

# Introduction to Markov chains

In this worksheet we simulate a random process called *Markov chain*. Our source of randomness will be the function `random()` that returns a random floating point number between 0 and 1 uniformly

```
sage: print random()
sage: print random()
sage: [random() for i in range(10)]
```

## Discrete random variables from `random()`

### Coin tossing

Write a Sage function `toss()` that returns 0 or 1, each with probability 1/2.

```
sage: # edit here
```

### Biased coin toss (a.k.a. Bernoulli measure)

Write a Sage function `biased_toss(p)` that, given a number  $p \in [0, 1]$  returns 0 with probability  $p$  and 1 with probability  $1 - p$ .

```
sage: # edit here
```

### Finite probabilities

Let  $p = (p_0, p_1, \dots, p_{n-1}) \in [0, 1]^n$  be such that  $\sum_{j=0}^{n-1} p_j = 1$ . Write a function `multi_toss(p)` so that given a vector  $p$  returns an integer  $\{0, 1, \dots, n-1\}$  with probability  $p_j$ .

```
sage: # edit here
```

## Markov chain simulation

Now we consider a *stochastic matrix*, ie a square matrix  $P = (p_{ij})_{0 \leq i, j < n}$  with non-negative coefficients and such that

$$\forall i \in \{0, 1, \dots, n-1\}, \quad \sum_{j=0}^{n-1} p_{ij} = 1$$

Here are four examples

```
sage: P1 = 1/3 * matrix(4, [0, 3, 0, 0, 1, 0, 2, 0, 0, 2, 0, 1, 0, 0, 3, 0])
sage: P2 = 1/12 * matrix(5, [0, 4, 4, 4, 0, 4, 0, 0, 4, 4, 6, 0, 0, 6, 0, 3, 3, 3, 0, 3, 0, 6, 0, 6, 0])
sage: P3 = 1/10 * matrix(5, [6, 4, 0, 0, 0, 3, 7, 0, 0, 0, 0, 5, 0, 5, 0, 0, 0, 0, 2, 8, 0, 0, 0, 6, 4])
sage: P4 = 1/3 * matrix(4, [0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0])
```

Write a function `is_stochastic_matrix(P)` that checks whether the input  $P$  is a stochastic matrix and returns True or False accordingly

```
sage: # edit here
```

Check that the matrices  $P_1, P_2, P_3$  and  $P_4$  are stochastic matrices

```
sage: # edit here
```

Let  $P = (p_{ij})_{0 \leq i, j < n}$  be a stochastic matrix. We define a random process  $X_0, X_1, \dots$  as follows. Each  $X_t$  is an element of  $\{0, 1, \dots, n-1\}$  (ie the indices of the stochastic matrix are "states" of the process). Assuming that  $X_t$  is in the state  $i$  the variable  $X_{t+1}$  is chosen randomly in  $\{0, 1, \dots, n-1\}$  according to the probability vector  $(p_{ij})_{0 \leq j < n}$  (ie the  $i$ -th row of the matrix  $P$ ).

Using the function `multi_toss(p)` that you already wrote, write a function `transition(P, i)` where  $P$  is a  $n \times n$  stochastic matrix and  $i$  an index in  $\{0, 1, \dots, n-1\}$ , returns an integer in  $\{0, 1, \dots, n-1\}$  according to the probabilities given by the  $i$ -th row of  $P$ .

```
sage: # edit here
```

Check that your function works using the matrices  $P_1, P_2, P_3$  and  $P_4$  that were previously defined.

```
sage: # edit here
```

Write a function `trajectory(P, x0, t)` that given a  $n \times n$  stochastic matrix, an initial condition  $x_0$  and a positive integer  $t$  return a simulation of the Markov chain of length  $t$  starting from  $x_0$ .

```
sage: # edit here
```

Use your function to get trajectories of length 20 for the four matrices  $P_1, P_2, P_3$  and  $P_4$ .

```
sage: # edit here
```

Make a graphics out of these trajectories (you can use the primitive `point2d` or `line2d`)

```
sage: # edit here
```

## Statistical properties of trajectories

Given a Markov chain  $X_0, X_1, \dots$  we will be interested in the number of times the chain visits a given states between time 0 and  $t$ . Let us define

$$S_N(i) := \#\{t \in \{0, 1, \dots, N-1\} : X_t = i\}$$

(Note that  $S_n(i)$  is a random variable that depends on the realisation of the Markov chain).

Write a function `statistics(P, x0, i, N)` that simulates the Markov chain given by  $P$  and returns the list of values  $S_0(i), S_1(i), S_2(i), \dots, S_{N-1}(i)$  obtained along this trajectory.

```
sage: # edit here
```

On the same graphics, plot several realisation of the sequence  $(S_n(0))_n$  for the Markov chain given by the matrix  $P_1$

```
sage: # edit here
```

Repeat the operation with  $S_n(0)$  replaced by  $S_n(i)$  for all possible values of  $i$ .

```
sage: # edit here
```

What do you observe?

Make the same experiment with the other matrices  $P_2, P_3$  and  $P_4$ .

```
sage: # edit here
```

## Equilibrium

We are now given a stochastic matrix  $P$  a fixed initial condition  $X_0$  and an integer  $t$  and are interested in the random variable  $X_t$ . That is to say the distribution of the Markov chain after  $t$  steps.

Write a function `distribution(P, x0, t)` that given a matrix  $P$ , an initial condition  $x_0$  and a time  $t$  return  $x_t$

```
sage: # edit here
```

Write a function `sample(P, x0, t, num)` that given a matrix  $P$  an initial condition  $x_0$ , a time  $t$  and a positive integer  $num$  return a sample of  $num$  simulation of  $x_t$

```
sage: # edit here
```

Using samples of size 1000, compute the empirical distribution of  $x_{100}$  for the matrix  $P_1$

```
sage: # edit here
```

For  $n = 0, 1, 2, \dots, 100$  compute  $u(P_1)^n$  where  $u$  is the row vector of floating point numbers  $u = (1.0, 0.0, 0.0, 0.0)$

```
sage: # edit here
```

What do you observe?

Repeat the operation starting from the row vector  $u = (0.25, 0.25, 0.25, 0.25)$

```
sage: # edit here
```

Make the same experiment with the matrices  $P_2$ ,  $P_3$  and  $P_4$

```
sage: # edit here
```