

Ολοκλήρωση - Μέθοδος Monte Carlo

Χρησιμοποίηση τυχαίων αριθμών για επίλυση ολοκληρωμάτων

Η μέθοδος Monte Carlo δίνει μια διαφορετική προσέγγιση για την επίλυση ενός ολοκληρώματος

Τυχαίοι αριθμοί

Η συνάρτηση `rand()` προσφέρει μια ακολουθία τυχαίων αριθμών ομοιόμορφα κατανεμημένων στο διάστημα $[0,1)$

Δύο βασικές μέθοδοι χρησιμοποιούνται για την επίλυση ολοκληρωμάτων

Μέθοδος επιλογής ή δειγματοληψίας

Μέθοδος μέσης τιμής

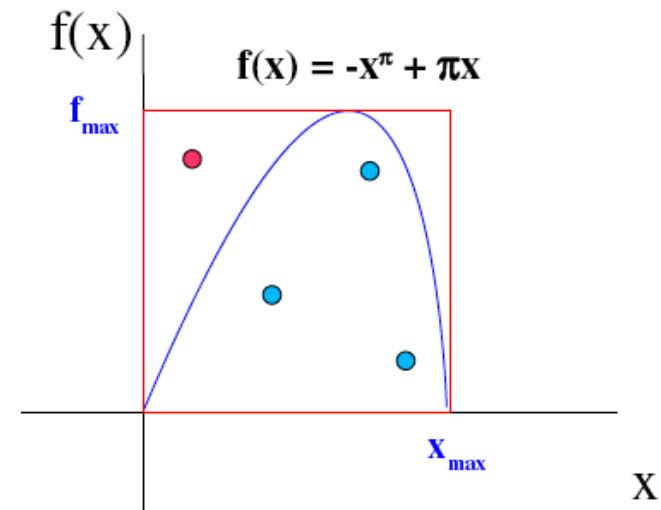
Monte Carlo μέθοδος δειγματοληψίας

- Περικλείουμε την συνάρτηση που θέλουμε να ολοκληρώσουμε μέσα σε ένα ορθογώνιο στο διάστημα της ολοκλήρωσης
 - ❑ Υπολογίζουμε το εμβαδό του ορθογωνίου
- Εισάγουμε τυχαία σημεία στο ορθογώνιο
- Μετρούμε τα σημεία που βρίσκονται μέσα στο ορθογώνιο και αυτά που περικλείονται από την συνάρτηση
- Το εμβαδό της συνάρτησης (ολοκλήρωμα) στο διάστημα ολοκλήρωσης δίνεται από

$$E_{f(x)} = E_{\text{ορθογ.}} \times \frac{N_{f(x)}}{N_{\text{ορθογ.}}}$$

Όπου $N_{f(x)}$ = αριθμός ●

$N_{\text{ορθογ.}}$ = αριθμός ●

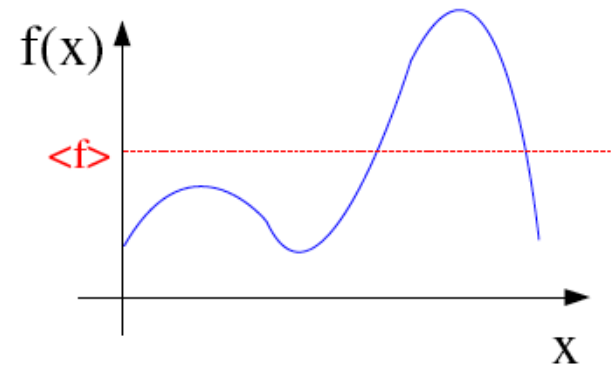


Monte Carlo μέθοδος μέσης τιμής

Η ολοκλήρωση με Monte Carlo γίνεται με το να πάρουμε τη μέση τιμή της συνάρτησης υπολογιζόμενη σε τυχαία επιλεγμένα σημεία μέσα στο διάστημα ολοκλήρωσης

$$I = \int_{x_a}^{x_b} f(x) dx = (b-a) \langle f(x) \rangle$$

$$\langle f(x) \rangle \approx \frac{1}{N} \sum_{i=0}^N f(x_i)$$



Το στατιστικό σφάλμα: $\delta I = \sigma_{\bar{f}}$ όπου $\sigma_{\bar{f}} = \frac{\sigma_f}{\sqrt{N}}$

Παράδειγμα κώδικα ολοκλήρωσης Monte Carlo

```

program random
integer iseed/12345/
call srand(iseed)      ! Ξεκίνημα ακολουθίας τυχαίων αριθμών με
                        ! αρχική τιμή 12345

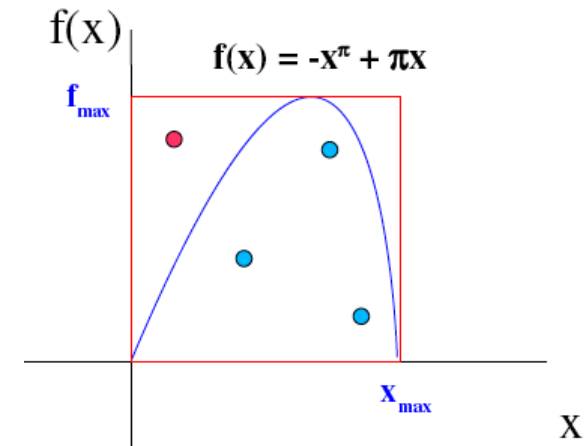
```

Τμήμα κώδικα για τη Μέθοδο δειγματοληψίας

```

Do I = 0, Npnts
  x = xmax * rand() ! Η συνάρτηση rand() επιστρέφει ψευδοτυχαίες
                    ! τιμές στο διάστημα [0,1)
  fRand = fMax * rand()
  if (fRand < myfunc(x)) then
    below = below + 1
  endif
Enddo
Print *, ' Apotelesma olokliwrsis = ', fmax*xMax *below/Npnts

```

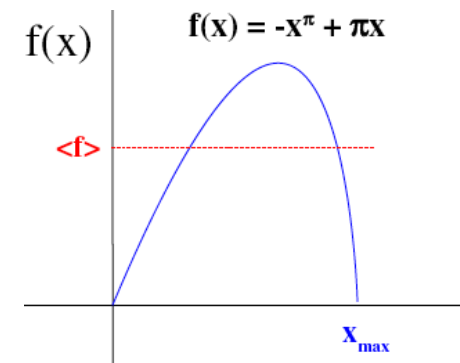


Τμήμα κώδικα για τη Μέθοδο μέσης τιμής

```

Sum = 0
Do I = 0, Npnts
  x = xmax * rand()
  sum = f(x) + sum
End do
Print *, ' Apotelesma olokliwrsis', xmax*sum/Npnts

```



Monte Carlo ολοκλήρωση σε πολλές διαστάσεις

Εύκολο να γενικεύσουμε τη μέθοδο της μέσης τιμής σε πολλές διαστάσεις

Το σφάλμα στη μέθοδο ολοκλήρωσης με Monte Carlo είναι στατιστικό

Ελαττώνεται ως $1/\sqrt{N}$

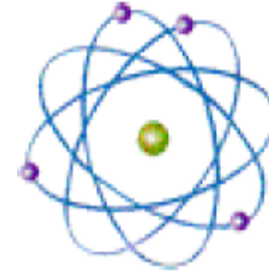
Για 2 διαστάσεις:

$$I = \int_a^b dx_1 \int_c^d dx_2 f(x, y) \simeq (b-a)(d-c) \times \frac{1}{N} \sum_i^N f(x_i, y_i)$$

Ολοκλήρωση - Πολλαπλά ολοκληρώματα

Για παράδειγμα, έστω τα ηλεκτρόνια του ατόμου του ${}^4\text{Be}$:

$$I = \int_0^1 dx_1 \int_0^1 dx_2 \int_0^1 dx_3 \cdots \int_0^1 dx_{12} f(x_1, x_2, \dots, x_{12})$$



Έχουμε 3 διαστάσεις για κάθε ηλεκτρόνιο x 4 ηλεκτρόνια = 12 διαστάσεις

$$I = (1-0)^{12} \times \frac{1}{N} \sum f(x_1^i, x_2^i, \dots, x_{12}^i)$$

Αλλά για $N=10^6$ σημεία στη ολοκλήρωση Monte Carlo έχουμε 10^6 υπολογισμούς

Για τα PCs υποθέτοντας 1Giga υπολογισμούς/sec θα χρειαστούμε 10^{-3} secs !!

Monte Carlo και τυχαίοι αριθμοί

Τα προσδιορισμένα συστήματα (**deterministic systems**) περιγράφονται εν γένει από κάποιο μαθηματικό κανόνα

Κάποια συστήματα ωστόσο δεν είναι προσδιορισμένα **Τυχαία ή στοχαστικά**

Οποιαδήποτε διεργασία ή αλγόριθμος χρησιμοποιεί τυχραίους αριθμούς και αντιτίθεται σε προσδιορισμένους αλγόριθμους ονομάζεται Monte Carlo

Η μέθοδος Monte Carlo χρησιμοποιείται ευρέως στις επιστήμες:

Φυσική: προσομοίωση φυσικών διεργασιών

Μαθηματικά: αριθμητική ανάλυση

Βιολογία: προσομοίωση κυττάρων

Οικονομικά: εκτίμηση της διακύμανσης του χρηματιστηρίου αξιών

Μηχανική: προσομοίωση πειραματικών διατάξεων

Σημασία των μεθόδων Monte Carlo

Οι μέθοδοι Monte Carlo αποτελούν ένα από τα σημαντικότερα εργαλεία στη Φυσική

- Ανάλυση δεδομένων
- Προσομοίωση φυσικών γεγονότων που στηρίζονται σε τυχαίες διεργασίες – πιθανότητες
- Σχεδιασμό ανιχνευτών, βελτιστοποίηση και προσομοίωση

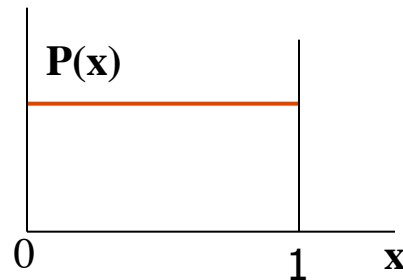
Επομένως ας μάθουμε μερικές από τις βασικές αρχές

- Σκοπός των γεννητόρων/προγραμμάτων Monte Carlo
- Γεννήτορες τυχαίων αριθμών
- Ολοκλήρωση
- Μερικά ιδιαίτερα δημοφιλή Monte Carlo προγράμματα

Τυχαίοι αριθμοί

Τυχαίος αριθμός είναι ένας αριθμός επιλεγμένος σαν να ήταν καθαρά τυχαία από μια συγκεκριμένη κατανομή

Σε μια ομοιόμορφη κατανομή τυχαίων αριθμών στο διάστημα $[0,1)$, κάθε αριθμός έχει την ίδια τύχη να επιλεχθεί



Για παράδειγμα: Όταν ρίχνετε ένα ζάρι οι αριθμοί που μπορείτε να πάρετε είναι ομοιόμορφα κατανεμημένοι μεταξύ 1 και 6.

Κάθε αριθμός έχει την ίδια πιθανότητα να "βγεί"

Γεννήτορες τυχαίων αριθμών

Ο καλύτερος τρόπος για να πάρουμε τυχαίους αριθμούς είναι να χρησιμοποιήσουμε μια διεργασία που συμβαίνει στη φύση.

- Ρίξιμο ενός ζαριού ή ενός νομίσματος
- Λόττο
- Τα αποτελέσματα του ποδοσφαίρου
- Η ραδιενεργός διάσπαση των πυρήνων

Φυσικά ο τρόπος αυτός για να διαλέξουμε τυχαίους αριθμούς δεν είναι ιδιαίτερα αποδοτικός

- Υπολογιστικές μέθοδοι αναπτύχθηκαν που κάνουν την ίδια διαδικασία

Πως μπορούμε όμως να κάνουμε κάποιο πρόγραμμα να υπολογίζει κάτι τυχαία;

- Με ένα γεννήτορα τυχαίων αριθμών

Γεννήτορες τυχαίων αριθμών

Γεννήτορας τυχαίων αριθμών είναι ένα υπο-πρόγραμμα το οποίο δημιουργεί μια ακολουθία τυχαίων αριθμών

Όλοι οι υπολογιστές σήμερα περιέχουν στην βιβλιοθήκη τους ένα μηχανισμό για την δημιουργία ακολουθίας τυχαίων αριθμών οι οποίοι είναι ομοιόμορφα κατανεμημένοι στο διάστημα $[0,1)$

Η ακολουθία των αριθμών αυτών μπορεί να θεωρηθεί σαν ψευδο-τυχαία ακολουθία αφού για κάθε εκτέλεση του προγράμματος, θα πάρουμε και πάλι την ίδια ακολουθία τυχαίων αριθμών για την ίδια αρχική τιμή του "σπόρου" ([seed](#)) της ακολουθίας

Στην πραγματικότητα οι συναρτήσεις που καλούμε στον υπολογιστή χρησιμοποιούν μια μέθοδο (Lehmer 1948) στηριγμένη σε 32-bit ακεραίους και επομένως έχουν περίοδο το πολύ $2^{31} \sim 10^9$.

Αυτό είναι το πλήθος των τυχαίων αριθμών που μπορούν να δημιουργηθούν μέσα σε λίγα δευτερόλεπτα σε ένα μοντέρνο υπολογιστή

Η μέθοδος που ακολουθείται χρησιμοποιεί μια εξίσωση της μορφής: $x_{n+1} = \text{mod}[(ax_n + b), m]$

όπου mod είναι το modulo. Οι σταθερές a, b και m διαλέγονται προσεκτικά ώστε η ακολουθία των αριθμών να γίνεται χαοτική και ομοιόμορφα κατανεμημένη

Γεννήτορες τυχαίων αριθμών

Κανόνες

- Η πρώτη αρχική τιμή, x_0 , (seed) επιλέγεται
- Η τιμή του $m > x_0$ και $a, b \geq 0$
- Το εύρος των τιμών είναι μεταξύ 0 και m (διαιρώντας με m μετατρέπεται μεταξύ 0 και 1)
- Η περίοδος του γεννήτορα αυτού είναι m-1.
Επομένως το m πρέπει να είναι αρκετά μεγάλο αφού η περίοδος δεν μπορεί ποτέ να γίνει μεγαλύτερη από m.

Για παράδειγμα:

Αν διαλέξουμε $a=b=x_0=7$ και $m = 10$ τότε θα πάρουμε την ακολουθία:

$$x_i = \text{mod}(a \cdot x_{i-1} + b, m)$$

7, 6, 9, 0 7, 6, 9, 0 7, 6, 9, 0

Και διαιρώντας με 10 θα έχουμε την ακολουθία

0.7, 0.6, 0.9, 0.0 0.7, 0.6, 0.9, 0.0 0.7, 0.6, 0.9, 0.0

Πολύ κακή ακολουθία

Γεννήτορες τυχαίων αριθμών

Από τους πλέον δημοφιλείς γεννήτορες είναι ο **RANDU** ο οποίος αναπτύχθηκε από την IBM το 1960 με τον ακόλουθο αλγόριθμο:

$$x_{n+1} = \text{mod}(65069x_n, 2^{31} - 1)$$

και αργότερα οι Park και Miller πρότειναν πως η εξίσωση $x_{n+1} = \text{mod}(16807x_n, 2^{31} - 1)$ δίνει την ελάχιστη συνθήκη για ένα ικανοποιητικό γεννήτορα

Πρόγραμμα [Randu](#)

Πρόγραμμα [Ran](#) (ελάχιστης απαίτησης)

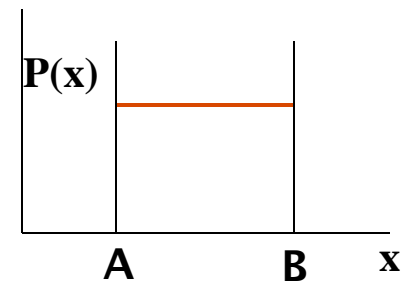
Τυχαίες κατανομές

Ομοιόμορφη (επίπεδη) Κατανομή

Αν το R είναι ένα τυχαίος πραγματικός αριθμός μεταξύ $[0,1)$ και αν τα A και B είναι πραγματικοί αριθμοί, και M, N είναι ακέραιοι αριθμοί τότε η τιμή:

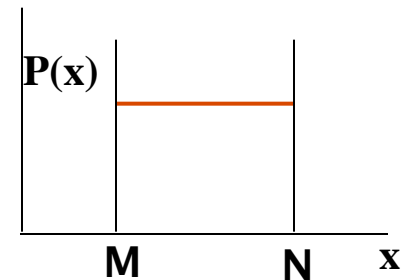
$$A + (B - A) * R$$

θα είναι ένα τυχαίος πραγματικός αριθμός στο διάστημα $[A,B)$



Η τιμή της $M + INT(N * R)$

θα είναι ένα τυχαίος ακέραιος αριθμός στο διάστημα $[M,N]$



Για παράδειγμα, για να δημιουργήσουμε την ομοιόμορφη τυχαία κατανομή που αντιστοιχεί στο ρίξιμο των ζαριών (τιμές μεταξύ $[1,6]$) θα γράφαμε:

$$zari = 1 + INT(6 * R)$$

Τυχαίες κατανομές – Probability Distribution Function (PDF)

□ Πως μπορούμε να βρούμε μια γενική Συνάρτηση Κατανομής Πιθανότητας (PDF)

Στις προσομοιώσεις μιας τυχαίας διεργασίας συχνά ζητάμε μια **μη** ομοιόμορφη (ισοπίθανη) κατανομή τυχαίων αριθμών.

Για παράδειγμα η ραδιενεργός διάσπαση χαρακτηρίζεται από κατανομή Poisson:

$$P(n;v) = \frac{v^n e^{-v}}{n!}$$

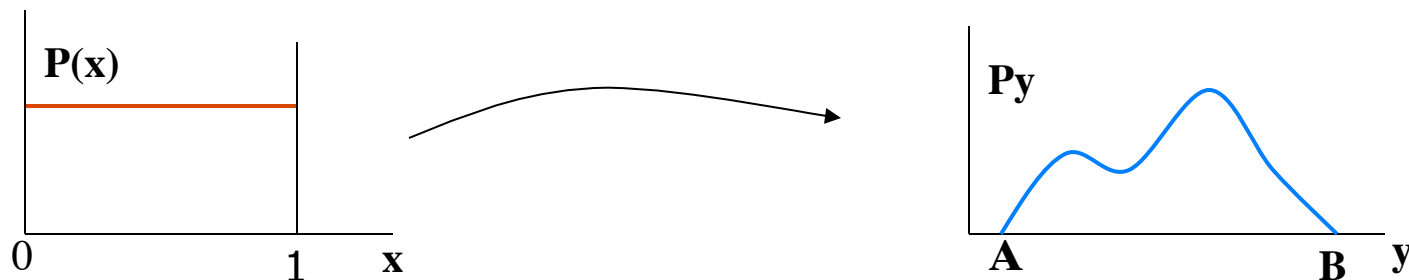
(πιθανότητα να βρούμε ακριβώς n γεγονότα όταν τα γεγονότα συμβαίνουν ανεξάρτητα το ένα από το άλλο και την ανεξάρτητη μεταβλητή x και με μέσο ρυθμό v στο διάστημα x)

➤ Χρησιμοποιούμε δυο μεθόδους (συνήθως)

□ Μέθοδος του μετασχηματισμού ([Transformation Method](#))

□ Μέθοδος της απόρριψης ([Rejection Method](#))

Σκοπός και των δυο μεθόδων είναι να μετατρέψουν μια ομοιόμορφη κατανομή τυχαίων αριθμών της μορφής $P_x(x)$ σε μια μη ομοιόμορφη της μορφής $P_y(y)$

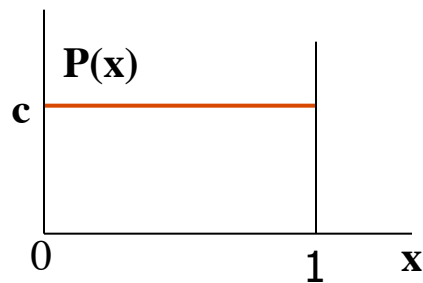


Τυχαίες κατανομές – Η μέθοδος του μετασχηματισμού

Θεωρήστε μια συλλογή από μεταβλητές $\{x_1, x_2, \dots, x_n\}$ που κατανέμονται σύμφωνα με μια συνάρτηση $P_x(x)$, τότε η πιθανότητα να βρούμε μια τιμή στο διάστημα x και $x+dx$ είναι $P_x(x)dx$

Αν y είναι κάποια συνάρτηση του x τότε μπορούμε να γράψουμε: $|P_x(x)dx| = |P_y(y)dy|$

Όπου $P_y(y)$ είναι η πυκνότητα πιθανότητας που περιγράφει την συλλογή $\{y_1, y_2, \dots, y_n\}$



$$P_x(x) = c \quad \text{Ομοιόμορφη κατανομή}$$

$$\text{και επομένως: } P_y(y) = c \left| \frac{dx}{dy} \right|$$

Για να βρούμε μια ακολουθία που χαρακτηρίζεται από την $P_y(y)$ θα πρέπει να βρούμε τη συνάρτηση μετασχηματισμού $y=f(x)$ που ικανοποιεί την εξίσωση:

$$\left| \frac{dx}{dy} \right| = P_y(y) \quad \text{Η σταθερά } C \text{ μπορεί να αγνοηθεί αφού απλά πολλαπλασιάζει την συνάρτηση}$$

Επομένως:

❑ Θέλουμε την $P_y(y)$

➤ Βρίσκουμε την $y=f(x)$ (συνάρτηση μετασχηματισμού) ώστε $|dx/dy| = P_y(y)$

Τυχαίες κατανομές - Η μέθοδος του μετασχηματισμού

Παραδείγματα τέτοιων συναρτήσεων μετασχηματισμού είναι τα ακόλουθα:

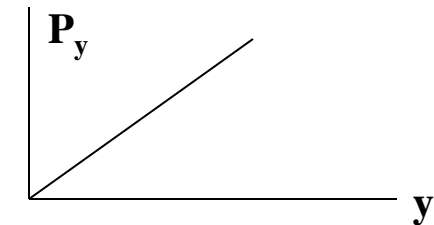
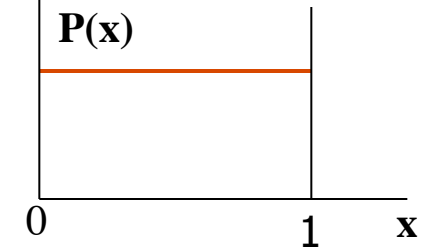
Επιθυμητή κατανομή $P_y(y)$	Συνάρτηση μετασχηματισμού $f(x)$
k	x/k
y	$\sqrt{2x}$
y^n	$[(n+1)x]^{1/(n+1)}$
$1/y$	e^x
e^y	$-\ln(x)$
$\cos y$	$\arcsin x$
\sqrt{y}	$[3/2x]^{2/3}$

Τυχαίες κατανομές - Η μέθοδος του μετασχηματισμού

Παράδειγμα: Θεωρήστε ότι θέλετε την κατανομή $P_y(y) = y$

Η συνάρτηση μετασχηματισμού είναι τότε:

$$\left| \frac{dx}{dy} \right| = P_y(y) = y \Rightarrow x = \int y dy$$



Λύνοντας ως προς y έχουμε: $\frac{1}{2}y^2 = x \Rightarrow y = \sqrt{2x}$

Όπου τα x είναι τυχαίοι αριθμοί στο διάστημα $[0,1)$

Ο κώδικας για το παραπάνω παράδειγμα είναι: [transform.f](#)

Τυχαίες κατανομές – Η μέθοδος της απόρριψης

Η μέθοδος του μετασχηματισμού είναι χρήσιμη όταν η συνάρτηση $f(x)$ μπορεί να υπολογιστεί

Ωστόσο υπάρχουν περιπτώσεις που η επιθυμητή συνάρτηση μπορεί να μην είναι γνωστή σε αναλυτική μορφή.

Για παράδειγμα, θεωρήστε ότι θέλουμε μια Gaussian κατανομή: $P_y(y) = e^{-y^2}$

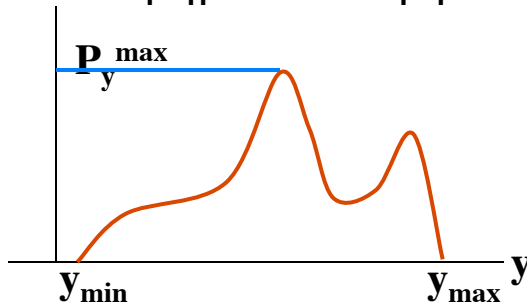
Για την περίπτωση αυτή δεν βρίσκουμε την συνάρτηση $y=f(x)$ γιατί:

$$\frac{dx}{dy} = P_y(y) = e^{-y^2} \Rightarrow x = \int e^{-y^2} dy \quad \text{Το ολοκλήρωμα αυτό δεν υπολογίζεται αναλυτικά}$$

Για τέτοιου είδους προβλήματα χρησιμοποιούμε τη μέθοδο της απόρριψης η οποία μπορεί να δημιουργήσει την επιθυμητή κατανομή για οποιαδήποτε συνάρτηση.

Με τη μέθοδο αυτή, η ακολουθία των τυχαίων αριθμών $\{y_1, y_2, \dots, y_n\}$ δημιουργείται με μια ομοιόμορφη κατανομή στο διάστημα $[y_{\min}, y_{\max}]$ που ενδιαφερόμαστε.

Υποθέστε ότι ο σκοπός μας είναι να δημιουργήσουμε μια ακολουθία αριθμών κατανεμημένων σύμφωνα με τη συνάρτηση $P_y(y)$



Προχωρούμε με την ακολουθία των αριθμών $\{y_1, y_2, \dots, y_n\}$ και δεχόμαστε τιμές που είναι πολλαπλάσια της $P_y(y)$

Τυχαίες κατανομές – Η μέθοδος της απόρριψης

➤ Ο αλγόριθμος που ακολουθούμε είναι ο ακόλουθος:

- Θέτουμε $m = 0$ (μετρητής)
- Προσδιορίζουμε την μέγιστη τιμή της επιθυμητής κατανομής $P_y(y)$ και το διάστημα $[y_{\min}, y_{\max}]$ της επιθυμητής κατανομής
- Δημιουργούμε τυχαίο ζευγάρι z_{ran}, y_{ran} από μια ομοιόμορφη κατανομή
- Για κάθε τιμή του y θέτουμε: $y = y_{\min} + (y_{\max} - y_{\min}) * y_{ran}$
 - ❑ Δημιουργούμε ένα τυχαίο αριθμό p_{test} ομοιόμορφα κατανεμημένο στο $[0, P_y^{\max}]$

$$P_{y_test} = P_{y_max} * z_{ran}$$

❑ Ελέγχουμε αν η τιμή της κατανομής $P(y) > P_{y_test}$

Αν ναι: Αυξάνουμε τον μετρητή $m = m+1$

Κρατάμε τη τιμή y μια και δίνει επιτρεπτή τιμή: θέτουμε $z(m) = y$

Αν όχι: Απορρίπτουμε την τιμή y

- Επαναλαμβάνουμε την διαδικασία N φορές:

Οι αριθμοί $z(m)$ που απομένουν στην ακολουθία κατανέμονται επομένως σύμφωνα με την συνάρτηση $P_y(y)$

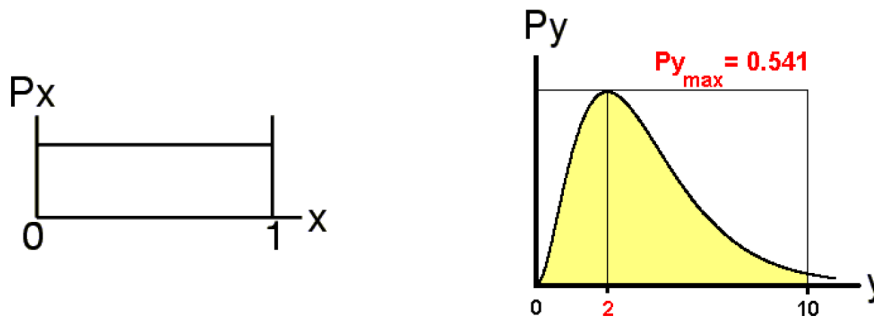
Τυχαίες κατανομές – Η μέθοδος της απόρριψης

Παράδειγμα: Έστω ότι θέλουμε την κατανομή $P_y(y) = e^{-y} y^2$ στο διάστημα $y=[0,10]$

Η μέγιστη τιμή της συνάρτησης P_y βρίσκεται ζητώντας $\frac{dP_y}{dy} = 0$

Η συνάρτηση έχει μέγιστο στο σημείο $y=2$ το οποίο αντιστοιχεί σε $P_{\max}=0.541$

Επομένως οι τιμές του P_{y_test} θα δημιουργηθούν στο διάστημα $[0,0.541]$



Το πρόγραμμα [rejection.f](#) δείχνει τον τρόπο που χρησιμοποιείται η μέθοδος της απόρριψης για το παραπάνω παράδειγμα.

Monte Carlo βελτιστοποίηση

Μπορούμε να χρησιμοποιήσουμε τυχαίους αριθμούς για να βρούμε τη μέγιστη ή ελάχιστη τιμή μιας συνάρτησης πολλών μεταβλητών

Τυχαία αναζήτηση

Η μέθοδος αυτή υπολογίζει την συνάρτηση πολλές φορές σε τυχαία επιλεγμένες τιμές των ανεξάρτητων μεταβλητών.

Αν συγκεντρώσουμε ένα ικανοποιητικό αριθμό δειγμάτων τότε προφανώς θα έχουμε εντοπίσει και το ακρότατο.

Παράδειγμα: Χρησιμοποιήστε τη μέθοδο Monte Carlo για να υπολογίσετε το ελάχιστο της συνάρτησης $f(x) = x^2 - 6x + 5$ στο διάστημα $x \in [1, 5]$

Η ακριβής λύση είναι $f_{\min} = -4.0$ για $x = 3.0$

Αλγόριθμος για ελάχιστα:

- Προσδιορισμός του πλήθους των πειραμάτων (N)
- Προσδιορισμός του διαστήματος [A,B]
- Αρχική τιμή για το ελάχιστο $f_{\min} = 9E9$ (πολύ μεγάλη τιμή)
- Επανάληψη της ακόλουθης διεργασίας N φορές)

☐ Δημιουργία ενός τυχαίου αριθμού x στο [A,B]

☐ Έλεγχος αν $F(x) < f_{\min}$

Αν ναι Βρήκαμε νέο ελάχιστο και κρατάμε τη τιμή του x

[min_mc.f](#) έχει το κώδικα για το παράδειγμα

Πιθανότητες

Αν ένα νόμισμα ριχθεί δεν είναι σίγουρο αν θα πάρουμε την πάνω όψη του. Ωστόσο αν συνεχίσουμε να επαναλαμβάνουμε το πείραμα αυτό, έστω N φορές και παίρνουμε την πάνω όψη S φορές, τότε ο λόγος S/N γίνεται σταθερός μετά από ένα μεγάλο πλήθος επανάληψης του πειράματος.

Η **πιθανότητα** P ενός γεγονότος A ορίζεται ως ακολούθως:

Αν το A μπορεί να συμβεί με S τρόπους από συνολικά K ισότιμους τρόπους τότε:

$$P = \frac{S}{K}$$

Παράδειγμα: Ρίχνοντας ένα νόμισμα, η πάνω όψη μπορεί να συμβεί μια φορά από τις δυνατές δύο περιπτώσεις. Επομένως η πιθανότητα είναι $P = 1/2$

Ο Αλγόριθμος για να λύσουμε το πρόβλημα αυτό:

- Προσδιορίζουμε το αριθμό των πειραμάτων N
- Μηδενίζουμε το μετρητή των επιτυχημένων αποτελεσμάτων
- Εκτελούμε τα N πειράματα (διαδικασία loop)
 - Δημιουργούμε ένα τυχαίο αριθμό x (ομοιόμορφη κατανομή)
 - Ελέγχουμε αν το $x < P$ και αν ναι αυξάνουμε το μετρητή κατά 1 (επιτυχημένη προσπάθεια)

Το πρόγραμμα [coin.f](#) περιέχει το παραπάνω παράδειγμα

Ρίχνοντας ένα νόμισμα

Ένα νόμισμα ρίχνεται 6 φορές. Ποιά είναι η πιθανότητα να πάρουμε

(α) ακριβώς 4 φορές την πάνω όψη

(β) τουλάχιστον 4 φορές την πάνω όψη

Τα ακριβή αποτελέσματα δίνονται από τη διωνυμική κατανομή:

Η διωνυμική κατανομή: Μια τυχαία διεργασία με ακριβώς δυο πιθανά αποτελέσματα τα οποία συμβαίνουν με συγκεκριμένες πιθανότητες καλείται διεργασία Bernoulli. Αν η πιθανότητα να πάρουμε κάποιο αποτέλεσμα (“επιτυχία”) σε κάθε προσπάθεια είναι p , τότε η πιθανότητα να πάρουμε ακριβώς r επιτυχίες ($r=0,1,2,\dots,N$) σε N ανεξάρτητες προσπάθειες χωρίς να παίζει ρόλο η σειρά με την οποία παίρνουμε επιτυχία ή αποτυχία, δίνεται από τη διωνυμική κατανομή

$$f(r; N, p) = \frac{N!}{r!(N-r)!} p^r q^{N-r}$$

Για το πρόβλημά μας θα έχουμε επομένως:

(α) $r=4$ για $N=6$ ενώ $p=1/2$ ($q=1-p$) και επομένως από τη διωνυμική κατανομή:

$$P = \frac{6!}{4!2!} \left(\frac{1}{2}\right)^4 \left(\frac{1}{2}\right)^2 = \frac{30}{2} \frac{1}{64} = \frac{15}{64} = 0.234375$$

(β) $r \geq 4$ για $N=6$ ενώ $p=1/2$ ($q=1-p$) και επομένως θα έχουμε σαν ολική πιθανότητα το άθροισμα για $P(r=4) + P(r=5) + P(r=6)$

$$P = \frac{6!}{4!2!} \left(\frac{1}{2}\right)^4 \left(\frac{1}{2}\right)^2 + \frac{6!}{5!1!} \left(\frac{1}{2}\right)^5 \left(\frac{1}{2}\right)^1 + \frac{6!}{6!0!} \left(\frac{1}{2}\right)^6 = \frac{11}{32} = 0.34375$$

Το πρόγραμμα [coin6.f](#) περιέχει τη MC λύση του προβλήματος αυτού