

# Cadenas de Markov

Armando Ocampo

## Librerías de trabajo

Antes de iniciar vamos a llamar a las siguiente librerías. Para su instalación necesitamos la función *install.packages()* y el nombre de la paquetería entre comillas. Por ejemplo, *install.packages('markovchain')*.

```
library(markovchain)
library(moments)
library(clipr)
library(readr)
```

## Dudas de la clase previa

### Copiando resultados desde R a Excel

A continuación se comparten algunos ejemplos de cómo se pueden copiar los datos de los objetos en R y pegar la información en un libro de excel. El primer ejemplo copia la información en el pisa papeles de la máquina (o clipboard) y permite su pegado posterior. Para ello se utiliza la función *write\_clip()* de la paquetería *clipr*. En esta función solo se coloca el nombre del objeto que se desea copiar.

```
vacunas_df <- read_csv('../data/dataset_vacunas.csv')
# write_clip(vacunas_df)
```

Después de correr este comando abrimos un libro en excel y pegamos los datos.

Otra manera de realizar este proceso es mediante la función *write.table()* de la paquetería *utils*. Además del objeto, se agrega el destino de la información, en este caso de nuevo al *clipboard* del sistema. Se añade la separación por tabulador y se elimina el nombre de las filas para no redundar esta información.

```
# write.table(vacunas_df, "clipboard",
#             sep="\t", row.names=FALSE)
```

Al igual que en el código previo, abrimos una hoja de Excel y pegamos la información.

### Transformación de los datos

Durante el proceso de minería de datos y desarrollo de modelos podemos encontrar información que no sigue una distribución normal o con valores extremos. En este caso se sugiere realizar una transformación de los datos. No obstante, queda la duda sobre el tipo de transformación a realizar. La función *skewness()* de la paquetería *moments* calcula el sesgo de la información y facilita este proceso. En ella se coloca el vector del cual se desea conocer el tipo de sesgo que presenta. De la misma manera, es válido realizar la transformación de la información al número de variables que sea necesario, previo al desarrollo de algoritmos o análisis de datos.

```
# Sesgo de PIB
skewness(vacunas_df$GDP)
```

```
## [1] 1.731123
```

```
# Sesgo de esperanza de vida
skewness(vacunas_df$Life_expectancy)
```

```
## [1] -0.5082224
```

```
# Sesgo de longitud de sépalo
skewness(iris$Sepal.Length)
```

```
## [1] 0.3117531
```

```
# Sesgo de ancho de pétalo
skewness(iris$Petal.Width)
```

```
## [1] -0.1019342
```

Nota: si el sesgo es positivo se sugiere realizar una transformación cuadrática, logarítmica o inversa ( $1/x$ ). Por el contrario, para sesgos negativos se recomiendan las siguientes transformaciones:  $\sqrt{\max(x+1) - x}$ ,  $\log_{10}(\max(x+1) - x)$ ,  $1/(\max(x+1) - x)$

## Cadenas de Markov

Las cadenas de Markov permiten conocer la probabilidad de que suceda un evento dado la aparición de un evento previo. Para esto se necesitan las siguientes características:

1. Existe un número finito de estados.
2. La probabilidad de ocurrencia de un evento depende del suceso inmediatamente anterior.
3. Las probabilidades permanecen constantes en el tiempo.

Para ello, se define como un estado a los posibles eventos que puedan ocurrir en un experimento. Cabe destacar que debe conocerse la probabilidad de ocurrencia de cada evento.

```
consumo = matrix(c(0.2,0.3,0.5,0.4,0.2,0.4,0.1,0.6,0.3),
                 nrow = 3,byrow = TRUE)
consumo
```

```
##      [,1] [,2] [,3]
## [1,]  0.2  0.3  0.5
## [2,]  0.4  0.2  0.4
## [3,]  0.1  0.6  0.3
```

```
mc <- new('markovchain', transitionMatrix = consumo,
         states = c('frutas', 'verduras', 'carne'),
         name = 'Cadena1')
```

```
str(mc) # lista de argumentos
```

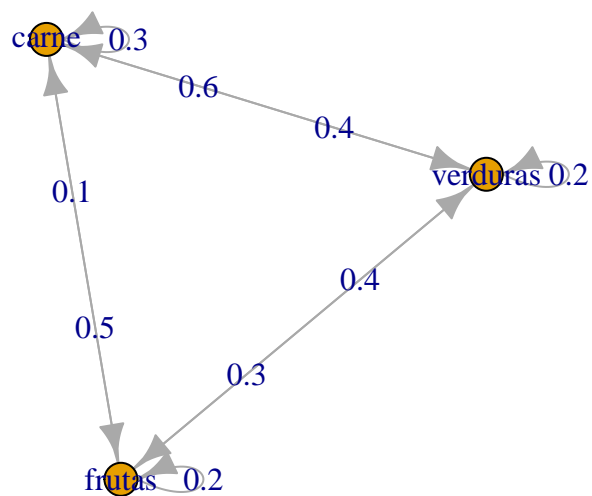
```
## Formal class 'markovchain' [package "markovchain"] with 4 slots
##   ..@ states      : chr [1:3] "frutas" "verduras" "carne"
##   ..@ byrow       : logi TRUE
##   ..@ transitionMatrix: num [1:3, 1:3] 0.2 0.4 0.1 0.3 0.2 0.6 0.5 0.4 0.3
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. $ : chr [1:3] "frutas" "verduras" "carne"
##   .. .. $ : chr [1:3] "frutas" "verduras" "carne"
##   ..@ name        : chr "Cadena1"
```

```
summary(mc)
```

```
## Cadena1 Markov chain that is composed by:
## Closed classes:
```

```
## frutas verduras carne
## Recurrent classes:
## {frutas,verduras,carne}
## Transient classes:
## NONE
## The Markov chain is irreducible
## The absorbing states are: NONE

set.seed(123)
plot(mc)
```



```
recurrentClasses(mc)

## [[1]]
## [1] "frutas" "verduras" "carne"

transitionProbability(object = mc, t0 = 'frutas',
                      t1 = 'carne')

## [1] 0.5
mc[1,3]

## [1] 0.5
n <- 3
mc ^ n

## Cadena1^3
## A 3 - dimensional discrete Markov Chain defined by the following states:
```

```

## frutas, verduras, carne
## The transition matrix (by rows) is defined as follows:
##      frutas verduras carne
## frutas  0.247    0.369 0.384
## verduras 0.240    0.380 0.380
## carne    0.228    0.381 0.391

X0 = c(200,300,500)
n = 6
Xn = X0*(mc^n)
Xn

##      frutas verduras  carne
## [1,] 237.0122 377.7969 385.1909

```