



# Evolución de enfermedades (COVID-19)



- GERARDO RODRIGO CHAVEZ CARCAMO
- RAMIRO ALEJANDRO HERNANDEZ ORTEGA
- JOSE FRANCISCO MELCHOR LOERA
- SALVADOR GERARDO CARACOZA GAMEZ



# CADENAS DE MARKOV

## PRINCIPIOS

Las leyes de Markov se refieren a los principios fundamentales que rigen las cadenas de Markov. Estas leyes establecen que el estado futuro de un sistema en una cadena de Markov depende únicamente del estado actual y no de los estados anteriores.

## USO DEL ESTADO DISCRETA

Las cadenas de Markov se utilizan para modelar sistemas que evolucionan en estados discretos a lo largo del tiempo. Las cadenas de Markov se utilizan en problemas de optimización, como la planificación de rutas o la asignación de recursos. Ayudan a encontrar la secuencia de acciones o decisiones óptimas en función de los estados y las probabilidades de transición.

# DESCRIPCION DEL PROYECTO

---

Este proyecto se enfoca en el estudio, análisis y simulación de la propagación de la enfermedad COVID-19 en México. Para lograr este objetivo, el proyecto utiliza un script en R que recopila datos de COVID-19 disponibles públicamente, analiza estos datos y luego emplea un modelo de cadena de Márkov para simular la propagación de la enfermedad a lo largo del tiempo.



## METAS

- La meta principal es entender la dinámica de la propagación de COVID-19 en México, incluyendo las tasas de infección, recuperación y muerte.
- Continuar desarrollando y mejorando el modelo de simulación basado en cadenas de Márkov para que sea más preciso y refleje más fielmente la realidad.
- Utilizar el modelo para predecir cómo puede evolucionar la pandemia de COVID-19 en México en el futuro, lo que puede ayudar a prepararse y responder de manera más eficaz.



## Descripción del modelo

---

### Nuestro planteamiento:

El problema que abordaremos se enfoca en determinar la probabilidad de recuperación o mortalidad en personas infectadas con COVID-19. La pandemia del COVID-19 ha afectado a millones de personas en todo el mundo, y conocer la probabilidad de resultados como la recuperación o la muerte es esencial para evaluar el impacto de la enfermedad.



## Descripción del modelo

### Instalar y cargar los paquetes necesarios:

Este script utiliza los paquetes dplyr, readr, curl y markovchain, que proporcionan funcionalidades para la manipulación de datos, lectura de archivos, descarga de archivos de internet y análisis de cadenas de Márkov, respectivamente.

```
# Instalar y cargar los paquetes necesarios
library(dplyr)
library(readr)
library(curl)
library(markovchain)
```



## Descripción del modelo

### Descargar y descomprimir el archivo ZIP:

El script descarga un archivo ZIP que contiene datos sobre COVID-19 desde una URL específica. Luego, descomprime este archivo en una carpeta llamada "datos\_abiertos\_covid19".

```
# Descargar y descomprimir el archivo ZIP
url <- "https://datosabiertos.salud.gob.mx/gobmx/salud/datos_abiertos/datos_abiertos_covid19.zip"
destfile <- "datos_abiertos_covid19.zip"
curl_download(url, destfile)
unzip(destfile, exdir = "datos_abiertos_covid19")
```



## Descripción del modelo

### Leer el archivo CSV:

El script lee un archivo CSV que contiene los datos de COVID-19 descomprimidos. Se obtiene el nombre del archivo CSV de la carpeta "datos\_abiertos\_covid19".

```
# Leer el archivo CSV
csv_file <- list.files("datos_abiertos_covid19", pattern = "*.csv", full.names = TRUE)
covid_data <- read_csv(csv_file, col_types = cols())
```

### Contar los registros existentes:

Calcula y muestra el número total de registros en el conjunto de datos.

```
# Contar los registros existentes
total_registros <- nrow(covid_data)
cat("Total de registros: ", total_registros, "\n")
```





## Descripción del modelo

### Creación de una matriz de transición:

Este es un paso preparatorio para el análisis de la cadena de Márkov. La matriz de transición representa las probabilidades de pasar de un estado a otro en la cadena de Márkov.

```
# Crear una matriz de transición
NombredeEstados3 = c("Sano","Infectados","Recuperados","Muertos")
matrix_transicion <- new("markovchain", transitionMatrix = matrix(c(1 - TasaInfeccion, TasaInfeccion, 0, 0,
0, 1 - TasaRecuperacion - TasaMuerte, TasaRecuperacion, TasaMuerte,
0, 0, 1, 0,
0, 0, 0, 1),
byrow = TRUE, nrow = 4, dimnames = list(NombredeEstados3, NombredeEstados3)))

plot(matrix_transicion)
```



## Descripción del modelo

### Cálculo de tasas:

Calcula las tasas de infección, recuperación y muerte basándose en los números de casos totales, recuperados y muertes.

```
Sanos <- 126014024 #Población de México según el INEGI al 2020
Infectados <- total_registros
Muertos <- registros_fecha_def
Recuperados <- Infectados - Muertos

TasaInfeccion <- Infectados/Sanos
TasaRecuperacion <- Recuperados/Infectados
TasaMuerte <-Muertos/Infectados

TasaInfeccion
TasaRecuperacion
TasaMuerte
```



## Descripción del modelo

**Contar los registros en la columna "FECHA\_DEF" que sean diferentes a "9999-99-99":**

Calcula y muestra el número de registros que tienen una fecha de defunción registrada. En este conjunto de datos, una fecha de "9999-99-99" se usa para indicar que no se ha registrado una fecha de defunción.

```
# Contar los registros en la columna "FECHA_DEF" que sean diferentes a "9999-99-99"
registros_fecha_def <- covid_data %>%
  filter(FECHA_DEF != 9999-99-99) %>%
  nrow()

print(total_registros)
print(registros_fecha_def)
cat("Registros con fecha de defunción diferente a '9999-99-99': ", registros_fecha_def, "\n")
```



## Descripción del modelo

### Simulación de la cadena de Márkov:

Simula la propagación de la enfermedad a lo largo del tiempo, suponiendo que la enfermedad se propaga de acuerdo con la matriz de transición. Se realiza una simulación de 100 días y se repite este proceso 100 veces.

```
summary(matrix_transicion)

for (i in 1:100) {
  # Defina el número de pasos de tiempo que desea simular
  n_pasos <- 100 #días

  # Simule la cadena de Markov
  simulacion <- rmarkovchain(n = n_pasos, object = matrix_transicion, t0 = "Sano")

  # Ver la simulación
  print(i)
  print(simulacion)
}
```



## Descripción del modelo

### Probabilidad de transición:

Calcula las probabilidades de transición entre diferentes estados (por ejemplo, de "Sano" a "Infectados") utilizando la matriz de transición.

```
transitionProbability(object = matrix_transicion, t0 = "Sano", t1 = "Infectados")  
transitionProbability(object = matrix_transicion, t0 = "Infectados", t1 = "Recuperados")  
transitionProbability(object = matrix_transicion, t0 = "Infectados", t1 = "Muertos")
```



# Descripción del modelo

## Scripts de resultados:

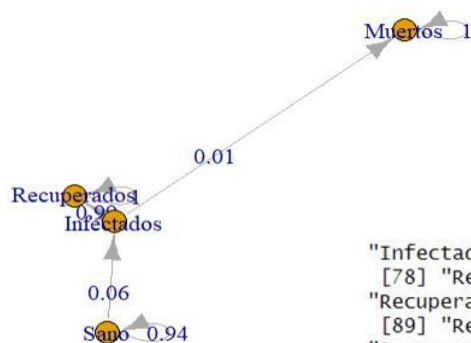
```
Warning: One or more parsing issues, call `problems()` on your data frame for
details, e.g.:
  dat <- vroom(...)
  problems(dat)Total de registros: 7266739
[1] 7266739
[1] 46515
Registros con fecha de defunción diferente a '9999-99-99': 46515
[1] 0.05766611
[1] 0.9935989
[1] 0.006401083
Unnamed Markov chain Markov chain that is composed by:
Closed classes:
Recuperados
Muertos
Recurrent classes:
{Recuperados},{Muertos}
Transient classes:
{Sano},{Infectados}
The Markov chain is not irreducible
The absorbing states are: Recuperados Muertos
[1] 1
[1] "Sano" "Infectados" "Recuperados" "Recuperados" "Recuperados"
"Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
[12] "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
"Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
[23] "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
"Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
```

[illegible]



## Descripción del modelo

### Scripts de resultados:



```
"Infectados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
[78] "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
"Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
[89] "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
"Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados" "Recuperados"
[100] "Recuperados"
[1] 0.05766611
[1] 0.9935989
[1] 0.006401083
```





## CONCLUSION

En conclusión, la utilización de las cadenas de Márkov para la simulación de la propagación de enfermedades como el COVID-19 ofrece una herramienta valiosa para anticipar la dinámica de la enfermedad. A través de este análisis, podemos obtener una mejor comprensión de las tasas de infección, recuperación y muerte, y prever cómo estas tasas pueden cambiar con el tiempo.

El proyecto demostró ser eficaz al proporcionar una visión de la situación de COVID-19 en México. Al utilizar datos públicos y análisis estadístico, se logró modelar la progresión de la enfermedad, aportando información esencial para la planificación de estrategias de salud pública.



## BIBLIOGRAFIAS

Instituto Nacional de Estadística y Geografía. (s.f.). Estructura de la población. Recuperado el 13 de mayo de 2023, de

<https://www.inegi.org.mx/temas/estructura/>

COVID-19 Tablero México - CONACYT - CentroGeo - GeoInt - DataLab. (2023) Recuperado el 13 de mayo, 2023, de <https://datos.covid-19.conacyt.mx/#DOView>

Taha, H. A. Investigación de operaciones. Pearson Educación.

Hillier, F. S., Lieberman, G. J., & Osuna, M. A. G. Introducción a la Investigación de Operaciones. McGraw-Hill.

Markovchain R Package, <https://cran.r-project.org/web/packages/markovchain/markovchain.pdf>