Practica 4 2024

Ignacio Fernández Contreras

ifcau3z@uma.es
Planificación de Proyectos y Análisis de Riesgos . E.T.S Informática.

1 Problema 1

Supongamos que una agencia de alquiler de automóviles tiene tres ubicaciones en Ottawa: ubicación en el centro, ubicación al este y ubicación en el extremo oeste. La agencia tiene un grupo de conductores de entrega para atender las tres ubicaciones. Las estadísticas de la empresa han determinado lo siguiente:

1. De las llamadas a la ubicación del centro, el 30% se entrega en el área del centro, el 30% se entrega en el extremo este y el 40% se entrega en el extremo oeste.
2. De las llamadas a la ubicación East End, el 40% se entrega en el área del Downtown, el 40% se entrega en el East End y el 20% se entrega en el West End.
3. De las llamadas a la ubicación del extremo oeste, el 50% se entrega en el área del centro de la ciudad, el 30% se entrega en el extremo este y el 20% se entrega en el extremo oeste.

Después de hacer una entrega, un conductor va al lugar más cercano para hacer la próxima entrega. De esta manera, la ubicación de un controlador específico está determinada solo por su ubicación anterior.

1. Determina la matriz de transición de estados.
2. Crea la matriz de estado en R, usando el paquete markovchain.
3. Dibuja el diagrama de transición de estados.
4. Supongamos que estamos en Downtown, ¿Cuál es la probabilidad de regresar a Downtown después de 2 viajes?
5. Calcula la distribución estacionaria.

1.1 apartado 1

Vamos a utilizar el paquete 'markovchain':

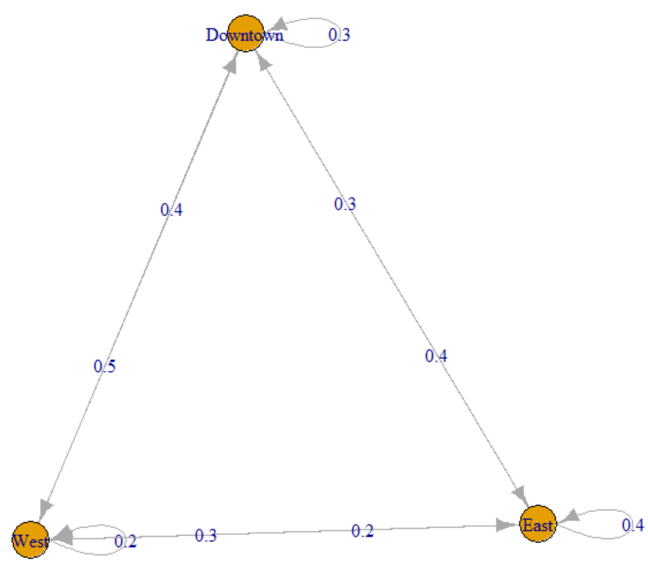
```r
rentalStates <- c("Downtown", "East", "West")
rentalTransition <- matrix(c(0.3, 0.3, 0.4, 0.4, 0.4, 0.2, 0.5, 0.3, 0.2),
                           byrow=T, nrow=3, dimnames=list(rentalStates, rentalStates))
```

1.2 apartado 2

```r
mcRental <- new ("markovchain", states=rentalStates, byrow=T,
                 transitionMatrix=rentalTransition, name="Rental Cars")
```

1.3 apartado 3

```
plot(mcRental)
```



1.4 apartado 4

```
probabilidad_Inicial = matrix(data=c(1, 0, 0), ncol=3)
dias = 2
probabilidad_Final = probabilidad_Inicial %*% (rentalTransition^dias)
```

```
> probabilidad_Final
     Downtown East West
[1,]     0.09 0.09 0.16
>
```

Solución: 0.09

1.5 apartado 5

```
distribucion_Estacionaria <- steadyStates(mcRental)
```

```
> distribucion_Estacionaria
      Downtown      East      West
[1,] 0.3888889 0.3333333 0.2777778
```

2 Problema 2

Los modelos de cadena de Markov fueron introducidos en la literatura médica por Beck y Pauker y proporcionan un medio muy cómodo, y que ha sido utilizado en numerosos análisis posteriores, para dar cuenta de las opciones de un tratamiento médico y los riesgos que se producen no sólo en el presente, sino también en el futuro cercano o lejano. En concreto, en el artículo escrito por Minion et al. se lleva a cabo un análisis del coste-efectividad del tratamiento de quimioterapia frente a la quimioterapia combinado con un medicamento denominado Bevacizumab (B), vendido con el nombre de Avastin4 , en pacientes con cáncer de cuello uterino persistente o metastásico. Dicho análisis se modeliza mediante un proceso de Markov, en el que se establecen cinco posibles estados de salud (Figura 1): respuesta (R), complicaciones limitadas (CL), progreso (P), complicaciones graves (CG) y muerte (M), ver Figura 1. Este es el ejemplo que las autoras van a utilizar para ilustrar cómo implementar un proceso de Markov con recompensa en distintas herramientas software. En dicho modelo un paciente que está en un estado durante un mes, podrá bien permanecer en el mismo estado o bien hacer transición a un estado diferente, con una cierta probabilidad (diferente para cada uno de los dos tratamientos en evaluación), en el siguiente mes (Tabla 1).

	De i a j	Respuesta R	Complicaciones Limitadas CL	Progreso P	Complicaciones Grave CG	Muerte M
Quimio	R	0,8671	0,0024	0,1280	0,0035	0
	CL	1	0	0	0	0
	P	0	0	0,8623	0	0,1377
	CG	0	0	0,9	0,1	0
	M	0	0	0	0	1
Quimio + B	R	0,8720	0,0273	0,0823	0,0184	0
	CL	1	0	0	0	0
	P	0	0	0,8771	0	0,1229
	CG	0	0	0,9	0,1	0
	M	0	0	0	0	1

Para Quimio y Quimio+B,:

1. Determina la matriz de transición de estados.
2. Crea la matriz de estado en R, usando el paquete markovchain.
3. Dibuja el diagrama de transición de estados.
4. Supongamos que estamos en Progresa, ¿Cuál es la probabilidad de Morir después de 3 meses?
5. Calcula la distribución estacionaria

2.1 Apartado 1

Vamos a utilizar el paquete 'markovchain':

```r
mquimio = matrix(c(0.8671, 0.0024, 0.1270, 0.0035, 0, 1, 0, 0, 0, 0, 0, 0, 0.8623,
                   0, 0.1377, 0, 0, 0.9, 0.1, 0, 0, 0, 0, 0, 1),
                 nrow = 5, ncol = 5, byrow = TRUE)
mquimioB = matrix(c(0.8720, 0.0273, 0.0823, 0.0184, 0, 1, 0, 0, 0, 0, 0, 0, 0.8771,
                    0, 0.1229, 0, 0, 0.9, 0.1, 0, 0, 0, 0, 0, 1),
                  nrow = 5, ncol = 5, byrow = TRUE)
```

```
> mquimio
       [,1]   [,2]   [,3]   [,4]   [,5]
[1,] 0.8671 0.0024 0.1270 0.0035 0.0000
[2,] 1.0000 0.0000 0.0000 0.0000 0.0000
[3,] 0.0000 0.0000 0.8623 0.0000 0.1377
[4,] 0.0000 0.0000 0.9000 0.1000 0.0000
[5,] 0.0000 0.0000 0.0000 0.0000 1.0000
```

```
> mquimioB
      [,1]   [,2]   [,3]   [,4]   [,5]
[1,] 0.872 0.0273 0.0823 0.0184 0.0000
[2,] 1.000 0.0000 0.0000 0.0000 0.0000
[3,] 0.000 0.0000 0.8771 0.0000 0.1229
[4,] 0.000 0.0000 0.9000 0.1000 0.0000
[5,] 0.000 0.0000 0.0000 0.0000 1.0000
```

2.2 Apartado 2

```r
quimio <- new ("markovchain", states=c("R", "CL", "P", "CG", "M"), transitionMatrix=mquimio, name="Quimio")
quimioB <- new ("markovchain", states=c("R", "CL", "P", "CG", "M"), transitionMatrix=mquimioB, name="Quimio + B")
```

2.3 Apartado 3

```r
plot(quimio, package="diagram", box.size=0.04)
plot(quimioB, package="diagram", box.size=0.04)
```

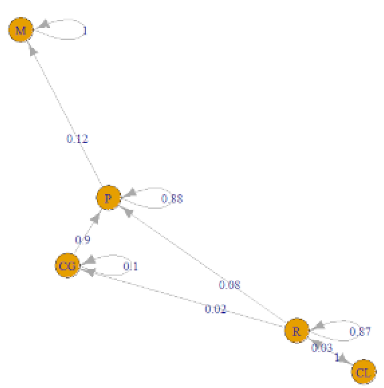


Diagrama de transición de quimio

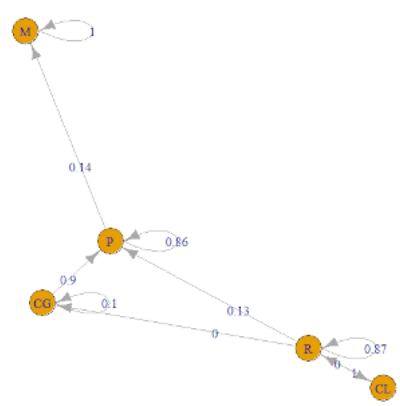


Diagrama de transición de quimioB

2.4 Apartado 4

```
probQ = matrix(c(0, 0, 1, 0, 0), ncol=5)
probQB = matrix(c(0, 0, 1, 0, 0), ncol=5)
meses = 3

# Matrices de transición de 3 meses
quimio3meses = quimio^meses
quimioB_3_meses = quimioB^meses

# Extraer las matrices de transición como objetos numéricos
quimio3meses_matrix = quimio3meses@transitionMatrix
quimioB_3_meses_matrix = quimioB_3_meses@transitionMatrix

# Multiplicación de probabilidad inicial por la matriz de 3 meses
probQ = probQ %*% quimio3meses_matrix
probQB = probQB %*% quimioB_3_meses_matrix
```

```
> probQ
     R CL         P CG         M
[1,] 0  0 0.6411729  0 0.3588271
```

```
> probQB
     R CL         P CG         M
[1,] 0  0 0.6747569  0 0.3252431
```

2.5 Apartado 5

```
distribucionEstacionaria = steadyStates(quimio)
distribucionEstacionariaB = steadyStates(quimioB)
```

```
> distribucionEstacionaria
     R CL P CG M
[1,] 0  0 0  0 1
```

```
> distribucionEstacionariaB
     R CL P CG M
[1,] 0  0 0  0 1
```