BAREKENG: Journal of Mathematics and Its Applications
December 2023 Volume 17 Issue 4 Page 2397–2410
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol17iss4pp2397-2410

INTEGRATED OF WEB APPLICATION RSHINY FOR MARKOV CHAIN AND ITS APPLICATION TO THE DAILY CASES OF COVID-19 IN WEST SUMATERA

Putri Monika[1], **Budi Nurani Ruchjana**[2*], **Kankan Parmikanti**[3], **Atje Setiawan Abdullah**[4]

[1]Doctoral Program of Mathematics, Faculty of Mathematics and Natural Sciences, Padjadjaran University
[2,3]Department of Mathematics, Faculty of Mathematics and Natural Sciences, Padjadjaran University
[4]Department of Computer Science, Faculty of Mathematics and Natural Sciences, Padjadjaran University, Sumedang, 45363, Indonesia

*Corresponding author's e-mail: ** **budi.nurani@unpad.ac.id**

ABSTRACT

Article History:

Received: 24th August 2023
Revised: 21st October 2023
Accepted: 20th November 2023

Keywords:

COVID-19;
Markov Chain;
Stationary Distribution;
Web Application Rshiny..

Discrete-time of Markov chains, starting now referred to as Markov chains, have been widely used by previous researchers in predicting the phenomenon. The predictions were made by manual calculations and using separate software, including Maple, Matlab, and Microsoft Excel. The analysis takes a relatively long time, especially in calculating the number of transitions from each state. This research built an integrated R script for the Markov chain based on the web application RShiny to quickly, easily, and accurately predict a phenomenon. The Markov chain integrated R script is built via command-command to predict the day-n distribution with the n-step distribution and long-term probability using a stationary distribution. The RShiny web application built is limited to states two and three. The integrated web application RShiny for the Markov chain is used to predict the daily cases of COVID-19 in West Sumatra. Based on the analysis carried out in predicting the daily cases of COVID-19 in West Sumatra from March 26, 2020, to October 20, 2020, for the next three days and in the long term, the results show that there is a 51.2% probability of an increase in COVID-19 cases, a 43% probability that cases will decrease, and 5.8% chance of stagnant cases

How to cite this article:

P. Monika, B. N. Ruchjana, K. Parmikanti and A. Setiawan., "INTEGRATED OF WEB APPLICATION RSHINY FOR MARKOV CHAIN AND ITS APPLICATION TO THE DAILY CASES OF COVID-19 IN WEST SUMATERA," *BAREKENG: J. Math. & App.*, vol. 17, iss. 4, pp. 2397-2410, December, 2023.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

Research Article · Open Access

1. INTRODUCTION

Corona disease-2019, also known as COVID-19, is an infectious disease caused by the SARS-Cov-2 virus. This disease was discovered at the end of 2019 in Wuhan, China. The virus quickly spread throughout numerous countries, and the World Health Organization proclaimed Covid-19 a pandemic in March 2020. According to [1] COVID-19 transmission can occur through direct contact with respiratory droplets of an infected individual (produced by coughing and sneezing) and touching surfaces contaminated with the virus. Fever, coughing, and shortness of breath are some of the symptoms. The infection can lead to pneumonia or difficulty breathing in more severe situations [2].

The disease quickly spread around the world, including in Indonesia. On March 2, 2020, the first positive case of COVID-19 was discovered in Indonesia. In a reasonably short time, COVID-19 spread to 34 provinces in Indonesia. One of them is West Sumatra Province, COVID-19 has spread since March 26, 2020, with the discovery of five positive cases [3]. A lot of research has been done in modelling the COVID-19 phenomenon. This paper uses the Markov chain method to predict the long-term chances of COVID-19 cases.

Markov chain is a stochastic model that describes a succession of possible occurrences in which the probability of each event is determined solely by the state obtained in the preceding event [4]–[8]. The Markov chain method has been widely used in research. K. J. Doubleday and J. N. Esunge apply the Markov chain discrete-time stochastic model to determine the relationship between various stock portfolios and the market as a whole [9]. F. Kachapova represents the Markov chain with transition diagrams for each state [10]. In education, Mardiyah. Maiyastri and D. Devianto use the Markov chain for Forecasting the Student Achievement Index [11]. F. N. Masukum, Y.A.R. Langi, and C. Mongi introduced Markov chain analysis to predict consumer movement of airline route Manado-Jakarta [12]. Markov chains are also used to forecast rainfall following the data mining approach [13]. In the case of COVID-19, Markov chains are used to forecast future conditions using stationary distributions [14], [15].

However, in these studies, manual calculations were carried out using software such as Maple, Ms Excel, and Matlab, which require a relatively long time to process the data. Along with the development of science, processing data can use the help of software. One is software R. Software R is software for data analysis purposes. Software R has advantages, including being open-source and easy to understand compared to other programming languages, can speed up data processing, and provide more accurate results [16]. The features offered by R software are complete for data processing needs. The R script built can be used as a Web Application with RShiny [17]. RShiny is a package from RStudio used to create interactive web applications. Other users in processing data can use web applications that have been created with RShiny [18]. The user uploads the data to be processed, and the output of the processing results can be immediately displayed on the web application RShiny display. The benefits of using RShiny for researchers are speeding up data processing and an interactive display.

This paper builds commands into an R script for discrete-time Markov chains to simplify and speed up data processing. Besides that, the R script does not require other software. The algorithm for composing the R script begins with inputting data which the Markov chain will then process. Data is defined in several states and expressed by a transition probability matrix. The nth-day probability will calculate by using the n-step distribution. The R script was created to classify the Markov chain's state, including irreducible, positive recurrent, and aperiodic conditions for an ergodic Markov chain—calculating long-term probabilities using the Markov chain stationary distribution. The R script was developed into an integrated web application RShiny for the Markov chain. In this research, the developed RShiny web application is limited to two and three states. Furthermore, the web application RShiny is published for data processing. The data used in this study is daily case data of COVID-19 in West Sumatra from March 26, 2020, to October 20, 2020.

2. RESEARCH METHODS

2.1 Software R and Web Application RShiny

R is statistical software for data analysis purposes [16]. Software R has advantages, including many data processing options with complete features; it is easier to learn and run to process data than other programming languages. R is free and open-source, so there is no need for licensing fees which are usually very expensive for data processing software. Suitable for both beginners and experts. Packages "markovchain" is a library that makes it easy to process data using the discrete Markov time chain method. The "markovchain" packages are equipped with functions to perform data analysis. For example, these packages can calculate the stationary distribution and classify state space [19].

Shiny is a set of RStudio tools for creating interactive web apps [11]. Shiny makes creating interactive web apps in R relatively straightforward. Applications that are sophisticated, accessible, and efficient can be constructed [20]. Shiny's official website can be found at https://shiny.rstudio.com.

Web applications RShiny make it very easy for users to process data. RShiny can be accessed via the web without running a script on RStudio. The web application is flexible because it can be accessed via PC, smartphone, and tablet. The web application RShiny consists of a user interface script and a server script as shown in **Figure 1** **[17]**. User interface scripts should be named ui.r, which controls input data, variables, and models. Display controls in the user interface script include sliders, radio buttons, and checkboxes. The user interface is used to enter input values and presentations, which can be graphics, numbers, and text. Server scripts are stored in a file named server.r. The server is the program's brain that contains all the data processing procedures and then sends the results to the output section. ShinyApp can be run with a combination of ui.r and server.r simultaneously. **Figure 1** represents the component structure in RShiny **[20]**.

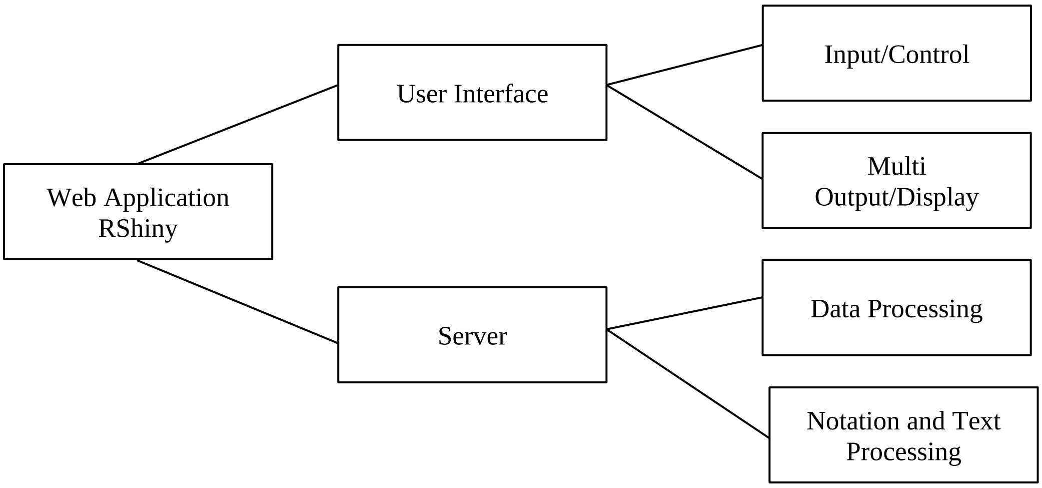


Figure 1. Component Web Application RShiny

2.2 Discrete-Time Markov Chains

According to Osaki [6], the discrete-time Markov chain is a stochastic process $\{X(n), n = 0,1,2 \ldots\}$ with a state space $i = 0,1,2, \ldots$, which satisfies,

$$P\{X_{n+1} = j \,|X(0) = i_0, X(1) = i_1, \ldots, X(n-1) = i_{n-1}, X(n) = i\} \\ = P\{X_{n+1} = j \,|X_n = i\} = p_{ij} \tag{1}$$

for all $i_0, i_1, \ldots, i_{n-1}, i, j, n$, and p_{ij} is a transition probability.

Note that $X(n) = i$ denotes that the process is in state i $(i = 0,1,2, \ldots)$ at time $(n = 0,1,2, \ldots)$. The conditional probability in **Equation (1)**, which describes the whole history is independent of the past history at time $0,1,2, \ldots, n-1$ and depends to the “future” state depends only on the “present” state and is independent of “past” history. Such a property is called the *Markov Property*. The probability of transition from state i to state j (p_{ij}) in **Equation (1)** can be expressed in the transition probability matrix $(\mathbf{P})$.

$$\mathbf{P} = [p_{ij}] = \begin{bmatrix} p_{00} & p_{01} & \cdots & p_{0n} \\ p_{10} & p_{11} & \cdots & p_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ p_{n0} & p_{n1} & \cdots & p_{nn} \end{bmatrix} \tag{2}$$

The n-step transition probability (p_{ij}^{n}) can be calculated by summing over all the intermediate state k at time r $(0 \leq r \leq n)$ and moving to state j from state k at the remaining time $n-r$ with $p_{ij}^{0}=0, i \neq j$ and $p_{ii}^{0}=0$.

$$p_{ij}^{n}=\sum_{k=0}^{\infty} p_{ik}^{r}\, p_{kj}^{n-r} \tag{3}$$

Equation (3) is called the Chapman-Kolmogorov equation. Let $\pi(n)=[\pi_0(n), \pi_1(n), \ldots]$ is the n-step distributions, then,

$$\sum_{j=0}^{\infty} \pi_j(n)=1 \tag{4}$$

so that

$$\pi(n)=\pi(0)\mathbf{P}^{n} \tag{5}$$

An ergodic Markov chain must meet the requirements, such as irreducible, aperiodic, and positive recurrent as in [6].

a) Irreducible

If a Markov chain has only one equivalent class, the Markov chain is called irreducible. For the irreducible Markov chain, all the states communicate with each other.

b) Recurrent state

The state i is said to be recurrent if and only if,

$$\sum_{n=0}^{\infty} p_{ii}^{n}=\infty \tag{6}$$

State i is transient if and only if,

$$\sum_{n=0}^{\infty} p_{ii}^{n}<\infty \tag{7}$$

c) Periodicity

The period of the state i is denoted by being the greatest common divisor (gcd) of the integers where $d(i) n \geq 1 p_{ii}^{n}>0$.

If $d(i)=1$, the state i is said to be aperiodic.

If $d(i)>1$, the state i is said to be periodic.

d) Positive Recurrent

If the state j is recurrent and aperiodic, then

$$p_{jj}^{n} \rightarrow \frac{1}{\mu_j}, \text{to } n \rightarrow \infty \tag{8}$$

If the state j is recurrent and periodic, then

$$p_{jj}^{nd(j)} \rightarrow \frac{d(j)}{\mu_j}, \text{to } n \rightarrow \infty \tag{9}$$

interpreted $\frac{1}{\mu_j}=0, \mu_j=\infty$, it means a null recurrent state. For an ergodic Markov chain there is a limit of chance,

$$\lim_{n \rightarrow \infty} p_{ij}^{n}=\pi_j>0 \qquad (i, j=0,1,2, \ldots) \tag{10}$$

which is free from the initial state i, with $\{\pi_j, j=0,1,2, \ldots\}$ it is singular and it is the positive solution of

$$\pi_j=\sum_{i=0}^{\infty} \pi_i p_{ij}, j \geq 0 \tag{11}$$

and

$$\sum_{j=0}^{\infty} \pi_j = 1 \tag{12}$$

$\{\pi_j\}$ is called the Markov chain stationary distribution [15].

3. RESULTS AND DISCUSSION

3.1 Build Script R for the Markov Chain

Building script R for the Markov chain is required packages for processing data, such as "markovchain" packages, "diagrams", and "dplyr". The Markov chain integrated R script was built for three states in this research. Data were defined and classified using criteria based on the difference between day and day data. Let is the nth day of data if it is defined as state 0 (downstate), if it is defined as state 1 (stagnation state), and if it is defined as state 2 (upward state). First, we define the data into several states and calculate the transition probability. Script R is built to define and classify each state. The transition probability of each state is expressed in the transition probability matrix following Equation (2). The transition probability of each step is calculated quickly and automatically with the built R Script. The transition probability is calculated using an if-else statement and process looping by following the steps on the pseudocode that is presented in Algorithm 1.

Algorithm 1 Algorithm for Define Data and Calculating Transition Probability

Input: Data COVID-19
Ouput: Transition Probability Matrix **P**

```
for i in 1: nrow(data) do
   if (data < 0)
   data ← “0”
   else if (data > 0)
   data ← “2”
   else
   data ← “1”
end for
for (i in 1:n)
      Define the matrix 3 × 3
      Define n  ← nrow(data)-1
      Calculate the transition probability
      matrix P according to (2)
end for
```

The transitions of each state are shown using a transition diagram. The script R represents the transition probability matrix, and the transition diagram is presented in Algorithm 2.

Algorithm 2 Algorithm for Transition Diagram

Input: Transition Probability Matrix **P**
Ouput: Transition Diagram

```
data ← Transition Pobability Matrix P
   new(“markovchain”,
   state=c(“0”,”1”,”2”),
   transitionMatrix=matriks,
   name=”Positif Case 1”)
 Plot(data)
 Using Package “diagram”
```

To predict the day-n of distribution by following Equation (5). Defines the initial probability of a Markov chain and done looping, which stops when the state converges or reaches steady states. The n-step distribution is represented in Algorithm 3.

Algorithm 3 Algorithm for Calculate the n-step distribution

```
Input: Transition Probability Matrix P,
       initial state
Ouput: $n$- step distribution
initialState ← c(v/p, w/p)
Define the function:
  nprediction ← function
  (mchain, intialState,n)
    for (i in 0:n)
    Calculate the $n$ −step distribution
     by iteration
     out ← out[,c(3,1:2)]
     return (out)
i ←1
z ← nprediction(mchain=data,
initialState=c(v/p,w/p),n=i)
while(!z[i,2] == z[i+1,2]&
      z[i,3] == z[i+1,3])
    z ← nprediction(mchain=data,
    initialState=c(v/p,w/p),n=i+1)
    i ←i+1
return(z)
```

If the Markov chain is ergodic, long-term probability can be estimated. Long-term probability was determined using a stationary Markov chain distribution and **Equation (11)**. Algorithm 4 presents the pseudocode for examining the ergodic Markov chain and calculating the stationary distribution of the Markov chain.

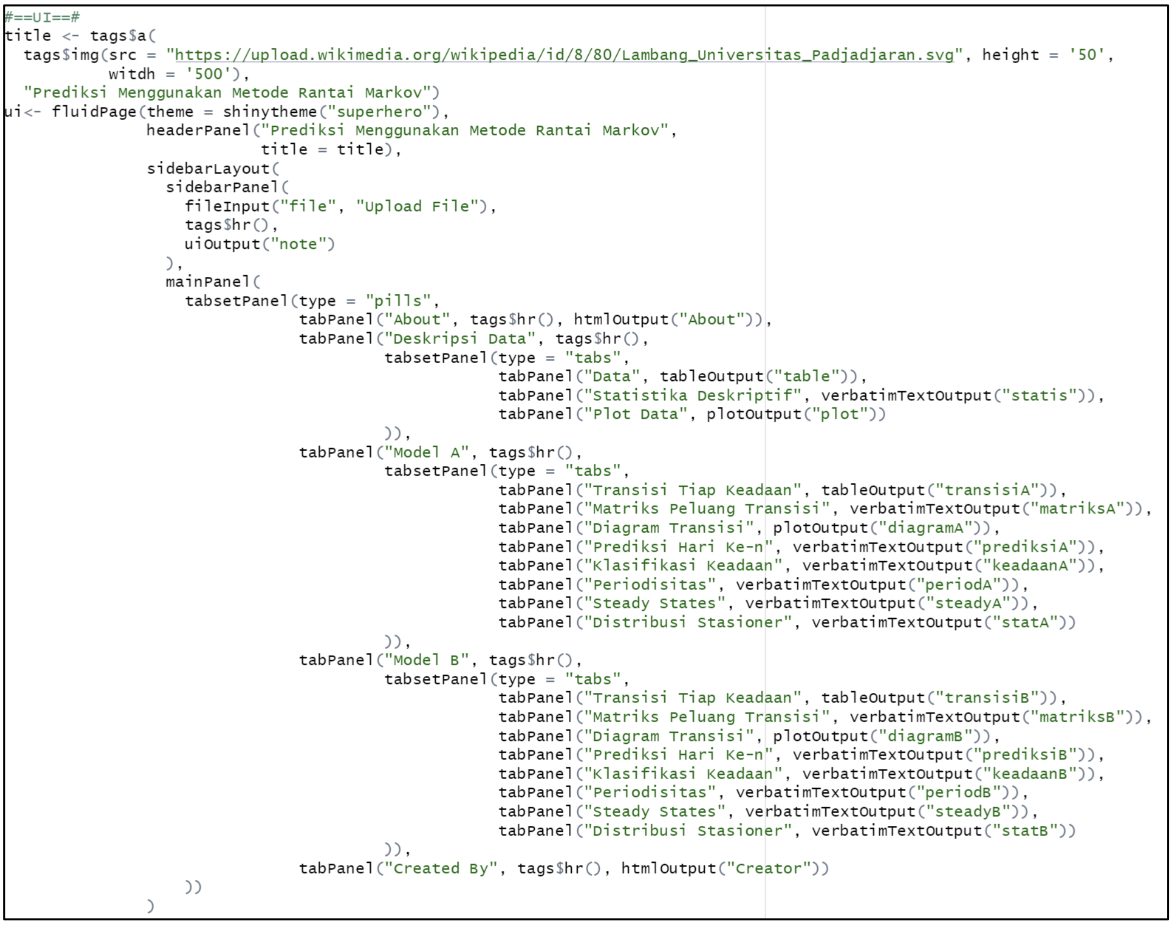
Algorithm 4 Algorithm for examine the ergodic Markov chain and calculate stasionary distribution

```
Input: data ← transition probability matrix P
Ouput: Summary Data, stasionary
       distribution
data ← Transition Pobability Matrix P
checking irreducible state
is.irreducible(data)
checking prediocity state
period(data)
checking recurrent state
recurrentStates(data)
summary(data)
Calculate stasionary distribution
by following (11)
stationary distribution ← data[nrow(data),]
```

3.2 Creating a Web Application for Markov Chain

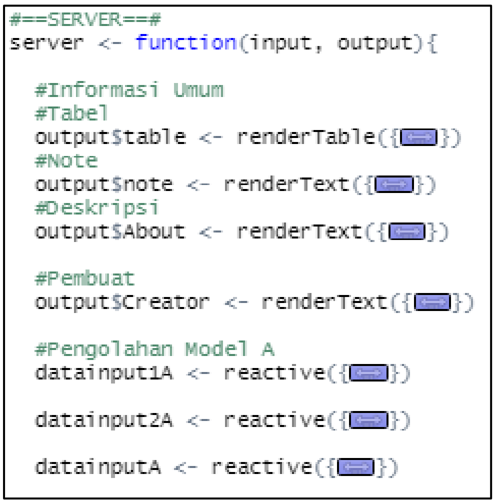
The first step to creating a web application using RShiny is to install and import the relevant packages. Two primary packages are typically utilized when constructing web apps using RShiny: the "shiny" package and the "shinythemes" package. The "shiny" package offers a framework for creating interactive user interfaces, while the "shinythemes" package allows you to make more visually appealing display themes. This web application is divided into the user interface (UI) script and the server script. The user interface (UI) configures the web app's visual appearance, including the layout, UI elements, and themes. Server scripts are responsible for processing data, executing computations, and integrating user activities with the user interface displays. **Figure 2** depicts an overview of the user interface (UI) script, which may comprise numerous elements such as input panels, graphics, text, and other components required to generate the desired interactive display. Add buttons, sliders, text input boxes, and so on. Meanwhile, **Figure 3** displays the server script, which performs numerous operations and calculations.

Server scripts may include data processing, analysis, or mathematical calculations to offer suitable answers to user interface activities.



```
#==UI==#
title <- tags$a(
  tags$img(src = "https://upload.wikimedia.org/wikipedia/id/8/80/Lambang_Universitas_Padjadjaran.svg", height = '50',
           witdh = '500'),
  "Prediksi Menggunakan Metode Rantai Markov")
ui<- fluidPage(theme = shinytheme("superhero"),
               headerPanel("Prediksi Menggunakan Metode Rantai Markov",
                           title = title),
               sidebarLayout(
                 sidebarPanel(
                   fileInput("file", "Upload File"),
                   tags$hr(),
                   uiOutput("note")
                 ),
                 mainPanel(
                   tabsetPanel(type = "pills",
                               tabPanel("About", tags$hr(), htmlOutput("About")),
                               tabPanel("Deskripsi Data", tags$hr(),
                                        tabsetPanel(type = "tabs",
                                                    tabPanel("Data", tableOutput("table")),
                                                    tabPanel("Statistika Deskriptif", verbatimTextOutput("statis")),
                                                    tabPanel("Plot Data", plotOutput("plot"))
                                        )),
                               tabPanel("Model A", tags$hr(),
                                        tabsetPanel(type = "tabs",
                                                    tabPanel("Transisi Tiap Keadaan", tableOutput("transisiA")),
                                                    tabPanel("Matriks Peluang Transisi", verbatimTextOutput("matriksA")),
                                                    tabPanel("Diagram Transisi", plotOutput("diagramA")),
                                                    tabPanel("Prediksi Hari Ke-n", verbatimTextOutput("prediksiA")),
                                                    tabPanel("Klasifikasi Keadaan", verbatimTextOutput("keadaanA")),
                                                    tabPanel("Periodisitas", verbatimTextOutput("periodA")),
                                                    tabPanel("Steady States", verbatimTextOutput("steadyA")),
                                                    tabPanel("Distribusi Stasioner", verbatimTextOutput("statA"))
                                        )),
                               tabPanel("Model B", tags$hr(),
                                        tabsetPanel(type = "tabs",
                                                    tabPanel("Transisi Tiap Keadaan", tableOutput("transisiB")),
                                                    tabPanel("Matriks Peluang Transisi", verbatimTextOutput("matriksB")),
                                                    tabPanel("Diagram Transisi", plotOutput("diagramB")),
                                                    tabPanel("Prediksi Hari Ke-n", verbatimTextOutput("prediksiB")),
                                                    tabPanel("Klasifikasi Keadaan", verbatimTextOutput("keadaanB")),
                                                    tabPanel("Periodisitas", verbatimTextOutput("periodB")),
                                                    tabPanel("Steady States", verbatimTextOutput("steadyB")),
                                                    tabPanel("Distribusi Stasioner", verbatimTextOutput("statB"))
                                        )),
                               tabPanel("Created By", tags$hr(), htmlOutput("Creator"))
                   ))
               )
```

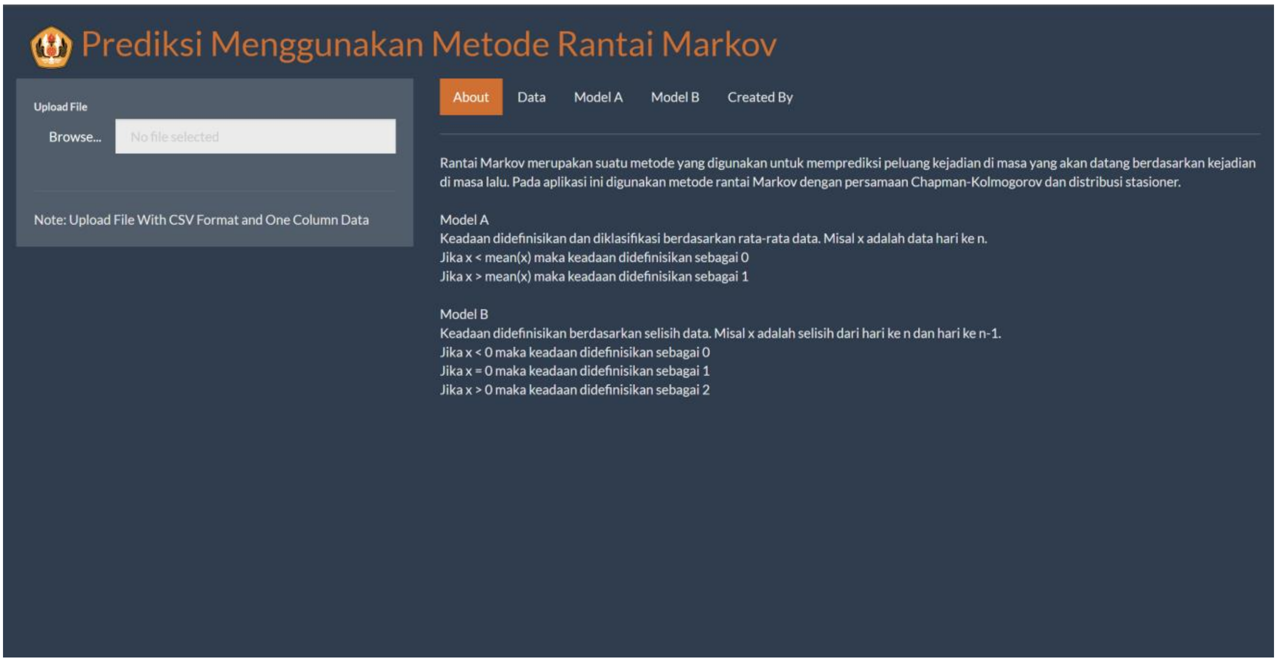
Figure 2. Script for User Interface RShiny



```
#==SERVER==#
server <- function(input, output){

  #Informasi Umum
  #Tabel
  output$table <- renderTable({⇔})
  #Note
  output$note <- renderText({⇔})
  #Deskripsi
  output$About <- renderText({⇔})

  #Pembuat
  output$Creator <- renderText({⇔})

  #Pengolahan Model A
  datainput1A <- reactive({⇔})

  datainput2A <- reactive({⇔})

  datainputA <- reactive({⇔})
```

Figure 3. Script for Server RShiny

The completed RShiny web application is published when the user interface and server scripts have been constructed and configured as needed. In this regard, **Figure 4** depicts a published RShiny web application that may be viewed online via links like https://putrimonika.shinyapps.io/rantaimarkov/. By publishing this web application, you enable others to access and interact with the application you have created via the internet. This process transforms an application that could only be run locally into something that users from various locations can access. As such, the steps involved in developing a web application with RShiny include installing and importing packages, scripting the user interface and server scripts, and the final stage of publishing the application so that it can be accessed online. In addition, you can organize various elements in the user interface and implement functionality that suits the needs of the application you are developing.



Prediksi Menggunakan Metode Rantai Markov

Note: Upload File With CSV Format and One Column Data

Rantai Markov merupakan suatu metode yang digunakan untuk memprediksi peluang kejadian di masa yang akan datang berdasarkan kejadian di masa lalu. Pada aplikasi ini digunakan metode rantai Markov dengan persamaan Chapman-Kolmogorov dan distribusi stasioner.

Model A
Keadaan didefinisikan dan diklasifikasi berdasarkan rata-rata data. Misal x adalah data hari ke n.
Jika x < mean(x) maka keadaan didefinisikan sebagai 0
Jika x > mean(x) maka keadaan didefinisikan sebagai 1

Model B
Keadaan didefinisikan berdasarkan selisih data. Misal x adalah selisih dari hari ke n dan hari ke n-1.
Jika x < 0 maka keadaan didefinisikan sebagai 0
Jika x = 0 maka keadaan didefinisikan sebagai 1
Jika x > 0 maka keadaan didefinisikan sebagai 2

Figure 4. Views Web Application Published RShiny

3.3 Numerical Experiments

This information is collected via the official website (https://corona.sumbarprov.go.id/). The data records the number of COVID-19 instances in the West Sumatra region daily. Table 1 displays descriptive statistics to provide a more in-depth look at the data. Furthermore, data on the number of people who tested positive for COVID-19 in West Sumatra has been processed into a graph, as shown in Figure 5. The descriptive statistics of daily COVID-19 case data in West Sumatra are shown in Table 1. In this situation, the minimum value is 0, suggesting no COVID-19 cases were reported on that day or period. The maximum value is 544, indicating the number of COVID-19 cases per day or time. The observed variable is associated with an average of 54 persons. The average number of persons is eleven. This means that half of the days or periods have fewer than or equal to 11 notable cases, while the other half have more significant issues than 11.

Table 1. Descriptive Statistics of Research Data

Descriptive Statistics	Value (Person)
Minimum	0
Maximum	544
Mean	54
Median	11

Meanwhile, Figure 5 depicts a graph of the number of people who tested positive for COVID-19 in West Sumatra. This graph shows the trend and variation of daily cases across the period analyzed. By looking at this graph, it is possible to see specific patterns, such as spikes in cases in certain periods, decreasing or increasing trends, as well as potential fluctuations that can provide additional insights for the interpretation of the research results.

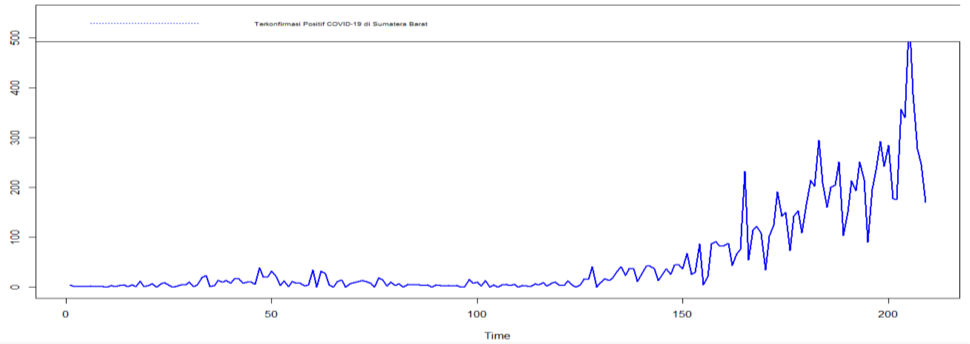
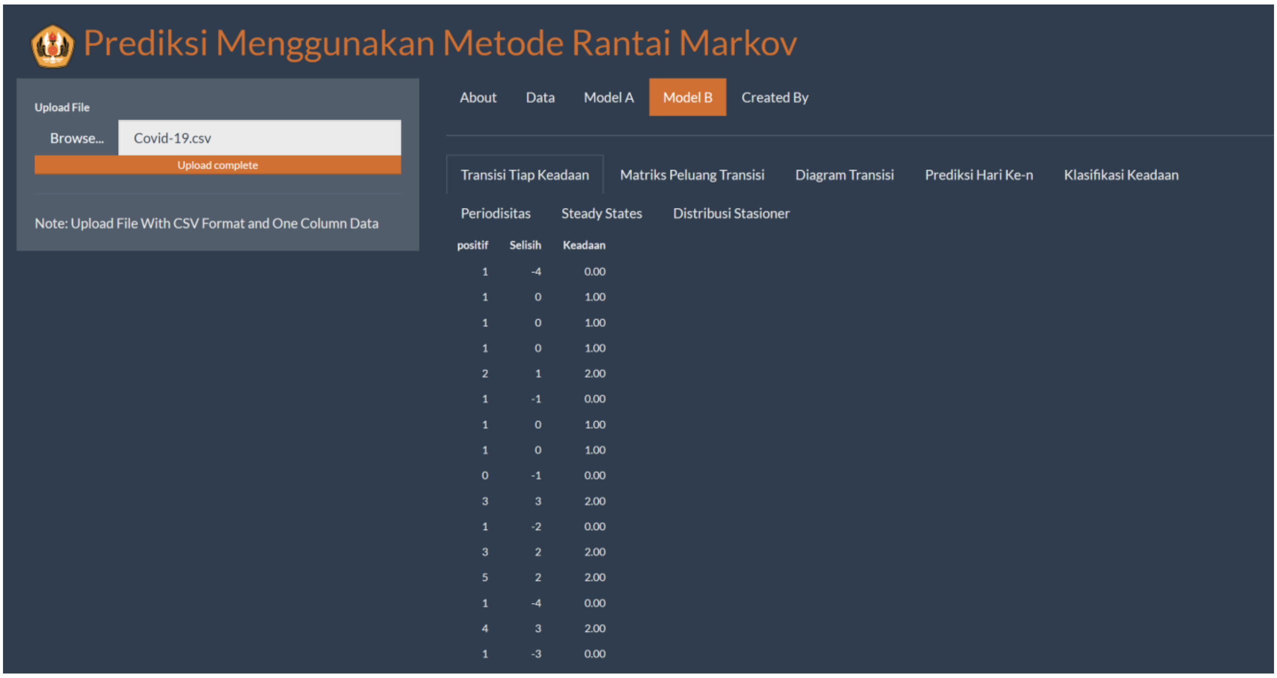


Figure 5. Plot of daily COVID-19 case data in West Sumatra

Data were defined and classified using criteria based on the difference between day$-n$ and day$-(n-1)$ data. Let x_n is the nth day of data. If $x_n < 0$ is defined as state 0 (downstate), if $x_n = 0$ is defined as state 1 (stagnation state), and if $x_n > 0$ is defined as state 2 (upward state). Figure 6 shows the transition of each state processed by script R.



positif	Selisih	Keadaan
1	-4	0.00
1	0	1.00
1	0	1.00
1	0	1.00
2	1	2.00
1	-1	0.00
1	0	1.00
1	0	1.00
0	-1	0.00
3	3	2.00
1	-2	0.00
3	2	2.00
5	2	2.00
1	-4	0.00
4	3	2.00
1	-3	0.00

Figure 6. The Display of Web Applications for The Transition of Each State

The probability of a one-step transition from state i to state j $(i, j = 0,1,2.)$ is shown in Table 2 below,

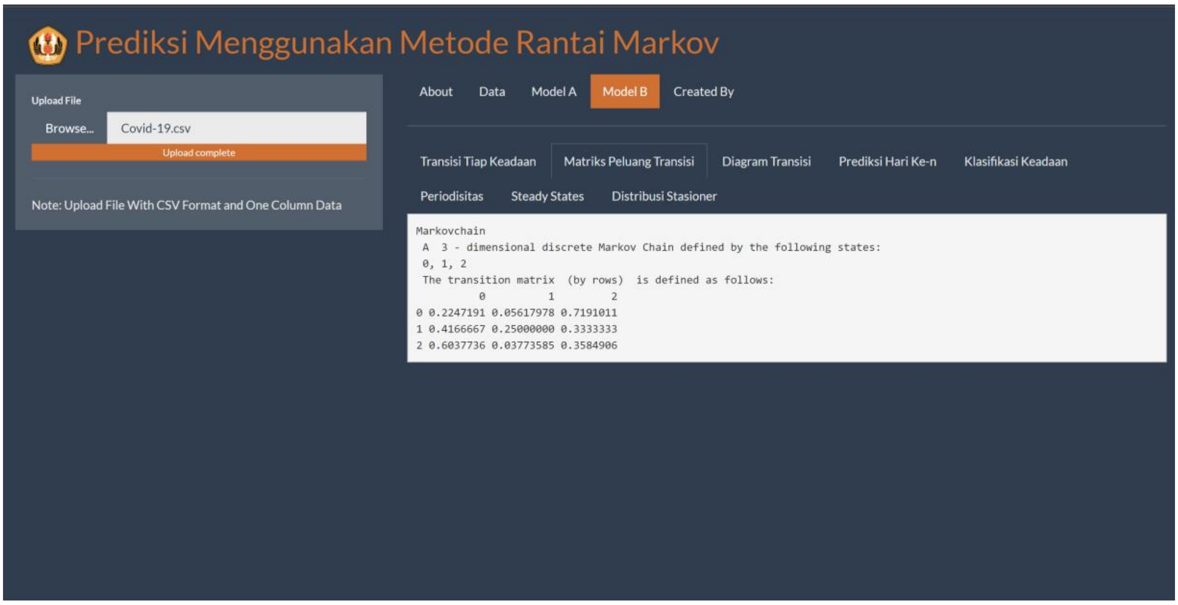
Table 2. Transition Probability form State *i* to State *j*

j ***i***	**0**	**1**	**2**
0	$\frac{20}{89}$	$\frac{5}{89}$	$\frac{64}{89}$
1	$\frac{5}{12}$	$\frac{3}{12}$	$\frac{4}{12}$
2	$\frac{64}{106}$	$\frac{4}{106}$	$\frac{38}{106}$

Probability from state i to state j can be expressed in transition probability matrix follows Equation (2).

$$\mathbf{P} = [p_{ij}] = \begin{bmatrix} \frac{20}{89} & \frac{5}{89} & \frac{64}{89} \\ \frac{5}{12} & \frac{3}{12} & \frac{4}{12} \\ \frac{64}{106} & \frac{4}{106} & \frac{38}{106} \end{bmatrix} \tag{13}$$

Figure 7 displays a visual representation of the web application designed to show the order transition probability matrix in a Markov chain that includes three states: 0, 1, and 2. This web application offers users an interactive platform to study and comprehend the probabilities associated with transitions between different states in a Markov chain. States 0, 1, and 2 represent other conditions or situations, and the transition probabilities show the possibility of moving from one state to another during a specific period.



```
Markovchain
 A  3 - dimensional discrete Markov Chain defined by the following states:
 0, 1, 2
 The transition matrix  (by rows)  is defined as follows:
          0          1         2
0 0.2247191 0.05617978 0.7191011
1 0.4166667 0.25000000 0.3333333
2 0.6037736 0.03773585 0.3584906
```

Figure 7. Web Application View for Transition Probability Matrix

The transition diagram in **Figure 8** depicts the state transitions in a Markov chain, which has three different states (0, 1, and 2). The figure represents how each state transitions dynamically between these stages. We may see the probability and patterns that control the passage from one state to another by observing the arrows and links in the diagram. This depiction aids in comprehending the underlying behavior and transitions associated with moving a Markov chain through several states.

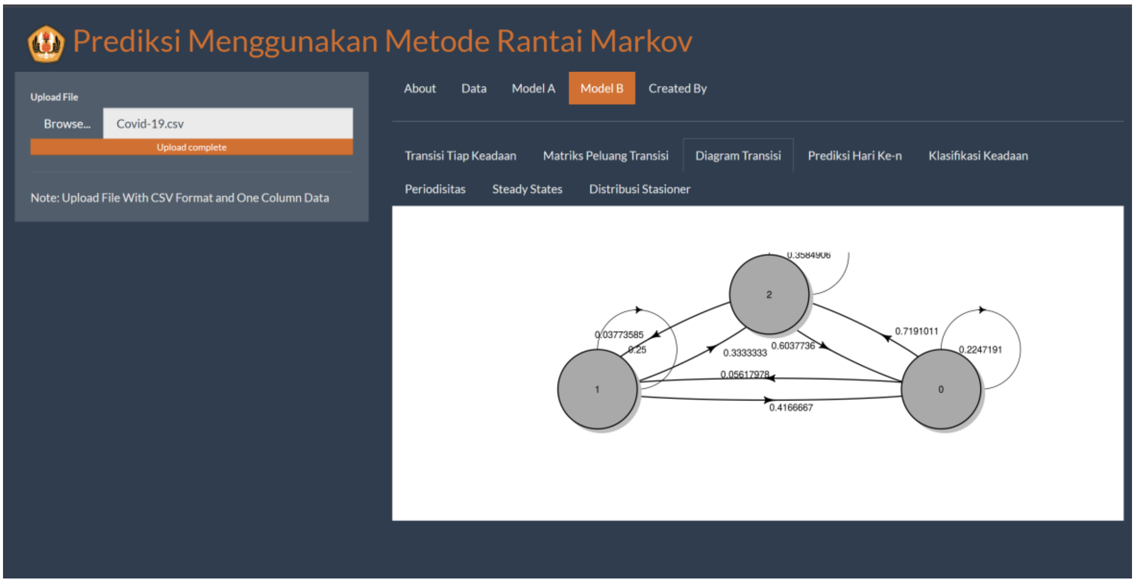


Figure 8. Web Application View for Diagram Transition

Then, using an n-step distribution, compute the probability of the next three days. Based on the data, it is known the initial distribution, it can be predicted the distribution of the next three days using **Equation (5)** as follows:

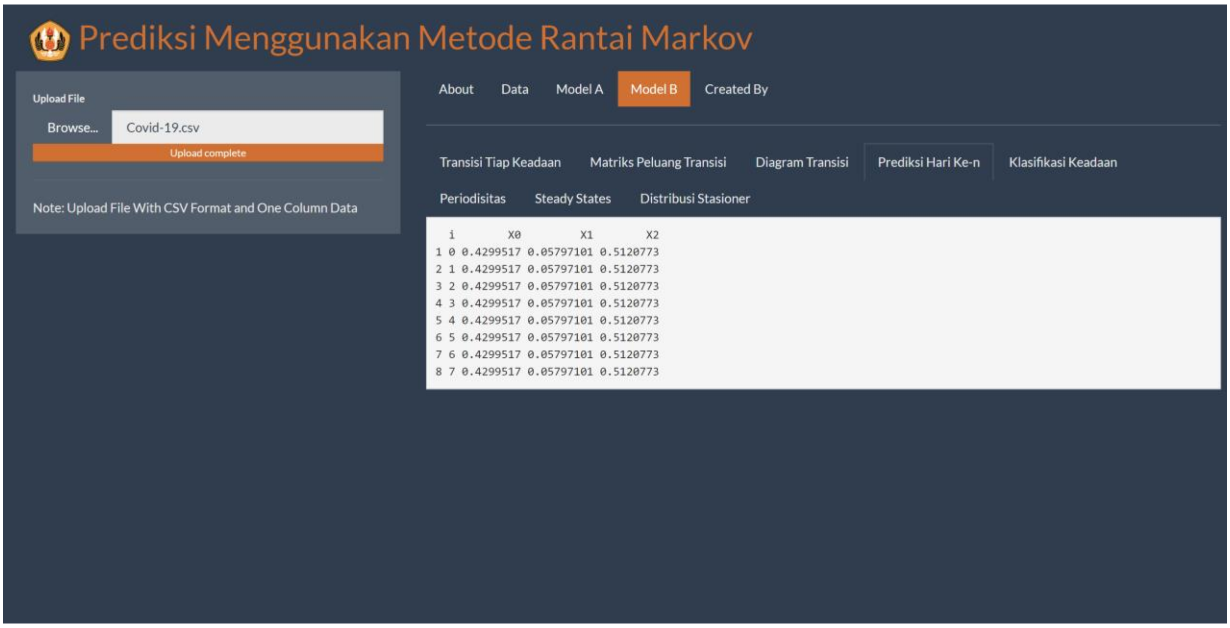
$$\pi(0) = [0.4299517 \quad 0.0579710 \quad 0.5120773]$$

$$\pi(3) = \pi(0)\mathbf{P}^3$$

$$= [0.4299517 \quad 0.0579710 \quad 0.5120773]\begin{bmatrix} 0.4011158 & 0.0585278 & 0.5403564 \\ 0.4340815 & 0.0660012 & 0.4999173 \\ 0.4536955 & 0.0565944 & 0.4897101 \end{bmatrix} \tag{14}$$

$$= [0.4299517 \quad 0.0579710 \quad 0.5120773]$$

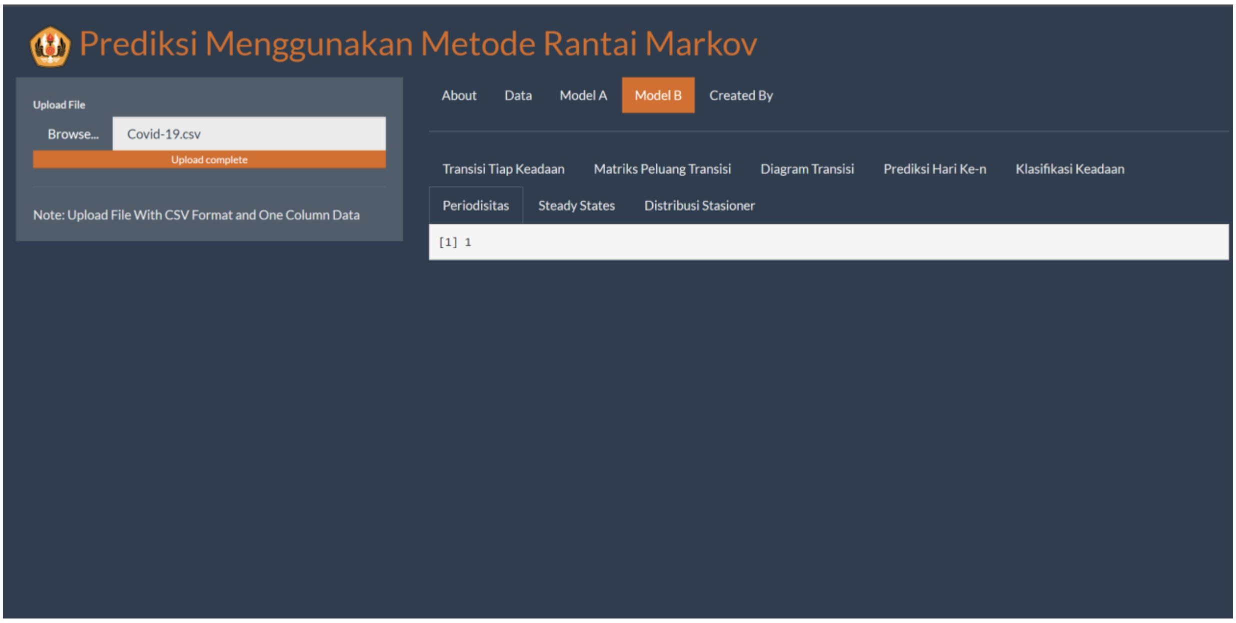
Calculations using a web application can be seen in **Figure 9**.



```
  i        X0         X1        X2
1 0 0.4299517 0.05797101 0.5120773
2 1 0.4299517 0.05797101 0.5120773
3 2 0.4299517 0.05797101 0.5120773
4 3 0.4299517 0.05797101 0.5120773
5 4 0.4299517 0.05797101 0.5120773
6 5 0.4299517 0.05797101 0.5120773
7 6 0.4299517 0.05797101 0.5120773
8 7 0.4299517 0.05797101 0.5120773
```

Figure 9. The Display of Web Application for Day-n Prediction

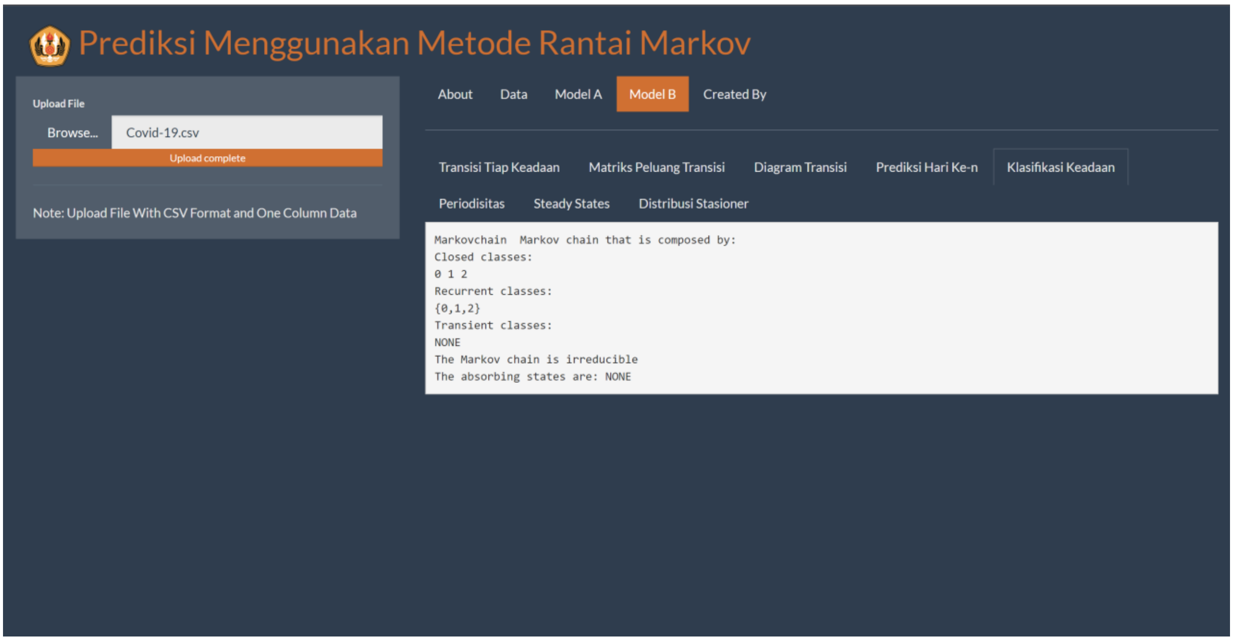
This section examines the Markov chain's state classification. **Figure 10** depicts the outcome of the computation using the web application. The result is 1, implying that $d(0) = d(1) = d(2) = 1$. As a result, states 0, 1, and 2 are aperiodic.



```
[1] 1
```

Figure 10. The Display of Web Application for Periodicity

Figure 11 shows the output of the web application. The result shows that the Markov chain is recurrent with communication class {0, 1, 2} and it's irreducible.

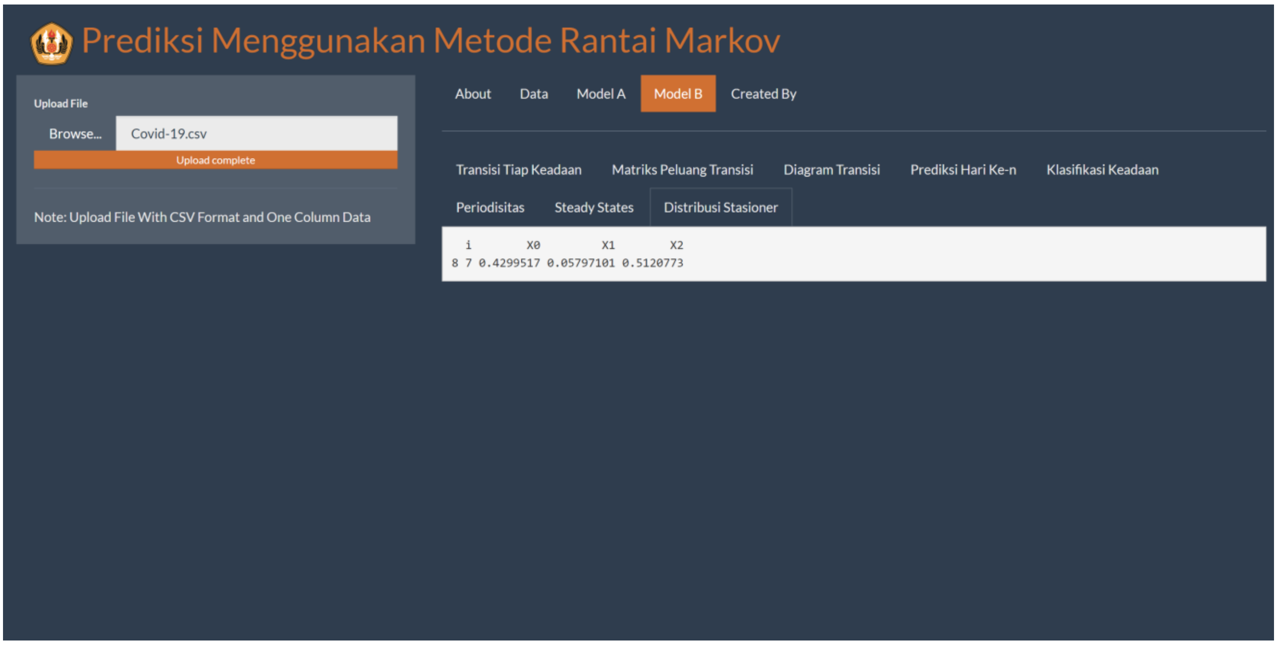


```
Markovchain  Markov chain that is composed by:
Closed classes:
0 1 2
Recurrent classes:
{0,1,2}
Transient classes:
NONE
The Markov chain is irreducible
The absorbing states are: NONE
```

Figure 11. The Display of Web Application for State Classification

This Markov chain is considered an ergodic Markov chain because it is irreducible, positive recurrent, and aperiodic, as determined by studying its state classification. Equation (11) can be used to calculate the long-term probability using the stationary distribution. The following is the long-term probability:

$$\begin{aligned}\pi &= [\pi_0, \pi_1, \pi_2] \\ &= [0.4299517, \quad 0.05797101, \quad 0.5120773]\end{aligned} \tag{15}$$



```
  i        X0         X1        X2
8 7 0.4299517 0.05797101 0.5120773
```

Figure 12. Web Application View for Stationery Distribution

Figure 12 shows the calculation results from the web application for the stationary distribution. In the long-term, the probability of people testing positive for COVID-19 in West Sumatra in State 0 is 43%, state 1 is 5.8%, and state 2 is 51.2%. State 2 has the most considerable probability, which is an increasing state. This means that the number of people testing positive for COVID-19 in West Sumatra today is higher than the previous day.

An ergodic Markov chain refers to a system that will reach a stationary distribution in the long-term, where the probability for each state will remain constant. This stationary distribution reflects the long-term equilibrium in the transition between states. In this case, the stationary distribution probabilities indicate the relative comparison of COVID-19 cases in three states in West Sumatra.

The calculation results and depiction in **Figure 12** provided a more precise knowledge of the long-term probability of switching between states in this system via the web application. It explains the switching tendency between states and the long-term state of the system represented by this ergodic Markov chain.

4. CONCLUSIONS

Based on the results of the previous chapter, a web application that integrates with a Markov chain can be built. This web application has been published and can be found at https://putrimonika.shinyapps.io/rantaimarkov/. Other researchers can use this application to process data using Markov chains quickly, automatically, and accurately. This online Markov chain integration application is helpful for data analysis, especially during state transitions. The program uses the Markov chain method to predict future events based on historical data, such as daily COVID-19 cases in West Sumatra. Forecast results can provide insight into potential future trends. The results of this integrated web application illustrate probability in the example of COVID-19 data in West Sumatra. The probability of an increase in COVID-19 cases is 51.2%, indicating a potential increase in the future. There is a 43% chance that fewer cases will occur, suggesting that the number of COVID-19 cases may decline. There is a 5.8% chance that the number of cases has not changed, meaning the number of cases may remain stable. These findings are useful for scholars and practitioners interested in understanding the changing dynamics of COVID-19 cases in West Sumatra. This web application allows users to understand state transitions and potential changes quickly. In short, this application facilitates the application of Markov chain methods in data analysis and prediction based on transition patterns discovered in historical data.

ACKNOWLEDGMENT

The authors are grateful to Universitas Padjadjaran's Rector and the Directorate of Research and Community Service (DRPM) for providing research funding support through the Padjadjaran Excellence Fastrack Scholarship (BUPP) and the Academic Leadership Grant (ALG) with contract number 1549/UN6.3.1/PT.00/2023. This research also part of RISE_SMA project which is funded by European Union year 2019-2024.

REFERENCES

[1] M. Ciotti, M. Ciccozzi, A. Terrinoni, W. C. Jiang, C. Bin Wang, and S. Bernardini, “The COVID-19 pandemic,” *Crit Rev Clin Lab Sci*, pp. 365–388, 2020, doi: 10.1080/10408363.2020.1783198.

[2] S. J. Daniel, “Education and the COVID-19 pandemic,” *Prospects (Paris)*, vol. 49, no. 1–2, pp. 91–96, Oct. 2020, doi: 10.1007/s11125-020-09464-3.

[3] N. A. Putri, A. E. Putra, and R. Mariko, “Hubungan usia, Jenis kelamin dan gejala dengan kejadian COVID-19 di Sumatera Barat,” *Majalah Kedokteran Andalas*, vol. 44, no. 2, pp. 104–111, 2021, [Online]. Available: http://jurnalmka.fk.unand.ac.id

[4] K. L. Chung, *Markov chains.* New York: Springer-Verlag, 1967.

[5] S. M. Ross, *Introduction to probability models*. Academic press, 2014.

[6] S. Osaki, *Applied stochastic system modeling*. Springer Science & Business Media, 2012.

[7] N. Privault, *Understanding markov chains. Examples and Applications*. Singapore: Publisher Springer-Verlag, 2013.

[8] L. Zeng *et al.*, “Markov-chain-based probabilistic approach to optimize sensor network against deliberately released pollutants in buildings with ventilation systems,” *Build Environ*, vol. 168, no. 106534, 2020.

[9] K. J. Doubleday and J. N. Esunge, “Application of Markov chains to stock trends,” *ournal of Mathematics and Statistics*, vol. 7, no. 2, pp. 103–106, 2011.

[10] F. Kachapova, “Representing markov chains with transition diagrams,” *J Math Stat*, vol. 9, no. 3, pp. 149–154, May 2013, doi: 10.3844/jmssp.2013.149.154.

[11] P. Studi Matematika, “PERAMALAN INDEKS PRESTASI MAHASISWA MENGGUNAKAN RANTAI MARKOV (STUDI KASUS: INDEKS PRESTASI MAHASISWA,” *Jurnal Matematika UNAND*, vol. 4, no. 3, pp. 105–111, 2019.

[12] F. N. Masuku, Y. A. R. Langi, and C. Mongi, “ANALISIS RANTAI MARKOV UNTUK MEMPREDIKSI PERPINDAHAN KONSUMEN MASKAPAI PENERBANGAN RUTE MANADO-JAKARTA,” *Jurnal Ilmiah Sains*, vol. 18, no. 2, 2018.

[13] A. Azizah, R. Welastika, A. N. Falah, B. N. Ruchjana, and A. S. Abdullah, “An Application of Markov Chain for Predicting Rainfall Data at West Java using Data Mining Approach,” in *IOP Conference Series: Earth and Environmental Science*, Institute of Physics Publishing, Aug. 2019. doi: 10.1088/1755-1315/303/1/012026.

[14] K. Aritonang *et al.*, “Analisis Pertambahan Pasien COVID-19 di Indonesia Menggunakan Metode Rantai Markov,” *Jurnal Rekayasa Sistem Industri*, vol. 9, no. 2, pp. 69–76, Jul. 2020, doi: 10.26593/jrsi.v9i2.3998.69-76.

[15] A. Luqmanul Hakim Achmad, B. Nurani Ruchjana, S. Supian, and C. Author, “Stationary Distribution Markov Chain for COVID-19 Pandemic,” *International Journal of Global Operations Research*, vol. 1, no. 2, pp. 71–74, 2020, [Online]. Available: http://www.iorajournal.org/index.php/ijgor/index

[16] Suhartono, *Analisis Data Statistik dengan R*. Surabaya: Lab. Komputasi Statistik, ITS, 2008.

[17] P. U. Gio and A. R. Effendie, *Belajar Bahasa Pemrograman R*. 2018.

[18] C. Beeley, *Web application development with R using Shiny*. Packt Publishing Ltd, 2016.

[19] G. Alfredo Spedicato, T. Seung Kang, S. Bhargav Yalamanchi, D. Yadav, and I. Cordón, “The markovchain Package: A Package for Easily Handling Discrete Markov Chains in R.” [Online]. Available: www.linhi.com

[20] N. Satyahadewi and H. Perdana, “Web Application Development for Inferential Statistics using R Shiny,” 2021.