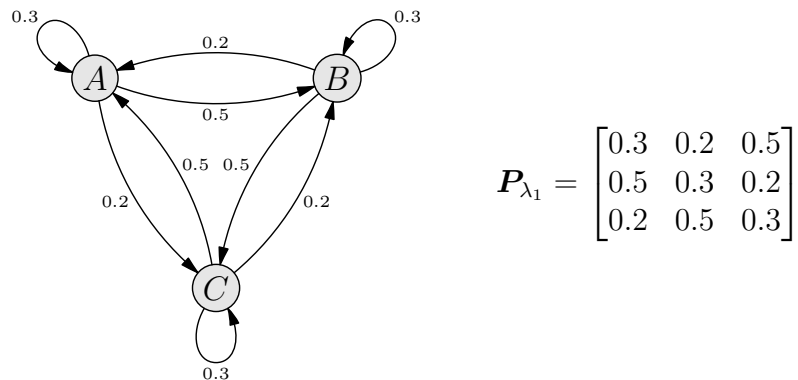


## Inference with Markov models

Recall the state transition diagram and transition probability matrix of a Markov chain  $\lambda_1$  which models the behavior of an NPC who is “patrolling the map”



The probability of the event  $E_1$  = “the NPC goes from room  $C$  to room  $B$  and then to room  $A$ ” amounts to

$$p(E_1) = p(X_1 = B \mid X_0 = C) \cdot p(X_2 = A \mid X_1 = B)$$

and the corresponding log-likelihood amounts to

$$\mathcal{L}(E_1) = \ln p(X_1 = B \mid X_0 = C) + \ln p(X_2 = A \mid X_1 = B)$$

Compute both these values for the model  $\lambda_1$ . Round your results to *two* decimal places.

$$p(E_1) =$$

$$\mathcal{L}(E_1) =$$

Also for model  $\lambda_1$ , compute the probability and log-likelihood of the event  $E_2$  = “the NPC goes from room  $C$  to room  $A$  and then to room  $B$ ”

$$p(E_2) =$$

$$\mathcal{L}(E_2) =$$

## Sampling a Markov model

Given a given Markov chain  $\lambda$  over some state space  $\mathcal{S}$ , the following procedure generates a sequence of states  $s = s[0]s[1]s[2] \dots$

manually initialize  $s[0] \in \mathcal{S}$

if  $s[t-1] = s_i$ , *draw*  $s[t] = s_j$  according to  $p(s_j \mid s_i)$ , that is select

$$s_j \sim (P_\lambda)_{ji}$$

where  $P_\lambda$  is the state transition probability matrix of the process  $\lambda$ .

Using matrix  $P_{\lambda_1}$ , generate a sequence  $s$  of 200 states where  $s[0] = A$ . That is, replace the dots in the following expression by 199 symbols sampled according to the above algorithm.

$$s = A \dots$$

Using matrix  $P_{\lambda_1}$ , generate  $n = 1000$  sequences  $s_l$  of length 200 where  $s_l[0] = A$  for all  $l$ . Compute their average log-likelihood

$$\frac{1}{n} \sum_{l=1}^n \mathcal{L}(s_l) =$$

## Learning a Markov model

Given a state sequence  $s$  such as the one you just created, we may use maximum likelihood estimation techniques to estimate or “learn” a Markov model  $\lambda_{MLE}$  that is most likely to have produced  $s$ .

Maximum likelihood estimates of the entries of the corresponding matrix  $P_{\lambda_{MLE}}$  are given by

$$(P_{\lambda_{MLE}})_{ji} = \frac{n_{ji}}{\sum_j n_{ji}}$$

where the quantity  $n_{ji}$  counts how often the transition  $s_i \rightarrow s_j$  occurs in sequence  $s$ .

Given the “training” sequence  $s$  you created above, compute the entries of  $P_{\lambda_{MLE}}$ . Round your results to two decimals.

$$P_{\lambda_{MLE}} =$$

## Stationary distribution a Markov model

Compute the stationary distribution

$$\pi = P_{\lambda_1} \pi$$

of the Markov chain  $\lambda_1$ . Proceed as follows: introduce

$$A = \begin{bmatrix} I - P_{\lambda_1} \\ \mathbf{1}^\top \end{bmatrix} \quad \text{and} \quad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

where the matrix  $I$  and the vector  $0$  and  $1$  have to be of appropriate sizes. Then solve

$$A \pi = b$$

for  $\pi$ . Round your result to two decimals.

$$\pi =$$

If you like, paste the code you used to practically compute your result after the following `import` statements

```
import numpy as np
import numpy.linalg as la
```