb{F}_2$ -linearni generatori

Druga ideja za ubrzanje implementacije je iskorištavanje binarnog prikaza brojeva u računalu. Stoga tražimo algoritme koji koriste samo 0 i 1. Zbog toga radimo u polju  $\mathbb{F}_2$  s elementima 0, 1, tj. sve operacije se izvode modulo 2. Zbrajanje u  $\mathbb{F}_2$  je upravo binarna ekskluzivna disjunkcija (XOR) operacija,  $\oplus$ :

$$(x + y) \bmod 2 \iff x \oplus y, \quad x, y \in \mathbb{F}_2.$$

Kako bi niz bio pokrenut,  $\mathbb{F}_2$  linearni generatori moraju biti inicijaliziran s po-



četnom vrijednošću, točnije s vektorom  $\mathbf{x}_0 = (x_0, x_1, \dots, x_{k-1}) \in \mathcal{S}$ . Skup stanja možemo definirati kao

$$\mathcal{S} = \{0, 1\}^k \setminus \{0\}.$$

Imajući ovo na umu, period generatora ne može biti veći od  $2^k - 1$  jer je to maksimalan broj mogućih vrijednosti za  $\mathbf{x}_n$  isključujući nulti vektor. Duljina perioda može se odrediti analizom karakterističnog polinoma matrice  $\mathbf{A}$ ,

$$P(z) = \det(z\mathbf{I} - \mathbf{A}) = z^k - \alpha_1 z^{k-1} - \dots - \alpha_1 z^1 - \alpha_k,$$

gdje je  $\alpha_j \in \mathbb{F}_2, j = 1, \dots, k, \alpha_1 = \text{trag}(\mathbf{A}), \alpha_k = \det \mathbf{A}$ . Sljedećim polinomom možemo definirati linearnu rekurziju unutar  $\mathbb{F}_2$ :

$$v_n = \alpha_1 v_{n-1} + \dots - \alpha_{k-1} v_{n-k+1} + \alpha_k v_{n-k}.$$

Ukoliko je  $\alpha_k = 1$ , što znači da je rang  $\mathbf{A}$  maksimalan, tada je gore navedena rekurzija  $k$ -tog reda i svaki  $x_{n,i}, n > 0, i = 0, \dots, k-1$ , iz Algoritma 1 slijedi ovu rekurziju. Ako je polinom  $P(z)$  primitivan nad  $\mathbb{F}_2$ , rekurzija ima maksimalan period  $2^k - 1$  i shodno tomu rekurziju iz Algoritma 1. Matrica  $\mathbf{B}$  često se koristi kako bi unaprijedila distribuciju izlaza (izlaznih vrijednosti), ali u mnogim slučajevima to je samo  $w \times w$ -jedinična matrica s dodanih  $k - w$  nul-stupaca.

---

#### Algoritam 1 $\mathbb{F}_2$ linearni generatori

---

$$x_{n+1} = \mathbf{A}x_n, x_n \in \mathbb{F}_2^k, \mathbf{A} \in \mathbb{F}_2^{k,k}$$

$$y_{n+1} = \mathbf{B}x_{n+1}, y_{n+1} \in \mathbb{F}_2^w, \mathbf{B} \in \mathbb{F}_2^{w,k}$$

- $x_n$  je  $k$ -bitni **vektor stanja** u  $n$ -tom koraku,  $n \in \mathbb{N}, k \in \mathbb{N}, k \in \mathbb{N} \setminus \{0\}$ ,
- $y_n$  je  $w$ -bitni **izlazni vektor**,  $w \in \mathbb{N} \setminus \{0\}$ ,
- $\mathbf{A}$  je **matrica tranzicije**,
- $\mathbf{B}$  je **matrica izlaznih vrijednosti**,

Izlazna vrijednost  $u_n \in [0, 1)$  je generirana na sljedeći način:

$$U_n = \sum_{i=1}^w y_{n,i-1} 2^{-i} = 0, y_{n,0}, y_{n,1}, \dots, y_{n,w-1} \text{ (su u binarnoj reprezentaciji).}$$


---

## 3 | Monte Carlo metoda

Glavna ideja Monte Carlo metode je ta da aproksimira očekivanu vrijednost  $\mathbb{E}(X)$  uz pomoć aritmetičke sredine rezultata velikog broja nezavisnih eksperimenata koji svi imaju jednaku distribuciju kao  $X$ . Osnova ove metode jedan je od najvažnijih rezultati teorije vjerojatnosti, **zakon velikih brojeva**.

Podsjetimo se teorema o zakonu velikih brojeva.

**Teorem 1.** *Neka je  $(X_n)_{n \in \mathbb{N}}$  niz integrabilnih, realnih slučajnih varijabli koje su nezavisne i jednako distribuirane, te definirane na vjerojatnosnom prostoru  $(\Omega, \mathcal{F}, \mathbb{P})$ . Nadalje*

$$\mu = \mathbb{E}(X_1).$$

Tada imamo  $\forall \omega \in \Omega$   $\mathbb{P}$ -g.s.

$$\frac{1}{n} \sum_{i=1}^n X_i(\omega) \xrightarrow{n \rightarrow \infty} \mu,$$

aritmetička sredina realizacija  $X_i$  konvergira prema teoretskom očekivanju svakog od  $X_i$ , tj. prema očekivanju  $\mu$ .

Važno je uočiti da svaka konvergencija u jakom zakonu velikih brojeva je gotovo sigurno i prava konvergencija. Ovo je  $\omega$ -konvergencija, tj. konvergencija aritmetičkih sredina u sklopu Teorema 1 koja se može reducirati na konvergenenciju nizova **realnih brojeva**  $\frac{1}{n} \sum_{i=1}^n X_i(\omega)$ .

Također, postoji malo općenitiji oblik Teorema 1. U tom obliku umjesto pretpostavke stroge nezavisnosti možemo koristiti nezavisnost po parovima, tj. svi parovi  $X_i$  i  $X_j$  moraju biti nezavisni za  $i \neq j$ .

### 3.1 Grubi oblik Monte Carlo metode

---

**Algoritam 2** Grubi oblik Monte Carlo metode

$\mathbb{E}(X)$  aproksimiramo pomoću aritmetičke sredine  $\frac{1}{N} \sum_{i=1}^N X_i(\omega)$  za neki  $N \in \mathbb{N}$ . Ovdje su  $X_i(\omega)$  rezultati  $N$  nezavisnih eksperimenata koji imaju jednaku vjerojatnosnu distribuciju kao  $X$ .

---

Što se tiče točnosti Monte Carlo metode, moramo istaknuti da kao stohastička metoda, različita pokretanja Monte Carlo metode obično dovode do različitih rezultata (iako mogu biti prilično bliski jedan drugome!) prilikom aproksimacije određenog izraza. Stoga se moramo nositi sa stohastičkom pogreškom. Prvo recimo da Monte Carlo metoda aproksimira relevantno očekivanje ispravno u prosjeku.

**Teorem 2 (Nepriistranost Monte Carlo procjenitelja).**

*Neka je  $(X_n)_{n \in \mathbb{N}}$  niz realnih integrabilnih slučajnih varijabli koje su nezavisne i jednako distribuirane kao  $X$ . Sve slučajne varijable su definirane na vjerojatnosnom prostoru  $(\Omega, \mathcal{F}, \mathbb{P})$ .*

*Tada je Monte Carlo aproksimacija*

$$\bar{X}_N := \frac{1}{N} \sum_{i=1}^N X_i, N \in \mathbb{N},$$

*nepristrana za  $\mu = \mathbb{E}(X)$ , tj.*

$$\mathbb{E}(\bar{X}_N) = \mu.$$

Iako ovo već osigurava da je Monte Carlo aproksimacija u prosjeku ispravna, to nam ne pomaže da steknemo osjećaj za apsolutnu vrijednost pogreške. Stoga promatramo standardnu devijaciju razlike između  $\bar{X}_N$  i  $\mu$ .

Kako imamo

$$\text{Var}(\bar{X}_N - \mu) = \text{Var}(\bar{X}_N) = \frac{1}{N^2} \sum_{i=1}^N \text{Var}(X_i) = \frac{\sigma^2}{N},$$

standardna devijacija pogreške je reda veličine  $O(1/\sqrt{N})$ . To znači da se pogreška smanjuje kako se broj nezavisnih eksperimenata (ili dimenzija uzoraka) povećava. Konkretno, udvostručavanje dimenzije uzoraka  $N$  smanjuje pogrešku za faktor  $\sqrt{2}$ . Kako je standardna devijacija mjera za (prosječnu) točnost grube Monte Carlo metode, ovaj izračun ima sljedeću važnu posljednicu:

**Povećanje točnosti Monte Carlo aproksimacije**

Povećanje (prosječne) točnosti grube Monte Carlo aproksimacije za jedan decimalni broj (tj. smanjenje standardne devijacije za faktor 0.1) zahtijeva povećanje broja Monte Carlo pokretanja za faktor 100.

To naime znači da jednostavno ponavljanje Monte Carlo pokretanja određeni broj puta ne poboljšava značajno točnost aproksimatora. U smislu gore navedenog uvida, ako želimo postići veću preciznost potrebno je značajno ulaganje truda. Korištenje standardne devijacije pogreške kao mjere za točnost Monte Carlo aproksimacije može se opravdati centralnim graničnim teoremom.

**Teorem 3 (Centralni granični teorem (n.j.d.)).**

*Neka je  $(X_n)_{n \in \mathbb{N}}$  niz integrabilnih, realnih slučajnih varijabli koje su nezavisne i jednako distribuirane, te definirane na vjerojatnosnom prostoru  $(\Omega, \mathcal{F}, \mathbb{P})$  gdje je  $\mu$  pripadno očekivanje, a  $\sigma^2$  varijanca. Također, pretpostavimo da su varijance svih varijabli konačne,*

$\sigma^2 = \text{Var}(X)$ . Tada, standardizirani zbroj tih slučajnih varijabli konvergira u distribuciji prema standardnoj normalnoj distribuciji, tj. imamo

$$\frac{\sum_{i=1}^N X_i - N\mu}{\sqrt{N}\sigma} \xrightarrow{D} \mathcal{N}(0,1) \text{ kada } N \rightarrow \infty.$$

Iz centralnog graničnog teorema možemo zaključiti da za velike vrijednosti  $N$  gruba forma Monte Carlo aproksimacije približno ima  $\mathcal{N}(\mu, \frac{\sigma^2}{N})$  distribuciju. Budući da standardna devijacija  $\sigma$  jedinstveno karakterizira raspršenost vrijednosti normalne distribucije oko njenog očekivanja  $\mu$ , korištenje standardne devijacije kao mjere za točnost Monte Carlo aproksimacije je opravdano. Kako znamo da je asimptotska distribucija Monte Carlo aproksimatora otprilike normalna, dobivamo **asimptotski**  $(1 - \alpha)$  **pouzdan interval za očekivanje**  $\mu$

$$\left[ \frac{1}{N} \sum_{i=1}^N X_i - z_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{N}}, \frac{1}{N} \sum_{i=1}^N X_i + z_{1-\frac{\alpha}{2}} \frac{\sigma}{\sqrt{N}} \right],$$

gdje je  $z_{1-\frac{\alpha}{2}}$   $1 - \frac{\alpha}{2}$ -kvantil standardne normalne distribucije. Kako 97.5%-kvantil standardne normalne iznosi 1.96, a popularan izbor za asimptotski simetrični 95%-kvantil za očekivanje koje je procijenjeno Monte Carlo metodom u primjenama je dan od strane

**$2\sigma$  pravilo za asimptotski 95% -pouzdan interval za  $\mu$**

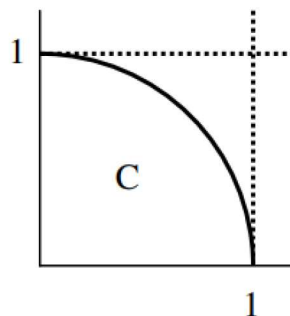
$$\left[ \frac{1}{N} \sum_{i=1}^N X_i - 2 \frac{\sigma}{\sqrt{N}}, \frac{1}{N} \sum_{i=1}^N X_i + 2 \frac{\sigma}{\sqrt{N}} \right].$$

## 3.2 Primjeri korištenja Monte Carlo metode

Osnova primjene Monte Carlo metode je izračunavanje ili aproksimacija određenih izraza pomoću nasumičnog odabira obično velikog broja prikladnih slučajnih brojeva. Kako nam jaki zakon velikih brojeva opravdava Monte Carlo metodu, potrebno je da se izraz koji proučavamo može povezati s očekivanjem koje se zatim samo po sebi aproksimira aritmetičkom sredinom gore navedene sekvence slučajnih brojeva. Ovu ideju ćemo ilustrirati s nekoliko primjera.

### Primjer 3 (Izračun $\pi$ pomoću Monte Carlo metode).

Eksperimentalni način za približno izračunavanje  $\pi$  je promatranje dijela jediničnog kruga  $C$  sa središtem u ishodištu koji se presijeca s pozitivnim jediničnim kvadratom  $[0, 1]^2$  (Slika 3.1).



Slika 3.1: Procjena  $\pi$  uz pomoć Monte Carlo metode

Naš eksperiment se sastoji od nasumičnog odabira točaka  $P_1, \dots, P_N$  unutar jediničnog kvadrata i promatranja

$$X_i = \mathbb{1}_{\{P_i \in C\}},$$

funkcije koje označavaju je li  $P_i$  unutar kruga ili ne. Time implicitno pretpostavljamo da su odabrane točke uniformno raspoređene na  $[0, 1]^2$ .

Tada imamo

$$\mathbb{P}(P \in C) = \pi/4$$

jer je vjerojatnost pogotka  $C$  jednaka njegovoj površini (primjetimo da površina  $[0, 1]^2$  iznosi 1). Znamo da **indikator funkcija**  $1_{P_i}$  zadovoljava

$$\mathbb{E}(1_{P_i}) = \mathbb{P}(P_i \in C) = \pi/4,$$

pa iz tog razloga možemo procijeniti  $\pi$  kroz odgovarajuću aritmetičku sredinu svih  $P_i$  kako bismo dobili Monte Carlo procjenu

$$\hat{\pi}(\omega) = \frac{4}{N} \sum_{i=1}^N 1_{P_i \in C}(\omega).$$

Brzina konvergencije metode prikazana je rezultatima u Tablici 3.1 gdje smo izabrali vrijednosti za  $N$  odabrali 100, 10000 i 100000. Imajmo na umu da čak ni za naizgled velik

N	100	10000	100000
$\hat{\pi}$	2.84	3.1268	3.14144

Tablica 3.1: Grube Monte Carlo aproksimacije za  $\pi$ 

broj ponavljanja kao što je  $N = 10,000$  nećemo nužno dobiti prve tri znamenke ispravno! Ovo treba smatrati impresivnim pokazateljem da Monte Carlo metoda konvergira uistinu poprilično sporo. No, također treba napomenuti da je relativna pogreška ove aproksimacije manja od 0.5%. U usporedbi s ovim, aproksimacija za  $N = 100000$  je onda izvanredno precizna!

Čak štoviše, može se dogoditi da zbog slučajnosti Monte Carlo aproksimacije dobijemo precizniji rezultat za  $N = 100$  nego za  $N = 100000$ . Zbog toga je iznimno važno da se izračunaju granice pouzdanih intervala  $[\hat{\pi}_{donja}, \hat{\pi}_{gornja}]$  za Monte Carlo aproksimaciju.

N	100	10000	100000
$\hat{\pi}_{donja}$	2.477	3.0938	3.13105
$\hat{\pi}_{gornja}$	3.203	3.1598	3.15183

Tablica 3.2: Monte Carlo 95%-pouzdanе granice za  $\pi$ 

U Tablici 3.2 su prikazane granice 95%-pouzdanog intervala. Uistinu, sve sadrže stvarnu vrijednost  $\pi$ , no njihove duljine se razlikuju.

Kao u slučaju aproksimacije koja je uglavnom zasnovana na aproksimaciji vjerojatnosti određenih događaja postoji jednostavnija formula jer je općenito za izradu pouzdanog intervala potrebna varijanca. Pojasnit ćemo to u sljedećem primjeru.

**Primjer 4 (Aproksimacija vjerojatnosti događaja).** Ovim primjerom želimo formalizirati da je aproksimacija vjerojatnosti događaja važna kao primjer primjene Monte Carlo metode. Stoga, neka je  $A$  određen događaj. Želimo aproksimirati vjerojatnost događaja  $A$ , tj.  $\mathbb{P}(A)$ . Korištenjem poveznice između očekivanja indikator funkcije  $\mathbb{1}_A$

$$\mathbb{1}_A(\omega) = \begin{cases} 1 & , \omega \in A \\ 0 & , \omega \notin A \end{cases} \quad (3.1)$$

i vjerojatnosti od  $A$

$$\mathbb{E}(\mathbb{1}_A) = \mathbb{P}(A),$$

Monte Carlo aproksimacija za  $\mathbb{P}(A)$  je jednostavno relativna frekvencija pojavljivanja događaja  $A$  u  $N$  nezavisnih eksperimenata.

Formalno, neka  $A_i$  označava pojavljivanje događaja  $A$  u  $i$ -tom eksperimentu. Tada definiramo Monte Carlo aproksimaciju za  $\mathbb{P}(A)$  kao

$$rf_N(A) = \frac{1}{N} \sum_{i=1}^N \mathbb{1}_{A_i}.$$

Također vrijedi

$$\text{Var}(1_{A_i}) = \mathbb{P}(A)(1 - \mathbb{P}(A)),$$

označimo li s

$$\hat{\sigma}_N^2 = rf_N(A)(1 - rf_N(A))$$

možemo zapisati približan 95%-pouzdan interval za  $\mathbb{P}(A)$  kao

$$\left[ rf_N(A) - \frac{1.96}{\sqrt{N}}\hat{\sigma}_N, rf_N(A) + \frac{1.96}{\sqrt{N}}\hat{\sigma}_N \right].$$

**Primjer 5 (Monte Carlo integracija [1]).**

Vrlo jednostavna, ali često učinkovita primjena Monte Carlo pristupa je izračunavanje vrijednosti determinističkih integrala oblika

$$\int_{[0,1]^d} g(x) dx,$$

gdje je  $g$  realna ograničena funkcija. Uvođenjem funkcije gustoće  $f(x)$   $d$ -dimenzionalne uniformne distribucije na  $[0, 1]^d$

$$f(x) = 1_{[0,1]^d}(x), x \in \mathbb{R}^d,$$

možemo prepraviti gornji integral kao očekivanje od  $g(X)$  gdje je  $X$  slučajna varijabla koja je uniformno distribuirana na  $[0, 1]^d$ , tj.

$$I = \int_{[0,1]^d} g(x) dx = \int g(x) f(x) dx = \mathbb{E}(g(X)).$$

Ovo nam dopušta da izračunamo Monte Carlo aproksimaciju za navedeni integral uz simuliranje realizacije  $N$  slučajnih varijabli  $X_1, \dots, X_N$  koje su sve nezavisne i uniformno distribuirane na  $[0, 1]^d$  i zatim dobiti

$$\hat{I}_n(\omega) = \frac{1}{N} \sum_{i=1}^N g(X_i(\omega)).$$

Treba napomenuti da iako se integral računa na podskupu od  $\mathbb{R}^d$ , realne slučajne varijable  $Z_i = g(X_i)$  dozvoljavaju primjenu **jakog zakona velikih brojeva**. Brzina konvergencije reda  $O(N^{-\frac{1}{2}})$  Monte Carlo aproksimacije ostaje **valjana** neovisno o dimenziji  $d$ . Budući da determinističke kvadraturne formule obično imaju brzinu konvergencije reda  $O(N^{-\frac{2}{d}})$ , očekujemo da će Monte Carlo metoda nadmašiti ove formule (barem u prosjeku) za dimenzije  $d > 4$ . Ovo se često parafrazira kao: "**Monte Carlo metode pobjeđuju prokletstvo dimenzionalnosti**".

Primjena Monte Carlo integracije nije ograničena na jedinični interval. Potpuno ista metoda se može prenijeti i na općenite ograničene  $d$ -dimenzionalne pravokutnike. Naravno, na tom pravokutniku slučajne varijable  $X_i$  moraju biti uniformno distribuirane.

U slučaju neograničene domene potrebna je prikladna transformacija  $h^{-1}$  koja preslikava domenu na jedinični interval. Tada je intergal jednak  $\mathbb{E}(g(h(X))h'(X))$  gdje je  $X$  uniformno distribuiran na jediničnom intervalu.

Metoda Monte Carlo integracije može se imitirati za izračunavanje diskretnih suma funkcije  $g(x)$  nad prebrojivim skupom  $A$ . Uistinu, uz

$$\sum_{x \in A} g(x) = \sum_{x \in A} \frac{g(x)}{p(x)} p(x) = \mathbb{E} \left( \frac{g(X)}{p(X)} \right), \quad (3.2)$$

gdje je  $\mathbb{P}$  diskretna vjerojatnosna distribucija na  $A$

$$\mathbb{P}(X = x) = p(x) > 0, \quad \forall x \in A,$$



svaka diskretna suma može se interpretirati kao očekivanje.

Monte Carlo metoda može se primijeniti tako da generiramo veliki broj ( $N$ ) realizacija slučajnih varijabli iz ove distribucije  $\mathbb{P}$  i zatim za izračunavanje grubog Monte Carlo aproksimatora za sumu

$$\hat{S}_{N,\mathbb{P}} = \frac{1}{N} \sum_{i=1}^N \frac{g(X_i)}{p(X_i)}$$

kako bi procijenili očekivanje iz Jednadžbe 3.2.

Naravno, izbor odgovarajuće vjerojatnosne distribucije u slučaju beskonačne sume nije jednostavan jer ne postoji uniformna distribucija na skupu s beskonačno mnogo elemenata.

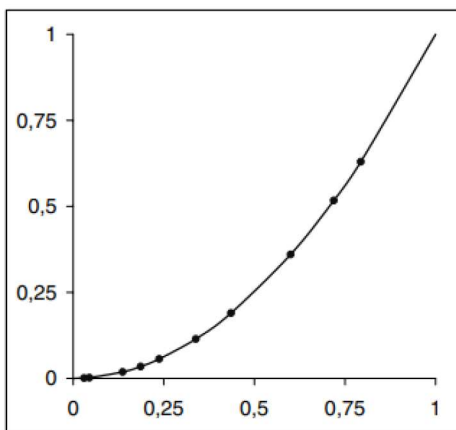
Općenito, za neograničene domene integracije  $D \subset \mathbb{R}^d$  može se koristiti distribucija vjerojatnosti koja idealno ima nosač točno jednak  $D$ -u. Budući da, ovisno o obliku  $D$ , može biti prilično teško pronaći takvu distribuciju, vrlo gruba metoda bi bila koristiti višedimenzionalnu normalnu distribuciju s identičnom matricom  $I$  kao matricom kovarijanci i vektorom očekivanja  $\mu$  koji leži unutar  $D$  (idealno na nekoj središnjoj poziciji). S  $\rho_{\mu,I}(x)$  kao odgovarajućom funkcijom gustoće, tada dobivamo

$$\int_D g(x) dx = \int_{\mathbb{R}^d} 1_D(x) \frac{g(x)}{\rho_{\mu,I}(x)} \rho_{\mu,I}(x) dx = \mathbb{E}(\tilde{g}(X))$$

iz koje konstrukcija Monte Carlo aproksimacije funkcionira na uobičajen način, gdje je

$$\tilde{g}(x) := 1_D(x) \frac{g(x)}{\rho_{\mu,I}(x)}.$$

Međutim, treba imati na umu da sada moramo zahtijevati da ovo posljednje očekivanje postoji i bude konačno jer više nije ograničeno.



Slika 3.2: Monte Carlo integracija  $g(x) = x^2$  na  $[0, 1]$ , točke predstavljaju uzorkovane vrijednosti za  $N = 10$  [1]

Na Slici 3.2 prikazana je ilustracija kako se integral funkcije  $g(x) = x^2$  na  $[0, 1]$  aproksimira pomoću

$$\frac{1}{N} \sum_{i=1}^{10} g(X_i(w)) = \frac{1}{N} \sum_{i=1}^{10} X_i(w)^2,$$

aritmetičke sredine svih vrijednosti izračunatih na 10 nasumično odabranih točaka. Treba napomenuti da točke označavaju vrijednosti funkcije na nasumično generiranim točkama. Konkretno, postoje dvije točke koje su istovremeno blizu ishodištu. Također integriramo ovu funkciju na  $[0, 1]$  pomoću Monte Carlo metode za dodatne vrijednosti  $N$ . Odgovarajuće vrijednosti zajedno sa donjim i gornjim granicama 95%-pouzdanog intervala su dane u Tablici 3.3. Treba napomenuti vrlo lošu izvedbu za  $N = 10$  koja proizlazi iz činjenice da ne postoji nijedna točka uzorkovanja blizu 1. Međutim, čak i u ovom slučaju, 95%-pouzdan interval uključuje točnu vrijednost.

No i dalje samo u slučaju  $N = 10000$  vidimo zadovoljavajuće ponašanje, što opet ukazuje na sporu konvergenciju grube Monte Carlo metode. Nadalje, ovaj vrlo jednostavan primjer jasno ističe jedan problem Monte Carlo integracije: Ako postoji mala dominantna regija (kao što je okolina 1 u našem primjeru), tada gruba Monte Carlo metoda zahtijeva ogroman broj uzorkovanih točaka da bi pružila zadovoljavajuće rezultate. Iz toga bi trebalo biti jasno da se obično ne bi trebale koristiti grube Monte Carlo metode za jednadimenzionalnu integraciju.

N	10	100	10000
$\hat{I}_{donja}$	0.047	0.297	0.325
$\hat{I}_N$	0.192	0.360	0.331
$\hat{I}_{gornja}$	0.338	0.423	0.337

Tablica 3.3: Monte Carlo integracija s 95%-pouzdanim intervalom za  $x^2$  na  $[0, 1]$  (točna vrijednost iznosi  $1/3$ )

## 3.3 Poboljšanje brzine konvergencije

### 3.3.1 Metode smanjivanja varijance

Glavni nedostatak grube Monte Carlo metode je njena spora konvergencija. U vjerojatnosnim terminima, to se izražava činjenicom da standardna devijacija pogreške opada samo kao kvadratni korijen u odnosu na potreban broj simulacija. Stoga, ako je moguće modificirati metodu tako da rezultira bržim smanjenjem varijance, moglo bi se ubrzati izračune u smislu da postizanje željene točnosti zahtijeva manje pokretanja simulacija. Svaka takva modifikacija grube Monte Carlo metode naziva se metodom smanjenja varijance.

#### Antitetičke varijable

Ovo je najlakša metoda za smanjivanje varijance. Temelji se na ideji kombiniranja nasumičnog izbora točaka sa sustavnim izborom. Glavni princip ove metode je smanjenje varijance uvođenjem simetrije. Pretpostavimo da želimo izračunati

$\mathbb{E}(f(X))$  gdje je  $X$  slučajna varijabla uniformno distribuirana na intervalu  $[0, 1]$ . Gruba Monte Carlo aproksimacija bi bila

$$\tilde{f}(X) = \frac{1}{N} \sum_{i=1}^N f(X_i),$$

gdje je  $X_i$  nezavisna kopija  $X$ , a u antitetičkoj metodi se također koriste i brojevi  $1 - X_i, \dots, 1 - X_N$  i uvodimo antitetičku Monte Carlo aproksimaciju

$$\tilde{f}_{anti}(X) = \frac{1}{2} \left( \frac{1}{N} \sum_{i=1}^N f(X_i) + \frac{1}{N} \sum_{i=1}^N f(1 - X_i) \right).$$

Napomenimo da su  $X$  i  $1 - X$  jednako distribuirani, obje sume s desne strane jednadžbe su nepristrani procjenitelji za  $\mathbb{E}(f(X))$ . Stoga, antitetička aproksimacija je također nepristrana. Neka je  $\sigma^2 = \text{Var}(f(X))$ . Tada je varijanca antitetičke aproksimacije dana izrazom

$$\text{Var}(\tilde{f}(X)) = \frac{\sigma^2}{2N} + \frac{1}{2N} \text{Cov}(f(X), f(1 - X)),$$

tj. imamo smanjenje varijance u usporedbi sa grubom Monte Carlo aproksimacijom temeljenom na  $2N$  slučajnih brojeva ako su  $f(X)$  i  $f(1 - X)$  negativno korelirani. Nadalje, treba napomenuti da također štedimo računalni napor jer trebamo generirati samo  $N$  slučajnih brojeva umjesto  $2N$ .

**Teorem 4** (Čebiševljeva nejednakost kovarijanci). [[1], str. 67.]

Neka je  $X$  realna slučajna varijabla, te neka su  $f, g$  neopadajuće funkcije s konačnom  $\text{Cov}(f(X), g(X))$ . Tada imamo:

$$\mathbb{E}(f(X)g(X)) \geq \mathbb{E}(f(X))\mathbb{E}(g(X)).$$

Ako izaberemo za  $g(X) = -f(1 - x)$  tada je direktno implicirana sljedeća tvrdnja.

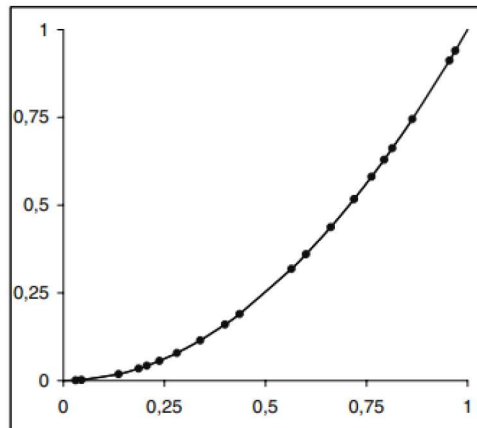
#### Korolar 1.

Neka je  $f$  neopadajuća ili nerastuća funkcija, te neka je  $X$  uniformno distribuirana na  $[0, 1]$  s konačnom  $\text{Cov}(f(X), f(1 - X))$ . Tada vrijedi:

$$\text{Cov}(f(X), f(1 - X)) \leq 0.$$

Uistinu, antitetička Monte Carlo aproksimacija bazirana na  $N$  slučajnih brojeva ima manju varijancu nego gruba Monte Carlo aproksimacija bazirana na  $2N$  slučajnih brojeva.

Demonstrirat ćemo korisnost ove metode na primjeru integracije na Slici 3.3. U prošlom dijelu loša izvedba je bila u slučaju korištenja  $N = 10$  slučajnih brojeva najviše iz razloga što je nedostajalo slučajnih brojeva oko 1. Ako dodamo točke  $1 - X_i$ , dobivamo antitetičku Monte Carlo aproksimaciju koja iznosi 0.340 što je čak i bolji rezultat nego gruba Monte Carlo aproksimacija na  $N = 100$  slučajnih



Slika 3.3: Antitetička Monte Carlo integracija funkcije  $g(X) = x^2$  na  $[0, 1]$ , točke predstavljaju uzorkovane vrijednosti za  $N = 10$  [1]

brojeva. Razlog leži u tome što je zbog simetričnosti skupa slučajnih brojeva sada i kritična okolina 1 pokrivena uzorkom. Ista činjenica je vidljiva i na sljedećoj slici.

Sada navedimo smanjenje varijance u normalnoj distribuciji. Neka je  $f$  neopadajuća ili nerastuća funkcija, te neka je  $X$  slučajna varijabla iz  $\mathcal{N}(\mu, \sigma^2)$  s konačnom  $Cov(f(X), f(2\mu - X))$ . Tada vrijedi

$$Cov(f(X), f(2\mu - X)) \leq 0.$$

Varijanca za antitetičku Monte Carlo aproksimaciju je

$$\tilde{\sigma}_{anti}^2 = \frac{1}{N-1} \sum_{i=1}^N \left( \frac{1}{2} (f(X_i) + f(\tilde{X}_i) - \tilde{f}_{anti}(X)) \right)^2$$

što vodi do aproksimacije za 95%-pouzdan interval za  $\mathbb{E}(f(X))$

$$\left[ \tilde{f}_{anti}(X) - 1.96 \frac{\tilde{\sigma}_{anti}}{\sqrt{N}}, \tilde{f}_{anti}(X) + 1.96 \frac{\tilde{\sigma}_{anti}}{\sqrt{N}} \right].$$

### Kontrolne varijable

Ova metoda koristi dodatne informacije o povezanim varijablama koje imaju poznatu očekivanu vrijednost kako bi se smanjila varijabilnost aproksimacije. Ako imamo varijablu  $Y$  za koju znamo očekivanu vrijednost  $\mu_Y$ , možemo koristiti ovu informaciju za poboljšanje aproksimacije ciljane varijable  $X$ .

Točnije, ako znamo slučajnu varijablu  $Y$  koja je (u nekom smislu) bliska  $X$  i za koju možemo točno izračunati  $\mathbb{E}(Y)$ , te onda se ta varijabla može koristiti kao **kontrolna varijabla**, tj. koristimo

$$\mathbb{E}(X) = \mathbb{E}(X - Y) + \mathbb{E}(Y),$$

što nas dovodi do **Monte Carlo aproksimacije s kontrolnim varijablama**

$$\tilde{X}_Y = \frac{1}{N} \sum_{i=1}^N (X_i - Y_i) + \mathbb{E}(Y)$$

za  $\mathbb{E}(\tilde{X}_Y)$  gdje su  $X_i, Y_i$  nezavisne kopije  $X$  i  $Y$ . Pomoću jednadžbe

$$\text{Var}(\tilde{X}_Y) = \frac{1}{N} \text{Var}(X - Y) = \frac{1}{N} (\text{Var}(X) + \text{Var}(Y) - 2\text{Cov}(X, Y))$$

dobivamo smanjenje varijance za aproksimaciju s kontrolnim varijablama u usporedbi s grubom aproksimacijom ako imamo

$$\text{Var}(X) \geq \text{Var}(X - Y).$$

Količina za koju se smanjuje  $\text{Var}(X)$  dana je s

$$2\text{Cov}(X, Y) - \text{Var}(Y).$$

Ako je  $Y$  vrlo blizu  $X$  to može dovesti do eliminacije gotovo cijele varijance grubog Monte Carlo aproksimatora korištenjem njegove varijante s kontrolnim varijablama. Međutim, učinkovitost kontrolne varijable također ovisi o činjenici

- da smo u mogućnosti direktno simulirati razliku  $X_i - Y_i$  kao jednu slučajnu varijablu (tj. znamo njezinu točnu distribuciju i dovoljno je simulirati samo jedan slučajni broj i koristiti odgovarajuću transformaciju)
- ili da moramo koristiti metodu inverzne transformacije za obje  $X_i$  i  $Y_i$  zasebno (ali i dalje koristimo isti slučajni broj za obje ili barem možemo izvući  $X_i$  i  $Y_i$  iz njihove zajedničke distribucije), što naravno zahtijeva veći napor nego u prethodnom slučaju.

Kako bi dobili interval pouzdanosti za Monte Carlo aproksimaciju s kontrolnim varijablama, treba koristiti interval pouzdanosti za grubu Monte Carlo aproksimaciju za  $\mathbb{E}(X - Y)$  i zatim jednostavno dodati  $\mathbb{E}(Y)$  ovom intervalu, tj. dobivamo približni 95% -interval pouzdanosti

$$\left[ \tilde{X}_Y - 1.96 \frac{\sigma_{\tilde{X}-Y}}{\sqrt{N}}, \tilde{X}_Y + 1.96 \frac{\sigma_{\tilde{X}-Y}}{\sqrt{N}} \right]$$

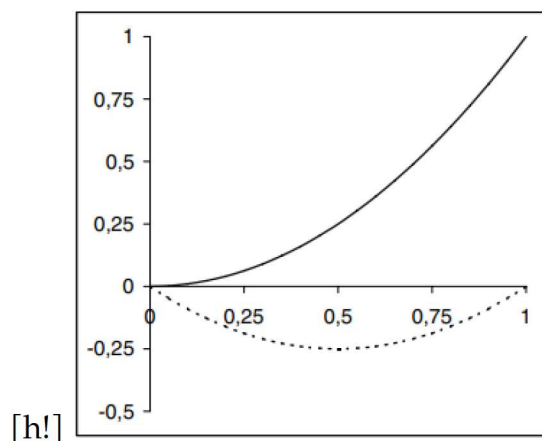
gdje je

$$\sigma_{\tilde{X}-Y}^2 = \frac{1}{N-1} \sum_{i=1}^N \left( X_i - Y_i - \frac{1}{N} \sum_{j=1}^N (X_j - Y_j) \right)^2.$$

Prvo ćemo ilustrirati metodu pomoću našeg standardnog primjera integracije. Za to jednostavno biramo  $Y$  da bude uniformno distribuiran na  $[0, 1]$ , dakle imamo  $X_i = Y_i^2$ . Budući da znamo da je  $E(Y) = 0.5$ , samo trebamo simulirati  $X_i$

$$X_i - Y_i = Y_i^2 - Y_i.$$

Gledajući grafikon ove razlike i uspoređujući ga s integrandom, jasno možemo očekivati smanjenje varijance (Slika 3.4). To je jednostavno zbog činjenice da  $g(x) = x^2$  ima mnogo veću varijaciju nego  $\hat{g}(x) = x^2 - x$ .



Slika 3.4: Funkcija  $g(x) = x^2$  i verzija s kontrolnim varijablama  $\tilde{g}(x) = x^2 - x$  (istcrtana linija) na  $[0, 1]$

Doista, brojevi prikazani u Tablici 3.3 impresivno ukazuju na smanjenje varijance u usporedbi s učinkom grube Monte Carlo aproksimacije (vidi Tablicu 3.4). Duljina pouzdanog intervala značajno je smanjena, što naglašava snagu metode kontrolnih varijabli kada je prikladna kontrola dostupna. Napomenimo da zbog zaokruživanja, pouzdani intervali nisu točno simetrični. Nadalje, točna aproksimacija za  $N = 10$  je nešto bolja od one za  $N = 100$ . Međutim, pouzdani interval za  $N = 100$  je znatno kraći nego onaj za  $N = 10$ , što naglašava da imamo pouzdaniju aproksimaciju u slučaju  $N = 100$ .

N	10	100	10000
$\hat{I}_{donja}$	0.292	0.327	0.331
$\hat{I}_N$	0.340	0.343	0.333
$\hat{I}_{gornja}$	0.389	0.359	0.334

Tablica 3.4: Integracija Monte Carlo metodom s kontroliranim varijablama  $x$  i granice 95%-pouzdanog intervala za  $\int_0^1 x^2 dx$  na  $[0, 1]$  (točna vrijednost=1/3) [1]

### Optimizacija kontrolne varijable

Ako smo pronašli kandidata  $Y$ , tada se zbog konstrukcije aproksimacije kontrolne varijable također može koristiti  $aY$  kao kontrolna varijabla za  $a > 0$ . To je zbog činjenice da je nova aproksimacija kontrolne varijable još uvijek nepristrana zbog linearnosti očekivanja. Također generira smanjenje varijance za  $a$  pozitivan  $a$  ako je  $Y$  već doveo do smanjenja varijance (primijetite da bi negativan  $a$  povećao varijancu aproksimacije kontrolne varijable). Dakle, optimalna upotreba kontrolne varijable  $Y$  postiže se uvođenjem multiplikatora  $a^*$  koji minimizira varijancu

$$g(a) = \text{Var}(X - aY) = \text{Var}(X) + a^2 \text{Var}(Y) - 2a \text{Cov}(X, Y) = \sigma^2 + a^2 \sigma_Y^2 - 2a \sigma_{X,Y}.$$

Iz ovoga dobivamo

$$a^* = \frac{\sigma_{XY}}{\sigma_Y^2}$$

sa stvarnim smanjenjem varijance

$$2a^* \text{Cov}(X, Y) - (a^*)^2 \text{Var}(Y) = \frac{\sigma_{XY}^2}{\sigma_Y^2}.$$

Označimo li korelaciju između  $X$  i  $Y$  s  $\rho_{X,Y}$  i iskoristimo li relaciju

$$\sigma_{XY} = \rho_{X,Y} \sigma_X \sigma_Y$$

dobivamo da je maksimalno smanjenje varijance dano izrazom

$$\frac{2a^* \text{Cov}(X, Y) - (a^*)^2 \text{Var}(Y)}{\text{Var}(X)} = \rho_{X,Y}^2.$$

Uočimo da maksimalno moguće smanjenje varijance opada kvadratno s korelacijom između  $X$  i njegove kontrolne varijable  $Y$ . Dakle, samo korištenje kontrolne varijable s visokim vrijednostima  $\rho_{X,Y}$  će učiniti metodu kontrolne varijable efektivnom. Ako imamo kontrolu sa recimo  $\rho_{X,Y} = 0.4$ , tada je maksimalno moguće smanjenje varijance samo 16% od originalne  $\sigma_X^2$ !

Ako su poznati i varijanca  $Y$  i kovarijanca između  $X$  i  $Y$ , tada možemo direktno koristiti  $a^*Y$  kao kontrolnu varijablu. Ako to nije slučaj, onda imamo mogućnost da procijenimo oba (ili samo  $\sigma_{XY}$  ako je barem  $\sigma_Y^2$  poznata) putem simulacijske procedure koja se može izvesti prije početka procedure kontrolne varijable. Također, možemo procijeniti nepoznate parametre tokom izvođenja procedure kontrolne varijable kao nusproizvod, a zatim kontinuirano ažurirati parametar  $a^*$  za aproksimaciju kontrolne varijable. U našem standardnom primjeru integracije s kontrolnom varijablom  $Y$ , optimalan izbor  $a$  može se eksplicitno izračunati kao  $a^* = 1$ . Dakle, zaista smo koristili najbolju **linearnu kontrolnu varijablu**.

### Višestruke kontrole

Zbog načina na koji je konstruirana aproksimacija s kontrolnim varijablama, možemo dodati još jednu varijablu  $Z$  u obliku

$$\tilde{X}_{Y,Z} = \tilde{X}_Y - \frac{1}{N} \sum_{i=1}^N Z_i + \mathbb{E}(Z).$$

Ovo nam opet daje nepristranog procjenitelja za  $\mu = E(X)$ . Dalje, dovodi do smanjenja varijance ako imamo

$$\text{Var}(Z) < 2\text{Cov}(X_Y, Z).$$

U posebnom slučaju nekoreliranih  $Z_i$  i  $Y_i$ , ovo se svodi na zahtjev

$$\text{Var}(Z) < 2\text{Cov}(X, Z).$$

Dakle, može se koristiti onoliko kontrolnih varijabli koliko se poželi. Konkretna primjena više kontrola na multivarijatnu situaciju kao što je

$$X = f(Y_1, \dots, Y_d)$$

bit će metoda nekondicionalnih kontrolnih varijabli.

### Kontrolne varijable i aproksimacije nizova

U ovom slučaju, pronalaženje dobre kontrolne varijable može biti izazovno. U našem primjeru integracije, već smo vidjeli da je kontrolna varijabla  $X$  najbolja linearna kontrolna varijabla (kao funkcija od  $X$ ). Mogli bismo također definirati najbolje aproksimacije višeg stupnja polinoma. Stoga pretpostavljamo da želimo aproksimirati

$$\mu = E(f(X))$$

i da imamo Taylorovu aproksimaciju reda  $k$  u sljedećem obliku:

$$f_k(x) = \sum_{j=0}^k \frac{f^{(j)}(x_0)}{j!} (x - x_0)^j.$$

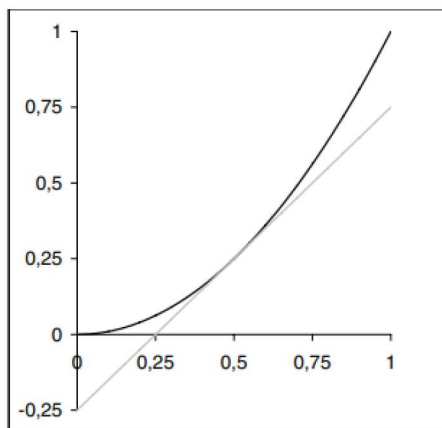
Glavno pitanje tada je možemo li odrediti optimalnu vrijednost  $x_0$  kako bismo postigli najveće moguće smanjenje varijance prilikom korištenja  $f_k(X)$  kao kovarijance. Naravno, to uvelike ovisi o našoj sposobnosti da izračunamo sve momente varijable  $X$  do reda  $k$  i, posebno, da izračunamo sve kovarijance  $Cov(X, f(X))$ . U našem primjeru integracije, implicitno smo učinili to jer smo mogli zamijeniti kontrolnu varijablu  $X$  za  $f(X) = X^2$  kontrolnom varijablom

$$f_1(X) = f'\left(\frac{1}{2}\right) \left(X - \frac{1}{2}\right) + f\left(\frac{1}{2}\right) = X - \frac{1}{4}.$$

Ova promjena je izvedena zato što se konstanta  $-\frac{1}{4}$  poništava u dijelu koji se odnosi na kovarijantu. Ovo nam omogućuje da bolje kontroliramo varijaciju u aproksimaciji, što vodi do smanjenja varijance i posljedično povećanja preciznosti Monte Carlo aproksimacije. Zamjenom originalne funkcije kontrolom koja bolje odgovara funkciji koju aproksimiramo, postiže se učinkovitija aproksimacija s manje simulacija, čime se smanjuje računsko opterećenje.

Dakle, najbolja linearna aproksimacija koju smo postigli u ovom slučaju je Taylorova aproksimacija prvog reda u točki  $x_0 = 1/2$ , koja se nalazi točno u središtu raspona varijable  $X$  (pogledajte Sliku 3.5 za ilustraciju).





Slika 3.5: Integrand  $x^2$  (crno) i najbolja linearna Taylorova aproksimacija prvog reda (sivo) [1]

Međutim, treba napomenuti da je Taylorova aproksimacija dobra samo lokalno. Stoga bi  $x_0$  trebao biti, na neki način, u središtu distribucije varijable  $X$ . Nadalje, kvaliteta ove aproksimacije naravno ovisi o kvaliteti aproksimacije Taylorovim polinomom. U nekim posebnim slučajevima, kao što su konveksne/konkavne funkcije  $f$  ili funkcije koje imaju reprezentaciju u obliku reda, možemo imati dobre granice iz eksplicitnog oblika ostatka Taylorovog polinoma. S druge strane, ne možemo ovdje iznijeti općeniti rezultat.

### Kontrolne varijable bezuvjetne srednje vrijednosti

Metoda bezuvjetnih kontrolnih varijabli prirodni je pristup za aproksimaciju očekivanja univarijatnih funkcija

$$E(g(X)) = E \left[ g \left( X^{(1)}, \dots, X^{(d)} \right) \right]$$

korištenjem  $d$  univarijatnih kontrola. Ove kontrole definirane su kao:

$$Y^{UM_j}(X) = g \left( \mu_1, \dots, \mu_{i-1}, X^{(i)}, \mu_{i+1}, \dots, \mu_d \right), \quad j = 1, \dots, d,$$

gdje je  $\mu_j = E \left[ X^{(j)} \right]$ , takozvana **bezuovjetna srednja kontrolna varijabla** (*engl.* "UMC"). Dakle, koriste se univarijatne verzije funkcije  $g(X)$  gdje je slobodno variranje dopušteno samo komponenti  $i$ , dok su ostale komponente postavljene na njihove srednje vrijednosti. Da bi metoda bila uspješna, moramo biti u mogućnosti izračunati sve očekivane vrijednosti kontrolnih varijabli. Možemo uvesti **aproksimaciju kontrolnih varijabli bezuvjetnih srednjih vrijednosti** kao

$$\tilde{X}_N^{UMC} = \frac{1}{N} \sum_{i=1}^N \left( g(X_i) - \sum_{j=1}^d Y^{UM_j}(X_i) \right) + \sum_{j=1}^d \mathbb{E}(Y^{UM_j}(X)).$$

U kontekstu financija, ovu metodu je uveo Paolo Pellizzari 2001. godine. Što se tiče intervala pouzdanosti i mogućeg smanjenja varijance sve što vrijedi za višestruke kontrolne varijable vrijedi i ovdje jer je ova metoda samo poseban primjer strategije višestrukih kontrolnih varijabli. Ilustriramo je jednostavnim primjerom gdje imamo slučajnu varijablu  $X = (X^{(1)}, X^{(2)}, X^{(3)})$  s multivarijatnom normalnom distribucijom, tj.

$$X \sim \mathcal{N} \left( \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 & 0.8 & 0.8 \\ 0.8 & 0.64 & 0.8 \\ 0.8 & 0.8 & 1 \end{pmatrix} \right).$$

Želimo aproksimirati

$$\mathbb{E}(g(X)) = \mathbb{E}(X^{(1)} \cdot X^{(2)} \cdot X^{(3)}).$$

Ovo implicira da su kontrolne varijable komponente  $X^{(i)}$ . Stoga je aproksimacija kontrolnih varijabli bezuvjetnih srednjih vrijednosti dana s

$$\tilde{X}_N^{UMC} = \frac{1}{N} \sum_{i=1}^N (X^{(1)} \cdot X^{(2)} \cdot X^{(3)} - X^{(1)} - X^{(2)} - X^{(3)}) + 3.$$

Simulacija s  $N = 10,000$  rezultirala je brojevima navedenim u Tablici 3.5. Primitimo da je varijanca smanjena za otprilike 25%, što je također rezultiralo kraćim pouzdanim intervalom.

Metoda	Srednja vrijednost	Donji kvantil	Gornji kvantil
CMC	3.240	3.120	3.361
UMCV	3.201	3.096	3.306

Tablica 3.5:  $\mathbb{E}(X^{(1)} \cdot X^{(2)} \cdot X^{(3)})$  aproksimirano s grubom Monte Carlo metodom ("CMC") i metodom kontrolnih varijabli bezuvjetnih srednjih vrijednosti (UMCV),  $N = 10000$  [1]

### Kontrolne varijable nisu uvijek dobre

U stvarnosti je to trivijalno, ali postoje i loše kontrolne varijable. Kad god postoji dobra kontrolna varijabla  $Y$  (mjerena visokom vrijednošću  $\rho_{X,Y}$ ), postoji i loša, a to je  $-Y$ . Da bi vidjeli, jednostavno primijetite da imamo

$$\rho_{X,Y} = -\rho_{X,-Y},$$

što znači da bi korištenje  $-Y$  kao kontrolne varijable rezultiralo povećanjem varijance odgovarajuće aproksimacije. Svrha ovog umjetnog primjera je, međutim, istaknuti da bi trebalo imati barem heuristički argument da postoji pozitivna korelacija između slučajne varijable  $X$  i kontrolne varijable  $Y$  ako ne znamo  $\rho_{X,Y}$  eksplicitno.

Ako smo zaista nesigurni u učinak smanjenja varijance kontrolne varijable, trebali bismo uzeti nekoliko realizacija obje varijable  $X$  i  $Y$ , procijeniti kovarijancu  $\sigma_{XY}$  i  $\sigma_Y^2$  iz tih realizacija, te na kraju koristiti  $aY$  s  $a = \hat{\sigma}_{XY} / \hat{\sigma}_Y^2$  kao kontrolnu varijablu.

### 3.3.2 Uzorkovanje po važnosti

"Uzorkovanje po važnosti" (engl. *Importance sampling*) se temelji na izravnoj transformaciji funkcije gustoće  $X$  (ili transformaciji funkcije distribucije vjerojatnosti, u slučaju da je  $X$  diskretna slučajna varijabla). Glavna ideja **uzorkovanja po važnosti** je jednostavno pronaći distribuciju za temeljnu slučajnu varijablu koja dodjeljuje visoku vjerojatnost onim vrijednostima koje su važne za izračunavanje objekta našeg interesa odnosno očekivanje naše transformirane varijable  $E(g(X))$ . Kako bismo motivirali metodu, ponovno ćemo se osvrnuti na Monte Carlo integraciju (Primjer 5). Funkcija  $f(x)$  je funkcija gustoće uniformne distribucije na  $U[0, 1]^d$ , pa vrijedi relacija

$$\int_{[0,1]^d} g(x)dx = \int g(x)f(x)dx = E(g(X))$$

koja nam omogućuje aproksimaciju determinističkog integrala korištenjem  $N$  nezavisnih, na  $[0, 1]^d$  uniformno distribuiranih slučajnih varijabli  $X_1, \dots, X_N$  kako bismo dobili grubu Monte Carlo aproksimaciju oblika

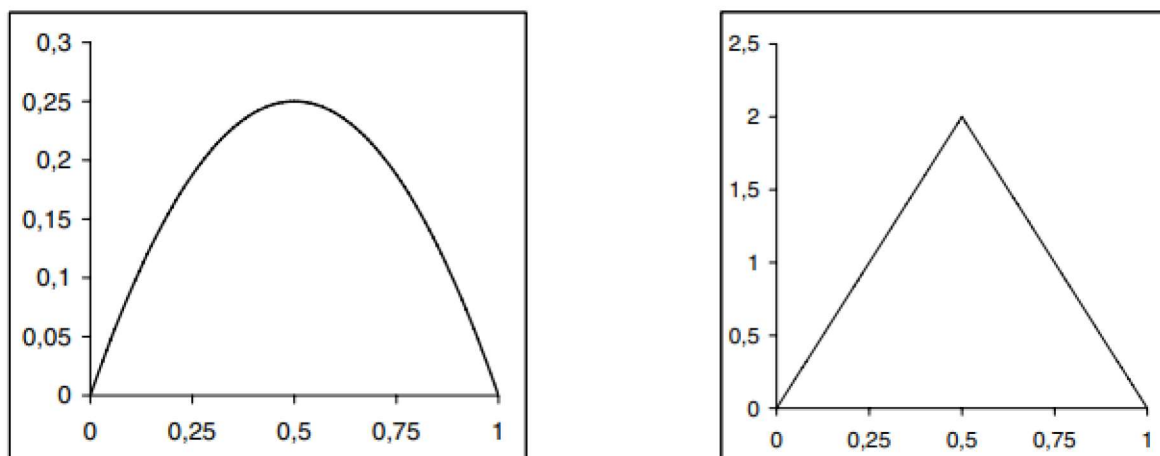
$$\hat{I}_N(\omega) = \frac{1}{N} \sum_{i=1}^N g(X_i(\omega)).$$

Međutim, ako pažljivo proučimo funkciju  $g(x)$ , možemo poboljšati točnost gornje aproksimacije tako što ćemo generirati više slučajnih brojeva u područjima gdje  $g(x)$  ima velike vrijednosti (u apsolutnom smislu), dok u područjima gdje je  $g(x)$  blizu 0, samo je nekoliko uzoraka potrebno za predviđanje doprinosa funkcije na tom području integracije. Jasno je da, kako bismo iskoristili naše znanje o funkciji  $g(x)$  na ovaj način, moramo promijeniti distribuciju slučajne varijable  $X$  na odgovarajući način, jer bismo inače promijenili vrijednost očekivanja.

Točnije, poboljšanje Monte Carlo metode se može ostvariti generiranjem više uzoraka u područjima gdje funkcija  $g(x)$  ima značajne vrijednosti, a manje uzoraka tamo gdje je  $g(x)$  blizu nule. Na taj način možemo smanjiti broj nepotrebnih uzoraka i povećati preciznost aproksimacije integrala. No, kako bismo postigli taj cilj, moramo promijeniti distribuciju slučajne varijable  $X$  iz koje uzorkujemo, kako bismo osigurali da je uzorkovanje više fokusirano na važna područja funkcije  $g(x)$ . Ali pritom moramo biti oprezni jer promjena distribucije može utjecati na izračun očekivanja, što želimo izbjeći ili nadoknaditi na odgovarajući način (ovo je temelj metode **uzorkovanja po važnosti**).

Kao primjer, razmotrimo slučaj kada je  $d = 1$ , a  $g(x) = x \cdot (1 - x)$ , što je očito nenegativna i simetrična funkcija na  $[0, 1]$ , jednaka je 0 za  $x \in \{0, 1\}$  te postiže svoj maksimum od 0.25 za vrijednost  $x = 0.5$ . U smislu prethodne rasprave, umjesto korištenja uniformne distribucije na  $[0, 1]$ , bilo bi bolje koristiti trokutastu distribuciju za slučajnu varijablu  $X$ , tj. trebala bi imati funkciju gustoće vjerojatnosti (slika 3.6)

$$\tilde{f}(x) = \begin{cases} 0, & \text{za } x \leq 0 \text{ ili } x \geq 1, \\ 4x, & \text{za } 0 < x < \frac{1}{2}, \\ 4 - 4x, & \text{za } \frac{1}{2} \leq x < 1. \end{cases}$$



Slika 3.6: Prikaz integranda  $x(1-x)$  (lijevo) i funkcije gustoće (desno)

Kako se vrijednost integrala ne smije promijeniti pri uzorkovanju varijable s funkcijom gustoće  $\tilde{f}$ , moramo podijeliti  $g(x)$  s  $\tilde{f}(x)$ :

$$\int_0^1 x(1-x)dx = \int_0^1 \frac{x(1-x)}{\tilde{f}(x)} \tilde{f}(x)dx.$$

To znači da kada koristimo "novu" distribuciju zapravo uzrokuje varijablu  $X(1-X)/\tilde{f}(X)$  kako bi dobili novu Monte Carlo aproksimaciju

$$\bar{I}_{\text{"nova"}} = \frac{1}{N} \sum_{i=1}^N \frac{X_i(1-X_i)}{\tilde{f}(X_i)}.$$

Treba napomenuti da je distribucija varijabli  $X_i$  sada određena funkcijom gustoće  $\tilde{f}$ . Jednostavna usporedba korištenja Monte Carlo aproksimatora za  $N = 1000$  između aproksimacije korištenjem uniformne i aproksimacije korištenjem trokutaste ("nove") distribucije pokazuje superiornost "nove" metode. Rezultat "nove" metode iznosi 0.168 s 95%-pouzdanim intervalom od  $[0.166, 0.170]$ , dok rezultat grube metode iznosi 0.163 s 95%-pouzdanim intervalom od  $[0.158, 0.166]$  pri čemu je točna vrijednost  $1/6$ . "Nova" metoda također rezultira manjom varijansom aproksimacije

$$\begin{aligned} \text{Var}(\bar{I}_{\text{gruba}}) &= \frac{1}{180N}, \\ \text{Var}(\bar{I}_{\text{"nova"}}) &= \frac{1}{1152N} = \frac{1}{64 \cdot 18N}. \end{aligned}$$

Tako smo smanjili varijancu na manje od jedne šestine varijance originalne grube Monte Carlo aproksimacije koristeći predloženu metodu. U općenitom slučaju pokušavamo formalizirati ovaj pristup kako bismo primijenili ideju **importance sampling-a** za izračunavanje:

$$E(g(X)) = \int g(x)f(x) dx,$$

gdje je  $X$  slučajna varijabla s vrijednostima u  $\mathbb{R}^d$  koja ima gustoću  $f(x)$ , i gdje pretpostavljamo da očekivanje postoji za funkciju  $g : \mathbb{R}^d \rightarrow \mathbb{R}$ . Za svaku funkciju gustoće  $\tilde{f}(x)$  na  $\mathbb{R}^d$  sa svojstvom

$$\tilde{f}(x) > 0, \forall x \text{ za koji vrijedi } f(x) > 0$$

i njenu pridruženu vjerojatnosnu mjeru  $\tilde{\mathbb{P}}$ , uvodimo  $\tilde{g}(x)$  putem relacije

$$\begin{aligned} E(g(X)) &= \int g(x)f(x)dx = \int \frac{g(x)f(x)}{\tilde{f}(x)}\tilde{f}(x)dx \\ &= \tilde{\mathbb{E}}\left(\frac{g(X)f(X)}{\tilde{f}(X)}\right) = \tilde{\mathbb{E}}(\tilde{g}(X)). \end{aligned} \quad (3.3)$$

Ovdje  $\tilde{\mathbb{E}}$  označava očekivanje s obzirom na mjeru  $\tilde{\mathbb{P}}$ . Funkcija težine  $\frac{f(X)}{\tilde{f}(X)}$  naziva se funkcijom omjera vjerojatnosti u gore navedenoj promjeni mjere s  $\mathbb{P}$  na  $\tilde{\mathbb{P}}$ . Aproximacija metodom **uzorkovanja po važnosti** (s obzirom na  $\tilde{f}(x)$  za  $\mu = \mathbb{E}(g(X))$ ) definira se kao:

$$\bar{I}_{\text{nova}, \tilde{f}, N}(g(X)) = \frac{1}{N} \sum_{i=1}^N \tilde{g}(X_i) = \frac{1}{N} \sum_{i=1}^N \frac{g(X_i)f(X_i)}{\tilde{f}(X_i)}$$

gdje su  $X_i$  neovisni i distribuirani prema funkciji gustoće **importance sampling-a**  $\tilde{f}$ .

Dakle, aproksimator **importance sampling-a** je ponderirani grubi Monte Carlo aproksimator, gdje su težine za svako opažanje  $X_i$  određene funkcijom omjera vjerojatnosti. Primijetite da je, zbog reprezentacije (3.3) aproksimacija nepristrana i konzistentna. Njena varijanca je dana izrazom:

$$\begin{aligned} \sigma_{\text{nova}, \tilde{f}, N}^2 &= \tilde{Var}\left(\bar{I}_{\text{nova}, \tilde{f}, N}(g(X))\right) \\ &= \frac{1}{N} \tilde{Var}(\tilde{g}(X)) = \frac{1}{N} \left(\tilde{\mathbb{E}}(\tilde{g}(X)^2) - \mu^2\right) \\ &= \frac{1}{N} \left(\int g(x)^2 \frac{f(x)}{\tilde{f}(x)} \tilde{f}(x) dx - \mu^2\right). \end{aligned} \quad (3.4)$$

Budući da je aproksimacija metodom uzorkovanja po važnosti opet prosjek nezavisnih i identično distribuiranih slučajnih varijabli, centralni granični teorem (Teorem 3) daje pouzdani interval

95% – **pouzdan interval za  $\mathbb{E}(g(X))$**

$$\left(\bar{I}_{\text{nova}, \tilde{f}, N}(g(X)) - \frac{1.96\tilde{\sigma}_{\text{nova}, \tilde{f}, N}}{\sqrt{N}}, \bar{I}_{\text{nova}, \tilde{f}, N}(g(X)) + \frac{1.96\tilde{\sigma}_{\text{nova}, \tilde{f}, N}}{\sqrt{N}}\right)$$

gdje  $\tilde{\sigma}_{\text{nova}, \tilde{f}, N}$  označava standardnu devijaciju aproksimacije metodom uzorkovanja po važnosti. Ako pretpostavimo da je  $g(x) \geq 0$  za sve  $x \in \mathbb{R}^d$ , tada postoji

jedna posebno zanimljiva karakteristika u ovoj reprezentaciji. Naime, ako odaberemo

$$\tilde{f}(x) = c \cdot f(x) \cdot g(x) = \frac{f(x) \cdot g(x)}{\int f(y) \cdot g(y) dy}$$

tada je  $\tilde{f}(x)$  funkcija gustoće na  $\mathbb{R}^d$  i imali bismo  $\tilde{g}(X) = 1/c$ , tj.

$$\tilde{Var} \left( \bar{I}_{\text{„nova“}, \tilde{f}, N}(g(X)) \right) = 0.$$

Međutim, nedostatak ovog pristupa je da je konstanta  $c$  zapravo vrijednost koju pokušavamo izračunati našim Monte Carlo pristupom, jer imamo  $\mu = 1/c$ . Dakle, ako već znamo  $c$ , ne bi nas zanimalo izvođenje uzorkovanja po važnosti. S pozitivne strane, ovaj odabir ima sljedeću posljedicu.

**Teorem 5** (Smanjenje varijance pomoću uzorkovanja po važnosti).

Neka je  $g(x)$  nenegativna funkcija. Tada postoje odabiri funkcija gustoća vjerojatnosti za uzorkovanje po važnosti  $\tilde{f}$  takvi da vrijedi:

$$\tilde{Var} \left( \bar{I}_{\text{„nova“}, \tilde{f}, N}(g(X)) \right) < Var \left( \bar{I}(g(X))_N \right),$$

gdje je  $\bar{I}(g(X))_N$  grubi Monte Carlo aproksimator za  $E(g(X))$ . Nadalje, za sve funkcije  $\tilde{f}$  koje zadovoljavaju uvjet  $\tilde{f}(x) > 0, \forall x$  gdje  $f(x) > 0$  dobivamo:

$$\begin{aligned} Var \left( \bar{I}(g(X))_N \right) - \tilde{Var} \left( \bar{I}_{\text{„nova“}, \tilde{f}, N}(g(X)) \right) \\ = \frac{1}{N} \int g(x)^2 \left( 1 - \frac{f(x)}{\tilde{f}(x)} \right) f(x) dx. \end{aligned} \quad (3.5)$$

Primijetimo da zadnja relacija proizlazi iz reprezentacije (3.4), činjenice da su obje aproksimacije nepristrane, te uobičajene reprezentacije za  $E(g(X)^2)$ .

Nadalje, ova zadnja relacija daje naslutiti strukturu dobre gustoće za importance sampling  $\tilde{f}$ . Kako bismo postigli smanjenje varijance, trebali bismo zadovoljiti sljedeće uvjete:

- $\tilde{f}(x)$  bi trebala biti velika (u smislu  $\tilde{f}(x) > f(x)$ ) kad god je  $g(x)^2 f(x)$  veliko.
- $\tilde{f}(x)$  bi trebala biti mala (u smislu  $\tilde{f}(x) < f(x)$ ) kad god je  $g(x)^2 f(x)$  malo.

Naravno, moraju postojati vrijednosti  $x$  za koje je  $\tilde{f}(x) < f(x)$ . U svrhu smanjenja varijance, to bi se trebalo događati samo kada produkt  $g(x)^2 f(x)$  nema veliki utjecaj. Osim dva navedena uvjeta, gustoća za metodu uzorkovanja po važnosti trebala bi zadovoljiti i praktične zahtjeve kao što su:

- $\tilde{f}(x)$  bi trebala biti jednostavna za izračunavanje.
- Slučajne varijable s gustoćom  $\tilde{f}$  trebale bi biti jednostavne za simulaciju.

Dakle, dobra gustoća za importance sampling trebala bi biti kompromis između sličnosti s  $g^2 \cdot f$  i praktične primjenjivosti.

### 3.3.3 Primjena metoda smanjenja varijanci

Vidjeli smo različite metode za ubrzanje konvergencije grube Monte Carlo metode. Naravno, najzanimljivija pitanja su koja se metoda kada koristi i koja je najbolja. Ne postoji jasan odgovor na ta pitanja. Međutim, pokušat ćemo formulirati nekoliko jednostavnih savjeta za primjenu Monte Carlo metoda:

- Ako imamo očiglednu dobru kontrolnu varijablu, onda ju trebamo iskoristiti. Često, u toj situaciji, korisno je koristiti antitetičke varijable prije primjene metode kontrolnih varijabli.
- Za izračunavanje očekivanja gdje rijetki događaji igraju važnu ulogu, metoda **uzorkovanja po važnosti** je obično poželjna metoda, ponekad i jedina koja funkcionira.
- Ako ne postoji očigledan razlog za primjenu metode smanjenja varijance, budimo oprezni s njezinom primjenom jer može dovesti do uzaludnog trošenja vremena za računanje.

Odlučiti treba li uvesti stratificirani uzorak ovisi o problemu. U jednodimenzionalnom slučaju često je lako uvesti stratifikaciju koja znatno smanjuje varijancu kada treba procijeniti  $E(g(X))$ . Važno je koristiti detaljno znanje o distribuciji  $X$  i ponašanju funkcije  $g(x)$ . U takvim slučajevima, obično će stratificirani uzorak<sup>1</sup> biti bolji od pristupa kontrolnih varijabli. Međutim, u višedimenzionalnim problemima stratificirani uzorak pati od 'prokletstva dimenzionalnosti'.

Još jedan važan čimbenik u odlučivanju o razini sofisticiranosti tehnike smanjenja varijance je daljnja primjena razvijenog algoritma. Ako će se algoritam često koristiti na problemskim situacijama koje se parametarski mijenjaju (kao što je određivanje cijena opcija u financijskom softveru), gdje je također važno brzo izračunati pouzdano rješenje, tada je preporučljivo provesti detaljnu analizu problema. U takvim slučajevima bi trebalo razmotriti kombinaciju različitih tehnika smanjenja varijance kako bi se smanjila što je više moguće. Ipak, treba ponovno naglasiti da smanjenje varijance nije jedini aspekt učinkovitosti.

S druge strane, ako se Monte Carlo pristup odabire samo kako bi se izračunalo jedno jedino očekivanje, ponekad potraga za prikladnom metodom smanjenja varijance može trajati duže nego korištenje grubog Monte Carlo aproksimatora s vrlo velikim brojem ponavljanja  $N$ .

---

<sup>1</sup>Stratificirani uzorak vrsta je slučajnog uzorka koji unutar uzorka odražava točno postotak pojedinih slojeva (stratuma) populacije.

## 3.4 Važnost generatora slučajnih brojeva

Generatori slučajnih brojeva su ključni za Monte Carlo metodu aproksimacije i integracije zbog sljedećih razloga:

- **1. Osnova Monte Carlo metode:**

Monte Carlo metoda se oslanja na generiranje velikog broja slučajnih brojeva kako bi se simulirali slučajni uzorci iz određene distribucije. Ovi uzorci se koriste za aproksimaciju matematičkih očekivanja i rješavanje integrala. Bez pouzdanih generatora slučajnih brojeva, rezultati Monte Carlo simulacija ne bi bili precizni ili pouzdani.

- **2. Kvaliteta slučajnih brojeva:**

Kvaliteta generatora slučajnih brojeva direktno utječe na točnost i pouzdanost rezultata Monte Carlo integracije. Loši generatori mogu proizvesti sekvence brojeva s korelacijama ili neadekvatnom distribucijom, što može dovesti do netočnih rezultata i pogrešnih zaključaka.

- **3. Uniformna distribucija:**

Za Monte Carlo integraciju je važno da slučajni brojevi budu uniformno distribuirani na intervalu  $[0, 1]$ . Uniformna distribucija osigurava da svaki dio integrala bude adekvatno uzorkovan, čime se postiže točna aproksimacija vrijednosti integrala.

- **4. Reproductivnost:**

Korištenje istih početnih uvjeta odnosno vrijednosti (seed) u generatorima omogućava reproductivnost rezultata. To je važno za validaciju i verifikaciju rezultata simulacija, kao i za uspoređivanje različitih metoda ili parametara.

- **5. Veliki broj generiranih slučajnih brojeva:**

Monte Carlo metoda zahtijeva generiranje velikog broja slučajnih brojeva. Učinkoviti i brzi generatori su stoga neophodni kako bi se simulacije mogle izvršiti u razumnom vremenskom roku.

- **6. Smanjenje varijance:**

Različite tehnike smanjenja varijance mogu se primijeniti kako bi se poboljšala točnost Monte Carlo metode. Kvalitetni generatori slučajnih brojeva omogućuju primjenu ovih tehnika, što rezultira boljim rezultatima i bržom konvergencijom.



## 3.5 Primjena Monte Carlo metode

**Primjer 6 (Aproksimacija integrala s funkcijom  $xe^x$ ).**

Recimo da želimo izračunati integral

$$\int_0^1 xe^x dx.$$

Integral možemo riješiti pomoću parcijalne integracije, no možemo primjeniti i Monte Carlo metodu. Kako je opći oblik našeg integrala

$$\int_a^b h(x)g(x)dx$$

moramo okarakterizirati funkcije  $h$  i  $g$ . Prvi korak je da za funkciju  $h$  uzmemo intengrand odnosno u našem primjeru

$$h(x) = xe^x.$$

Drugi korak je da odredimo funkciju  $g$  koja je u ovom slučaju

$$g(x) = 1, x \in (a, b).$$

Iz ovoga možemo također odrediti funkciju distribucije vjerojatnosti  $p(x)$  unutar intervala  $(a, b) = (0, 1)$ .

$$p(x) = \frac{g(x)}{\int_a^b g(x)dx} = \frac{1}{b-a}.$$

U trećem koraku možemo vidjeti da izraz s desne strane predstavlja definiciju uniformne distribucije  $\mathcal{U}(0, 1)$ . Također, primijetite da je konstanta  $C = 1$ , gdje je  $\int_a^b g(x) = C < \infty$ . Četvrti korak je izračunavanje Monte Carlo aproksimacije na sljedeći način:

$$I = C \times \mathbb{E}_p h(X) \approx \frac{1}{N} \sum_{i=1}^N x_i e^{x_i},$$

gdje je svaki  $x_i$  uzorkovan iz standardne uniformne distribucije. Ispod se nalazi primjer simulacije u R-u s različitim brojem slučajnih brojeva

**Kod 1** (Monte Carlo aproksimacija integrala za dva uzorka slučajnih brojeva).

```
## Monte Carlo aproksimacija integrala
## za dva razlicita uzorka slucajnih brojeva

## I. Prvi uzorak
N1 = 100
x1 = runif(N1, 0, 1)
I_1 = sum((x1 * exp(x1))) / N1
I_1 = 1.03545
```

```
## II. Drugi uzorak
N2 = 10000
x2 = runif(N2, 0, 1)
I_2 = sum((x2 * exp(x2))) / N2
I_2 = 1.000887
```

**Primjer 7 (Optimizacija integrala funkcije  $e^{-\frac{(x-4)^2}{2}}$ ).**

Recimo da želimo pronaći vrijednost  $x_{opt}$  koja maksimizira funkciju

$$g(x) = e^{-((x-4)^2)/2}.$$

Točnije, želimo riješiti problem

$$x_{opt} = \operatorname{argmax}_{x \in \mathbb{R}} e^{-\frac{(x-4)^2}{2}}.$$

Ovaj problem je moguće riješiti klasičnim računskim metodama, no možemo biti dosjetljivi i iskoristiti Monte Carlo metodu aproksimacije. U tom slučaju proenstveno primjećujemo da je funkcija  $g(x)$  skalirana verzija normalne distribucije čije je očekivanje 4, a varijanca 1, tj.

$$g(x) = C \times \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-4)^2}{2}}$$

gdje je  $C = \sqrt{2\pi}$ .

To znači da možemo pronaći rješenje za  $x_{opt}$  na način da uzorkujemo slučajne brojeve iz normalne distribucije i odredimo gdje ti uzorci imaju najveću gustoću.

Ispod možemo vidjeti i R kod za rješavanje pripadnog problema.

**Kod 2** (Monte Carlo optimizacija integrala).

```
# MONTE CARLO aproksimacija za exp(x-4)^2

set.seed(12345) # Postavi seed za reproducibilnost

# Inicijalizacija
N <- 100000
x <- seq(0, 6, by = 0.1)
C <- sqrt(2 * pi)
g <- function(x) exp(-0.5 * (x - 4)^2)

# Plot g(x)/C i g(x)
plot(x, g(x) / C, type = "l", col = "darkred",
     lwd = 3, ylim = c(0, 1), ylab = "Gustoca",
     xlab = "x", ,main="Graficki prikaz")
lines(x, g(x), col = "navy", lwd = 2)
```

```

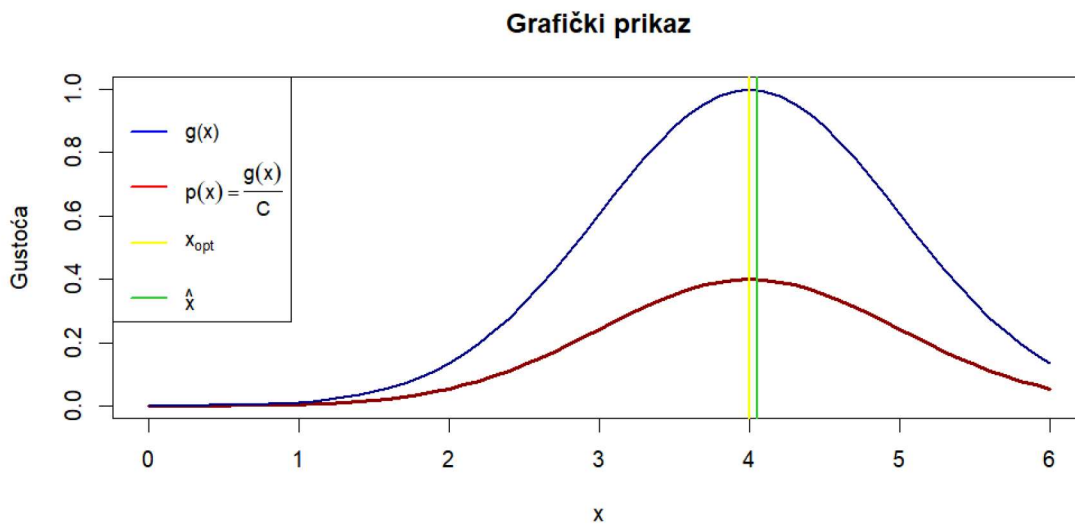
#N(4,1)
y <- dnorm(x, mean = 4, sd = 1)

# Racunanje MONTE CARLO aproksimacije
x_samples <- rnorm(N, mean = 4, sd = 1)
# Generiraj normalno distribuirane slucajne brojeve
bins <- seq(min(x_samples), max(x_samples), length.out = 100)
counts <- hist(x_samples, breaks = bins, plot = FALSE)$counts
optIdx <- which.max(counts) # Pronadi indeks maksimalne vrijednosti
xHat <- bins[optIdx] # Pronadi x koji odgovara maksimalnom binu

# Optimalna vrijednost i procijenjena optimalna vrijednost
abline(v = 4, col = "yellow", lwd = 2) # Pravac na tocki x_opt = 4
abline(v = xHat, col = "limegreen", lwd = 2)
# Pravac na procijenjenoj tocki xHat

#Legenda plota
legend("topleft", legend = c("g(x)", expression(p(x)==frac(g(x), C)),
expression(x[opt]), expression(hat(x))),
col = c("blue", "red", "yellow", "limegreen"),
lwd = 2, cex =0.9)

```



Slika 3.7: Grafički prikaz provedene aproksimacije

U gornjem prikazu vidimo funkciju  $g(x)$  (ona koju želimo optimizirati) u plavoj boji te normalnu distribuciju  $p(x)$  iz koje uzimamo uzorke u crvenoj boji. Monte Carlo metoda pruža dobru aproksimaciju (zelena boja) stvarnom rješenju (žuta boja).

### Primjeri primjene Monte Carlo aproksimacije u financijama

Metoda Monte Carlo aproksimacije ima široku primjenu u financijama i jedan je od dosta važnih načina izračuna u tom području, a aproksimacija cijene opcija je jedan od klasičnih primjera.

Opcije su financijski instrumenti čija cijena ovisi o budućim kretanjima cijena osnovne imovine, što uključuje određenu razinu nesigurnosti. Monte Carlo metoda se koristi kako bi se simulirala buduća kretanja cijena te na temelju tih simulacija aproksimirala cijena te opcije.

Radi lakše shvaćanja primjera podsjetimo se samo definicije Europske call opcije.

**Definicija 3.** *Europska call opcija je ugovor koji vlasniku daje pravo, ali ne nameće obavezu, da kupi određeni financijski instrumentu trenutku  $T > 0$  po cijeni  $K$  određenoj u trenutku  $t = 0$ .*

**Primjer 8** (Aproksimacija cijene Europske call opcije).

Koristimo **Black-Scholesov model** za kretanje cijene dionice, gdje je kretanje cijene dionice  $S_t$  opisano sljedećim diferencijalnim jednadžbama:

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

- $S_t$  je cijena osnovne imovine u trenutku  $t$ ,
- $\mu$  je očekivani prinos (drift),
- $\sigma$  je volatilitnost (st. devijacija povrata imovine),
- $W_t$  je Wienerov proces (Brownovo gibanje).

Korištenjem ove jednadžbe možemo simulirati cijene dionica kroz vrijeme, a zatim pomoću Monte Carlo metode aproksimirati cijenu opcije.

### Monte Carlo simulacija

Cijena Europske call opcije može se aproksimirati kao očekivana vrijednost isplate, diskontirana na sadašnji trenutak, gdje isplata opcije na datumu dospijeca  $T$  ovisi o tome hoće li cijena osnovne imovine ispuniti uvjet  $S_T > K$ :

$$C = e^{-rT} \mathbb{E}[\max(S_T - K, 0)],$$

gdje je  $r$  bezrizična kamatna stopa,  $T$  je vrijeme do dospijeca, a  $K$  je dogovorena cijena. Koristit ćemo Monte Carlo metodu da aproksimiramo cijenu Europske call opcije uz sljedeće parametre:

- $S_0 = 100$  (početna cijena osnovne imovine),
- $K = 105$  (dogovorena cijena),
- $r = 0.01$  (godišnja kamatna stopa),
- $\sigma = 0.2$  (volatilitnost),

- $T = 1$  godina (vrijeme do dospijeca),
- $N = 10,000$  (broj simulacija).

**Kod 3** (Aproksimacija cijene Europske call opcije).

```
# Postavke
set.seed(123)
S0 <- 100      # Pocetna cijena
K <- 105       # Dogovoreni iznos iznos
r <- 0.01      # Kamatna stopa
sigma <- 0.2   # Volatilnost
T <- 1         # Vrijeme do dospijeca (u godinama)
N <- 10000     # Broj simulacija

# Simuliranje buduće cijene pomoću Black-Scholesovog modela
Z <- rnorm(N)  # Generiraj N nasumičnih brojeva iz
standardne normalne distribucije
ST <- S0 * exp((r - 0.5 * sigma^2) * T + sigma * sqrt(T) * Z)
# Buduće cijene

# Izračunaj isplatu opcije
payoff <- pmax(ST - K, 0)

# Monte Carlo aproksimacija cijene opcije
C <- mean(payoff) * exp(-r * T) # Diskontiraj
isplatu na sadašnji trenutak

# Ispis procijenjene cijene opcije
C=6.262963
```

*Gore se nalazi R kod čijim pokretanjem dobivamo procijenjenu cijenu Europske call opcije. Na temelju parametara iz našeg primjera, Monte Carlo simulacija može proizvesti približnu cijenu od otprilike 6.263 (ovisno o broju simulacija).*

*Pojasnimo još jednom kako točno rješavamo ovaj problem:*

- Simuliramo 10,000 različitih mogućih cijena  $S_T$  na kraju godine.
- Zatim izračunamo isplatu za svaku simulaciju (razlika između  $S_T$  i dogovorene cijene  $K$ , ako je veća od nule).
- Konačno, diskontiramo prosječnu isplatu na sadašnji trenutak, koristeći kamatnu stopu  $r$ .

*Ova metoda je vrlo fleksibilna i može se primijeniti na složenije opcije, kao što su barijerne opcije, azijske opcije i mnoge druge financijske instrumente.*

**Primjer 9 (Aproksimacija vrijednosti portfelja).**

U sljedećem primjeru ćemo napraviti aproksimaciju vrijednosti portfelja koji sadrži više vrsta imovina.

**Definicija 4 (Vrijednost pod rizikom).** Vrijednost pod rizikom (VaR) je mjera rizika koja procjenjuje maksimalni gubitak portfelja unutar određenog vremenskog razdoblja, s određenom razinom pouzdanosti (obično 95% ili 99%).

Monte Carlo metoda koristi se za simulaciju mnogih mogućih scenarija kretanja cijena imovine u portfelju, kako bismo izračunali koliko možemo izgubiti u "najgorim" slučajevima.

Parametri:

- Portfelj se sastoji od tri vrste imovina: dionica A, B i C.
- Početna vrijednost portfelja je 100.000 EUR raspoređena na sljedeći način:
  - 40% u dionici A
  - 35% u dionici B
  - 25% u dionici C
- Promatrano vremensko razdoblje od jednog dana.
- Pretpostavljamo da povrati dionica slijede normalnu distribuciju sa sljedećim parametrima:
  - Dionica A:  $\mu_A = 0.001$ ,  $\sigma_A = 0.02$ ,
  - Dionica B:  $\mu_B = 0.0015$ ,  $\sigma_B = 0.015$ ,
  - Dionica C:  $\mu_C = 0.002$ ,  $\sigma_C = 0.01$ .
- - Želimo izračunati 99% VaR koristeći Monte Carlo simulaciju.

Koristit ćemo Monte Carlo simulaciju kako bismo simulirali povrate za svaku dionicu, te zatim izračunali koliko portfelj može izgubiti u najgorem scenariju.

**Kod 4 (Simulacija vrijednosti VaR-a).**

```
# Postavke portfelja
initial_value <- 100000 # Pocetna vrijednost portfelja u eurima
weights <- c(0.4, 0.35, 0.25) # Udio dionica A, B, C u portfelju

# Parametri povrata (srednja vrijednost i volatilitnost)
mu <- c(0.001, 0.0015, 0.002) # Ocekivani povrati
sigma <- c(0.02, 0.015, 0.01) # Volatilitnost

# Broj simulacija i vremenski horizont
N <- 10000 # Broj simulacija
```

```
period <- 1 # Jednodnevni period

# Monte Carlo simulacija povrata
set.seed(123)
simulated_returns <- matrix(rnorm(N * 3,
                                   mean = mu * period,
                                   sd = sigma * sqrt(period)), ncol = 3)

# Izracunaj povrate portfelja u svakoj simulaciji
portfolio_returns <- simulated_returns %*% weights

# Izracunaj krajnju vrijednost portfelja u svakoj simulaciji
portfolio_values <- initial_value * (1 + portfolio_returns)

# Izracunaj promjenu u vrijednosti portfelja
portfolio_losses <- initial_value - portfolio_values

# 99% VaR (najveci gubitak u 1% najgorih slucajeva)
VaR_99 <- quantile(portfolio_losses, 0.99)

# Ispis rezultata
VaR_99
VaR_99=1965.407
```

Pokretanjem ovog koda u R-u, dobit ćemo aproksimaciju vrijednost pod rizikom (VaR) za portfelj pri razini pouzdanosti od 99%. Primjerice, simulacija može dati rezultat da je 99% VaR  $\approx$  2000 EUR. To znači da, s 99% pouzdanosti, najveći mogući jednodnevni gubitak portfelja neće premašiti 2000 EUR ili s druge strane, postoji 1% šanse da će gubitak biti veći od tog iznosa. Objašnjenje provođenja metode je sljedeće:

- Generirali smo 10,000 nasumičnih povrata za svaku dionicu u portfelju, koristeći normalnu distribuciju sa zadanim srednjim povratima i volatilnošću.
- Povrat portfelja izračunat je kao ponderirani prosjek povrata pojedinih dionica (prema udjelima u portfelju).
- Za svaki simulirani povrat izračunat je gubitak portfelja, te je zatim 99% VaR procijenjen kao 99. percentil tih gubitaka.

Sami zaključak nakon provedene metode je sljedeći:

Monte Carlo metoda omogućava aproksimaciju rizika portfelja simulacijom mnogih mogućih scenarija kretanja tržišta. VaR je samo jedna od mjera rizika, ali je vrlo korisna za aproksimaciju potencijalnih gubitaka u najgorim slučajevima.

Treba naglasiti da se oba navedena primjera iz financija svode na izračune kompliciranih integrala, gdje Monte Carlo metoda integracije omogućuje jednostavnu aproksimaciju kompliciranih izraza.

# Literatura

- [1] RALF KORN, ELKE KORN, GERALD KROISANDT, *Monte Carlo Methods and Models in Finance and Insurance*, Taylor and Francis Group, LLC CRC Press, 2010.
- [2] MICHAEL EVANS, TIM SWARTZ, *Approximating Integrals via Monte Carlo and Deterministic Methods*, Oxford University Press Inc., New York, 2000.
- [3] MIRTA BENŠIĆ, NENAD ŠUVAK, *Uvod u vjerojatnost i statistiku*, Sveučilište J.J. Strossmayera, Odjel za matematiku, Osijek, 2014.
- [4] PAUL GLASSERMAN, *Monte Carlo Methods in Financial Engineering*, Springer, 2003.
- [5] *The OG Clever Machine*, dostupno na <https://theclevermachine.wordpress.com/2012/09/22/monte-carlo-approximations/>
- [6] *StatLect*, dostupno na <https://www.statlect.com/asymptotic-theory/Monte-Carlo-method>





## Sažetak:

Tema ovog rada je Monte Carlo metoda koja koristi slučajne brojeve za aproksimaciju, što je posebno korisno za složene ili visokodimenzionalne funkcije. Ključna prednost ove metode je njena sposobnost da se nosi s problemima visoke dimenzionalnosti, gdje druge metode postaju neučinkovite. Glavni nedostatak Monte Carlo metode je njena spora konvergencija. Točnost aproksimacije se sporo povećava s povećanjem broja simulacija. U ovom kontekstu, generatori slučajnih brojeva su ključni za stvaranje nasumičnih uzoraka koji se koriste u simulacijama. Kvaliteta rezultata Monte Carlo metode uvelike ovisi o kvaliteti generatora slučajnih brojeva, jer loše generirani uzorci mogu dovesti do netočnih aproksimacija. Dakle, pri izboru i testiranju generatora slučajnih brojeva moramo biti vrlo pažljivi jer su oni od velike važnosti za uspješnost primjene Monte Carlo metode aproksimacije.

## Ključne riječi

slučajni brojevi, generatori, integracija, konvergencija, aproksimacija



# Monte Carlo approximation method

## Summary

The topic of this thesis is the Monte Carlo method, which uses random numbers to estimate the value of integrals, particularly for complex or high-dimensional functions. The key advantage of this method is its ability to handle high-dimensional problems where other methods become inefficient. However, the main drawback of the Monte Carlo method is its slow convergence, meaning that the accuracy of the estimate increases slowly with the number of simulations. In this context, random number generators play a crucial role in generating the random samples used in the simulations. The quality of the Monte Carlo method's results largely depends on the quality of the random number generators, as poorly generated samples can lead to inaccurate estimates. Therefore, careful selection and testing of random number generators are essential for the successful application of the Monte Carlo method.

## Keywords

random numbers, generators, integration, convergence, approximation



# Životopis

Rođen sam 20. travnja 2000. godine u Šiklošu u Mađarskoj. Pohađao sam i završio Osnovnu školu Josipa Kozarca u Slatini i opću gimnaziju u Srednjoj školi Marka Marulića, također u Slatini. U osnovnoj i srednjoj školi redovito sam sudjelovao na natjecanjima iz matematike gdje sam ostvarivao odlične rezultate. Nakon završene srednje škole upisujem preddiplomski studij *Matematika* na Odjelu za matematiku u Osijeku (sadašnji Fakultet primijenjene matematike i informatike). Studij završavam 2022. godine i na istom fakultetu upisujem diplomski studij *Financijska matematika i statistika*.