

ΠΑΡΑΔΕΙΓΜΑΤΑ

ΚΟΛΟΒΟΥ ΑΘΑΝΑΣΙΑ

<http://users.uoa.gr/~akolovou/>

Βήματα (γενικά)

- Μοντελοποίηση του προβλήματος
- Ορισμός τυχαίων μεταβλητών
- Ορίζουμε σαν παράμετρο τον αριθμό των επαναλήψεων του πειράματος

Παράδειγμα 1

- Αν ρίξουμε ένα ζάρι 4 φορές τουλάχιστον 1 φορά θα εμφανιστεί το 6 .
- Αρχικά ορίζουμε ένα πίνακα που αποτελείται από 4 στήλες, και έχει τόσες γραμμές όσες και οι επαναλήψεις του πειράματος

1rst throw	2nd throw	3rd throw	4rth throw
2	5	1	6
3	4	1	5
2	1	6	6
...

- Ορίζουμε την τυχαία μεταβλητή $sixes$ η οποία παίρνει τιμή 1 αν πήραμε τουλάχιστον μία φορά 6 και 0 αν όχι.

Παράδειγμα 1 (συνέχεια)

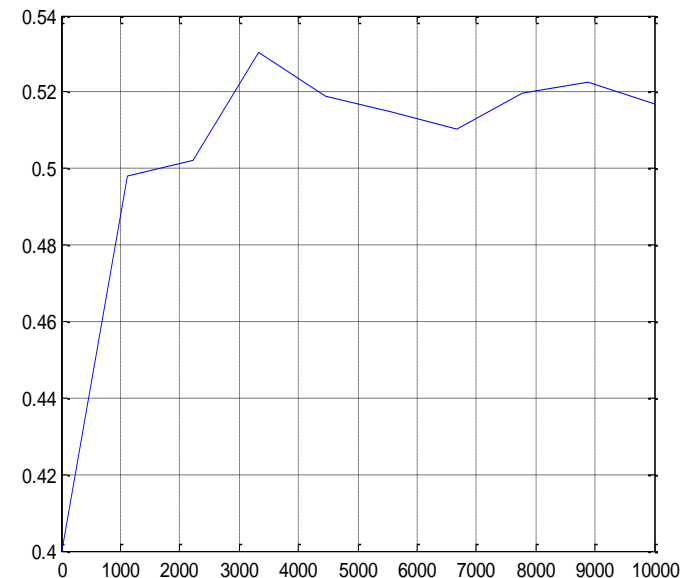
- Έπειτα θα χρησιμοποιήσουμε λογικές εκφράσεις για να μετατρέψουμε τις τιμές των τυχαίων αριθμών σε τιμές των τυχαίων μεταβλητών.
- Η έκφραση της μορφής `sixes == 6` δημιουργεί ένα νέο πίνακα ίσων διαστάσεων με τον πίνακα `sixes` που έχει 1 εκεί όπου ισχύει η συνθήκη και 0 αλλού.
- Αρκεί λοιπόν να βρούμε τις γραμμές του νέου αυτού πίνακα που έχουν 1 .

Παράδειγμα 1 (matlab implementation)

```
roll=100;  
out=floor(6*(rand(roll,4)))+1; %4 ρίψεις ενός ζαριού  
sixes = (out==6); %πόσες φορές πήραμε 6 στις ρίψεις  
success=sum(sixes,2)>=1; %επιτυχία αν πήραμε 6 τουλάχιστον 1 φορά  
sum(success)/roll
```

- Θα τρέξουμε το πείραμα για διαφορετικό αριθμό επαναλήψεων από 10 έως 10000

```
roll=linspace(10,10000,10);  
for i=1:size(roll,2)  
    out=floor(6*(rand(roll(i),4)))+1;  
    sixes = (out==6);  
    success=sum(sixes,2)>=1;  
    results(i)=sum(success)/roll(i);  
end  
plot(roll,results);grid on
```



Παράδειγμα 2

- Στο Las Vegas, ο τροχός μία ρουλέτας έχει 38 slots που αριθμούνται ως εξής 0, 00, 1, 2, ..., 36. Οι θέσεις 0 και 00 είναι πράσινες και οι μισές από τις υπόλοιπες 36 είναι κόκκινες και οι άλλες μισές είναι μαύρες. Ο κρουπιέρης γυρίζει τη ρουλέτα και πετάει τη μπίλια στον τροχό. Αν ποντάρεις ένα ευρώ στο κόκκινο κερδίζεις ένα ευρώ αν η μπίλια σταματήσει στο κόκκινο, διαφορετικά χάνεις ένα ευρώ.
- Να γραφτεί πρόγραμμα που υπολογίζει τα κέρδη για ένα παίχτη που ποντάρει 500 φορές στο κόκκινο.

Παράδειγμα 2

- Ορισμός τυχαίων μεταβλητών `wins` που παίρνει τιμή 1 αν κερδίζω στο στοίχημα ή 0 αν χάνω, και `loses` ($\text{loses} = \text{wins} - 1$) που εξαρτάται από την `wins` και παίρνει τιμές 0 αν κερδίζω ή -1 αν χάνω στο στοίχημα.
- Το αποτέλεσμα θα είναι $\text{winnings} = (\text{wins} + \text{loses});$

Παράδειγμα 2

- Αρχικά ορίζουμε ένα πίνακα που αποτελείται από μία στήλη, και έχει τόσες γραμμές όσες και οι επαναλήψεις του πειράματος.
- Έπειτα θα χρησιμοποιήσουμε λογικές εκφράσεις για να μετατρέψουμε τις τιμές των τυχαίων αριθμών σε τιμές των τυχαίων μεταβλητών.
- Μία έκφραση της μορφής `reds < 18 / 38` στο Matlab παράγει έναν πίνακα ίσων διαστάσεων με τον πίνακα `reds` που έχει ένα εκεί όπου ισχύει η συνθήκη $P(1) = 18/38$ και 0 αλλού, $P(0) = 20/38$.

Παράδειγμα 2 (συνέχεια)

- Ας το δούμε τώρα ολοκληρωμένο για 500 επαναλήψεις

```
runs=500;
```

```
red=rand(runs,1);
```

```
wins=red<18/38;
```

```
loses=wins-1;
```

```
winnings=sum(wins)+sum(loses);
```

- Ένα αποτέλεσμα για παράδειγμα είναι

```
winnings=-50
```

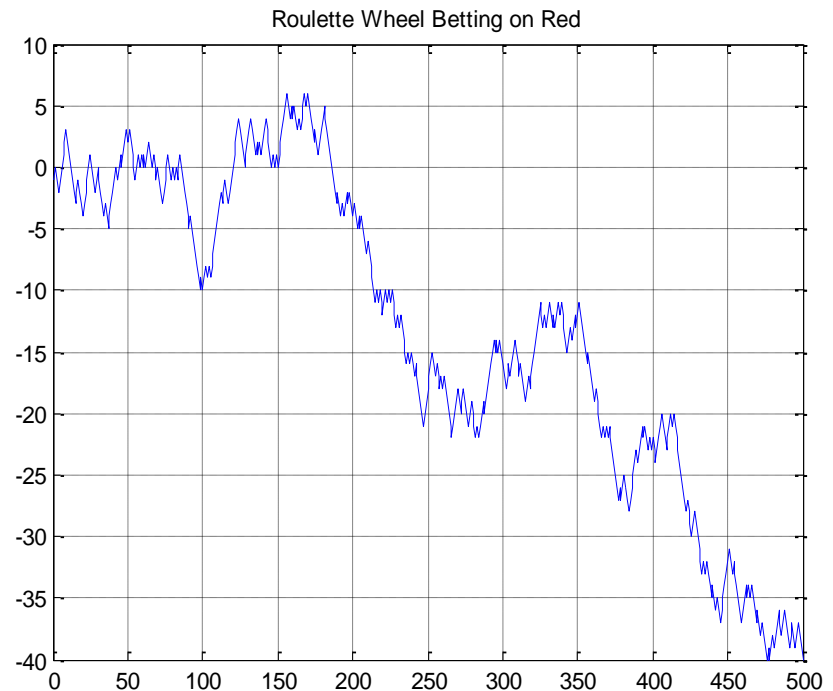
☹ no luck at all!!

Παράδειγμα 2 – plot results

- Μπορούμε να χρησιμοποιήσουμε την εντολή `cumsum` (cumulative sum) που υπολογίζει τα μερικά αθροίσματα σε ένα πίνακα (για παράδειγμα σε μία ακολουθία $\{a,b,c \dots\}$ τα μερικά αθροίσματα θα είναι $a, a+b, a+b+c \dots$)

```
winnings=cumsum(wins+loses);
```

```
plot(1:runs,winnings);grid;
```



Παράδειγμα 3

- Ένα άλλο πρόβλημα θα ήταν να πούμε ότι ποντάρουμε ότι ένας συγκεκριμένος αριθμός, πχ το 15 θα εμφανιστεί. Αν η μπίλια σταματήσει στον αριθμό παίρνεις πίσω το ευρώ σου συν 50 ευρώ ακόμα. Αν όχι, χάνεις το ευρώ σου.
- Να γραφτεί πρόγραμμα που μας υπολογίζει τα κέρδη για κάποιον που ποντάρει 500 φορές στο 15.

Παράδειγμα 3

- Αρχικά ορίζουμε ένα πίνακα που αποτελείται από μία στήλη, και έχει τόσες γραμμές όσες και οι επαναλήψεις του πειράματος.
- Έπειτα θα χρησιμοποιήσουμε λογικές εκφράσεις για να μετατρέψουμε τις τιμές των τυχαίων αριθμών σε τιμές των τυχαίων μεταβλητών.
- Μία έκφραση της μορφής `number_15 < 1/38` στο Matlab παράγει έναν πίνακα ίσων διαστάσεων με τον πίνακα `number_15` που έχει ένα εκεί όπου ισχύει η συνθήκη $P(1)=1/38$ και 0 αλλού, $P(0)=37/38$.

Παράδειγμα 3 (συνέχεια)

```
runs=500;
```

```
bet=rand(runs,1);
```

```
number_15=bet<1/38;
```

```
other_numbers=number_15-1;
```

```
winnings=cumsum(number_15*50+other_numbers);
```

```
plot(1:runs,winnings);grid;
```



Παράδειγμα 4

- Μία μηχανή παγωτού λειτουργεί κανονικά όταν και τα τέσσερα υποσυστήματα της (A,B,C,D) λειτουργούν κανονικά.
- Τα υποσυστήματα είναι ανεξάρτητα και καθένα από αυτά έχει πιθανότητα βλάβης $1/4$
- Δεδομένου ότι η μηχανή σήμερα σήμερα δεν λειτουργεί, ποια είναι η πιθανότητα μόνο το A να ευθύνεται για αυτό ?

Παράδειγμα 4 (συνέχεια)

- Ορισμός τυχαίων μεταβλητών A, B, C, D που παίρνουν την τιμή 1 ή 0.
- Ορίζουμε επίσης και την τυχαία μεταβλητή ICE , που παίρνει τιμή 1 ή 0 ανάλογα με το αν λειτουργεί ή όχι η μηχανή.
- Ορίζουμε σαν παράμετρο τον αριθμό των επαναλήψεων του πειράματος

Παράδειγμα 4 (συνέχεια)

- Αρχικά ορίζουμε ένα πίνακα που αποτελείται από μία τετράδα τυχαίων αριθμών, και έχει τόσες γραμμές όσες και οι επαναλήψεις του πειράματος.
- Έπειτα θα χρησιμοποιήσουμε λογικές εκφράσεις για να μετατρέψουμε τις τιμές των τυχαίων αριθμών σε τιμές των τυχαίων μεταβλητών.
- Μία έκφραση της μορφής `subsystem>0.25` στο Matlab παράγει έναν πίνακα που έχει 1 εκεί όπου ισχύει η συνθήκη, $P(1)=3/4$ και 0 αλλού, $P(0)=1/4$.

Παράδειγμα 4 (συνέχεια)

- Το αποτέλεσμα του πειράματος μπορεί να προκύψει από μία τετράδα τυχαίων τιμών που συμβολίζουν αν τα ανεξάρτητα υποσυστήματα λειτουργούν ή όχι

A	B	C	D
0	1	1	0
1	0	0	1
...

- Η τυχαία μεταβλητή ICE γίνεται 1 όταν μία ολόκληρη γραμμή στον παραπάνω πίνακα είναι 1.
- Χρησιμοποιούμε την εντολή `sum` για να το εκφράσουμε αυτό και συγκεκριμένα αν $sum < 4$ η μηχανή δεν λειτουργεί.

Παράδειγμα 4 (συνέχεια)

- Με την συνθήκη $\text{sum} < 4$ θα πάρουμε ένα νέο πίνακα που περιέχει μόνο τις τετράδες που το άθροισμα είναι μικρότερο του 4.
- Στη συνέχεια από αυτές τις τετράδες θέλουμε μόνο να κρατήσουμε εκείνες μόνο που έχουν $A=0$, $B=C=D=1$.
- Έτσι θα πάρουμε ένα νέο πίνακα που περιέχει μόνο τις τετράδες που μας ενδιαφέρουν (A δεν λειτουργεί και όλα τα άλλα λειτουργούν)

Παράδειγμα 4 (matlab implementation)

```
runs=100;
subsystem=rand(runs,4);
failure=subsystem>0.25;
x=find ( sum(failure,2)<4 ) ; % sum(A,2) means sum each row
%want to keep only the lines with sum<4 , so we construct a new
array
new=failure(x(:,1),:);
%choose the lines from new with A=0, B=C=D=1 (sum of row
equals 3 at this case)
result= find ( new(:,1)==0 & sum(new,2)==3 );
p=length(result)/length(new)
```