



UNIVERSIDAD AUTÓNOMA DE COAHUILA

FACULTAD DE SISTEMAS

Proyecto final
Predicción de clima del los municipios de Coahuila

PRESENTA

Sergio López Ortiz - 20304163
Jesús Daniel Fernández Ortiz - 20304047
Gabriel Eduardo Salas Jaramillo - 20302777
Erick Ivan Toledo Galeana - 19017291
Mariana de Jesus Arellano Rodriguez - 20308725

DOCENTE

Dra. Valeria Soto Mendoza

ASIGNATURA

Modelos Computacionales
Agosto-Diciembre 2023

10 de noviembre de 2023

Ejemplo 1.

En este problema, se presenta una serie temporal de temperaturas para Saltillo, Coahuila, en octubre de 2022. La variable aleatoria es la temperatura, y se definen tres estados climáticos: Frío, Moderado y Cálido.

Analizando la redacción del problema se observa que: – Variable Aleatoria = Temperatura. – Estados = Frío, Moderado, Cálido. – Temporalidad(paso) = Un día.

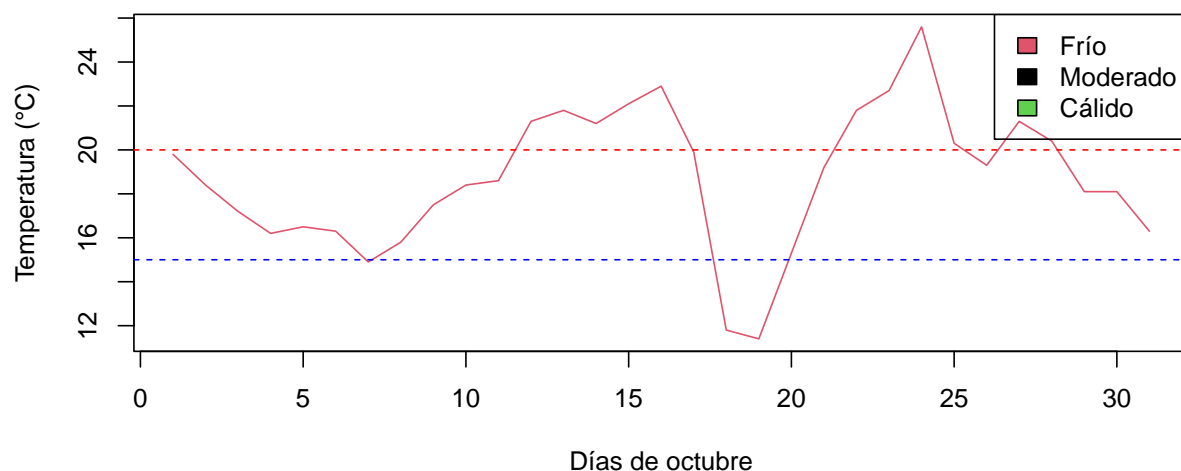
```
# Datos de temperatura para Saltillo, Coahuila, en octubre de 2022
temperaturas_octubre <- c(19.8, 18.4, 17.2, 16.2, 16.5, 16.3, 14.9, 15.8,
                          17.5, 18.4, 18.6, 21.3, 21.8, 21.2, 22.1, 22.9,
                          19.9, 11.8, 11.4, 15.3, 19.2, 21.8, 22.7, 25.6,
                          20.3, 19.3, 21.3, 20.4, 18.1, 18.1, 16.3)

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_octubre, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear cadena de Markov
cadena_markov_clima <- markovchainFit(data = estados_climaticos)

# Visualizar temperaturas y estados climáticos a lo largo del tiempo
plot(temperaturas_octubre, type = "l", col = estados_climaticos, xlab = "Días de octubre",
     ylab = "Temperatura (°C)")
legend("topright", legend = levels(estados_climaticos), fill = unique(estados_climaticos))

# Añadir líneas de referencia para los límites entre estados climáticos
abline(h = 15, col = "blue", lty = 2)
abline(h = 20, col = "red", lty = 2)
```



```

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_octubre, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear una matriz de transición
matriz_transicion <- table(estados_climaticos[-length(estados_climaticos)],
                          estados_climaticos[-1])

# Convertir a probabilidades
matriz_transicion_prob <- prop.table(matriz_transicion, margin = 1)

# Redondear hacia arriba la matriz de transición
matriz_transicion_redondeada <- ceiling(matriz_transicion_prob)

# Ajustar manualmente para asegurar que las sumas de las filas sean 1
matriz_transicion_redondeada[1, ] <-
  matriz_transicion_redondeada[1, ] / sum(matriz_transicion_redondeada[1, ])
matriz_transicion_redondeada[2, ] <-
  matriz_transicion_redondeada[2, ] / sum(matriz_transicion_redondeada[2, ])
matriz_transicion_redondeada[3, ] <-
  matriz_transicion_redondeada[3, ] / sum(matriz_transicion_redondeada[3, ])

# Convertir la matriz a fracciones
matriz_transicion_fracciones <- as.fractions(matriz_transicion_redondeada)

m <- matrix(c(1/2, 1/2, 0, 1/3, 1/3, 1/3, 0, 1/2, 1/2),
            ncol = 3, byrow = TRUE)
mc <- new("markovchain",
          states = c("Frío", "Moderado", "Cálido"), transitionMatrix = m)

```

■ a)

– Matriz de transición de estados:

```

# Imprimir la matriz de transición en fracciones
print(matriz_transicion_fracciones)

```

```

##
##           Frío Moderado Cálido
## Frío      1/2  1/2      0
## Moderado  1/3  1/3     1/3
## Cálido    0    1/2     1/2

```

– Resumen de clases de la cadena de Markov:

```

# Se muestra el resumen de la cadena de Markov
summary(mc)

```

```

## Unnamed Markov chain  Markov chain that is composed by:
## Closed classes:
## Frío Moderado Cálido
## Recurrent classes:

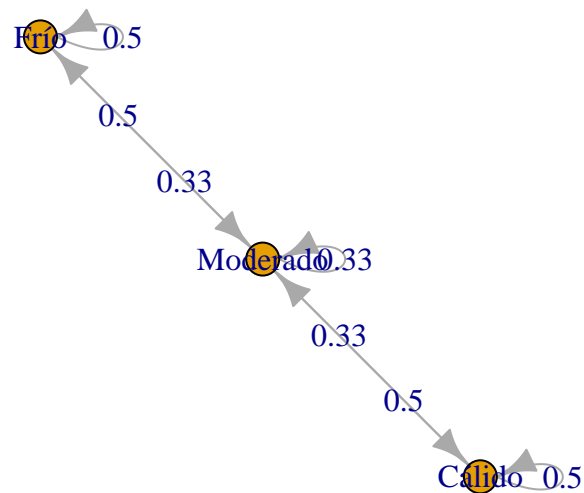
```

```
## {Frío,Moderado,Cálido}  
## Transient classes:  
## NONE  
## The Markov chain is irreducible  
## The absorbing states are: NONE
```

La cadena de Markov consta de tres estados climáticos: Frío, Moderado y Cálido. Es irreducible, lo que significa que se puede pasar de cualquier estado a otro. No tiene estados absorbentes, y todos los estados son recurrentes, lo que implica que el sistema eventualmente regresa a cualquier estado en el que se encuentre.

- b) Genere el diagrama de transición de estados.

```
plot(mc)
```



Ejemplo 2.

En este problema, se presenta una serie temporal de temperaturas para el municipio de Piedras negras, Coahuila, en Noviembre de 2022. La variable aleatoria es la temperatura, y se definen tres estados climáticos: Frío, Moderado y Cálido.

Analizando la redacción del problema se observa que: – Variable Aleatoria = Temperatura. – Estados = Frío, Moderado, Cálido. – Temporalidad(paso) = Un día.

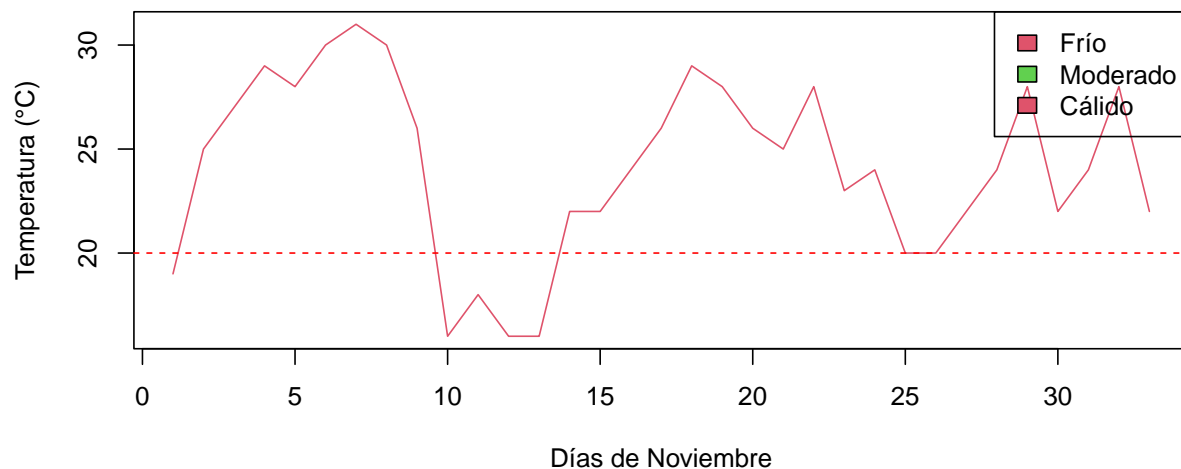
```
# Datos de temperatura para Piedras Negras, Coahuila, en Noviembre de 2022
temperaturas_noviembre <- c(19, 25, 27, 29, 28, 30, 31, 30,
                             26, 16, 18, 16, 16, 22, 22, 24,
                             26, 29, 28, 26, 25, 28, 23, 24,
                             20, 20, 22, 24, 28, 22, 24, 28,
                             22)

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_noviembre, breaks = c(-Inf, 15, 20, Inf),
                           labels = c("Frío", "Moderado", "Cálido"))

# Crear cadena de Markov
cadena_markov_clima <- markovchainFit(data = estados_climaticos)

# Visualizar temperaturas y estados climáticos a lo largo del tiempo
plot(temperaturas_noviembre, type = "l", col = estados_climaticos, xlab = "Días de Noviembre",
      ylab = "Temperatura (°C)")
legend("topright", legend = levels(estados_climaticos), fill = unique(estados_climaticos))

# Añadir líneas de referencia para los límites entre estados climáticos
abline(h = 15, col = "blue", lty = 2)
abline(h = 20, col = "red", lty = 2)
```



```

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_noviembre, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear una matriz de transición
matriz_transicion <- table(estados_climaticos[-length(estados_climaticos)],
                          estados_climaticos[-1])

# Convertir a probabilidades
matriz_transicion_prob <- prop.table(matriz_transicion, margin = 1)

# Redondear hacia arriba la matriz de transición
matriz_transicion_redondeada <- ceiling(matriz_transicion_prob)

# Ajustar manualmente para asegurar que las sumas de las filas sean 1
matriz_transicion_redondeada[1, ] <-
  matriz_transicion_redondeada[1, ] / sum(matriz_transicion_redondeada[1, ])
matriz_transicion_redondeada[2, ] <-
  matriz_transicion_redondeada[2, ] / sum(matriz_transicion_redondeada[2, ])
matriz_transicion_redondeada[3, ] <-
  matriz_transicion_redondeada[3, ] / sum(matriz_transicion_redondeada[3, ])

# Convertir la matriz a fracciones
matriz_transicion_fracciones_1 <- as.fractions(matriz_transicion_redondeada)

m <- matrix(c(0, 0, 1, 0, 1/2, 1/2, 0, 1/2, 1/2),
            ncol = 3, byrow = TRUE)
mc <- new("markovchain",
         states = c("Frío", "Moderado", "Cálido"), transitionMatrix = m)

```

■ a)

– Matriz de transición de estados:

```

# Imprimir la matriz de transición en fracciones
print(matriz_transicion_fracciones_1)

```

```

##
##           Frío Moderado Cálido
##  Frío      NaN  NaN      NaN
##  Moderado   0  1/2      1/2
##  Cálido     0  1/2      1/2

```

– Resumen de clases de la cadena de Markov:

```

# Se muestra el resumen de la cadena de Markov
summary(mc)

```

```

## Unnamed Markov chain  Markov chain that is composed by:
## Closed classes:
## Moderado Cálido
## Recurrent classes:

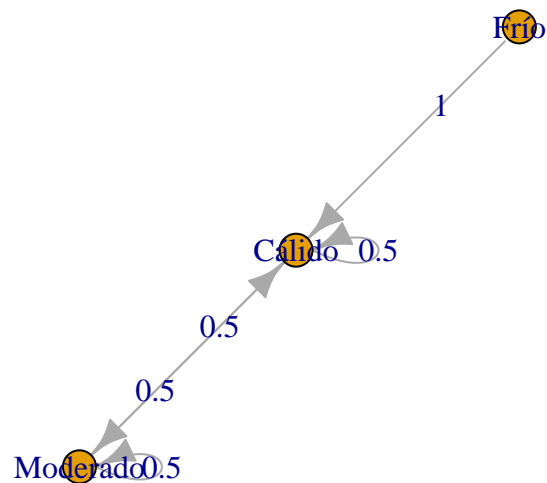
```

```
## {Moderado,Cálido}  
## Transient classes:  
## {Frío}  
## The Markov chain is not irreducible  
## The absorbing states are: NONE
```

La cadena de Markov modela tres estados climáticos: Frío, Moderado y Cálido. No es irreducible, indicando que no todos los estados son accesibles desde cualquier otro estado en el sistema. Presenta una clase transitoria, específicamente “Frío”, lo que significa que una vez que el sistema entra en este estado, no regresará a otros estados.

- b) Genere el diagrama de transición de estados.

```
plot(mc)
```



Ejemplo 3.

En este problema, se presenta una serie temporal de temperaturas para el municipio de Sabinas, Coahuila, en Marzo de 2023. La variable aleatoria es la temperatura, y se definen tres estados climáticos: Frío, Moderado y Cálido.

Analizando la redacción del problema se observa que: – Variable Aleatoria = Temperatura. – Estados = Frío, Moderado, Cálido. – Temporalidad(paso) = Un día.

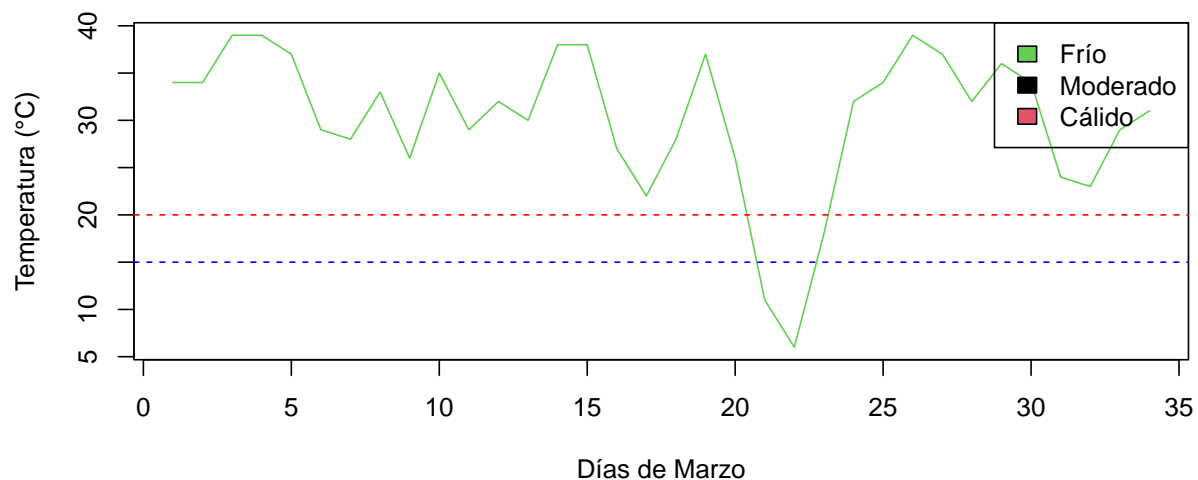
```
# Datos de temperatura para Sabinas, Coahuila, en Marzo de 2023
temperaturas_marzo <- c(34, 34, 39, 39, 37, 29, 28, 33,
                        26, 35, 29, 32, 30, 38, 38, 27, 22,
                        28, 37, 26, 11, 6, 18, 32, 34,
                        39, 37, 32, 36, 34, 24, 23, 29,
                        31)

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_marzo, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear cadena de Markov
cadena_markov_clima <- markovchainFit(data = estados_climaticos)

# Visualizar temperaturas y estados climáticos a lo largo del tiempo
plot(temperaturas_marzo, type = "l", col = estados_climaticos, xlab = "Días de Marzo",
     ylab = "Temperatura (°C)")
legend("topright", legend = levels(estados_climaticos), fill = unique(estados_climaticos))

# Añadir líneas de referencia para los límites entre estados climáticos
abline(h = 15, col = "blue", lty = 2)
abline(h = 20, col = "red", lty = 2)
```




```

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_marzo, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear una matriz de transición
matriz_transicion <- table(estados_climaticos[-length(estados_climaticos)],
                          estados_climaticos[-1])

# Convertir a probabilidades
matriz_transicion_prob <- prop.table(matriz_transicion, margin = 1)

# Redondear hacia arriba la matriz de transición
matriz_transicion_redondeada <- ceiling(matriz_transicion_prob)

# Ajustar manualmente para asegurar que las sumas de las filas sean 1
matriz_transicion_redondeada[1, ] <-
  matriz_transicion_redondeada[1, ] / sum(matriz_transicion_redondeada[1, ])
matriz_transicion_redondeada[2, ] <-
  matriz_transicion_redondeada[2, ] / sum(matriz_transicion_redondeada[2, ])
matriz_transicion_redondeada[3, ] <-
  matriz_transicion_redondeada[3, ] / sum(matriz_transicion_redondeada[3, ])

# Convertir la matriz a fracciones
matriz_transicion_fracciones_1 <- as.fractions(matriz_transicion_redondeada)

m <- matrix(c(1/2, 1/2, 0, 0, 0, 1, 1/2, 0, 1/2),
            ncol = 3, byrow = TRUE)
mc <- new("markovchain",
         states = c("Frío", "Moderado", "Cálido"), transitionMatrix = m)

```

■ a)

– Matriz de transición de estados:

```

# Imprimir la matriz de transición en fracciones
print(matriz_transicion_fracciones_1)

```

```

##
##           Frío Moderado Cálido
##  Frío      1/2  1/2      0
##  Moderado   0    0      1
##  Cálido    1/2   0     1/2

```

– Resumen de clases de la cadena de Markov:

```

# Se muestra el resumen de la cadena de Markov
summary(mc)

```

```

## Unnamed Markov chain  Markov chain that is composed by:
## Closed classes:
## Frío Moderado Cálido
## Recurrent classes:

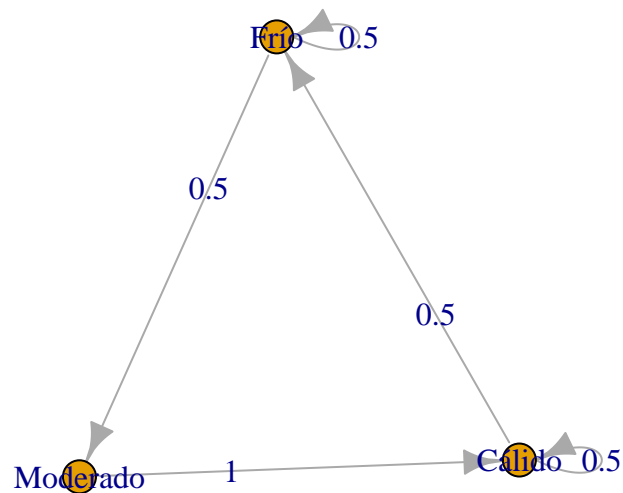
```

```
## {Frío,Moderado,Cálido}  
## Transient classes:  
## NONE  
## The Markov chain is irreducible  
## The absorbing states are: NONE
```

La cadena de Markov modela tres estados climáticos: Frío, Moderado y Cálido. Es irreducible, lo que significa que todos los estados son accesibles desde cualquier otro estado en el sistema. No presenta clases transitorias, indicando que no hay estados desde los cuales el sistema no pueda regresar.

- b) Genere el diagrama de transición de estados.

```
plot(mc)
```



Ejemplo 4.

En este problema, se presenta una serie temporal de temperaturas para Torreón, Coahuila, en octubre de 2022. La variable aleatoria es la temperatura, y se definen tres estados climáticos: Frío, Moderado y Cálido.

Analizando la redacción del problema se observa que: – Variable Aleatoria = Temperatura. – Estados = Frío, Moderado, Cálido. – Temporalidad(paso) = Un día.

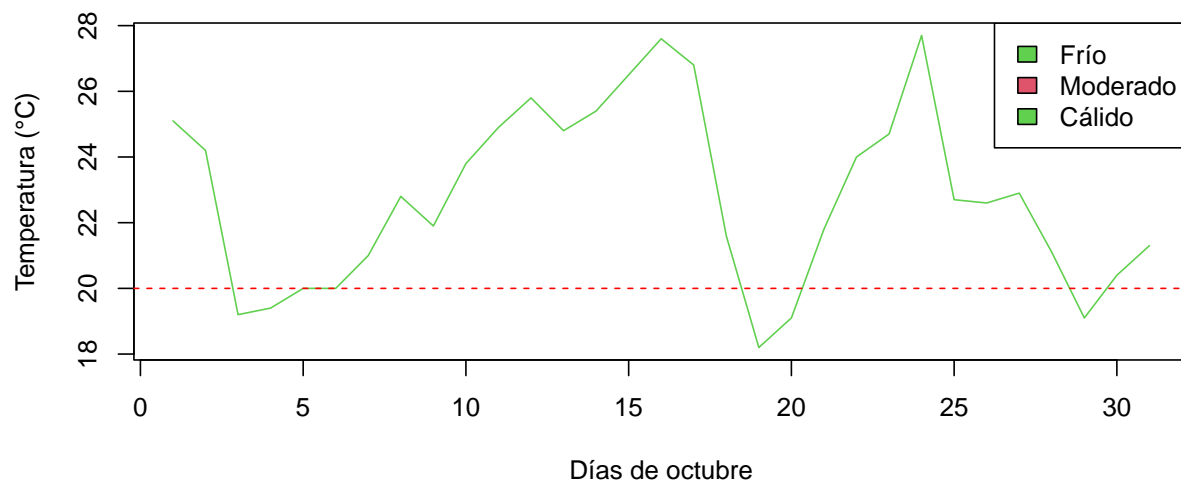
```
# Datos de temperatura para Torreón, Coahuila, en octubre de 2022
temperaturas_octubre <- c(25.1, 24.2, 19.2, 19.4, 20, 20, 21, 22.8,
                          21.9, 23.8, 24.9, 25.8, 24.8, 25.4, 26.5, 27.6,
                          26.8, 21.6, 18.2, 19.1, 21.8, 24, 24.7, 27.7,
                          22.7, 22.6, 22.9, 21.1, 19.1, 20.4, 21.3)

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_octubre, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear cadena de Markov
cadena_markov_clima <- markovchainFit(data = estados_climaticos)

# Visualizar temperaturas y estados climáticos a lo largo del tiempo
plot(temperaturas_octubre, type = "l", col = estados_climaticos, xlab = "Días de octubre",
     ylab = "Temperatura (°C)")
legend("topright", legend = levels(estados_climaticos), fill = unique(estados_climaticos))

# Añadir líneas de referencia para los límites entre estados climáticos
abline(h = 15, col = "blue", lty = 2)
abline(h = 20, col = "red", lty = 2)
```



```

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_octubre, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear una matriz de transición
matriz_transicion <- table(estados_climaticos[-length(estados_climaticos)],
                          estados_climaticos[-1])

# Convertir a probabilidades
matriz_transicion_prob <- prop.table(matriz_transicion, margin = 1)

# Redondear hacia arriba la matriz de transición
matriz_transicion_redondeada <- ceiling(matriz_transicion_prob)

# Ajustar manualmente para asegurar que las sumas de las filas sean 1
matriz_transicion_redondeada[1, ] <-
  matriz_transicion_redondeada[1, ] / sum(matriz_transicion_redondeada[1, ])
matriz_transicion_redondeada[2, ] <-
  matriz_transicion_redondeada[2, ] / sum(matriz_transicion_redondeada[2, ])
matriz_transicion_redondeada[3, ] <-
  matriz_transicion_redondeada[3, ] / sum(matriz_transicion_redondeada[3, ])

# Convertir la matriz a fracciones
matriz_transicion_fracciones <- as.fractions(matriz_transicion_redondeada)

m <- matrix(c(0,0,1, 0,1/2,1/2, 0, 1/2, 1/2),
            ncol = 3, byrow = TRUE)
mc <- new("markovchain",
          states = c("Frío", "Moderado", "Cálido"), transitionMatrix = m)

```

- a) – Matriz de transición de estados:

```

# Imprimir la matriz de transición en fracciones
print(matriz_transicion_fracciones)

```

```

##
##           Frío Moderado Cálido
##  Frío      NaN  NaN      NaN
##  Moderado   0  1/2      1/2
##  Cálido     0  1/2      1/2

```

– Resumen de clases de la cadena de Markov:

```

# Se muestra el resumen de la cadena de Markov
summary(mc)

```

```

## Unnamed Markov chain  Markov chain that is composed by:
## Closed classes:
## Moderado Cálido
## Recurrent classes:
## {Moderado,Cálido}
## Transient classes:

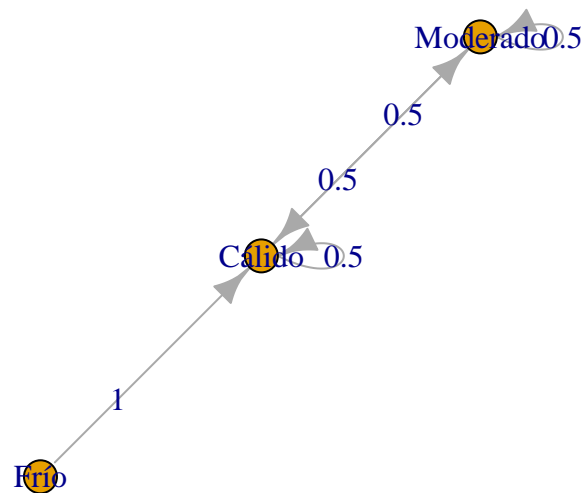
```

```
## {Frío}  
## The Markov chain is not irreducible  
## The absorbing states are: NONE
```

La cadena de Markov descrita comprende dos clases climáticas cerradas, Moderado y Cálido, ambas recurrentes, mientras que la clase Frío es transitoria. Esta transitoriedad impide la irreducibilidad de la cadena, ya que no se puede llegar desde cualquier estado a cualquier otro en un número finito de pasos. Aunque no hay estados absorbentes, lo que implica que ninguna condición climática se vuelve permanente, la presencia de la clase transitoria Frío establece limitaciones en la accesibilidad entre los estados recurrentes, Moderado y Cálido.

- b) Genere el diagrama de transición de estados.

```
plot(mc)
```



Ejemplo 5.

En este problema, se presenta una serie temporal de temperaturas para San Buenaventura, Coahuila, en octubre de 2022. La variable aleatoria es la temperatura, y se definen tres estados climáticos: Frío, Moderado y Cálido.

Analizando la redacción del problema se observa que: – Variable Aleatoria = Temperatura. – Estados = Frío, Moderado, Cálido. – Temporalidad(paso) = Un día.

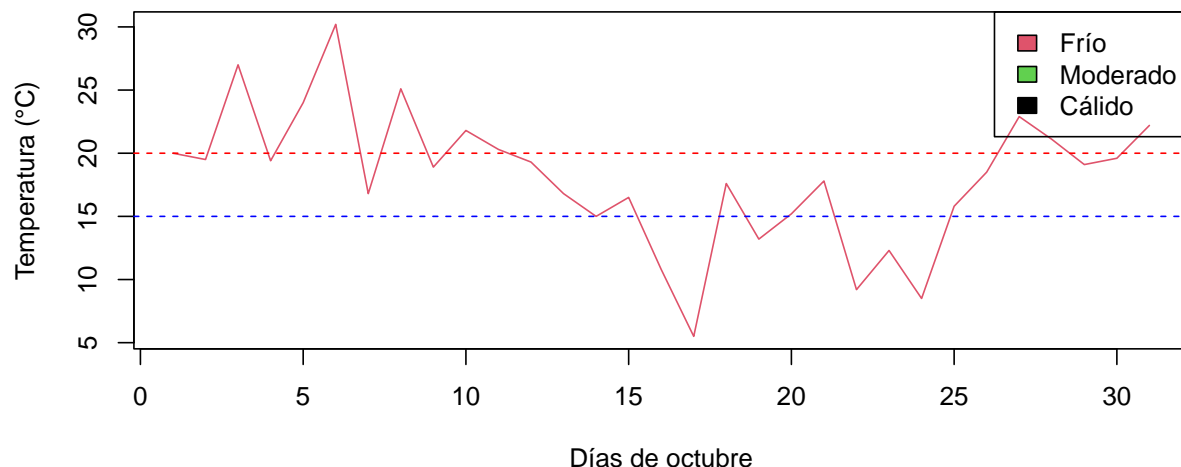
```
# Datos de temperatura para Torreón, Coahuila, en Noviembre de 2022
temperaturas_octubre <- c(20, 19.5, 27, 19.4, 24, 30.2, 16.8, 25.1,
                          18.9, 21.8, 20.3, 19.3, 16.8, 15, 16.5, 10.8,
                          5.5, 17.6, 13.2, 15.2, 17.8, 9.2, 12.3, 8.5,
                          15.8, 18.5, 22.9, 21.1, 19.1, 19.6, 22.2)

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_octubre, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear cadena de Markov
cadena_markov_clima <- markovchainFit(data = estados_climaticos)

# Visualizar temperaturas y estados climáticos a lo largo del tiempo
plot(temperaturas_octubre, type = "l", col = estados_climaticos, xlab = "Días de octubre",
     ylab = "Temperatura (°C)")
legend("topright", legend = levels(estados_climaticos), fill = unique(estados_climaticos))

# Añadir líneas de referencia para los límites entre estados climáticos
abline(h = 15, col = "blue", lty = 2)
abline(h = 20, col = "red", lty = 2)
```



```

# Definir estados climáticos
estados_climaticos <- cut(temperaturas_octubre, breaks = c(-Inf, 15, 20, Inf),
                          labels = c("Frío", "Moderado", "Cálido"))

# Crear una matriz de transición
matriz_transicion <- table(estados_climaticos[-length(estados_climaticos)],
                          estados_climaticos[-1])

# Convertir a probabilidades
matriz_transicion_prob <- prop.table(matriz_transicion, margin = 1)

# Redondear hacia arriba la matriz de transición
matriz_transicion_redondeada <- ceiling(matriz_transicion_prob)

# Ajustar manualmente para asegurar que las sumas de las filas sean 1
matriz_transicion_redondeada[1, ] <-
  matriz_transicion_redondeada[1, ] / sum(matriz_transicion_redondeada[1, ])
matriz_transicion_redondeada[2, ] <-
  matriz_transicion_redondeada[2, ] / sum(matriz_transicion_redondeada[2, ])
matriz_transicion_redondeada[3, ] <-
  matriz_transicion_redondeada[3, ] / sum(matriz_transicion_redondeada[3, ])

# Convertir la matriz a fracciones
matriz_transicion_fracciones <- as.fractions(matriz_transicion_redondeada)

m <- matrix(c(1/2, 1/2, 0, 1/3, 1/3, 1/3, 0, 1/2, 1/2),
            ncol = 3, byrow = TRUE)
mc <- new("markovchain",
          states = c("Frío", "Moderado", "Cálido"), transitionMatrix = m)

```

- a) – Matriz de transición de estados:

```

# Imprimir la matriz de transición en fracciones
print(matriz_transicion_fracciones)

```

```

##
##           Frío Moderado Cálido
## Frío      1/2  1/2      0
## Moderado  1/3  1/3      1/3
## Cálido    0   1/2      1/2

```

– Resumen de clases de la cadena de Markov:

```

# Se muestra el resumen de la cadena de Markov
summary(mc)

```

```

## Unnamed Markov chain Markov chain that is composed by:
## Closed classes:
## Frío Moderado Cálido
## Recurrent classes:
## {Frío,Moderado,Cálido}
## Transient classes:

```

```
## NONE
## The Markov chain is irreducible
## The absorbing states are: NONE
```

La cadena de Markov bajo consideración presenta tres clases cerradas: Frío, Moderado y Cálido, las cuales forman todas las clases recurrentes. No hay clases transitorias en esta cadena. La irreducibilidad de la cadena implica que es posible transitar entre cualquier par de estados en un número finito de pasos. Además, no existen estados absorbentes, lo que significa que no hay condiciones climáticas que se vuelvan permanentes.

- b) Genere el diagrama de transición de estados.

```
plot(mc)
```

