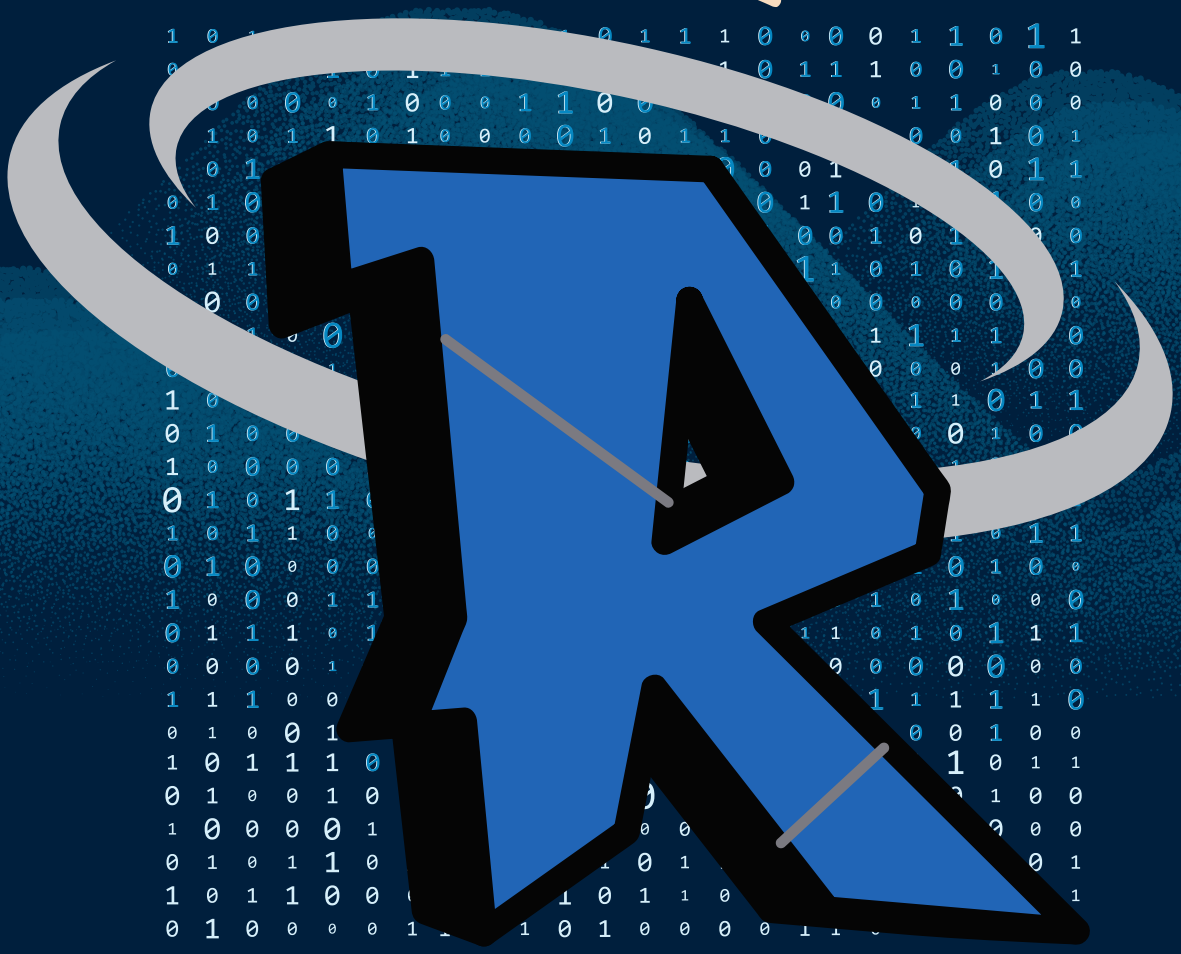
Problem Solving

With



PRESENTED BY

Ebtssam Hassan	20201491232
Rahma Ezzat	20201381362
Aya Mohammed	20201379022
Asmaa Hamdy	20201032790
Ziad Ahmed	20201376238
Abdelrahman salah	20201444840
Omar Mekkawy	20201375851

1. Box A contains 2 white balls and box B contains 4 red balls at time n, select one ball from each box and interchange them, considering X_n is the number of red balls in A, The transition matrix:

$$P=\begin{bmatrix} 0 & 1 & 0 \\ \frac{1}{8} & \frac{4}{8} & \frac{3}{8} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Answer :

Let's model this Markov Chain using R.

We will start by creating a transition matrix

Code:

```
1  library(markovchain)
2  library(diagram)
3  p <- matrix(c(0,1,0 ,1/8,0.5,3/8 ,0,0.5,0.5),nrow = 3,byrow = TRUE)
4
5  states <- c("0","1","2")
6
7  row.names(p) <- states
8  colnames(p) <- states
9  p
```

Output:

```
> p
      0   1     2
0 0.000 1.0 0.000
1 0.125 0.5 0.375
2 0.000 0.5 0.500
```

- ***In the above code,***

states refer to the state space of the Markov Chain; while P represents the transition matrix that gives the probabilities of moving from one state to another.

- There is a package in R ' Markov-chain' which can help us save time in implementing Markov Chains in R.

Code:

```
markov.p <- new("markovchain",
                states=states,
                transitionMatrix=
                  matrix(c(0,1,0 ,1/8,0.5,3/8 ,0,0.5,0.5),
                         nrow = 3,byrow = TRUE,
                         dimnames = list(states,states)))
markov.p
```

Output:

```
> markov.p
Unnamed Markov chain
 A  3 - dimensional discrete Markov Chain defined by the following states:
 0, 1, 2
 The transition matrix  (by rows)  is defined as follows:
       0   1     2
0 0.000 1.0 0.000
1 0.125 0.5 0.375
2 0.000 0.5 0.500
```

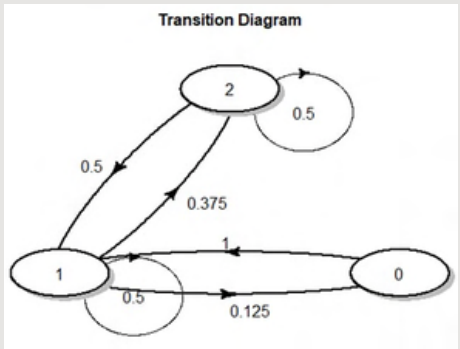
- Now, to plot the above transition matrix we can use the R package, "diagram".The "diagram" package has a function called "plot at" that can help us plot a state space diagram of the transition matrix in an easy-to-understand manner.

Code:

```
plotmat(t(p) |, box.prop = 0.6, self.shifty = -.06 ,self.shiftx = .15,main = "Transition Diagram")
```

Box.prob length/width ratio of label box
self.shiftx relative shift of self-arrow, in x-direction
self.shifty relative shift of self-arrow, in y-direction

Output:



- ***Now, the above Markov Chain can be used to answer some of the future state questions. Like P(X_2 =1 | X_1 = 0) = 1***

We can similarly calculate for two steps. Let's calculate the probability of coming back to the 0 state in the second step.

Code: `markov.p^2`

Output:

```
 The transition matrix  (by rows)  is defined as follows:
        0      1      2
0 0.1250 0.5000 0.3750
1 0.0625 0.5625 0.3750
2 0.0625 0.5000 0.4375
```

We have a probability of 0.0625 of returning to state 0 in the second.

The Markov Chain reaches an equilibrium called a stationary state. In this case, the starting point becomes completely irrelevant. The stationary state can be calculated using some linear algebra methods; however, we have a direct function, 'steady states', in R, which makes our lives easier.

Code:

```
> steadyStates(markov.p)
                0         1   2
[1,] 0.06666667 0.5333333 0.4
```

In the stationary state, we have a probability of 0.06666667 of ending up in the state 0 and a probability of 0.4 of reaching state 2.

- Let's say we have to complete 20 steps. Will 20 steps take our Markov Chain to the stationary state?

```
> markov.p^20
Unnamed Markov chain^20
 A  3 - dimensional discrete Markov Chain defined by the following states:
 0, 1, 2
 The transition matrix  (by rows)  is defined as follows:
           0         1   2
0 0.06666667 0.5333333 0.4
1 0.06666667 0.5333333 0.4
2 0.06666667 0.5333333 0.4

> markov.p^21
Unnamed Markov chain^21
 A  3 - dimensional discrete Markov Chain defined by the following states:
 0, 1, 2
 The transition matrix  (by rows)  is defined as follows:
           0         1   2
0 0.06666667 0.5333333 0.4
1 0.06666667 0.5333333 0.4
2 0.06666667 0.5333333 0.4
```

Since the probabilities in the 20th and 21st steps are coming to be equal, we can say that we have reached the stationary state.

2. A coin is flipped repeatedly until three heads appear in a row. what is the expected number of flips needed?

States are {No head, H, HH, HHH***}***

The transition matrix:

$$P=\begin{bmatrix} 0.5 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Answer :

Let's model this Markov Chain using R.

We will start by creating a transition matrix of the zone movement probabilities.

Code:

```
1 library(markovchain)
2 library(diagram)
3 states <- c("T","H","HH","HHH")
4 x <- matrix(c(0.5,0.5,0,0,0.5,0,0.5,0,0.5,0,0,0.5,0,0,0,1),nrow = 4,byrow = TRUE)
5 row.names(x) = states
6 colnames(x) =  states
7 x
```

Output:

```
> x
      T   H  HH HHH
T   0.5 0.5 0.0 0.0
H   0.5 0.0 0.5 0.0
HH  0.5 0.0 0.0 0.5
HHH 0.0 0.0 0.0 1.0
```

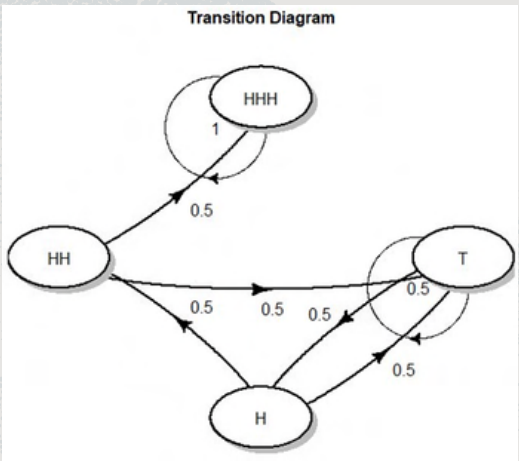
We called the "Markov chain" function

Now, to plot the above transition matrix we can use the R package, "diagram".The "diagram" package has a function called "plotmat" that can help us plot a state space diagram of the transition matrix in an easy-to-understand manner.

Code:

```
plotmat(t(x) , box.prop = 0.6,
        self.shifty = -.06
        ,self.shiftx = -.09,
        main = "Transition Diagram")
```

Output:



- ***Now, the above Markov Chain can be used to answer some of the future state questions. Like P(X_2=H | X_1 = HH) = 0***

Now, how can we find the
expected numbers of flips needed to reach the absorbing state (HHH)?
Simply we can use the Q matrix
it's a subset of the transition matrix X and it represents the probability of transitions between transient states and itself
We can do our task by the rule $[\![(I-Q)]\!]^{(-1)}$
Let's do this with R
We used the library MatLab to inverse our matrix by the 'solve' function

Code:

```
12  q = x[c(1,2,3),c(1,2,3)]
13  q
14  i = diag(3)
15  i
16  install.packages('matlab')
17  library(matlab)
18  exp = solve(i-q)
19  exp
```

Output:

```
> exp = solve(i-q)
> exp
   T H HH
T  8 4  2
H  6 4  2
HH 4 2  2
```

- ***We can conclude that (8+4+2) = 14***

is the expected number of absorbing starting from state 'T'

Thanks