Matrix Models: Markov Chains

Problems in the context of software development

PID_00266372

Marcel Serra Julià

Marcel Serra Julià

I got my degree in Mathematics at the Universitat Autònoma de Barcelona (UAB) in 2009 and did my eHealth Masters at the McMaster University (Canada) in 2011. While I did my graduate training I was also a teaching assistant in the Department of Computer Sciences at the McMaster University. In 2012, I developed my professional career as a Software Developer working for several companies in the digital sector in the Barcelona area, including Mascoteros (a marketplace for the pet industry) and YaEncontre (a real estate website). In these companies I specialized in web development with Symfony, a popular PHP framework based on MVC architecture. The rapidly changing digital industry was a catalyst which led me to look to new technologies such as Big Data, Computer Vision and Computational Modeling. As a result of my expertise and motivation for research I am currently working as a Data Manager at ICTA (Institut de Ciència i Tecnologia Ambiental, UAB).

The assignment and creation of this UOC Learning Resource have been coordinated by the lecturer: Cristina Cano Bastidas (2020)

First edition: February 2020

Av. Tibidabo, 39-43, 08035 Barcelona
Publishing: FUOC

1. Problems

MATRIX MODELS: MARKOV CHAINS

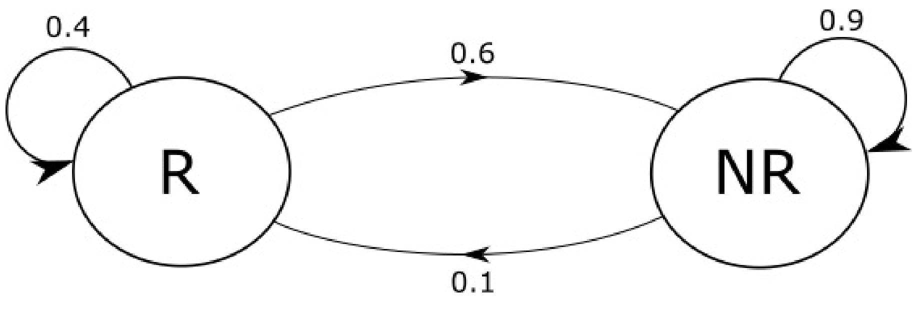
Topics: *Introduction to discrete matrix models. Discrete-time Markov Chains. States diagram and transition probability. Evolution over time of Markov Chains. Positive Matrices and dominant eigenvalues. Stationary states distribution. Applications.*

A Markov Chain is a stochastic process in which the system has no memory. This means that future states only depend on the present state and the transition probabilities between states. Hence, to determine a future state in a Markov Chain it is not necessary to know anything about the past states.

This module will focus on discrete and homogenous Markov Chains. *Homogenous* means that transition probabilities are independent of time. Transition probabilities will be expressed by a matrix P, where the element p_{ij} is the probability of moving from state j to state i. In addition, the state of the system on a given time will be represented by the vector X_t. Markov Chain iterations will be represented by $X_{t+1} = PX_t$.

The following example illustrates how Markov Chains can be used for modelling. After observing the weather for a long time, we have learnt that if we have a rainy day (R) the probability that it would be rainy a day later is 0.4, while the probability that it would be dry (NR) a day later is 0.6. We have also observed that when we have a dry day the probability that it would be rainy a day later is 0.1 and the probability that it would be dry is 0.9. Given this observation, the information gathered is depicted in the diagram below:

Figure 1. Rain diagram



This diagram can be transformed into a matrix format, where each column represents the probability of moving from each initial state to any other state. The following table summarizes these probabilities:

		Initial States	
		R	*NR*
Arrival States	*R*	0.4	0.1
	NR	0.6	0.9

The associated transition matrix for this Markov Chain is:

$$P = \begin{array}{c} \\ R \\ NR \end{array} \begin{array}{c} \begin{array}{cc} R & NR \end{array} \\ \begin{pmatrix} 0.4 & 0.1 \\ 0.6 & 0.9 \end{pmatrix} \end{array}$$

Any state is described by a vector. We assume it is not raining today, so today's state is $X_0 = (0,1)$. In order to compute probabilities for each state over the next 3 days, we will iterate as follows:

$$X_1 = \begin{array}{c} \begin{array}{cc} R & NR \end{array} \\ \begin{pmatrix} 0.4 & 0.1 \\ 0.6 & 0.9 \end{pmatrix} \end{array} \begin{array}{c} X_0 \\ \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{array} = \begin{array}{c} X_1 \\ \begin{pmatrix} 0.1 \\ 0.9 \end{pmatrix} \end{array} \begin{array}{c} \\ R \\ NR \end{array}$$

$$X_2 = \begin{array}{c} \begin{array}{cc} R & NR \end{array} \\ \begin{pmatrix} 0.4 & 0.1 \\ 0.6 & 0.9 \end{pmatrix} \end{array} \begin{array}{c} X_1 \\ \begin{pmatrix} 0.1 \\ 0.9 \end{pmatrix} \end{array} = \begin{array}{c} X_2 \\ \begin{pmatrix} 0.13 \\ 0.87 \end{pmatrix} \end{array} \begin{array}{c} \\ R \\ NR \end{array}$$

$$X_3 = \begin{array}{c} \begin{array}{cc} R & NR \end{array} \\ \begin{pmatrix} 0.4 & 0.1 \\ 0.6 & 0.9 \end{pmatrix} \end{array} \begin{array}{c} X_2 \\ \begin{pmatrix} 0.13 \\ 0.87 \end{pmatrix} \end{array} = \begin{array}{c} X_3 \\ \begin{pmatrix} 0.122 \\ 0.861 \end{pmatrix} \end{array} \begin{array}{c} \\ R \\ NR \end{array}$$

Before moving on to other examples, we will review some properties of Markov Chains and their transition matrices. These properties will not be proven but will be validated in our rain model.

Markov Chains are stochastic processes and, therefore, the transition matrix satisfies the definition of a stochastic process. This definition requires the following two properties:

- All elements in a given column add to 1.
- None of the entries are negative.

Considering this, the matrix *P* from our rain model satisfies these conditions.

For any stochastic process it can be proven that:

- If P is stochastic, then P^n is stochastic.
- The eigenvalues of P are all equal or smaller than 1.
- Given the transition matrix P and the vector of initial states, any future state of a Markov Chain can be determined as follows:

$$X_n = PX_{n-1} = PPX_{n-2} = \ldots = P^n X_0$$

Then, the computation of the aforementioned X_3 can be done as follows:

$$X_3 = P^3 X_0 = \overset{P^3}{\begin{pmatrix} 0.064 & 0.001 \\ 0.216 & 0.729 \end{pmatrix}} \overset{X_0}{\begin{pmatrix} 0 \\ 1 \end{pmatrix}} = \overset{X_3}{\begin{pmatrix} 0.122 \\ 0.861 \end{pmatrix}} \begin{array}{l} R \\ NR \end{array}$$

Hence, knowing the state n of a Markov Chain is equivalent to computing the n power of a matrix. Depending on the complexity of the transition matrix it may be useful to use the knowledge acquired on matrix diagonalization to compute the n power.

Let's recall what the **Diagonalization theorem** states: *A matrix P is diagonalizable if and only if it has linearly independent eigenvectors. Under these circumstances, it exists a matrix B such as $D = B^{-1}PB$, where matrix D is diagonal.*

We can now return to our example and compute the diagonal matrix. The eigenvalues of P are 0.3 and 1 and the associated eigenvectors (1, – 1) and (1,6), respectively. The Matrix B that satisfies $D = B^{-1}PB$ is

$$B = \begin{pmatrix} 1 & 1 \\ -1 & 6 \end{pmatrix} \quad \text{and the inverse will be} \quad B^{-1} = \begin{pmatrix} 6/7 & -1/7 \\ 1/7 & 1/7 \end{pmatrix}.$$

By computing $B^{-1}PB$ we obtain the diagonal matrix D with the eigenvalues of P in the diagonal. To compute the n power of a diagonal matrix we only need to raise the elements in the diagonal to the required power.

$$D = \begin{pmatrix} 0.3 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad D^3 = \begin{pmatrix} 0.3^3 & 0 \\ 0 & 1^3 \end{pmatrix}.$$

Hence, to compute P^3 we can also use the expression $P^3 = BD^3B^{-1}$. This could be extended to any power and matrix dimensions.

- The transition matrices of Markov Chains always have the eigenvalue 1, which is the dominant eigenvalue.
- Under certain circumstances, the steady state vector X that satisfies $PX = X$ coincides with the normalized eigenvector associated to the eigenvalue 1.
- Homogeneous Markov Chains always have a steady state.

In our example, we can see that by normalizing (1,6) the eigenvector associated to eigenvalue 1 we obtain $(1/\sqrt{37},6/\sqrt{37}) = (0.16439898,0,98639392)$

Notice that if we solved the following system and normalized the result, we would find the same solution

$$\begin{pmatrix} 0.4 & 0.1 \\ 0.6 & 0.9 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}$$

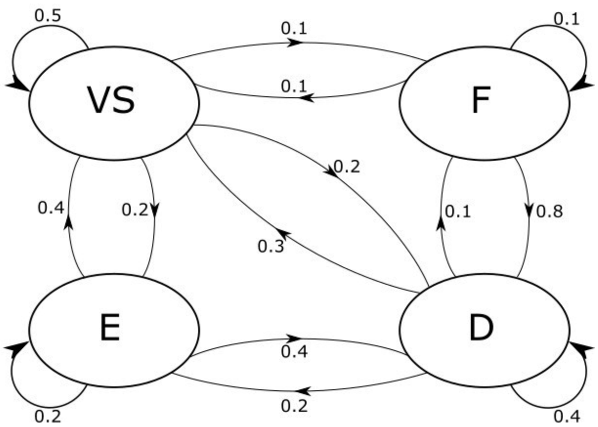
With the concepts we have introduced and the previous example we can now try to solve the following problems and self-assessment exercises.

1. An online university has 100,000 students around the world enrolled in some of their courses. The IT team of the university has developed 3 services for students to follow their courses:

- Video Streaming Service (VS). Students connect to it to have access to the content in video format.
- Online Forum (F). Students connect to it to interact with other students.
- Evaluation Platform (E). Students connect to it to answer quizzes and assess their learning progress.

At any point in time, students can be either connected to any of the services or disconnected (D). The team has been monitoring the migration of students from one service to another and have been able to generate the following diagram:

Figure 2. State Diagram



(a) Based on this diagram, find the transition matrix for the Markov Chain. Note: all columns on a Markov matrix add to 1.

(b) The following distribution of connections takes place at 9 am on Monday: Video Streaming (VS): 20%; Forum (F): 5%; Evaluation (E): 5%; Disconnected (D) 70%. What will the distribution be 1 hour later?

(c) How many connections to each service should be expected after 10 hours (7 pm)?

(d) How many students should be expected to be connected into each service in the long term?

(e) Using the software *R*, plot the expected evolution of connections over a period of 20 hours.

2. As software engineers, we have been asked to study an algorithm that wasn't written by us to evaluate the computing resources the algorithm needs. We have been able to determine that the algorithm has 3 states:

- Waiting or Idle (I)
- Processing Information (P)
- Accessing disk to store processed information (S)

The transition matrix is as follows:

$$\begin{array}{cc} & \begin{array}{ccc} I & P & S \end{array} \\ \begin{array}{c} I \\ P \\ S \end{array} & \begin{pmatrix} 1/2 & 1/10 & 2/3 \\ 1/2 & 1/2 & 0 \\ 0 & 4/10 & 1/3 \end{pmatrix} \end{array}$$

(a) Draw the state transition diagram for the given matrix.

(b) If at t_0 the algorithm was at the I state, what is the probability that the algorithm would be at state P for $t = 1$? And what is the probability that the algorithm would remain at state S for $t = 1$?

(c) Which are the probabilities for each of the states for $t = 5$?

(d) What is the expected behaviour of the given algorithm in the long term?

2. Solutions to the Problems

1. a) The transition matrix for the given diagram is:

$$M = \begin{array}{c} \\ VS \\ F \\ E \\ D \end{array} \begin{array}{c} \begin{array}{cccc} VS & F & E & D \end{array} \\ \begin{pmatrix} 0.5 & 0.1 & 0.4 & 0.3 \\ 0.1 & 0.1 & 0 & 0.1 \\ 0.2 & 0 & 0.2 & 0.2 \\ 0.2 & 0.8 & 0.4 & 0.4 \end{pmatrix} \end{array} .$$

b) By multiplying the initial distribution vector by the transition matrix M we will get the distribution one hour later.

$$X_1 = \underbrace{\begin{pmatrix} 0.5 & 0.1 & 0.4 & 0.3 \\ 0.1 & 0.1 & 0 & 0.1 \\ 0.2 & 0 & 0.2 & 0.2 \\ 0.2 & 0.8 & 0.4 & 0.4 \end{pmatrix}}_{M} \underbrace{\begin{pmatrix} 0.2 \\ 0.05 \\ 0.05 \\ 0.7 \end{pmatrix}}_{X_0} = \underbrace{\begin{pmatrix} 0.335 \\ 0.095 \\ 0.19 \\ 0.38 \end{pmatrix}}_{X_1} .$$

c) To compute the distribution of connections after 10 hours, we can iterate 10 times:

$$\begin{aligned} X_1 &= MX_0 \\ X_2 &= MX_1 \\ &\dots \\ X_{10} &= MX_9 \end{aligned}$$

which can also be computed as $X_{10} = M^{10}X_0$

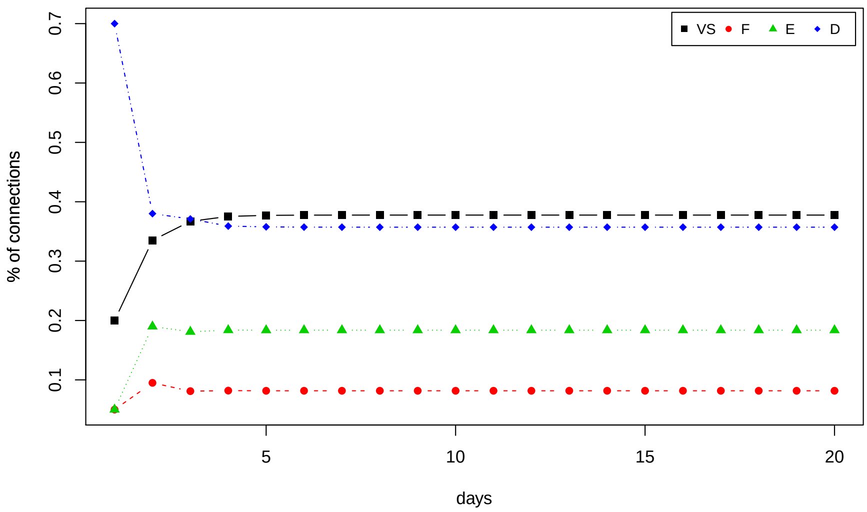
$$X_{10} = \begin{pmatrix} 0.5 & 0.1 & 0.4 & 0.3 \\ 0.1 & 0.1 & 0 & 0.1 \\ 0.2 & 0 & 0.2 & 0.2 \\ 0.2 & 0.8 & 0.4 & 0.4 \end{pmatrix}^{10} \begin{pmatrix} 0.2 \\ 0.05 \\ 0.05 \\ 0.7 \end{pmatrix} = \begin{pmatrix} 0.37755099 \\ 0.08163265 \\ 0.18367347 \\ 0.35714289 \end{pmatrix} .$$

d) In the long term, the distribution of students is given by the normalized eigenvector corresponding to the eigenvalue 1.

$$\begin{array}{c} VS \\ F \\ E \\ D \end{array} \begin{pmatrix} 0.37755102 \\ 0.08163265 \\ 0.18367347 \\ 0.35714286 \end{pmatrix}.$$

e) The evolution of connections over the first 20 hours is depicted in the following graph:

Figure 1. Rain diagram



The required *R* code used to solve this exercise is the following:

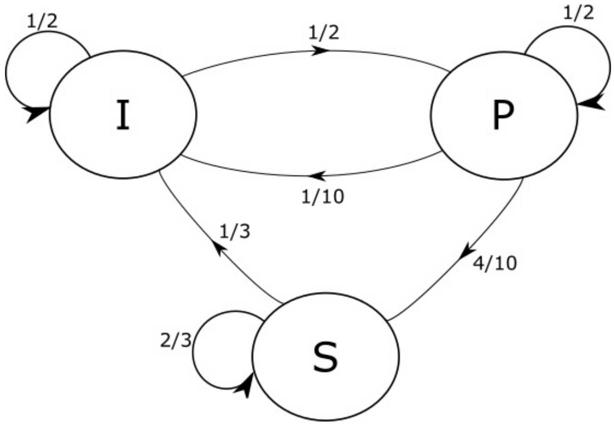
```
> stateLabels<-c("VS","F","E","D")
> M<-matrix(c(0.5,0.1,0.4,0.3,0.1,0.1,0,0.1,0.2,0,0.2,0.2,0.2,0.8,0.4,0.4),4,4,
 byrow=TRUE, dimnames = list(stateLabels, stateLabels))
> M
    VS   F   E   D
VS 0.5 0.1 0.4 0.3
F  0.1 0.1 0.0 0.1
E  0.2 0.0 0.2 0.2
D  0.2 0.8 0.4 0.4
> install.packages("markovchain")
> library(markovchain)
> MC_services<-new("markovchain",states = stateLabels,
 byrow=FALSE, transitionMatrix=M,name="Students connection to each service")
> MC_services
Students connection to each service
 A  4 - dimensional discrete Markov Chain defined by the following states:
 VS, F, E, D
```

```
 The transition matrix  (by cols)  is defined as follows:
    VS   F   E   D
VS 0.5 0.1 0.4 0.3
F  0.1 0.1 0.0 0.1
E  0.2 0.0 0.2 0.2
D  0.2 0.8 0.4 0.4

> install.packages("shape")
> library("shape")
> install.packages("diagram")
> library("diagram")

> plotmat(M, pos=c(1,2,1), lwd=1, box.lwd=1, cex.txt=0.7, box.size=0.09,
 box.type="circle", box.prop=0.75, box.col=0.5, self.shifty=0.01,
 self.shiftx=-0.13, main= "states diagram")

> library(Matrix)
> library(expm)

> X0<-matrix(c(2/10,1/20,1/20,7/10), 4, 1, dimnames = list(stateLabels))
> X0
   [,1]
VS 0.20
F  0.05
E  0.05
D  0.70

> X1<-M %*% X0
> X1
    [,1]
VS 0.335
F  0.095
E  0.190
D  0.380

> MC_services^10
Students connection to each service^10
 A  4 - dimensional discrete Markov Chain defined by the following states:
 VS, F, E, D
 The transition matrix (by cols) is defined as follows:
           VS          F          E          D
VS 0.37755107 0.37755067 0.37755117 0.37755097
F  0.08163265 0.08163266 0.08163265 0.08163265
E  0.18367347 0.18367347 0.18367347 0.18367347
D  0.35714281 0.35714320 0.35714271 0.35714291

> X10<- (M %^% 10) %*% X0
```

```
> X10
          [,1]
VS 0.37755099
F  0.08163265
E  0.18367347
D  0.35714289

> V<-eigen(M)
> V
eigen() decomposition
$values
[1]  1.0000000  0.2000000  0.1414214 -0.1414214

$vectors
          [,1]          [,2]       [,3]        [,4]
[1,] 0.6775602  7.071068e-01 -0.7158897 -0.05504412
[2,] 0.1464995  4.153925e-16 -0.1228273 -0.32082065
[3,] 0.3296239 -3.230830e-16  0.1737040 -0.45370891
[4,] 0.6409353 -7.071068e-01  0.6650130  0.82957368

> vaps <- V$values
> abs(vaps)
[1] 1.0000000 0.2000000 0.1414214 0.1414214

> veps<-V$vectors
> Vep1<-veps[,1]
> Vep1
[1] 0.6775602 0.1464995 0.3296239 0.6409353
> steadystate<-abs(Vep1)
> steadystate
[1] 0.6775602 0.1464995 0.3296239 0.6409353
> nomalizedSteadyState<-steadystate/sum(steadystate)
> nomalizedSteadyState
[1] 0.37755102 0.08163265 0.18367347 0.35714286
```

2. a) The transition diagram represented by the given matrix is:

Figure 4. Algorithm



b) Given an initial state with X_0 = (1,0,0), we do the following in order to compute X_1:

$$X_1 = \begin{array}{c} \begin{array}{ccc} I & P & S \end{array} \\ \begin{pmatrix} 1/2 & 1/10 & 2/3 \\ 1/2 & 1/2 & 0 \\ 0 & 4/10 & 1/3 \end{pmatrix} \end{array} \begin{array}{c} X_0 \\ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \end{array} = \begin{array}{c} X_1 \\ \begin{pmatrix} 1/2 \\ 1/2 \\ 0 \end{pmatrix} \end{array} \begin{array}{c} I \\ P \\ S \end{array} .$$

Therefore, the probability of being at state P for $t = 1$ is 0.5 and the probability of being at state S for $t = 1$ is 0.

c) To compute X_5:

$$X_5 = \begin{pmatrix} 1/2 & 1/10 & 2/3 \\ 1/2 & 1/2 & 0 \\ 0 & 4/10 & 1/3 \end{pmatrix}^5 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.3948 \\ 0.3755 \\ 0.2296 \end{pmatrix} .$$

d) To calculate the steady state, we can either normalize the eigenvector of the eigenvalue 1 or solve the following system (remember that they are equivalent):

$$\begin{pmatrix} 1/2 & 1/10 & 2/3 \\ 1/2 & 1/2 & 0 \\ 0 & 4/10 & 1/3 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} .$$

The result is:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0.3846 \\ 0.3846 \\ 0.2307 \end{pmatrix} .$$

The required R code used to solve this exercise is the following:

```
> M<-matrix(c(1/2,1/2,0,1/10,1/2,4/10,2/3,0,1/3),3,3)
> M
     [,1] [,2]      [,3]
[1,]  0.5  0.1 0.6666667
[2,]  0.5  0.5 0.0000000
[3,]  0.0  0.4 0.3333333
```

```
> X0<-matrix(c(1,0,0),3,1)
> X0
     [,1]
[1,]    1
[2,]    0
[3,]    0
> M %*% X0
     [,1]
[1,]  0.5
[2,]  0.5
[3,]  0.0

> library(Matrix)
> library("expm")
> M5<-M %^% 5
> M5 %*% X0
          [,1]
[1,] 0.3948148
[2,] 0.3755556
[3,] 0.2296296

> V<-eigen(M)
> V
eigen() decomposition
$values
[1] 1.0000000+0.0000000i 0.1666667+0.4149967i 0.1666667-0.4149967i

$vectors
              [,1]                  [,2]                  [,3]
[1,] -0.6509446+0i  0.6215816+0.0000000i  0.6215816+0.0000000i
[2,] -0.6509446+0i -0.3656362-0.4552134i -0.3656362+0.4552134i
[3,] -0.3905667+0i -0.2559453+0.4552134i -0.2559453-0.4552134i

> vaps <- V$values
> abs(vaps)
[1] 1.0000000 0.4472136 0.4472136
> veps<-V$vectors
> veps[,1]
[1] -0.6509446+0i -0.6509446+0i -0.3905667+0i
> steadystate<-abs(veps[,1])
> steadystate
[1] 0.6509446 0.6509446 0.3905667
> nomalizedSteadyState<-steadystate/sum(steadystate)
> nomalizedSteadyState
[1] 0.3846154 0.3846154 0.2307692
```

3. Self-Assessment Problems

1. A corporation has 100 potential users for an internal system at $t = 0$. 20 of these users are connected and 80 are disconnected. Due to the nature of the system, we know that 40% of connected users will log off and the rest will remain connected. We also know that 20% of disconnected users will connect to the system. What is the steady state for this system?

◯ Connected: 20, disconnected: 80.

◯ Connected: 33.33, disconnected: 66.66.

◯ Connected: 60, disconnected: 40.

◯ Connected: 50, disconnected: 50.

2. A worldwide company running software as a service has 100000 users globally. The company has 3 main data centres around the world, namely America (A), Europe (E) and Asia (A_s). The following matrix represents the transition of connections that move from one data centre to another on a daily basis.

$$\begin{array}{c c} & \begin{array}{ccc} A & E & A_s \end{array} \\ \begin{array}{c} A \\ E \\ A_s \end{array} & \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.2 & 0.9 & 0.2 \\ 0 & 0 & 0.7 \end{pmatrix} \end{array} .$$

What are the expected connections to the data centre in Asia in the long term?

◯ Connections to Asia: 70,000.
◯ Connections to Asia: 30,000.
◯ Connections to Asia: 0.
◯ Connections to Asia: 100,000.

3. Two friends, John and Mary, are playing the following game with a dice. If an even number comes out, the same player keeps throwing the dice. If an odd number comes out, it is the other player's turn to play. However, they are using an unbalanced dice with the following probabilities for each number to come out: $P(1) = 1/12, P(2) = 3/12, P(3) = 1/12, P(4) = 3/12, P(5) = 2/12, P(6) = 2/12$. If John starts playing, what is the probability that Mary plays at turn 5?

◯ The probability that Mary plays at turn 5 is: 0.5021.
◯ The probability that Mary plays at turn 5 is: 0.3333.
◯ The probability that Mary plays at turn 5 is: 0.6666.
◯ The probability that Mary plays at turn 5 is: 0.4979.

4. Suppose that there is a forest with two types of trees, type A and type B. The death rate is 1% for type A and 5% for type B every year. 75% of the space left by the dead trees is taken by type B trees and 25% by type A trees.

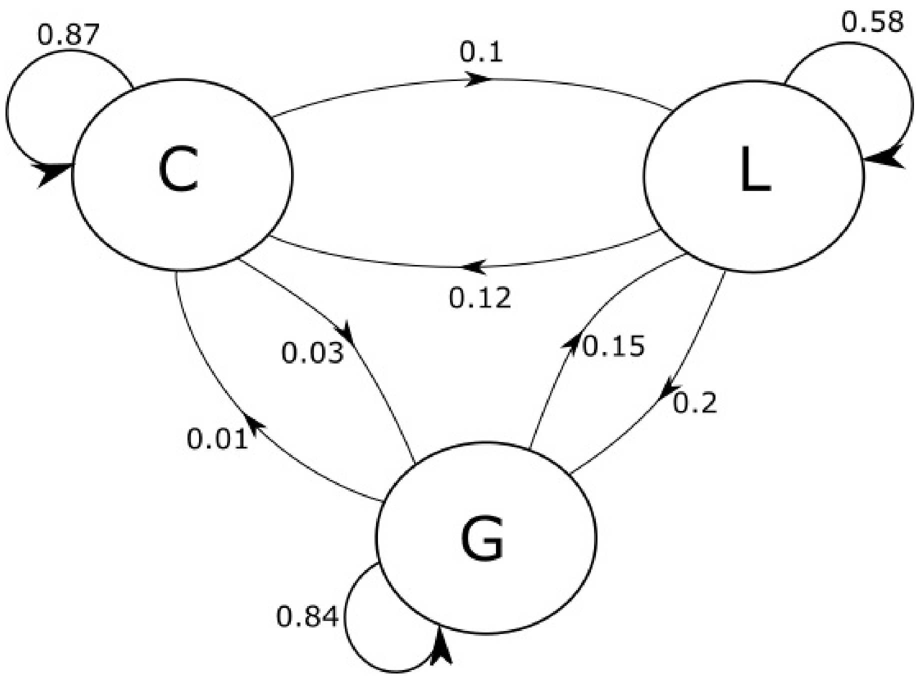
a) What is the expected distribution of trees after 2 years if at t_0 the distribution is 50% for each type of trees?

◯ Type A: 0.0125 . Type B: 0.9875.
◯ Type A: 0.5 . Type B: 0.5.
◯ Type A: 0.5049 . Type B: 0.4951.
◯ Type A: 0.625 . Type B: 0.375.

b) What is the expected distribution of trees in the long term?

◯ Type A: 0.0125 . Type B: 0.9875.
◯ Type A: 0.5 . Type B: 0.5.
◯ Type A: 0.5049 . Type B: 0.4951.
◯ Type A: 0.625 . Type B: 0.375.

5. We are interested in studying the election results in a province with 3 political parties: the Conservative party (C), the Liberal party (L) and the Green party (G). Several studies on election results have repeatedly shown that there is a constant flow of voters across these three parties. This flow can be represented by the following diagram:



What is the expected distribution of voters in the long term?

◯ Conservative: 0.85, Liberal: 0.65, Green: 0.8.
◯ Conservative: 0.3809, Liberal: 0.2619, Green: 0.3571.
◯ Conservative: 1, Liberal: 0.6875, Green: 0.9375.
◯ Conservative: 0.01, Liberal: 0.15, Green: 0.84.

6. The mobile operating system market consists of two main platforms: Android and Apple. A study has shown a constant flow of users across the two platforms. Every year, 13% of Apple users switch to Android and 4% of Android users switch to Apple.

a) Considering that this flow of users remains constant, what should the expected market share be in the long term?

◯ Apple: 0.8700, Android: 0.1300.
◯ Apple: 0, Android: 1.
◯ Apple: 0.2353, Android: 0.07647.
◯ Apple: 1, Android: 3.25.

b) If Android has 30% and Apple 70% in 2019, what was the market share in 2018?

◯ Apple: 0.8700, Android: 0.1300.
◯ Apple: 0.3243, Android: 0.6757.
◯ Apple: 0.3132, Android: 0.6867.
◯ Apple: 0.04, Android: 0.96.

4. Solutions to Self-Assessment Problems

1. The transition matrix for the described system is:

$$\begin{array}{cc} & \begin{array}{cc} C & D \end{array} \\ \begin{array}{c} C \\ D \end{array} & \begin{pmatrix} 0.6 & 0.2 \\ 0.4 & 0.8 \end{pmatrix} \end{array}$$

If we solve the following system:

$$\begin{pmatrix} 0.6 & 0.2 \\ 0.4 & 0.8 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}$$

we get the vector $(x,y) = (1,2)$. By normalizing this vector, we get the steady state distribution (1/3,2/3). Therefore, if we multiply by 100, the result is:

○ Connected: 20, disconnected: 80.
⊗ Connected: 33.33, disconnected: 66.66.
○ Connected: 60, disconnected: 40.
○ Connected: 50, disconnected: 50.

2. The eigenvalues for the given transition matrix

$$\begin{array}{cc} & \begin{array}{ccc} A & E & A_s \end{array} \\ \begin{array}{c} A \\ E \\ A_s \end{array} & \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.2 & 0.9 & 0.2 \\ 0 & 0 & 0.7 \end{pmatrix} \end{array}$$

are 0.7 and 1 with multiplicity 2 and 1, respectively. The eigenvectors are $(1,0,-1)$ and $(0,1,-1)$ corresponding to 0.7, and $(1,2,0)$ corresponding to 1. If we normalize the eigenvector corresponding to 1, we get (1/3,2/3,0) as a steady state distribution. Therefore, the expected number of connections to each data centre is 33333.3, 66666.6, and 0.

○ Connections to Asia: 70,000.
○ Connections to Asia: 30,000.
⊗ Connections to Asia: 0.
○ Connections to Asia: 100,000.

3. In order to build the transition matrix we need to know the probability that a player keeps throwing the dice or that it is the other player's turn.

$$\begin{aligned} P(\text{Change}) &= P(1) + P(3) + P(5) = 1/12 + 1/12 + 2/12 = 1/3 \\ P(\text{Continue}) &= P(2) + P(4) + P(6) = 3/12 + 3/12 + 2/12 = 2/3 \end{aligned} \tag{1}$$

The transition matrix is:

$$\begin{pmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{pmatrix}.$$

If John starts playing, we have to compute X_5:

$$\begin{pmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{pmatrix}^5 \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.5021 \\ 0.4979 \end{pmatrix}.$$

○ The probability that Mary plays at turn 5 is: 0.5021.
○ The probability that Mary plays at turn 5 is: 0.3333.
○ The probability that Mary plays at turn 5 is: 0.6666.
⊗ The probability that Mary plays at turn 5 is: 0.4979.

4. a) We need to find the transition matrix for this forest. We define:

$$\begin{aligned} a_n &= \text{percentage of type } A \text{ trees in the forest at year } n \\ b_n &= \text{percentage of type } B \text{ trees in the forest at year } n. \end{aligned} \tag{2}$$

Now, to compute the number of trees one year later (a_{n+1}) we first subtract the number of dead trees from the trees of the previous generation (a_n) and then we add the proportion from the space left that is taken by each of the species.

$$\begin{aligned} a_{n+1} &= a_n - 0.01\, a_n + 0.25(0.01 a_n + 0.05\, bn) \\ b_{n+1} &= b_n - 0.05\, a_n + 0.75(0.01 a_n + 0.05\, bn) \end{aligned} \tag{3}$$

$$\begin{aligned} a_{n+1} &= (1 - 0.01 + 0.0025)\, a_n + 0.0125 b_n \\ b_{n+1} &= (1 - 0.05 + 0.0375)\, b_n + 0.0075 a_n \end{aligned} \tag{4}$$

$$\begin{aligned} a_{n+1} &= 0.9925\, a_n + 0.0125\, b_n \\ b_{n+1} &= 0.0075\, a_n + 0.9875\, b_n . \end{aligned} \tag{5}$$

This system can be expressed as follows:

$$\begin{pmatrix} a_{n+1} \\ b_{n+1} \end{pmatrix} = \begin{pmatrix} 0.9925 & 0.0125 \\ 0.0075 & 0.9875 \end{pmatrix} \begin{pmatrix} a_n \\ b_n \end{pmatrix}.$$

To compute the evolution of the forest two years on, we will compute $X_2 = T^2 X_0$, where T is the transition matrix and $X_0 = (0.5,0.5)$ the initial state of the system.

$$X_2 = \begin{pmatrix} 0.9925 & 0.0125 \\ 0.0075 & 0.9875 \end{pmatrix}^2 \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} = \begin{pmatrix} 0.5049 \\ 0.4951 \end{pmatrix}$$

○ Type A: 0.0125, Type B: 0.9875.
○ Type A: 0.5, Type B: 0.5.
⊗ Type A: 0.5049, Type B: 0.4951.
○ Type A: 0.625, Type B: 0.375.

b) To compute the distribution in the long term, we can normalize the eigenvector associated to the eigenvalue 1. By solving the system $(A-Id)X = 0$ we get the eigenvector.

$$\left(\begin{pmatrix} 0.9925 & 0.0125 \\ 0.0075 & 0.9875 \end{pmatrix} - \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right) \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

The solution is $(x,y) = (1,0.6)$. By normalizing it, we get $(0.625,0.375)$.

○ Type A: 0.0125, Type B: 0.9875.
○ Type A: 0.5, Type B: 0.5.
○ Type A: 0.5049, Type B: 0.4951.
⊗ Type A: 0.625, Type B: 0.375.

5. Given the transition matrix:

$$\begin{pmatrix} 0.85 & 0.15 & 0.05 \\ 0.1 & 0.65 & 0.15 \\ 0.05 & 0.2 & 0.8 \end{pmatrix}$$

The eigenvector associated to the eigenvalue 1 is (1,0.6875,0.9375). If we normalize it, we get the following steady state distribution: (0.3809,0.2619,0.3571).

○ Conservative: 0.85, Liberal: 0.65, Green: 0.8.
⊗ Conservative: 0.3809, Liberal: 0.2619, Green: 0.3571.
○ Conservative: 1, Liberal: 0.6875, Green: 0.9375.
○ Conservative: 0.01, Liberal: 0.15, Green: 0.84.

6. a) Again, by normalizing the eigenvector with eigenvalue 1 of the transition matrix we get the steady state distribution:

$$\begin{pmatrix} 0.87 & 0.04 \\ 0.13 & 0.96 \end{pmatrix}$$

and the steady state is (0.2353,0.7647).

○ Apple: 0.8700, Android: 0.1300.
○ Apple: 0, Android: 1.
⊗ Apple: 0.2353, Android: 0.7647.
○ Apple: 1, Android: 3.25.

b) If we want to calculate the market share the year before having a market share of (0.3,0.7), we can solve the following system:

$$\begin{pmatrix} 0.87 & 0.04 \\ 0.13 & 0.96 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0.3 \\ 0.7 \end{pmatrix}.$$

The solution is: (0.3132,0.6867).

○ Apple: 0.8700, Android: 0.1300.
○ Apple: 0.3243, Android: 0.6757.
⊗ Apple: 0.3132, Android: 0.6867.
○ Apple: 0.04, Android: 0.96.