Εφαρμογές προσομοίωσης στο matlab_session_3_2026

1. κατανομή δειγματικού μέσου,
2. εκτίμηση του π,
3. προσομοίωση ζαριού με randi και
4. προσομοίωση ζαριού χωρίς randi.

Εφαρμογή 7: Monte Carlo: sampling distribution of the sample mean

Η εφαρμογή αυτή δείχνει πώς συμπεριφέρεται ο **δειγματικός μέσος** όταν επαναλαμβάνουμε πολλές φορές την ίδια δειγματοληψία.

Ορίζουμε έναν πληθυσμό με:

```
mu = 0;
sigma = 1;
```

δηλαδή θεωρούμε ότι οι παρατηρήσεις προέρχονται από κανονική κατανομή με μέσο 0 και τυπική απόκλιση 1.

Μετά ορίζουμε:

```
nSim = 1000;
vectorN = [10, 25, 100, 200];
```

Δηλαδή κάνουμε **1000 προσομοιώσεις** για τέσσερα διαφορετικά μεγέθη δείγματος: $N = 10$, $N = 25$, $N = 100$, $N = 200$.

Σε κάθε επανάληψη, το MATLAB δημιουργεί ένα τυχαίο δείγμα:

```
sample = normrnd(mu,sigma,N,1);
```

και υπολογίζει τον μέσο του δείγματος:

```
averageMatrix(i,j) = mean(sample);
```

Άρα κάθε στήλη του averageMatrix περιέχει 1000 δειγματικούς μέσους για ένα συγκεκριμένο μέγεθος δείγματος.

Στο τέλος, τα ιστογράμματα δείχνουν την **κατανομή των δειγματικών μέσων**. Όσο αυξάνεται το N, οι δειγματικοί μέσοι συγκεντρώνονται όλο και πιο κοντά στο πραγματικό $\mu = 0$. Αυτό φαίνεται και από την τυπική απόκλιση του δειγματικού μέσου, η οποία θεωρητικά είναι:

$$\frac{\sigma}{\sqrt{N}}$$

Άρα η εφαρμογή δείχνει ότι όσο μεγαλώνει το δείγμα, ο δειγματικός μέσος γίνεται πιο ακριβής εκτιμητής του πληθυσμιακού μέσου.

Εφαρμογή 8: Monte Carlo estimate of π

Η εφαρμογή αυτή χρησιμοποιεί προσομοίωση Monte Carlo για να εκτιμήσει τον αριθμό π.

Δημιουργούνται $N = 1{,}000{,}000$ τυχαία σημεία μέσα στο μοναδιαίο τετράγωνο:

```
x = rand(1,N);
y = rand(1,N);
```

Άρα κάθε σημείο έχει συντεταγμένες (x, y), όπου και τα δύο βρίσκονται στο διάστημα $(0,1)$.

Μετά ελέγχουμε ποια σημεία βρίσκονται μέσα στο τεταρτοκύκλιο ακτίνας 1:

```
counter = (x.^2 + y.^2) <= 1;
```

Η συνθήκη:

$$x^2 + y^2 \leq 1$$

είναι η εξίσωση του εσωτερικού κύκλου ακτίνας 1. Επειδή όμως τα σημεία είναι μόνο στο πρώτο τεταρτημόριο, μετράμε την επιφάνεια ενός **τεταρτοκύκλιου**.

Το εμβαδόν του τετραγώνου είναι 1, ενώ το εμβαδόν του τεταρτοκύκλιου είναι:

$$\frac{\pi}{4}$$

Άρα το ποσοστό των σημείων που πέφτουν μέσα στο τεταρτοκύκλιο προσεγγίζει το $\pi/4$. Γι' αυτό ο κώδικας γράφει:

```
pi_estimate = (N1/N)*4
```

Δηλαδή:

$$\hat{\pi} = 4 \cdot \frac{\text{points inside quarter circle}}{\text{total points}}$$

Όσο μεγαλύτερο είναι το N, τόσο πιο κοντά θα είναι η εκτίμηση στο πραγματικό π.

Εφαρμογή 9: Dice simulation using randi

Η εφαρμογή αυτή προσομοιώνει ρίψεις ζαριού με την εντολή randi.

```
dice = randi([1 6],100,1)
```

Η εντολή αυτή δημιουργεί 100 τυχαίους ακέραιους αριθμούς από το 1 έως το 6. Κάθε αριθμός αντιστοιχεί σε ένα πιθανό αποτέλεσμα ζαριού.

Δηλαδή το dice είναι ένα διάνυσμα 100 γραμμών, όπου κάθε γραμμή είναι μία ρίψη:

1, 2, 3, 4, 5 ή 6

Μετά ο κώδικας φτιάχνει ιστόγραμμα:

```
histogram(dice,1:7)
```

Το ιστόγραμμα δείχνει πόσες φορές εμφανίστηκε κάθε αποτέλεσμα. Επειδή το δείγμα έχει μόνο 100 ρίψεις, οι συχνότητες δεν θα είναι ακριβώς ίσες. Όμως, θεωρητικά, κάθε πλευρά έχει πιθανότητα:

$$\frac{1}{6}$$

Άρα, αν αυξάναμε πολύ τον αριθμό των ρίψεων, οι σχετικές συχνότητες θα πλησίαζαν το $1/6$ για κάθε αποτέλεσμα.

Εφαρμογή 10: Dice simulation without randi

Η εφαρμογή αυτή κάνει το ίδιο πράγμα, δηλαδή προσομοιώνει ρίψεις ζαριού, αλλά **χωρίς** να χρησιμοποιεί την randi.

Αρχικά δημιουργεί 100 τυχαίους αριθμούς από την ομοιόμορφη κατανομή στο $(0,1)$:

```
random = rand(100,1);
```

Μετά αρχικοποιεί όλα τα αποτελέσματα ως 1:

```
dice = ones(size(random));
```

Στη συνέχεια, για κάθε τυχαίο αριθμό, ελέγχει σε ποιο διάστημα ανήκει.

Η λογική είναι:

αν random <= 1/6 → dice = 1
αν random > 1/6 → dice = 2
αν random > 2/6 → dice = 3
αν random > 3/6 → dice = 4
αν random > 4/6 → dice = 5
αν random > 5/6 → dice = 6

Ουσιαστικά χωρίζουμε το διάστημα $(0,1)$ σε έξι ίσα υποδιαστήματα:

$$(0,1/6],\ (1/6,2/6],\ (2/6,3/6],\ (3/6,4/6],\ (4/6,5/6],\ (5/6,1)$$

και σε κάθε υποδιάστημα αντιστοιχούμε μία πλευρά του ζαριού.

Για παράδειγμα, αν ένας τυχαίος αριθμός είναι:

0.72

τότε είναι μεγαλύτερος από $1/6$, $2/6$, $3/6$, και $4/6$, αλλά όχι από $5/6$. Άρα στο τέλος παίρνει τιμή 5.

Η εφαρμογή αυτή είναι πιο αναλυτική από την randi, γιατί δείχνει **πώς μπορούμε να μετατρέψουμε μια ομοιόμορφη τυχαία μεταβλητή στο (0,1) σε διακριτή τυχαία μεταβλητή** με τιμές 1 έως 6.