Exercise 1: Simple Markov Chain Analysis

o Task: Create a Markov chain for a system with three states (e.g., "On", "Off", "Idle") and transition probabilities provided.

To Expected Output: Display the Markov chain object, calculate the stationary distribution, simulate the chain, and check for ergodicity.

Install.packages(“markovchain”)

Define states and transition matrix

states <- c("On", "Off", "Idle")

transition_matrix <- matrix(c(0.6, 0.3, 0.1, 0.1, 0.8, 0.1, 0.3, 0.3, 0.4), nrow = 3, byrow = TRUE)

Create Markov chain

mc <- new("markovchain", states = states, transitionMatrix = transition_matrix)

Print the Markov chain object

print(mc)

Calculate stationary distribution

steady_state <- steadyStates(mc)

print(steady_state)

Simulate the Markov chain

set.seed(456) # For reproducibility

sim <- rmarkovchain(n = 20, object = mc, t0 = "On")

print(sim)

Check ergodicity

if (is.irreducible(mc)) {

cat("The Markov chain is ergodic.\n")

} else {

cat("The Markov chain is not ergodic.\n")

}

<u>**RESULT**</u>:

```
>install.packages(“markovchain”)
> states <- c("On", "Off", "Idle")
> transition_matrix <- matrix(c(0.6, 0.3, 0.1, 0.1, 0.8, 0.1, 0.3, 0.3, 0.
4), nrow = 3, byrow =
+                               TRUE)
```

```
> mc <- new("markovchain", states = states, transitionMatrix = transition_
matrix)
> print(mc)
      On Off Idle
On   0.6 0.3  0.1
Off  0.1 0.8  0.1
Idle 0.3 0.3  0.4

> steady_state <- steadyStates(mc)
> print(steady_state)
            On Off      Idle
[1,] 0.2571429 0.6 0.1428571
> set.seed(456)
> sim <- rmarkovchain(n = 20, object = mc, t0 = "On")
> print(sim)
 [1] "On"  "On"  "Off" "On"  "Off" "Off" "Off" "Off" "Off" "Off" "Off" "Of
f" "Off"
[14] "On"  "On"  "Off" "On"  "On"  "Off" "Off"
> # Check ergodicity
> if (is.irreducible(mc)) {
+   cat("The Markov chain is ergodic.\n")
+ } else {
+   cat("The Markov chain is not ergodic.\n")
+ }
The Markov chain is ergodic.
```

Exercise 2: Real-World Application

o Task: Apply Markov chains to model a practical scenario (e.g., weather p
atterns,
stock market behavior) using data or assumptions. Formulate a Markov chain
,
analyze its properties, and interpret the results.

```
# Example: Modeling weather transitions
states <- c("Sunny", "Cloudy", "Rainy")
transition_matrix <- matrix(c(0.7, 0.2, 0.1, 0.3, 0.5, 0.2, 0.2, 0.3, 0.5)
, nrow = 3, byrow =
TRUE)
mc_weather <- new("markovchain", states = states, transitionMatrix = trans
ition_matrix)
# Print the Markov chain object
print(mc_weather)
# Calculate stationary distribution
steady_state_weather <- steadyStates(mc_weather)
print(steady_state_weather)
# Simulate the Markov chain
set.seed(789) # For reproducibility
sim_weather <- rmarkovchain(n = 30, object = mc_weather, t0 = "Sunny")
print(sim_weather)
# Check ergodicity
if (is.ergodic(mc_weather)) {
cat("The weather Markov chain is ergodic.\n")
} else {
cat("The weather Markov chain is not ergodic.\n")
}
```

RESULT:

```
> states <- c("sunny", "cloudy", "rainy")
> transition_matrix <- matrix(c(0.7, 0.2, 0.1, 0.3, 0.5, 0.2, 0.2, 0.3, 0.
5), nrow = 3, byrow =
+                                TRUE)
> mc_w <- new("markovchain", states = states, transitionMatrix = transitio
n_matrix)
> print(mc_w)
       sunny cloudy rainy
sunny    0.7    0.2   0.1
cloudy   0.3    0.5   0.2
```

```
rainy     0.2     0.3    0.5

> steady_state_w <- steadyStates(mc_w)
> print(steady_state_w)
         sunny    cloudy     rainy
[1,] 0.4634146 0.3170732 0.2195122
> set.seed(789)
> sim_w <- rmarkovchain(n = 30, object = mc_w, t0 = "sunny")
> print(sim_w)
 [1] "sunny"  "sunny"  "sunny"  "sunny"  "sunny"  "sunny"  "sunny"  "sunny
"  "sunny"
[10] "sunny"  "sunny"  "sunny"  "sunny"  "sunny"  "sunny"  "sunny"  "sunny
"  "sunny"
[19] "cloudy" "cloudy" "cloudy" "cloudy" "cloudy" "cloudy" "cloudy" "sunny
"  "sunny"
[28] "sunny"  "cloudy" "sunny"
> # Check ergodicity
> if (is.irreducible(mc_w)) {
+   cat("The Markov chain is ergodic.\n")
+ } else {
+   cat("The Markov chain is not ergodic.\n")
+ }
The Markov chain is ergodic.
```