

Protocol for the state-space model

1. Install “R” and “Rstudio”

<https://rstudio-education.github.io/hopr/starting.html>

2. Download R file, csv file, and stan file from our github

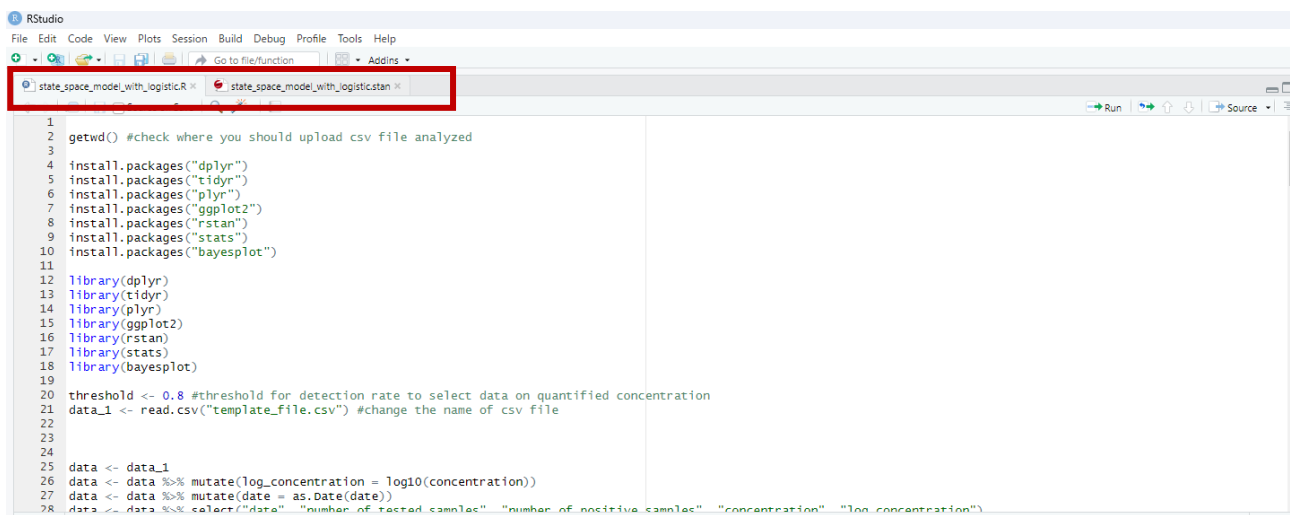
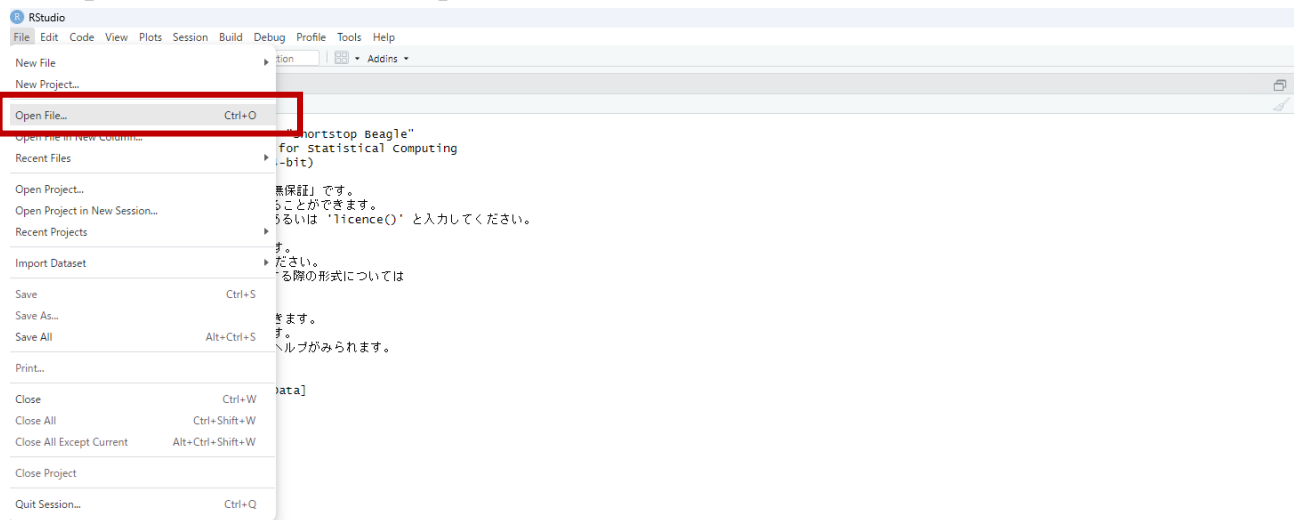
Hiroki-Ando1998 Add files via upload f750f96 · 4 minutes ago 46 Commits		
Analysis of real-world data	Update 20240823_Figure_4_RSV_2_substitution.R	2 weeks ago
Figure	Add files via upload	2 weeks ago
Raw data on wastewater concentration of IAV ...	Delete Raw data on wastewater concentration of IAV and RS...	2 weeks ago
Simulation	Update 20240809_WBE_ND_simulation_1.R	2 weeks ago
state_space_model_with_logistic.R	Update state_space_model_with_logistic.R	18 minutes ago
state_space_model_with_logistic.stan	Add files via upload	18 minutes ago
template_example_file.xlsx	Add files via upload	4 minutes ago
template_file.csv	Add files via upload	18 minutes ago

```
WBE_censored_data / state_space_model_with_logistic.R
Hiroki-Ando1998 Update state_space_model_with_logistic.R
Code Blame 182 lines (138 loc) · 6.05 KB Code 55% faster with GitHub Copilot
1
2 getwd() #check where you should upload csv file analyzed
3
4 install.packages("dplyr")
5 install.packages("tidyr")
6 install.packages("plyr")
7 install.packages("ggplot2")
8 install.packages("rtree")
9 install.packages("state")
10 install.packages("bayesplot")
11
12 library(dplyr)
13 library(tidyr)
14 library(plyr)
15 library(ggplot2)
16 library(rtree)
17 library(state)
18 library(bayesplot)
19
20 threshold <- 0.8 #threshold for detection rate to select data on quantified concentration
21 data_1 <- read.csv("template_file.csv") #change the name of csv file
22
23
24
25 data <- data_1
26 data <- data %>% mutate(log_concentration = log10(concentration))
27 data <- data %>% mutate(date = as.Date(date))
28 data <- data %>% select("date", "number_of_tested_samples", "number_of_positive_samples", "concentration", "log_concentration")
29 colnames(data) <- c("date", "count", "positive", "concentration")
30
```

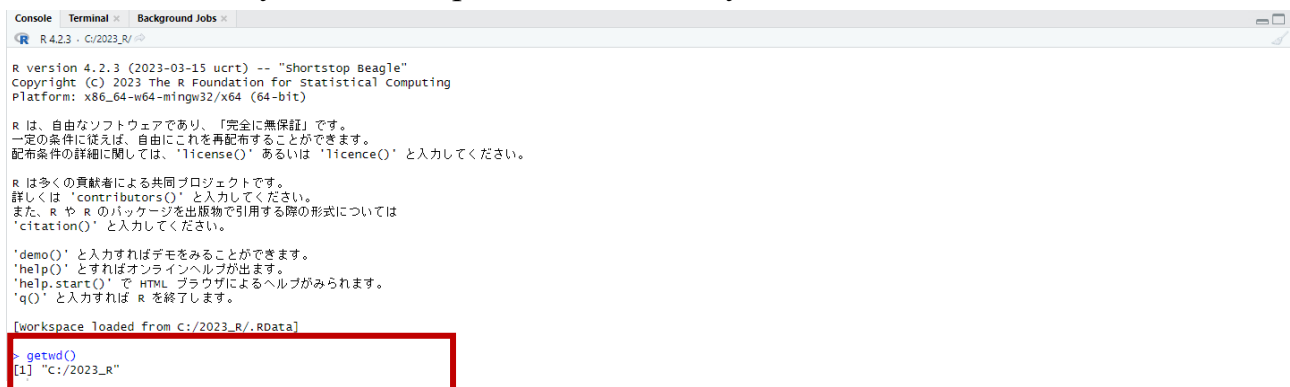
3. Input your wastewater data in the provided csv file and change the name of csv file

A	B	C	D	E
date	number_of_tested_sa	number_of_positive_samples	concentration	
2022/1/1				
2022/1/2				
2022/1/3	2	2	30955.34439	
2022/1/4	3	3	24419.80553	
2022/1/5				
2022/1/6	NA	NA	NA	
2022/1/7	NA	NA	NA	
2022/1/8	2	1	1334.963692	
2022/1/9	5	3	1842.572202	
2022/1/10	5	3	2632.292098	
2022/1/11	5	3	3171.828039	
2022/1/12	4	3	2149.002467	
2022/1/13	5	2	4058.253557	
2022/1/14	4	1	13460.97495	

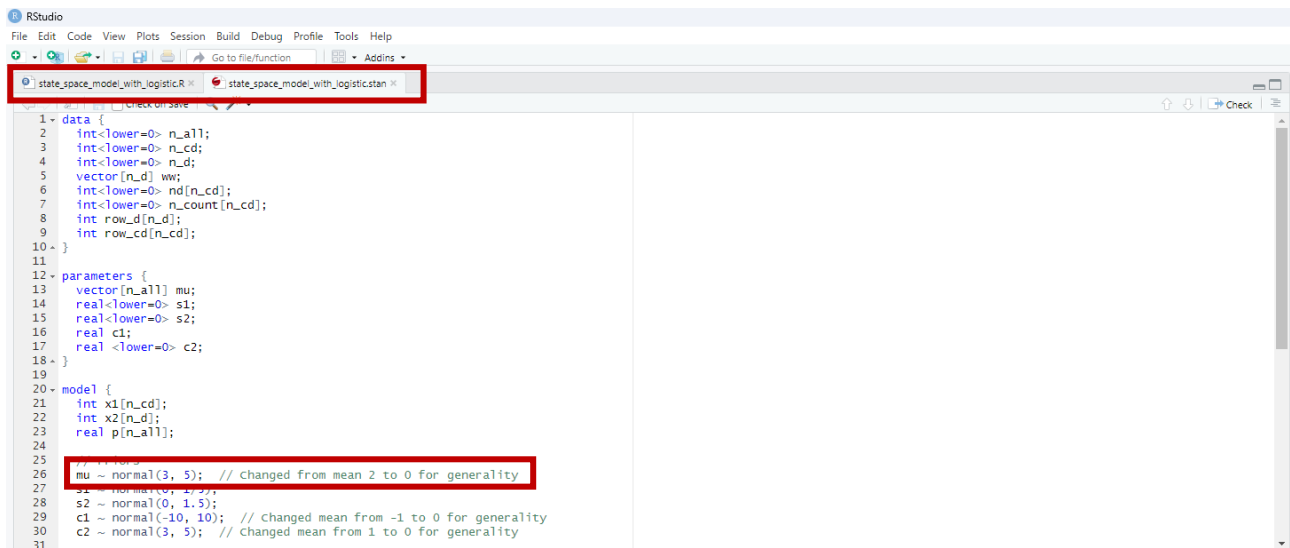
4. Open Rstudio and then, open R file and stan file



5. Check where you should upload csv file analyzed



6. Change the prior distribution of wastewater concentration (i.e., “mu”) in the Stan file

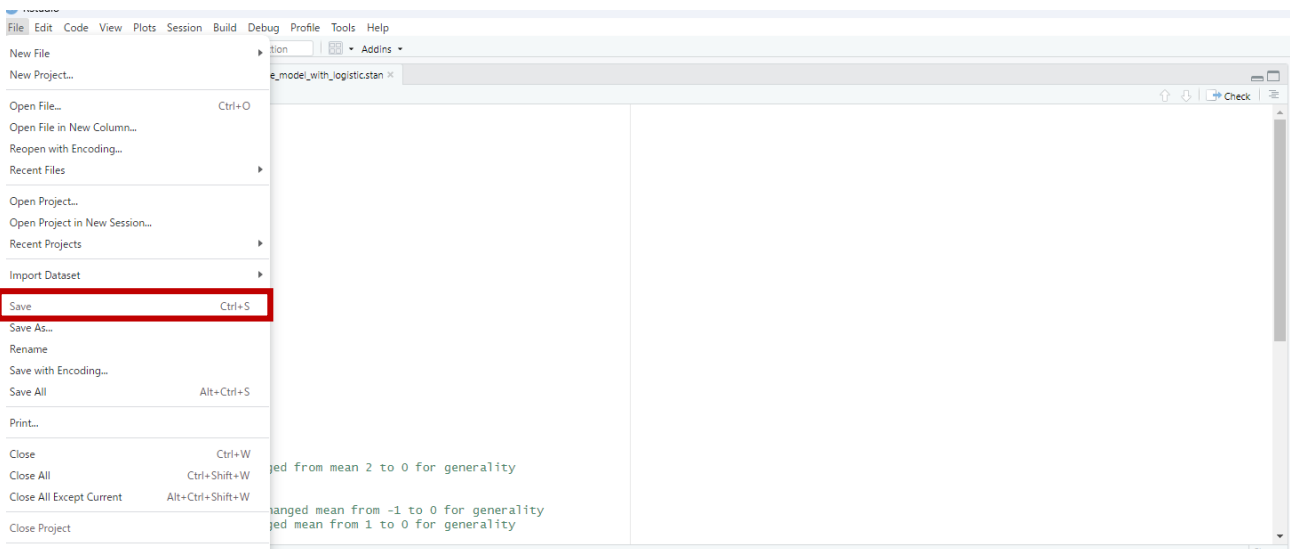


The screenshot shows the RStudio interface with a Stan file open. The file is named `state_space_model_with_logistic.stan`. The code defines a Stan model with data, parameters, and a model block. A red box highlights the line `mu ~ normal(3, 5); // changed from mean 2 to 0 for generality` in the `parameters` block.

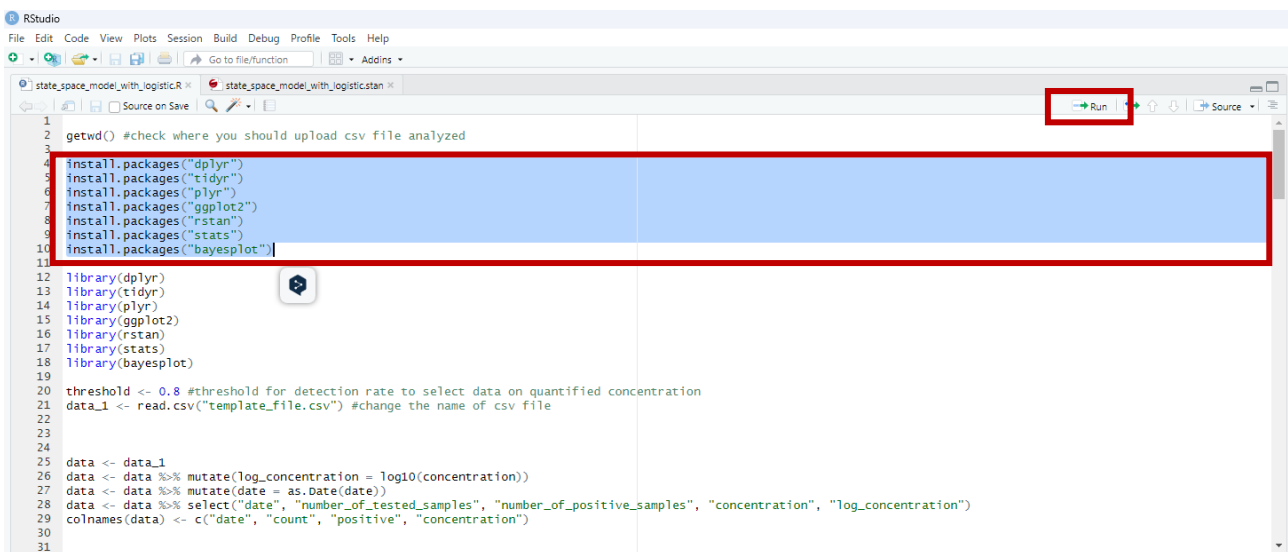
```
1 data {  
2   int<lower=0> n_all;  
3   int<lower=0> n_cd;  
4   int<lower=0> n_d;  
5   vector[n_d] ww;  
6   int<lower=0> nd[n_cd];  
7   int<lower=0> n_count[n_cd];  
8   int row_d[n_d];  
9   int row_cd[n_cd];  
10 }  
11  
12 parameters {  
13   vector[n_all] mu;  
14   real<lower=0> s1;  
15   real<lower=0> s2;  
16   real c1;  
17   real<lower=0> c2;  
18 }  
19  
20 model {  
21   int x1[n_cd];  
22   int x2[n_d];  
23   real p[n_all];  
24  
25   mu ~ normal(3, 5); // changed from mean 2 to 0 for generality  
26   s1 ~ normal(0, 1.5);  
27   s2 ~ normal(0, 1.5);  
28   c1 ~ normal(-10, 10); // changed mean from -1 to 0 for generality  
29   c2 ~ normal(3, 5); // changed mean from 1 to 0 for generality  
30 }  
31
```

Note

- `normal(a, b)`: `a` is mean and `b` is standard deviation
- As a guideline for “a”, the average of the data on log-transformed wastewater concentration (\log_{10} copies/L) might be a good choice.
- If you use viral load rather than viral concentration, the value of “a” is more than 10.
- If you changed the value, **please save the file**



7. Install R packages

A screenshot of the RStudio interface. The script editor on the left contains R code for installing and loading several packages. Lines 4 through 10, which are the installation commands, are highlighted with a blue background and enclosed in a red rectangular box. The console on the right is empty. The top toolbar shows the 'Run' button, which is also highlighted with a red box.

```
1 getwd() #check where you should upload csv file analyzed
2
3
4 install.packages("dplyr")
5 install.packages("tidyr")
6 install.packages("plyr")
7 install.packages("ggplot2")
8 install.packages("rstan")
9 install.packages("stats")
10 install.packages("bayesplot")
11
12 library(dplyr)
13 library(tidyr)
14 library(plyr)
15 library(ggplot2)
16 library(rstan)
17 library(stats)
18 library(bayesplot)
19
20 threshold <- 0.8 #threshold for detection rate to select data on quantified concentration
21 data_1 <- read.csv("template_file.csv") #change the name of csv file
22
23
24
25 data <- data_1
26 data <- data %>% mutate(log_concentration = log10(concentration))
27 data <- data %>% mutate(date = as.Date(date))
28 data <- data %>% select("date", "number_of_tested_samples", "number_of_positive_samples", "concentration", "log_concentration")
29 colnames(data) <- c("date", "count", "positive", "concentration")
30
31
```

Note

You no longer need to repeat this process once they are installed.

8. Check the threshold value and name of your csv file.

A screenshot of the RStudio interface, similar to the previous one. The script editor on the left shows the same R code. In this view, lines 20 and 21, which define the threshold and the CSV file name, are highlighted with a blue background and enclosed in a red rectangular box. The console on the right is empty. The top toolbar shows the 'Run' button.

```
1 getwd() #check where you should upload csv file analyzed
2
3
4 install.packages("dplyr")
5 install.packages("tidyr")
6 install.packages("plyr")
7 install.packages("ggplot2")
8 install.packages("rstan")
9 install.packages("stats")
10 install.packages("bayesplot")
11
12 library(dplyr)
13 library(tidyr)
14 library(plyr)
15 library(ggplot2)
16 library(rstan)
17 library(stats)
18 library(bayesplot)
19
20 threshold <- 0.8 #threshold for detection rate to select data on quantified concentration
21 data_1 <- read.csv("template_file.csv") #change the name of csv file
22
23
24
25 data <- data_1
26 data <- data %>% mutate(log_concentration = log10(concentration))
27 data <- data %>% mutate(date = as.Date(date))
28 data <- data %>% select("date", "number_of_tested_samples", "number_of_positive_samples", "concentration", "log_concentration")
29 colnames(data) <- c("date", "count", "positive", "concentration")
30
31
```

9. Run the state-space model (row from 25-87)

```

25 data <- data_1
26 data <- data %>% mutate(log_concentration = log10(concentration))
27 data <- data %>% mutate(date = as.Date(date))
28 data <- data %>% select("date", "number_of_tested_samples", "number_of_positive_samples", "concentration", "log_concentration")
29 colnames(data) <- c("date", "count", "positive", "concentration")
30
31
32 #state-space model with logistic
33 data_stan <- data
34 data_stan <- data_stan %>% mutate(positive_rate = positive/count)
35 sample_size <- nrow(data_stan)
36
37 #vector of row number used for the analysis
38 #pick row numbers for censored data
39 data_row_D <- data.frame(true = which((data_stan$positive_rate >= threshold)))
40 sample_size_D <- nrow(data_row_D)
41 #pick row numbers for censored data
42 data_row_CD <- data.frame(true = which(data_stan$count > 0))
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70 print(mcmc, pars = c("c1", "c2", "s1", "s2"), probe = c(0.025, 0.50, 0.975))
71
72 #check traceplots if you want
73 #mcmc_combo(mcmc, pars = c("c1", "c2", "s1", "s2"))
74
75
76 #MCMC samples and 95% credible intervals
77 mcmc_sample <- rstan::extract(mcmc)
78 state_name <- "mu"
79 result <- data.frame(t(apply(
80   X = mcmc_sample[[state_name]],
81   MARGIN = 2,
82   FUN = quantile,
83   probs = c(0.025, 0.5, 0.975) #credible interval can be changed
84 )))
85
86 colnames(result) <- c("low", "median", "upr")
87 data_estimated_concentration <- cbind(data, result)

```

10. Check the estimation result

```

92 #figure
93 data_fig <- data_data_estimated_concentration
94 data_fig <- data_fig %>% mutate(posi = if_else(count < 1, "no", if_else(positive >= threshold*count, "yes", "no")))
95
96 plot <- ggplot(data_fig, aes(x = as.Date(date)))
97 plot <- plot + geom_point(aes(y = concentration, color = posi, shape = posi))
98 plot <- plot + scale_shape_manual(values = c(24, 16))
99 plot <- plot + scale_color_manual(values = c("#FFC44F", "#0570B0"))
100 plot <- plot + geom_ribbon(aes(ymin = low, ymax = upr), fill = "#5990C0", alpha = 0.3)
101 plot <- plot + geom_line(aes(y = median), color = "#0570B0", size = 1)
102 plot <- plot + labs(x = "Date", y = "wastewater concentration")
103 plot <- plot + scale_x_date(limits = c(as.Date("2022-01-01"), as.Date("2023-06-01")), date_breaks = "2 months", date_labels = "%b") #change the date
104 plot <- plot + scale_y_continuous(limits = c(2.0, 5.5), breaks = seq(2.0, 5.5, 1)) #change the scale of y axis
105 plot <- plot + theme(
106   axis.line = element_line(linewidth = 1.0, lineend = "square"),
107   text = element_text(colour = "black", size = 14),
108   legend.position = "none",
109   axis.ticks = element_line(linewidth = 1.0),
110   axis.ticks.length = unit(-2, "mm"))
111 plot
112
113
114
115
116

```

Note

- Change the range of date (row 101)
- Change the scale of y axis (row 102)

11. Export the estimation result

```

113 #Export estimation result
114 #confirm the place where file should be loaded by using "getwd()": "c:/xxxx/"
115 #write file name: "2024xxx-xx-xx.csv"
116 write.csv(x = data_estimated_concentration, file = "c:/xxxx/2024xxx-xx-xx.csv")

```