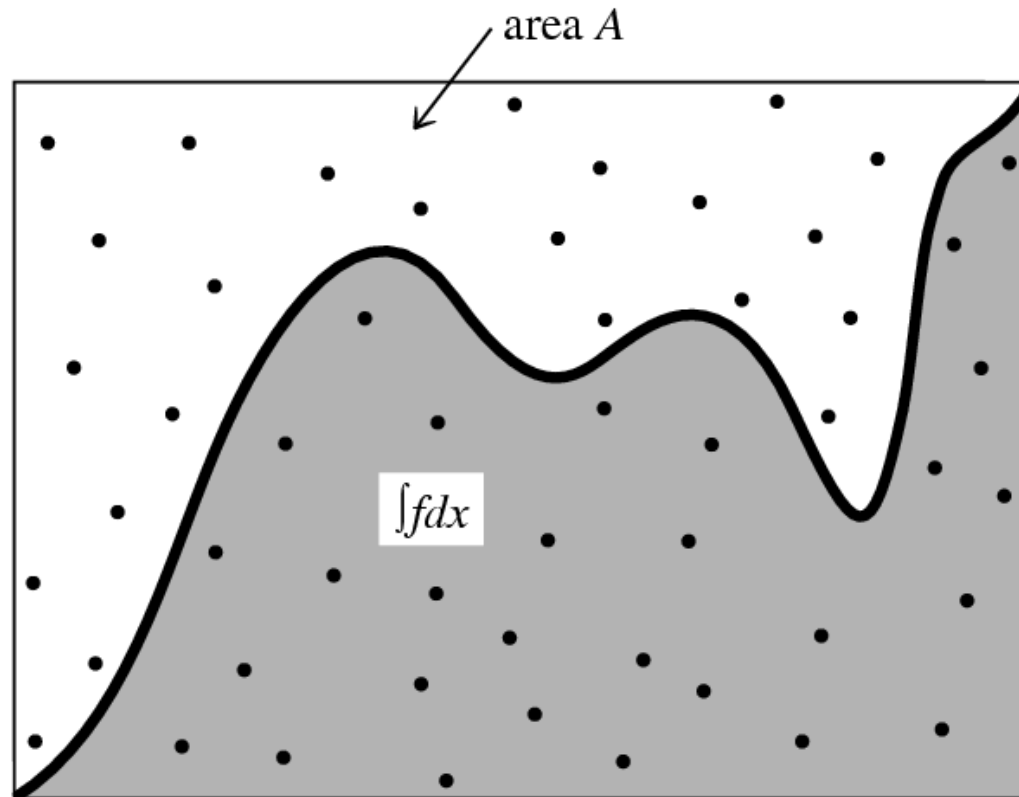


Monte Carlo metoda za numerično integriranje



Monte Carlo metode so se pojavile leta 1946 pri načrtovanju vodikove atomske bombe. Uporabili so jih za simulacijo gibanja nevtronov. Prvi avtorji so bili Stanislaw Ulam, John von Neumann in Nick Metropolis.

Kratka ponovitev verjetnosti

$U(0, 1)$: enakomerna porazdelitev na $[0, 1]$.

X_1, X_2, \dots : neodvisne slučajne spremenljivke, $S_N = X_1 + \dots + X_N$

$E(X) = \int_{-\infty}^{\infty} xp(x)dx$: matematično upanje, $p(x)$ je gostota porazdelitve

$D(X) = \int_{-\infty}^{\infty} (x - E(X))^2 p(x)dx$: varianca oz. disperzija

$\sigma(X) = \sqrt{D(X)}$: standardna deviacija

Krepki zakon velikih števil: $P \left(\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N (X_k - E(X_k)) = 0 \right) = 1.$

Ocena Čebiševa: $P \left(|X - E(X)| \geq \left(\frac{D(x)}{\epsilon} \right)^{1/2} \right) \leq \epsilon.$

Centralni limitni izrek: $\lim_{N \rightarrow \infty} P \left(\frac{S_N - E(S_N)}{\sigma(S_n)} < \epsilon \right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\epsilon} e^{-\frac{t^2}{2}} dt.$

Monte Carlo metoda

Naj bo f Lebesguovo integrabilna na $[0, 1]$ in $I[f] = \int_0^1 f(x) dx$.

Za slučajno spremenljivko X porazdeljeno po $U(0, 1)$ velja

$$E[f(X)] = I[f] \quad (\text{matematično upanje}).$$

Za vzorec X_1, \dots, X_N neodvisnih slučajnih spremenljivk porazdeljenih po $U(0, 1)$ je

$$I_N[f] = \frac{1}{N} \sum_{i=1}^N f(X_i)$$

cenilka za matematično upanje. Pri Monte Carlo metodi kot približek za $I[f]$ vzamemo $I_N[f]$. Cenilka $I_N[f]$ je nepristranska, saj je

$$E[I_N[f]] = I[f].$$

Posledica KZVŠ je

$$P\left(\lim_{N \rightarrow \infty} I_N[f] = I[f]\right) = 1.$$

Konvergenca 1

$\epsilon_N[f] = I[f] - I_N[f]$: napaka metode Monte Carlo

$D[f] = \int_0^1 (f(x) - I[f])^2 dx$: varianca f

$\sigma[f] = \sqrt{D[f]}$: standardna deviacija f

Pokazali bomo, da je za velik N velja

$$\epsilon_N[f] \approx \sigma[f] N^{-1/2} \nu,$$

kjer je ν slučajna spremenljivka porazdeljena po $N(0, 1)$.

Konvergenca 2

Ocena Čebiševa:
$$P\left(|X - E(X)| \geq \left(\frac{D(x)}{\epsilon}\right)^{1/2}\right) \leq \epsilon.$$

Za $S_N = f(X_1) + \dots + f(X_N) = NI_N[f]$ dobimo

$$P\left(|S_N - E(S_N)| \geq \left(\frac{D(S_N)}{\epsilon}\right)^{1/2}\right) \leq \epsilon,$$

od tod

$$P\left(|NI_N - NI[f]| \geq \left(\frac{ND[f]}{\epsilon}\right)^{1/2}\right) \leq \epsilon$$

in končno

$$P\left(|\epsilon_N[f]| \geq \left(\frac{D[f]}{N\epsilon}\right)^{1/2}\right) \leq \epsilon.$$

Od tod sledi, da pri fiksnem ϵ napaka pada z $\mathcal{O}(N^{-1/2})$.

Konvergenca 3

$$\text{CLI: } \lim_{N \rightarrow \infty} P \left(\frac{S_N - E(S_N)}{\sigma(S_N)} < \epsilon \right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\epsilon} e^{-\frac{t^2}{2}} dt.$$

Od tod za $S_N = f(X_1) + \dots + f(X_N) = NI_N[f]$ dobimo

$$\lim_{N \rightarrow \infty} P \left(\frac{NI_N[f] - NI[f]}{\sqrt{N}\sigma[f]} < \epsilon \right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\epsilon} e^{-\frac{t^2}{2}} dt$$

oziroma

$$\lim_{N \rightarrow \infty} P \left(I_N[f] - I[f] < \frac{\epsilon\sigma[f]}{\sqrt{N}} \right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\epsilon} e^{-\frac{t^2}{2}} dt$$

in končno

$$\lim_{N \rightarrow \infty} P \left(|\epsilon_N[f]| < \frac{\epsilon\sigma[f]}{\sqrt{N}} \right) = \frac{1}{\sqrt{2\pi}} \int_{-\epsilon}^{\epsilon} e^{-\frac{t^2}{2}} dt.$$

Izboljšave

Nekaj vrednosti za $a(\epsilon) := \frac{1}{\sqrt{2\pi}} \int_{-\epsilon}^{\epsilon} e^{-\frac{t^2}{2}} dt$:

$$a(0.6745) = 0.5,$$

$$a(2.576) = 0.99,$$

$$a(3.291) = 0.999,$$

Pri fiksnem ϵ je z verjetnostjo $a(\epsilon)$ napaka integrala pod

$$\frac{\epsilon \sigma[f]}{\sqrt{N}}.$$

To napako lahko zmanjšamo na različne načine, ki jih bomo predstavili v nadaljevanju:

- povečamo N ,
- zmanjšamo faktor σ ,
- z drugačno izbiro točk (kvazi Monte Carlo metode) namesto faktorja $N^{-1/2}$ dobimo

$$N^{-1}(\ln N)^k$$

za neko konstantno k .

Ocena variance

CLI lahko uporabimo za oceno, koliko točk potrebujemo, da bo napaka z dano verjetnostjo dovolj majhna. Pri tem uporabimo formulo

$$\frac{\epsilon\sigma[f]}{\sqrt{N}},$$

težava pa je, da ne poznamo standardne deviacije $\sigma[f]$.

To lahko ocenimo tako, da iz vzorca MN neodvisnih slučajnih spremenljivk sestavimo M cenilk $I_N^{(1)}, \dots, I_N^{(M)}$.

Potem je

$$\bar{\epsilon}_N = \left(\frac{1}{M} \sum_{j=1}^M \left(I_N^{(j)} - \bar{I}_N \right)^2 \right)^{1/2},$$

kjer je

$$\bar{I}_N = \frac{1}{M} \sum_{j=1}^M I_N^{(j)},$$

približek za ϵ_N in za oceno za $\sigma[f]$ vzamemo $\sigma = N^{1/2}\bar{\epsilon}_N$.

Primerjava z mrežnimi metodami

Za integral po območju $I^d = [0, 1]^d$ dobimo

$$I[f] = E[f(\mathbf{X})] = \int_{I^d} f(\mathbf{X}) d\mathbf{X},$$

kjer je \mathbf{X} vektor neodvisnih slučajnih spremenljivk X_1, \dots, X_d , razporejenih po $U(0, 1)$.

Za napako še vedno velja, da pada z $\mathcal{O}(N^{-1/2})$, neodvisno od d .

Če iz kvadrature formule reda k s tenzorskimi produkti izpeljemo mrežno kvadraturno formulo za integriranje po I^d , potem pri N točkah napaka pada z $\mathcal{O}(N^{-k/d})$, saj velja $N = n^d$, napaka pa pada kot n^{-k} .

Ostale razlike:

- Mreža pri mrežnih kvadrature formule je pri velikem d prevelika.
- Monte Carlo metode so tudi v eni dimenziji primernejše za računanje Fourierovih koeficientov.
- Monte Carlo metoda nima težav z nezveznostmi, ki sicer zmanjšajo red kvadrature pravil.

Generiranje naključnih števil

Z računalnikom ne moremo generirati pravih naključnih števil, saj jih vedno generiramo z nekim procesom, ki ga lahko ponovimo. Tako v resnici delamo s psevdonaključnimi števili.

Ena izmed preprostih metod za generiranje naključnih števil je linearni kongruenčni model, kjer računamo zaporedje

$$x_{r+1} = ax_r + c \pmod{m},$$

kjer sta a in c določeni pozitivni konstanti, izbiramo pa števila med 0 in $m - 1$. Pri tem a , c in m izberemo tako, da je perioda (števila se najkasneje po m korakih začnejo ponavljati) čim večja.

Model ima še to težavo, da če generiramo naključne d -dimenzionalne vektorje, potem točke ležijo na $(d - 1)$ -dimenzionalnih ravninah, ki jih je približno $m^{1/d}$.

Za velike probleme (npr. $N = 10^{13}$), so potrebni posebni generatorji. Generator v Matlabu se hvali, da naj bi bila perioda velika okrog 2^{1492} .

Za nadaljevanje predpostavimo, da imamo dober generator naključnih števil, ki so porazdeljena po $U(0, 1)$.

Neenakomerne porazdelitve

Pri Monte Carlo metodi potrebujemo tudi slučajne spremenljivke, ki niso enakomerno porazdeljene. Če npr. uporabimo slučajno spremenljivko z gostoto $p(x)$, potem se formula za matematično upanje spremeni v

$$E[f] = I[f] = \int f(x)p(x)dx.$$

Cenilka še naprej ostane

$$I_N[f] = \frac{1}{N} \sum_{i=1}^N f[X_i],$$

za napako

$$\epsilon_N[f] = I[f] - I_N[f]$$

pa tako kot prej velja, da je

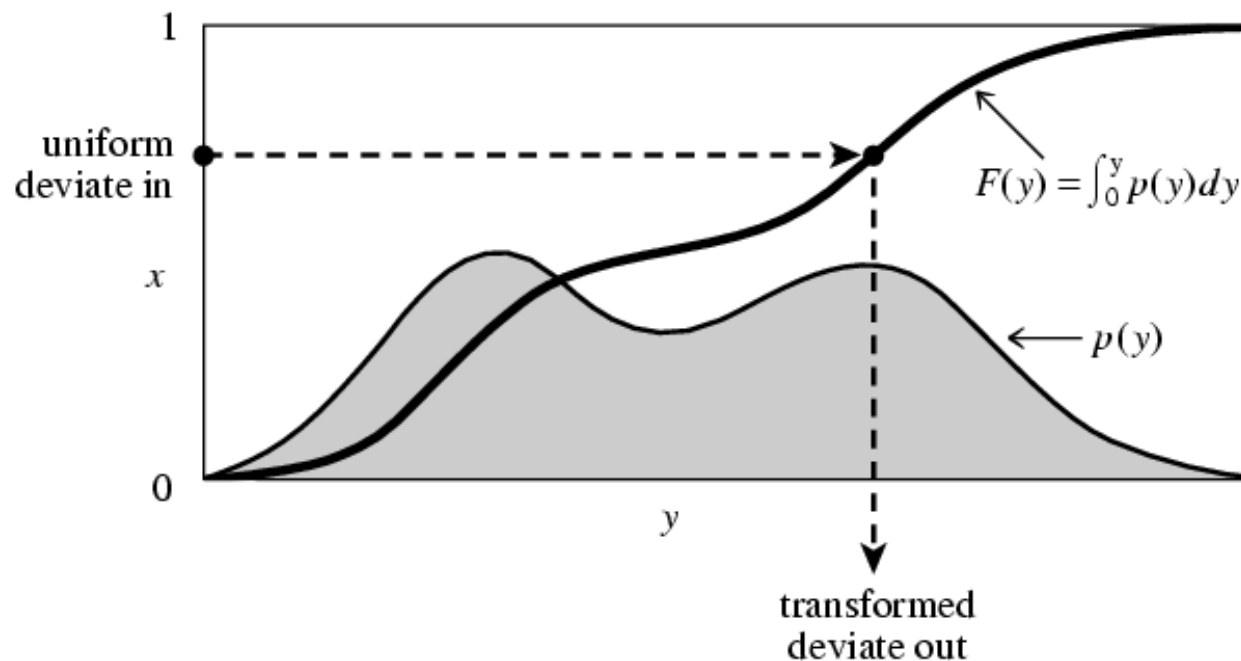
$$\epsilon_N[f] \approx N^{-1/2} \sigma[f] \nu,$$

kjer je ν slučajna spremenljivka porazdeljena po $N(0, 1)$ in

$$\sigma^2[f] = \int (f(x) - I[f])^2 p(x) dx.$$

Generiranje neenakomerno porazdeljenih naključnih števil 1

Denimo, da želimo zgenerirati naključno spremenljivko X , porazdeljeno z gostoto $p(x)$. Prvi pristop je transformacijska metoda.



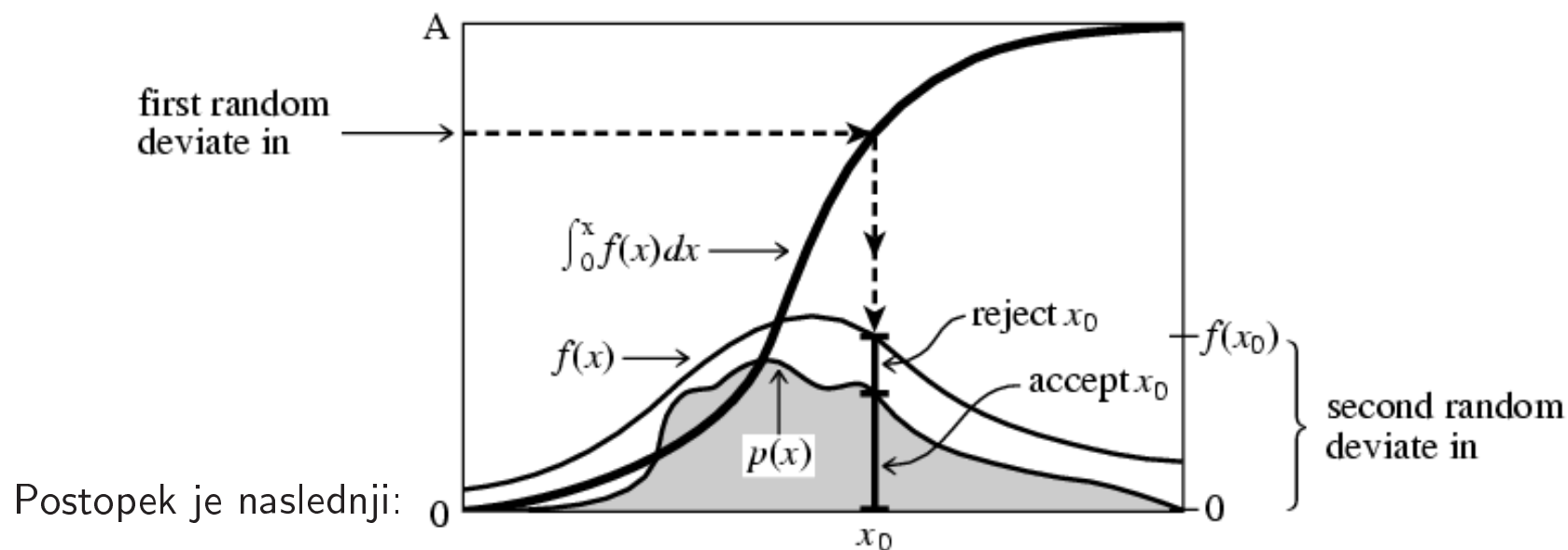
Ko zgeneriramo Y iz $U(0, 1)$, moramo poiskati X , pri katerem je $F(Y) = X$, kjer je

$$F(y) = \int_0^y p(t) dt.$$

To pomeni, da moramo poznati F^{-1} ali pa vsaj numerično dovolj hitro rešiti enačbo.

Generiranje neenakomerno porazdeljenih naključnih števil 2

Drugi pristop je metoda sprejemanja in zavračanja (acceptance-rejection method). Denimo, da poznamo funkcijo $f(x) \geq p(x)$, za katero se da preprosto izračunati inverzno funkcijo za $F(x) = \int_0^x f(x) dx$.



- Zgeneriramo y iz $U(0, F(1))$.
- Izračunamo $x_0 = F^{-1}(y)$.
- Zgeneriramo z iz $U(0, f(x_0))$. Če je $0 < z < p(x_0)$, potem izberemo $x = x_0$, sicer pa se vrnemo na točko a).

Redukcija variance 1

Prva metoda za redukcijo variance je porazdelitev po pomembnosti (importance sampling). Integral zapišemo kot

$$I[f] = \int f(x) dx = \int \frac{f(x)}{p(x)} p(x) dx$$

in si to predstavljamo kot integral z gostoto p . Sedaj za neodvisne spremenljivke X_1, \dots, X_N , porazdeljene z gostoto p , dobimo Monte Carlo cenilko

$$I_N[f] = \frac{1}{N} \sum_{k=1}^N \frac{f(X_k)}{p(X_k)}$$

za $I[f]$. Za napako $\epsilon_N[f] = I[f] - I_N[f]$ sedaj velja

$$\epsilon_N[f] \approx \sigma_p N^{-1/2},$$

kjer je

$$\sigma_p^2 = \int \left(\frac{f(x)}{p(x)} - I[f] \right)^2 p(x) dx.$$

Če torej dobimo $\sigma_p \ll \sigma[f]$, smo pomembno zmanjšali napako. Metoda deluje, če je f/p skoraj konstanta.

Redukcija variance 2

Druga metoda je uporaba kontrolnih vrednosti (control variates). Denimo, da poznamo g , ki ima podobno obliko kot f , integral $I[g] = \int g(x)dx$ pa znamo analitično izračunati. Potem pišemo

$$\int f(x)dx = \int (f(x) - g(x))dx + \int g(x)dx$$

in uporabimo Monte Carlo metodo za izračun integrala $f - g$. Dobimo

$$I_N[f] = \frac{1}{N} \sum_{k=1}^N (f(X_k) - g(X_k)) + I[g].$$

Za napako $\epsilon_N[f] = I[f] - I_N[f]$ sedaj velja

$$\epsilon_N[f] \approx \sigma_{f-g} N^{-1/2},$$

kjer je

$$\sigma_{f-g}^2 = \int (\bar{f}(x) - \bar{g}(x))^2 dx$$

in $\bar{f}(x) = f(x) - I[f]$, $\bar{g}(x) = g(x) - I[g]$.

Če torej dobimo $\sigma_{f-g} \ll \sigma[f]$, smo pomembno zmanjšali napako. Metoda deluje, če je $f - g$ skoraj konstanta.