

# MODELOS DE CADENAS DE MARKOV EN LA PRÁCTICA: UNA REVISIÓN DE OPCIONES DE SOFTWARE DE BAJO COSTE

Jiaru Bai<sup>1</sup>, Cristina del Campo<sup>\*\*2</sup>, L. Robin Keller<sup>\*3</sup>

\* Paul Merage School of Business, University of California, Irvine, 92697-3125, USA.

\*\* Corresponding author, Facultad de Ciencias Económicas y Empresariales, Campus de Somosaguas,  
Universidad Complutense de Madrid, 28223 Pozuelo de Alarcón (Madrid), SPAIN

## ABSTRACT

Markov processes (or Markov chains) are used for modeling a phenomenon in which changes over time of a random variable comprise a sequence of values in the future, each of which depends only on the immediately preceding state, not on other past states. A Markov process (PM) is completely characterized by specifying the finite set  $S$  of possible states and the stationary probabilities (i.e. time-invariant) of transition between these states. The software most used in medical applications is produced by TreeAge, since it offers many advantages to the user. But, the cost of the TreeAge software is relatively high. Therefore in this article two software alternatives are presented: Sto Tree and the zero cost add-in program "markovchain" implemented in R. An example of a cost-effectiveness analysis of two possible treatments for advanced cervical cancer, previously conducted with the Treeage software, is re-analyzed with these two low cost software packages.

You can find this Spanish version of the original English paper in the following link:  
<http://faculty.sites.uci.edu/lrkeller/publications/>

**KEYWORDS:** cervical cancer treatments, cost-effectiveness analysis, Markov decision trees, stationary transition probabilities

**MSC:** Primary 60J10; Secondary 90B99

## RESUMEN

Los procesos de Markov (o cadenas de Markov) se utilizan para el modelización de un fenómeno en el que el cambio con el tiempo de una variable aleatoria comprenden una secuencia de valores en el futuro, cada uno de los cuales depende únicamente del estado inmediatamente anterior, no de otros estados pasados. Un proceso de Markov (PM) se caracteriza por completo mediante la especificación del conjunto  $S$  finito de estados posibles y las probabilidades estacionarias (es decir, invariante en el tiempo) de transición entre esos estados. El software más utilizado en aplicaciones médicas es producido por TreeAge, ya que ofrece muchas ventajas para el usuario. Pero, el coste del software TreeAge es relativamente alto. Por lo tanto, en este artículo se presentan dos alternativas de software: Sto Tree y el paquete, gratuito, "markovchain" implementado en R. Un ejemplo de un análisis de coste-efectividad de dos tratamientos posibles para el cáncer de cuello uterino avanzado, llevado a cabo previamente con el software TreeAge, se volvió a analizar con estos dos paquetes de software de bajo coste.

Puede encontrar esta versión en español del artículo original en inglés en el siguiente enlace:  
<http://faculty.sites.uci.edu/lrkeller/publications/>

**PALABRAS CLAVE:** tratamientos del cáncer de cuello uterino, análisis coste-efectividad, árboles de decisión de Markov, probabilidades de transición estacionarias

## 1. INTRODUCCIÓN

Los procesos o cadenas de Markov son herramientas bien conocidas para la modelización de una amplia gama de fenómenos en los que los cambios en el tiempo de una variable aleatoria conforman una secuencia de valores temporales, en los que además su situación futura depende únicamente de su estado actual, no de sus estados pasados. Un proceso de Markov (PM) queda totalmente caracterizado cuando se determina el conjunto  $S$  (finito) de sus posibles estados y las probabilidades estacionarias (i.e. invariantes en el tiempo) de

---

<sup>1</sup> JiaruB@uci.edu

<sup>2</sup> campocc@ccee.ucm.es

<sup>3</sup> LRKeller@uci.edu

transición entre dichos estados ( $P_{ss'} = P(S_{t+1} = s'/S_t = s)$ ). Los procesos de Markov tienen aplicaciones en el modelado y análisis de una amplia gama de los fenómenos en todas las ramas de la ciencia, que van desde la lingüística o la biología, hasta las ciencias políticas, la medicina, la economía, la informática, y muchas otras.

Ahora bien, si a cada cambio de estado se le asigna una cantidad numérica tendremos un modelo de Markov con recompensa. Y así surgieron durante los años sesenta y setenta, los procesos de Markov con recompensa (PMR), principalmente por su aplicación en campos relacionados con la ingeniería (Howard [6]), mientras que en otros campos como la economía hubo que esperar un poco más (ver, por ejemplo Norberg [9]), donde se presenta una aplicación de los PMR en seguros de vida). Un PMR es simplemente un proceso de Markov en el que en cada estado se recibe una recompensa distinta ( $R_s = E[R_{t+1}/S_t = s]$ ). Si bien la definición más general también incluye un posible descuento ( $\gamma \in [0,1]$ ), puesto que con ello se evita un retorno infinito en los procesos de Markov cíclicos así como posibilita que, si la recompensa es de carácter financiero, las recompensas inmediatas tengan más interés que las recompensas retrasadas o futuras. Si el modelo no tiene descuento simplemente  $\gamma = 1$ .

Sin embargo, los procesos de Markov, con o sin recompensa, no se pueden utilizar para representar problemas en los que en cada uno de los estados del proceso markoviano haya un gran número de decisiones incrustadas, hecho que ocurre a menudo en situaciones que requieren toma de decisiones secuenciales donde un decisor puede elegir entre varias opciones, cada una de ellas con una posible recompensa y una probabilidad de transición diferentes. Dicho problema se modeliza como un proceso de decisión de Markov (PDM), que se define como un PMR en el que además del conjunto finito de estados,  $S$ , existe un conjunto (finito) de alternativas o acciones ( $A$ ) y cuyas probabilidades de transición dependerían también de la acción considerada  $P_{ss',a} = P(S_{t+1} = s'/S_t = s, A = a)$ . Evidentemente la función de recompensa también dependería de dicha alternativa considerada  $R_{s,a} = E[R_{t+1}/S_t = s, A = a]$ .

Para un estudio más en profundidad de estos métodos tan potentes y sistemáticos de análisis de sistemas estocásticos como son los procesos de Markov ver, por ejemplo, Gouberman y Siegle [3], Rubio-Terrés [10] o Soto Álvarez [11].

El software más utilizado en aplicaciones médicas es TreeAge, puesto que ofrece muchas ventajas al usuario como una interfaz gráfica amigable, modelización de procesos de Markov, gráficos e informes coste-efectividad, la posibilidad de realizar análisis de sensibilidad gráficamente, simulación Monte-Carlo integrada y otras características. Para una descripción detallada del software, TreeAge Pro Healthcare, consulte la documentación así como la versión de prueba en [www.treeage.com](http://www.treeage.com).

Pero el problema que presenta TreeAge es fundamentalmente su precio, y la necesidad de renovar la licencia todos los años (si la licencia es anual) o pagar la cuota de mantenimiento (si la licencia es estándar), lo que, en el medio o largo plazo, termina produciendo una cantidad que puede resultar excesiva. Por ello en este artículo se presentan dos alternativas bien de coste bajo, Sto Tree, bien gratuito, paquete “markovchain” sobre R, y se aplican para comparar los resultados del estudio de coste-efectividad de dos tratamientos para el cáncer de cuello uterino.

## 2. ANTECEDENTES

Las tecnologías de la información y de apoyo a las decisiones han evolucionado rápidamente en las últimas décadas. Miles de millones de dólares invertidos en la investigación de la tecnología han estimulado mejoras en la potencia de procesamiento informático, hardware, software, redes y telecomunicaciones. La rápida evolución de estas tecnologías, ha conseguido que la mayoría de los usuarios tengamos ahora ordenadores personales, bien en el trabajo, bien en casa o en ambos, y una alta proporción de ellos utilizamos paquetes de aplicaciones integradas, como, por ejemplo, Microsoft Office para Windows, LibreOffice u OpenOffice para Linux o la suite propietaria de Mac, en nuestro día a día.

Por supuesto, la idea de revisar los programas software que se utilizan como apoyo a la toma de decisiones no es nueva, pero según lo que las autoras han encontrado en su revisión de la literatura, Menn y Holle [7] han publicado, hasta el momento, la única revisión existente de software de ayuda a la toma de decisiones sanitarias basada en modelos de Markov que implementan y evalúan los costes-efectividad de distintos tratamientos médicos. Hay algunas otras evaluaciones, como por ejemplo Bhargava, Sridhar y Herrick [1], pero se refieren a Sistemas de ayuda a la decisión en general, sin centrarse únicamente en modelos de Markov. Sin embargo dicha revisión se basa fundamentalmente en la comparación de estos programas con respecto a la facilidad de uso e implementación de un modelo, el tiempo de ejecución, las características de la

presentación de resultados, y la flexibilidad, mientras que en el presente artículo las autoras están interesadas, en tres aspectos:

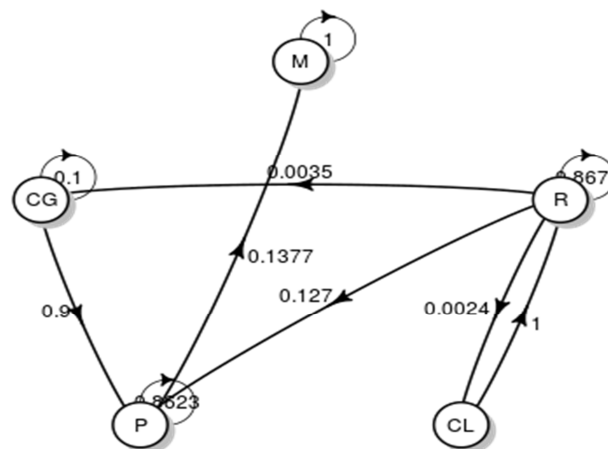
- Facilidad para aprender a utilizar el programa - muchos de los sistemas disponibles son demasiado complejos para que los usuarios los conozcan y los puedan utilizar con éxito dentro del poco tiempo que se suele tener disponible.
- Capacidad para examinar la lógica del sistema - Un usuario que utiliza un programa desea que sea transparente en su funcionamiento. Es difícil entender lo que está sucediendo a menos que se pueda comprender la lógica de las operaciones realizadas por el sistema.
- Coste - con un presupuesto disponible siempre limitado a la hora de invertirlo en un programa software de ayuda a la decisión se necesita un sistema libre, o al menos de coste reducido.

No parece haber una gran gama de sistemas de ayuda a la decisión médica y sanitaria propuestos en la abundante literatura puesto que la gran mayoría de los artículos revisados por las autoras del presente artículo utilizan TreeAge. Ello hace todavía más necesaria la revisión de las distintas opciones ahora mismo disponibles, dos de las cuales se presentan en este artículo, con la característica añadida de su bajo o nulo coste.

### 3. EL EJEMPLO

Los modelos de cadena de Markov fueron introducidos en la literatura médica por Beck y Pauker [2] 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. [8] 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 Avastin<sup>4</sup>, en pacientes con cáncer de cuello uterino persistente o metastásico. Dicho análisis se modeliza mediante un proceso de Markov con recompensa, 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.

Figura 1. Diagrama de transiciones de la cadena de Markov para quimioterapia (Creado con R)



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). 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 comunicación privada, Minion et al. señalaron errores tipográficos en su tabla 2 publicada [8] originalmente: En la tabla de probabilidad de las transiciones, la probabilidad de pasar de "Complicaciones Graves" a "Progreso" en un mes es 0,9 (no 0), y la probabilidad de pasar de "Complicaciones Graves" a "Muerte" es 0 (no 0.9), para ambas opciones de tratamiento. En el presente artículo se utilizan las probabilidades corregidas.

En un modelo de Markov, el paciente asocia una recompensa o utilidad a cada estado que representa la efectividad del tratamiento que depende del estado de salud en el que se encuentre el paciente durante cada mes. La efectividad total para cada paciente es la suma de esas utilidades durante todo el tratamiento. Dichas

<sup>4</sup> Avastin se administra como una infusión por vía intravenosa con el mismo ciclo que la quimioterapia.

utilidades, en el artículo de Minion et al. [8], son 1 para el estado Respuesta (R), indicando que el paciente ha vivido un mes en el estado de Respuesta al tratamiento de cáncer de cuello uterino, 0,75 para el estado Complicaciones Limitadas (CL) y 0,5 para los estados Progreso (P) y Complicaciones Graves (CG), ya que la calidad de vida se ajusta a la baja para ese mes, y finalmente 0 para el estado Muerte.

Tabla 1. Probabilidades de Transición del Estado i al Estado j en un Ciclo

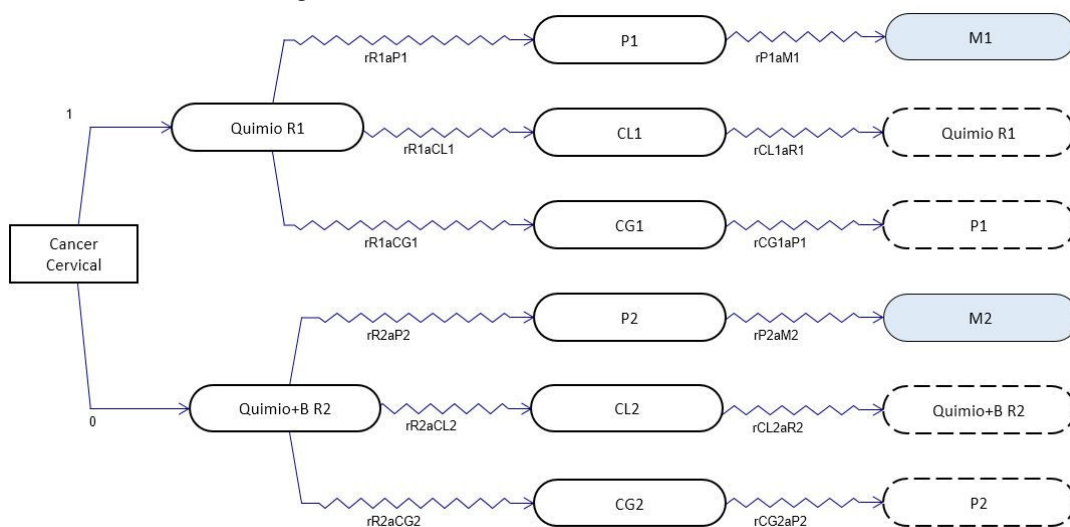
	De i a j	Respuesta R	Complicaciones Limitadas CL	Progreso P	Complicaciones Graves CG	Muerte M
Quimio	R	0,8671	0,0024	0,1270	0,0035	0
	CL	1	0	0	0	0
	P	0	0	0,8623	0	0,1377
	CG	0	0	0,9	0,1	0
	M	0	0	0	0	1
Quimio + B	R	0,8720	0,0273	0,0823	0,0184	0
	CL	1	0	0	0	0
	P	0	0	0,8771	0	0,1229
	CG	0	0	0,9	0,1	0
	M	0	0	0	0	1

#### 4. EL SOFTWARE STO TREE

Otra opción que permite formular un modelo de Markov es mediante el uso del software Sto Tree (Hazen [5], <http://users.iems.northwestern.edu/~hazen/StoTreeOv.html>), una macro que se incrusta en Microsoft Excel, por lo que otras características de Excel también están disponibles al mismo tiempo. Y tiene además otra ventaja competitiva frente a TreeAge, y es que el software Sto Tree sólo cuesta 20\$.

Respecto al uso concreto del software (consultar su manual de uso en [13]), lo primero que se debe reseñar es que Sto Tree no genera un diagrama de transición de Markov (como el de la Figura 1), pero, en cambio, utiliza un árbol de Markov similar al que utiliza TreeAge (Figura 2).

Figura 2. Árbol de Decisión de Markov en Sto Tree



La construcción de un modelo de Markov en Sto Tree es de un estilo más libre. No existe una formulación estándar en Sto Tree y los nodos y arcos se pueden arrastrar a cualquier lugar de la hoja de cálculo,

a gusto del usuario. Es fácil construir un modelo de Markov en el software, simplemente añadiendo nodos y arcos, no se necesita ningún trabajo de codificación. Lo único que hay que tener en cuenta es que para Sto Tree hay dos tipos de arcos: la flecha ondulada representa una transición con una determinada velocidad; mientras la flecha recta muestra una transición instantánea. El software está más adaptado a modelos de Markov continuos, y es algo menos eficiente en modelos de Markov discretos, aunque cualquier probabilidad  $p$  se puede convertir a una tasa  $r$  mediante el uso de la fórmula  $p = 1 - e^{-r\Delta s}$ , donde  $\Delta s$  es la duración del ciclo. El software no considera tiempo de duración para esas transiciones con la flecha recta, ya que son vistas como algo que ocurre instantáneamente.

Siempre que haya una transición a los estados pre-existentes anteriores, se tiene que seleccionar un nodo duplicado para representar dicha transición. Por ejemplo, del estado “Complicaciones limitadas” (CL) se puede volver al estado “Respuesta” (R), por lo que un nodo duplicado R (con el círculo punteado en la figura 2) se crea después de CL. A diferencia del software TreeAge, en el que el nodo terminal debe ser uno de los estados de Markov que aparecen en ese árbol de Markov particular, el software Sto Tree puede seleccionar cualquier estado que se haya creado anteriormente.

Los parámetros se pueden definir fácilmente usando el nombre de la función definido en Excel, lo que permite cambiar fácilmente los valores de las variables en las celdas con nombre, y permite también hacer análisis de sensibilidad manualmente de esta manera. Un análisis de sensibilidad automático se encuentra disponible en Sto Tree, permitiendo, además, el análisis de sensibilidad probabilístico.

Figura 3. Hoja de cálculo de Sto Tree para el cálculo de los costes (Costs) y años de vida ajustados por calidad (QALY, en inglés) durante 60 Ciclos de Tratamiento Mensual

Cycle Time =	1	Cancer Cervical:	Cancer Cervical	Quimio R1	P1	M1	CL1	CG1	Quimio+B R2	P2	M2	CL2	CG2		
Starting Time =	0	Chance/Decision?	1	0	0	0	0	0	0	0	0	0	0	0	
Time Horizon =	60	Initial Cohort =	1												
Discount Rate =	0														
Calculate Sheet															
Reset to Start Time		Time Counts													
	0	0	1	0	0	0	0	0	0	0	0	0	0	0	FALSE
Step ahead one cycle	1	0	0.8671	0.127	0	0.0024	0.0035								
		Quality Coefficients:	1	0.5	0	0.75	0.5		1	0.5	0	0.75	0.5		
		Cost Coefficients:		524	262	0	809	4157	7540	262	0	7825	4331		
		Incremental QALY Tolls:	0	0	0	0	0	0	0	0	0	0	0	0	
		Incremental QALYs:	0	0.93355	0.03175	0	0.0009	0.000875	0	0	0	0	0	0	
		Incremental Total:	0	0.93355	0.03175	0	0.0009	0.000875	0	0	0	0	0	0	
		Cum Discounted QALYs:	0	0.93355	0.03175	0	0.0009	0.000875	0	0	0	0	0	0	
		Overall Cum QALYs =	0.967												
		Incremental Cost Tolls:	0	0	0	0	0	0	0	0	0	0	0	0	
		Incremental Costs:	0	489.1802	16.637	0	0.9708	7.27475	0	0	0	0	0	0	
		Incremental Total:	0	489.1802	16.637	0	0.9708	7.27475	0	0	0	0	0	0	
		Cum Discounted Costs:	0	489.1802	16.637	0	0.9708	7.27475	0	0	0	0	0	0	
		Overall Cum Costs =	514												

La celda de la última columna es un valor lógico que mostraría “True” (verdadero) si esta fila tiene todas las sumas en el nodo de decisión o de azar inicial. En nuestro caso (Figura 3), el nodo de azar / decisión es "Cáncer de cuello uterino", pero en el momento 0 todos los cargos son en el estado “Quimio R1”, ya que la flecha recta emplea 0 tiempo. Así que la celda muestra el valor “False” (falso).

El software Sto Tree no permite hacer optimización de estos modelos de Markov, lo que significa que trata los nodos de decisión de igual manera que los nodos de azar. Se puede, eso sí, indicar la decisión elegida adjudicando la probabilidad de la rama como 1 y las demás como 0. Por ejemplo, en la Figura 2, el paciente puede elegir entre dos tipos de tratamiento: la quimioterapia solamente y quimioterapia más Bevacizumab. Asignamos la primera opción con probabilidad 1 y la segunda con 0, lo que significa que la rama con sólo quimioterapia está activa ahora y eso permite obtener los resultados para esa rama. Y lo mismo se podría hacer con la opción quimioterapia más Bevacizumab. Sin embargo, el software no compara el resultado de las dos opciones y por tanto tampoco indica cuál es la alternativa óptima, eso queda para el usuario.

Finalmente, se puede realizar un análisis por cohorte una vez que el árbol estocástico está completo, como se muestra en la Figura 3. Sólo hay que especificar el tiempo de ciclo y el horizonte de tiempo, y, en su caso, la tasa de descuento, puesto que el software también acepta tasa de descuento como en cualquier PMR. Los años de vida ajustados por calidad (AVAC o QALY, en inglés) y los costes se pueden calcular una vez que se han introducido los parámetros del modelo en la tabla. En la figura, cada ciclo es un mes, y por eso, el cálculo es de los meses de vida ajustados por calidad (MVAC). La hoja es sencilla y su implementación no presenta dificultades.

El usuario puede hacer un clic en "step ahead one cycle" (avanzar un ciclo) para comprobar los cálculos en cada etapa, o puede utilizar "calculate sheet" (textualmente, calcular la hoja) para obtener los resultados globales para el horizonte de tiempo especificado. Sin embargo, el software no proporciona una salida detallada del análisis de la cohorte, sino que el usuario tiene que registrar el resultado de cada paso y resumir los resultados incrementales y acumulativos en consecuencia.

Y como último detalle, una versión sin licencia permite crear los árboles estocásticos, pero no se puede realizar ninguna de las operaciones computacionales. Sin embargo, si un árbol estocástico se ha creado en una versión con licencia, dicho modelo puede ser compartido y utilizado en una versión sin licencia.

## 5. R: LA OPCIÓN GRATUITA A CONSIDERAR

Las tres autoras de este artículo son, todas ellas, profesoras en una facultad de Administración y Dirección de Empresas. En concreto, algunas de ellas dan clase a alumnos del grado de Administración y Dirección de Empresas y/o de Economía enseñando a dichos alumnos una amplia gama de técnicas estadísticas, tanto básicas como avanzadas, y el paquete estadístico que utilizan con más frecuencia es R. Las ventajas de dicho programa es que incluye todas las técnicas que se necesitan, es de código abierto y disponible sin coste para todas las plataformas (Windows, Linux o Mac), es fácil de instalar y utilizar, e incluso se puede ejecutar desde una unidad de CD o un USB.

Para aquellos que no han oído hablar de R o aquellos que sí han oído hablar pero se sienten atemorizados ante la idea de tener que utilizarlo escribiendo código, les recomendamos que se lo instalen (<https://www.r-project.org/>) y se bajen algunos de los manuales que existen en español (por ejemplo <https://cran.r-project.org/doc/contrib/R-intro-1.1.0-espanol.1.pdf>). Pero sobre todo les recomendamos que se instalen una de las interfaces gráficas de usuario especialmente diseñadas para trabajar con R y que lo que hacen es facilitar al usuario el acceso a una selección de técnicas de análisis de uso muy común utilizando unos sencillos menús (equivalentes, por ejemplo, a los que tiene SPSS). Las autoras les recomiendan el R-commander (comúnmente conocido como Rcmdr) por ser, en su opinión, la más utilizada y sobre la que más documentación podrán encontrar. Consulte, por ejemplo, la página web <http://www.uv.es/innovamide/14u/R/R0/R0.wiki>, en español, sobre cómo instalar R sobre Windows y cómo instalar Rcmdr y ver, por ejemplo, Guisande González, Vaamonde Liste y Barreiro Felpeo [4] sobre cómo usarlo para diversos análisis.

Volviendo ahora al problema que nos ocupa de los procesos de Markov. En este caso el paquete de R que se va a utilizar (markovchain) no está incorporado todavía a Rcmdr, pero a continuación explicaremos en detalle cómo usarlo (consultar también Spedicato, Kang y Yalamanchi [12]) y facilitaremos, en primer lugar, el código para poder repetir el análisis realizado en este artículo y, en segundo lugar, poder adaptarlo a cualquier problema de similares características que el lector pueda enfrentar. En el código que aparece debajo en el cuadro gris, el signo de sostenido (#) indica que el texto que aparece a continuación es un comentario. Las líneas que no tienen dicho signo contienen las sentencias que realmente se deben ejecutar.

El código que aparece a continuación calcula los resultados para el tratamiento únicamente con la quimioterapia. Para obtener los resultados del tratamiento de la quimioterapia con Bevacizumab, únicamente habría que repetir el código sustituyendo el nombre "quimio" por "quimio+B", y además sustituyendo las probabilidades de transición y los costes por los correspondientes a esta segunda opción.

```
# Instala la librería.
# Una vez instalada en un ordenador no se vuelve a ejecutar la sentencia
install.packages("markovchain")

# Carga la librería para poder utilizarla en la sesión actual.
# Esta sentencia hay que ejecutarla al principio de cada sesión
library(markovchain, pos=4)

# Crea la cadena de markov, utilizando el paquete "markovchain"
# Se especifican a continuación los nombres de los estados de la cadena (entre comillas)
# Después se escriben de manera continua las probabilidades de transición, por filas (byrow=TRUE)
# Se precisa la dimensión de la matriz (nrow = 5)
# Y por último, optativo, el nombre que se le quiera dar a la cadena de Markov
quimio <- new("markovchain", states = c("R", "CL", "P", "CG", "M"), transitionMatrix = matrix(data =
+ c(0.8671, 0.0024, 0.1270, 0.0035, 0, 1, 0, 0, 0, 0, 0, 0.8623, 0, 0.1377, 0, 0, 0.9, 0.1, 0, 0, 0, 0, 1), byrow
```

```

+= TRUE, nrow = 5), name = "Quimio")
# Verificamos que la matriz de transición es correcta, ya que la imprime en formato matriz
quimio

# Dibuja el diagrama de markov
plot(quimio)

# Dibuja otro diagrama de markov más bonito en blanco y negro (en concreto el de la Figura 1)
# Para ello necesita instalar otra librería. Una vez instalada en un ordenador no se vuelve a ejecutar la sentencia
install.packages("diagram")
# Se carga la librería, para poder utilizarla en esta sesión
library(diagram, pos=4)
# Y ahora se le pide a R que haga el gráfico
plot(quimio, package="diagram", box.size = 0.04)

# Guarda el gráfico en formato png
dev.copy(png,'C:/investigacion_operacional/figura_1.png')
dev.off()

# Cálculo de la matriz fundamental
Q_quimio = matrix(data = c(0.8671, 0.0024, 0.1270, 0.0035, 1, 0, 0, 0, 0, 0, 0.8623, 0, 0, 0, 0.9, 0.1), byrow =
+ TRUE, nrow = 4)
I4 = diag(4)
F_quimio = solve(I4-Q_quimio)
F_quimio
# Tiempo esperado para la absorción, considerando que se empieza en el estado respuesta
t_quimio = F_quimio%%c(1,0,0,0)
cat("Tiempo esperado de vida=")+ print(t_quimio)

# Cálculo del coste del tratamiento, como lo hace TreeAge
# Los costes son los mensuales en cada estado de la cadena
coste = cbind(c(524,262,809,4157,0))
utilidad = c(1,0.75,0.5,0.5,0)
x <- c(1,0,0,0,0)
coste0 = x%%coste
Cost=0
for(i in seq(1:60)) {
x = x*quimio
Rewardeff = utilidad*x
Rewardcost = coste0%%Rewardeff
Cost = sum(Rewardcost) + Cost
}
cat("Coste del tratamiento =")+ print(Cost)

```

## 6. CONCLUSIÓN

Sto Tree y el paquete markovchain para R son buenas opciones de software frente al coste relativamente elevado del software TreeAge para el análisis coste-efectividad de un árbol de decisión de Markov.

RECIBIDO: Marzo 2016.

## REFERENCIAS

- [1] BHARGAVA, H. K., SRIDHAR, S., y HERRICK, C. (1999): Beyond spreadsheets: Tools for building decision support systems. **IEEE Computer**, 3 (3), 31-39.
- [2] BECK, J.R., y PAUKER, S.G. (1983): The Markov process in medical prognosis. **Medical Decision Making**, 3(4), 419-58.
- [3] GOUBERMAN, A., y SIEGLE, M. (2014): Markov reward models and Markov decision processes in

- discrete and continuous time: Performance evaluation and optimization. En: Remke, A. y Stoelinga, M. (Eds.) **Stochastic Model Checking. Rigorous Dependability Analysis Using Model Checking Techniques for Stochastic Systems**, 156-241, Springer, Berlin Heidelberg.
- [4] GUISANDE GONZÁLEZ, C., VAAMONDE LISTE, A., y BARREIRO FELPETO, A. (2011): **Tratamiento de datos con R, Estadística y SPSS**, Ediciones Díaz de Santos, Madrid.
- [5] HAZEN, G.B. (2002): Stochastic trees and the StoTree modeling environment: Models and software for medical decision analysis, **Journal of Medical Systems**, 26(5), 399-413.
- [6] HOWARD, R. A. (1971): **Dynamic Probabilistic Systems**, Vol. 1-2, Wiley, New York.
- [7] MENN, P., y HOLLE, R. (2009): Comparing three software tools for implementing Markov models for health economic evaluations, **Pharmaeconomics**, 27(9), 745-753.
- [8] MINION, L. E., BAI, J., MONK, B. J., KELLER, L. R., ESKANDER, R. N., FORDE, G. K., CHAN, J.K., y TEWARI, K. S. (2015): A Markov model to evaluate cost-effectiveness of antiangiogenesis therapy using bevacizumab in advanced cervical cancer, **Gynecologic oncology**, 137(3), 490-496.
- [9] NORBERG, R. (1995): "Differential Equations for Moments of Present Values in Life Insurance," **Insurance: Mathematics and Economics**, 17(2), 171-180.
- [10] RUBIO-TERRÉS, C. (2000): Introducción a la utilización de los modelos de Markov en el análisis farmaeconómico. **Revista Farmacia Hospitalaria**, 24(4), 241-247.
- [11] SOTO ÁLVAREZ, J. (2012): **Evaluación económica de medicamentos y tecnologías sanitarias: Principios, métodos y aplicaciones en política sanitaria**, Springer Healthcare Iberica, Madrid.
- [12] SPEDICATO, G.A., KANG, T.S., y YALAMANCHI, S.B. (2015): The markovchain Package: A Package for Easily Handling Discrete Markov Chains in R, Disponible en [https://cran.r-project.org/web/packages/markovchain/vignettes/an\\_introduction\\_to\\_markovchain\\_package.pdf](https://cran.r-project.org/web/packages/markovchain/vignettes/an_introduction_to_markovchain_package.pdf), The Comprehensive R Archive Network (CRAN), Consultado 15-3, 2016.
- [13] **StoTree v.7 manual** (2015), Disponible en [http://users.iems.northwestern.edu/~hazen/Documentation/StoTree\\_7\\_3.pdf](http://users.iems.northwestern.edu/~hazen/Documentation/StoTree_7_3.pdf), Consultado 15-3, 2016.