

tubes pemstok

Kemas Veriandra Ramadhan

2025-11-18

```
# 1. Load & urutkan data
```

```
data <- read.csv("data_pemstok.csv", stringsAsFactors = FALSE)
colnames(data) <- c("tanggal", "waktu", "hari_ke", "jumlah_orang", "keterangan")
```

```
# Gabungkan tanggal + waktu
```

```
data$datetime <- paste(data$tanggal, data$waktu)
data$datetime <- as.POSIXct(data$datetime, format = "%m-%d-%Y %H:%M")
```

```
# Sort berdasarkan hari_ke dan datetime
```

```
data <- data[order(data$hari_ke, data$datetime), ]
```

```
data
```

| ## | tanggal | waktu | hari_ke | jumlah_orang | keterangan | datetime |
|-------|------------|-------|---------|--------------|------------|---------------------|
| ## 1 | 11-12-2025 | 17:02 | 1 | 43 | NA | 2025-11-12 17:02:00 |
| ## 2 | 11-12-2025 | 17:12 | 1 | 54 | NA | 2025-11-12 17:12:00 |
| ## 3 | 11-12-2025 | 17:22 | 1 | 46 | NA | 2025-11-12 17:22:00 |
| ## 4 | 11-12-2025 | 17:32 | 1 | 55 | NA | 2025-11-12 17:32:00 |
| ## 5 | 11-12-2025 | 17:42 | 1 | 31 | NA | 2025-11-12 17:42:00 |
| ## 6 | 11-12-2025 | 17:52 | 1 | 18 | NA | 2025-11-12 17:52:00 |
| ## 7 | 11-13-2025 | 17:02 | 2 | 35 | NA | 2025-11-13 17:02:00 |
| ## 8 | 11-13-2025 | 17:12 | 2 | 49 | NA | 2025-11-13 17:12:00 |
| ## 9 | 11-13-2025 | 17:22 | 2 | 65 | NA | 2025-11-13 17:22:00 |
| ## 10 | 11-13-2025 | 17:32 | 2 | 48 | NA | 2025-11-13 17:32:00 |
| ## 11 | 11-13-2025 | 17:42 | 2 | 34 | NA | 2025-11-13 17:42:00 |
| ## 12 | 11-13-2025 | 17:52 | 2 | 12 | NA | 2025-11-13 17:52:00 |
| ## 13 | 11-15-2025 | 17:00 | 3 | 61 | NA | 2025-11-15 17:00:00 |
| ## 14 | 11-15-2025 | 17:10 | 3 | 71 | NA | 2025-11-15 17:10:00 |
| ## 15 | 11-15-2025 | 17:20 | 3 | 63 | NA | 2025-11-15 17:20:00 |
| ## 16 | 11-15-2025 | 17:30 | 3 | 57 | NA | 2025-11-15 17:30:00 |
| ## 17 | 11-15-2025 | 17:40 | 3 | 27 | NA | 2025-11-15 17:40:00 |
| ## 18 | 11-15-2025 | 17:50 | 3 | 15 | NA | 2025-11-15 17:50:00 |

```
# 2. Buat state pakai quantile (3 state)
```

```
# hitung batas quantile
```

```
q <- quantile(data$jumlah_orang,
               probs = c(0, 1/3, 2/3, 1),
               na.rm = TRUE)
```

```
# mapping jumlah_orang -> state: Sepi / Sedang / Ramai
```

```
data$keterangan <- cut(
  data$jumlah_orang,
  breaks = q,
```

```
include.lowest = TRUE,
labels = c("Sepi", "Sedang", "Ramai")
)
```

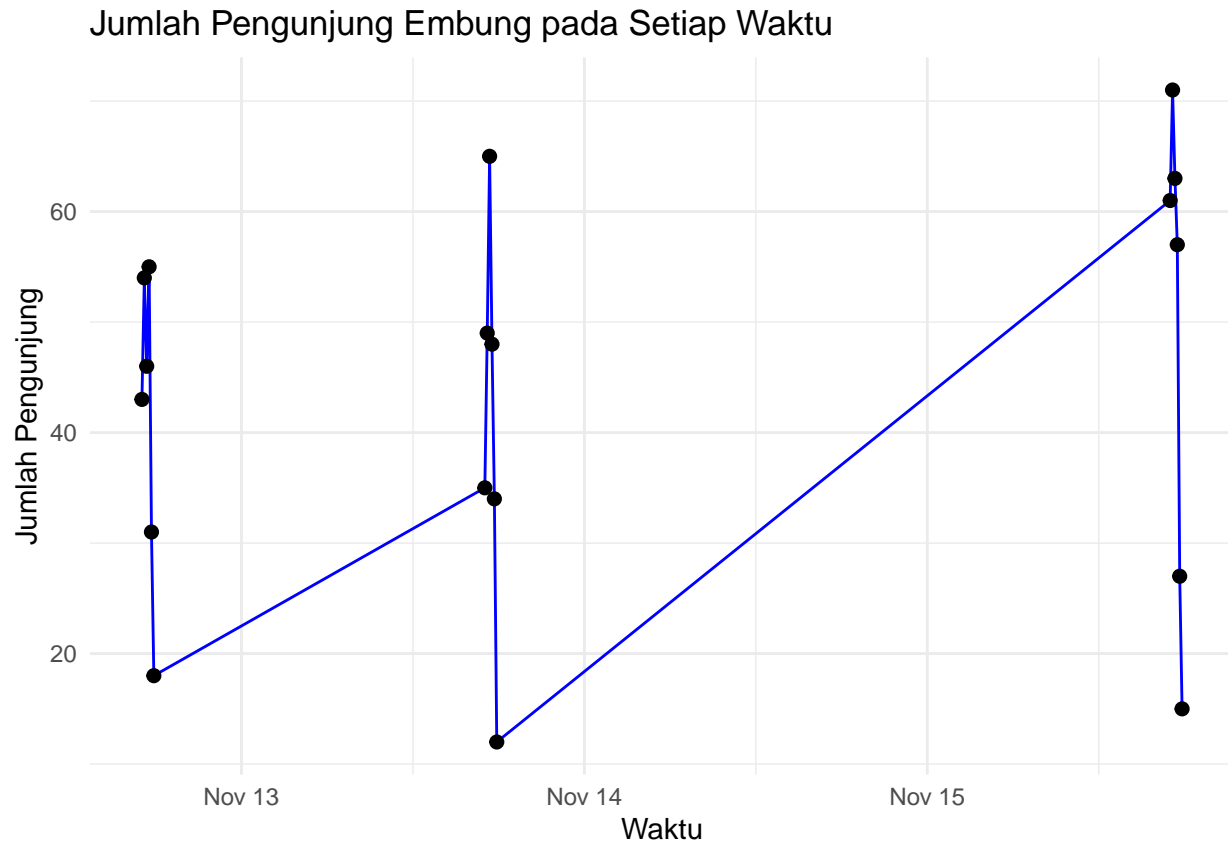
```
# lihat ringkasan state
table(data$keterangan)
```

```
##
##   Sepi Sedang Ramai
##     6     6     6
```

```
data
```

```
##      tanggal waktu hari_ke jumlah_orang keterangan      datetime
## 1  11-12-2025 17:02      1         43      Sedang 2025-11-12 17:02:00
## 2  11-12-2025 17:12      1         54      Sedang 2025-11-12 17:12:00
## 3  11-12-2025 17:22      1         46      Sedang 2025-11-12 17:22:00
## 4  11-12-2025 17:32      1         55      Ramai 2025-11-12 17:32:00
## 5  11-12-2025 17:42      1         31      Sepi 2025-11-12 17:42:00
## 6  11-12-2025 17:52      1         18      Sepi 2025-11-12 17:52:00
## 7  11-13-2025 17:02      2         35      Sedang 2025-11-13 17:02:00
## 8  11-13-2025 17:12      2         49      Sedang 2025-11-13 17:12:00
## 9  11-13-2025 17:22      2         65      Ramai 2025-11-13 17:22:00
## 10 11-13-2025 17:32      2         48      Sedang 2025-11-13 17:32:00
## 11 11-13-2025 17:42      2         34      Sepi 2025-11-13 17:42:00
## 12 11-13-2025 17:52      2          12      Sepi 2025-11-13 17:52:00
## 13 11-15-2025 17:00      3         61      Ramai 2025-11-15 17:00:00
## 14 11-15-2025 17:10      3         71      Ramai 2025-11-15 17:10:00
## 15 11-15-2025 17:20      3         63      Ramai 2025-11-15 17:20:00
## 16 11-15-2025 17:30      3         57      Ramai 2025-11-15 17:30:00
## 17 11-15-2025 17:40      3          27      Sepi 2025-11-15 17:40:00
## 18 11-15-2025 17:50      3          15      Sepi 2025-11-15 17:50:00
```

```
# grafik jumlah orang per waktu tanpa keterangan
# tambahkan titik tiap data point
library(ggplot2)
ggplot(data, aes(x = datetime, y = jumlah_orang)) +
  geom_line(color = "blue") +
  geom_point(size = 2) +
  labs(
    title = "Jumlah Pengunjung Embung pada Setiap Waktu",
    x = "Waktu",
    y = "Jumlah Pengunjung",
    color = "Keterangan"
  ) +
  theme_minimal()
```



State didefinisikan berdasarkan kategori tingkat keramaian. Pembagian dilakukan menggunakan quantile (tertile) agar setiap kategori memiliki jumlah data yang sama. Batas quantile:

| State | Rentang Jumlah Orang | Arti |
|------------------|----------------------|--------------------|
| 1. Sepi | 12 – 34 orang | Pengunjung sedikit |
| 2. Sedang | 35 – 54 orang | Keramaian normal |
| 3. Ramai | 55 – 71 orang | Pengunjung padat |

Setiap state berisi 6 observasi.

Ruang keadaan formal:

$S = \{1 \text{ (Sepi)}, 2 \text{ (Sedang)}, 3 \text{ (Ramai)}\}$

```
# 3. Susun rantai Markov (transisi antar state)
state_seq <- data$keterangan
# pasangan (state_t, state_{t+1})
trans_counts <- table(
  saat_ini = head(state_seq, -1),
  selanjutnya = tail(state_seq, -1)
)
# matriks probabilitas transisi (per baris = dari suatu state)
trans_prob <- prop.table(trans_counts, 1)

cat("Batas Quantile (Sepi/Sedang/Ramai)")
```

```
## Batas Quantile (Sepi/Sedang/Ramai)
```

```

print(q)

##          0% 33.33333% 66.66667%      100%
## 12.00000 34.66667 54.33333 71.00000

cat("Frekuensi Transisi")

## Frekuensi Transisi

print(trans_counts)

##          selanjutnya
## saat_ini Sepi Sedang Ramai
## Sepi      3      1      1
## Sedang    1      3      2
## Ramai     2      1      3

cat("Matriks Probabilitas Transisi (P)")

## Matriks Probabilitas Transisi (P)

print(round(trans_prob, 3))

##          selanjutnya
## saat_ini Sepi Sedang Ramai
## Sepi    0.600 0.200 0.200
## Sedang  0.167 0.500 0.333
## Ramai   0.333 0.167 0.500

```

| Dari → Ke | Sepi | Sedang | Ramai |
|-----------|------|--------|-------|
| Sepi | 3 | 1 | 1 |
| Sedang | 1 | 3 | 2 |
| Ramai | 2 | 1 | 3 |

Dari “Sepi”, 3 kali tetap Sepi dan masing-masing 1 kali naik ke Sedang/Ramai.

Dari “Sedang”, 3 kali tetap, 2 kali naik jadi Ramai.

Dari “Ramai”, 3 kali tetap, dan cukup sering turun (2 kali ke Sepi).

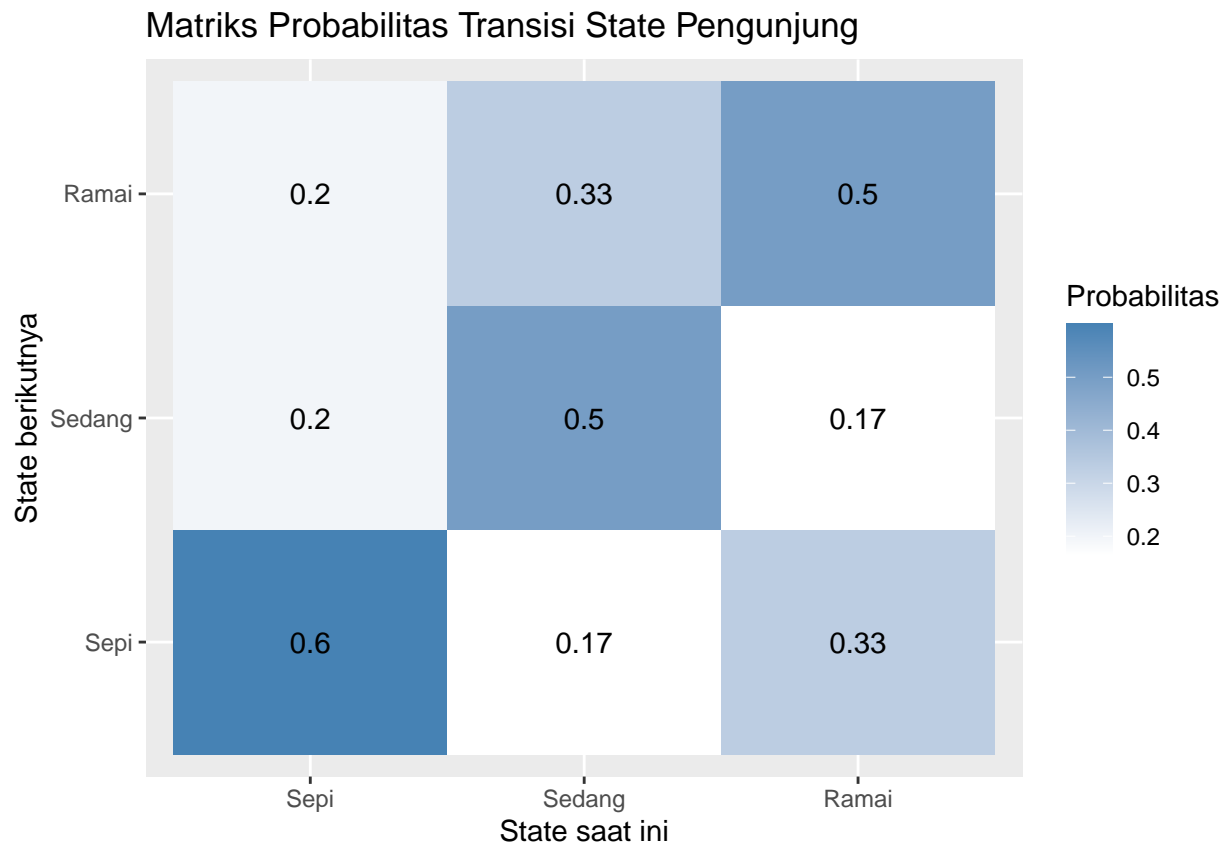
```

library(ggplot2)

df_trans <- as.data.frame(trans_prob)
colnames(df_trans) <- c("from", "to", "prob")

ggplot(df_trans, aes(x = from, y = to, fill = prob)) +
  geom_tile() +
  geom_text(aes(label = round(prob, 2))) +
  scale_fill_gradient(low = "white", high = "steelblue") +
  labs(
    title = "Matriks Probabilitas Transisi State Pengunjung",
    x = "State saat ini",
    y = "State berikutnya",
    fill = "Probabilitas"
  )

```



Jika SEKARANG Sepi

- 60% tetap sepi
- 20% naik ke Sedang
- 20% naik ke Ramai

Jika SEKARANG Sedang

- 16.7% turun
- 50% tetap
- 33.3% naik

Jika SEKARANG Ramai

- 33.3% turun ke Sepi
- 16.7% turun ke Sedang
- 50% tetap Ramai

State Sepi dan Ramai relatif stabil; sementara Sedang lebih “labil”.

```
# install.packages("igraph") # kalau belum ada
library(igraph)
```

```
##
## Attaching package: 'igraph'

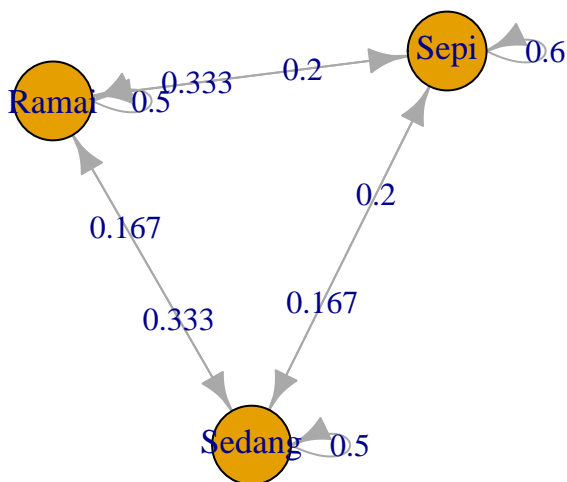
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
```

```
## The following object is masked from 'package:base':
##
##      union
# ubah ke matriks numeric untuk igraph
matP <- as.matrix(trans_prob)

g <- graph_from_adjacency_matrix(matP,
                                mode = "directed",
                                weighted = TRUE)

plot(
  g,
  edge.label = round(E(g)$weight, 3),
  vertex.size = 40,
  vertex.label.cex = 1.1,
  edge.arrow.size = 1,
  main = "Diagram Transisi Rantai Markov\nState Jumlah Pengunjung"
)
```

Diagram Transisi Rantai Markov State Jumlah Pengunjung



```
#Probabilitas Langkah Ke N
P <- as.matrix(trans_prob)

# contoh 2 langkah ke depan (20 menit)
P2 <- P %*% P

# umum: n langkah ke depan
library(expm)
```

```
## Warning: package 'expm' was built under R version 4.4.3
## Loading required package: Matrix
##
## Attaching package: 'expm'
## The following object is masked from 'package:Matrix':
```

```
##
##      expm
n <- 2
Pn <- P %^% n
Pn

##      selanjutnya
## saat_ini      Sepi      Sedang      Ramai
## Sepi  0.4600000 0.2533333 0.2866667
## Sedang 0.2944444 0.3388889 0.3666667
## Ramai  0.3944444 0.2333333 0.3722222
```

Jika saat ini Sedang, 2 langkah ke depan (20 menit):

- 25.6% Sepi
- 33.3% tetap
- 41.1% menjadi Ramai

```
#Distribusi Stasioner
P <- as.matrix(trans_prob)

eig <- eigen(t(P))
pi <- Re(eig$vectors[,1])
pi <- pi / sum(pi)
round(pi, 3)
```

```
## [1] 0.393 0.270 0.337
```

Artinya dalam jangka waktu sangat panjang:

- 39% waktu Embung dalam kondisi Sepi
- 27% waktu dalam kondisi Sedang
- 34% waktu dalam kondisi Ramai

Klasifikasi State

- Semua peluang antar state > 0
- Semua state saling terhubung (communicating)
- Tidak ada siklus tetap \rightarrow aperiodik
- State bersifat recurrent positif