EEB313 Final Project Report
Brucellosis Roundup: Wrangling Transmission with Vaccination and Culling Tactics
Group F: Hannah, Mia, Salwa, Hazel

Abstract

Brucellosis is responsible for significant global economic losses in both human health and animal production (Seleem et al., 2010). This report explores disease control strategies (vaccination and culling) for brucellosis transmission through mathematical modeling. The results indicate that selectively removing infected animals and widespread vaccination could significantly slow brucellosis spread.

Introduction

Brucellosis, a zoonotic disease, is a significant concern in the cattle industry, both in North America and globally (OIE 2009). In Canada and the United States, stringent brucellosis control measures, including rigorous testing, surveillance, and culling, have been implemented (CFIA 2018). These measures have proven particularly effective, with brucellosis now considered to have been eradicated in Canada (Seleem et al., 2010). In contrast, some regions in Sub-Saharan Africa, such as Chad, face challenges in adopting culling as a control strategy due to the essential role of cattle in livelihoods and food security. Economic and social factors in these areas make culling less feasible, resulting in persistent brucellosis issues that impact cattle health, milk production, and human health due to ongoing zoonotic transmission (Schelling 2007). This is why it is important to consider the use of vaccines as a disease control strategy.

Following our research into brucellosis, we decided to develop a mathematical model to determine the effect of culling and immunization on brucellosis rates in cattle populations. This modified Susceptible-Infected-Recovered (SIR) model aimed to answer questions about the effect of disease control strategies such as culling and vaccination on various indicators of brucellosis rates, including recovery and transmission.

Methods

Assumptions

This model simplifies brucellosis dynamics by not fixing the population size, allowing adaptable modeling for intervention exploration. Homogeneous mixing ensures equal interaction probabilities, simplifying transmission calculations. The model assumes immediate disease spread, complete immunity post-recovery, and uniform susceptibility. A constant contact rate and direct transmission streamline the analysis, while constant recovery rate and omitting external factors enhance simplicity in understanding disease progression.

Procedure

We chose to construct a mechanistic model to tease apart the dynamics of brucellosis spread in a livestock population, employing a suite of ordinary differential equations (ODE's). Initializing transmission rates, recovery rates, culling rate, and vaccination rates, the script defines a vector of parameter values. It then sets initial conditions for susceptible, infected, and recovered individuals in various scenarios, as well as parameters for effects of vaccination A, B, both of them together. The model's differential equations are encapsulated in a function, representing the rates of change for each compartment over time. The equations consider the interactions between vaccinated and unvaccinated populations, incorporating vaccination rates, recovery rates, and culling.

An R Shiny app (https://hannahflores.shinyapps.io/brucellosis_app/) was constructed to translate the model intricacies into an interactive and user-friendly interface, providing sliders for users to dynamically adjust model parameters. The "RUN" button triggers the simulation based on user inputs. The server logic includes a reactive function that dynamically updates simulation results and summary statistics based on user inputs. The app's output includes a Distribution Over Time Plot, Final Epidemic Size Table, Maximum Epidemic Size Table, and Time of Maximum Epidemic Size Table.

Results

Culling vs. No Culling

Our study started with an evaluation of the impact of culling versus no culling on brucellosis infection dynamics, without the influence of vaccination. In the initial phase, a steep ascent in infection density was observed in both scenarios, as depicted in Figure 1, with the x-axis tracking the days and the y-axis displaying the density of infections. The introduction of culling led to a more pronounced decline in infection density, highlighting its effectiveness. In contrast, the absence of culling resulted in a plateau of infection density, attributed to the lower inherent recovery rate without intervention measures.

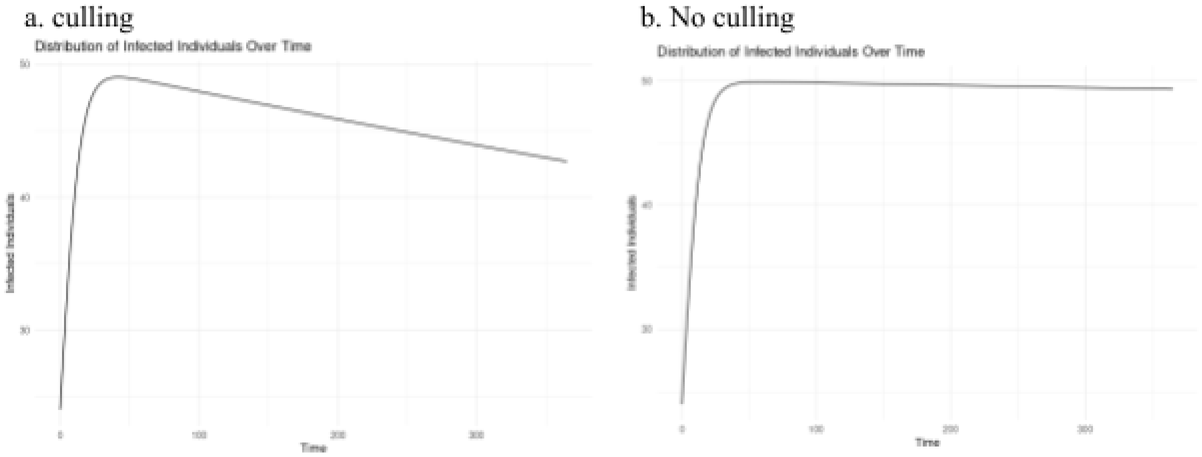
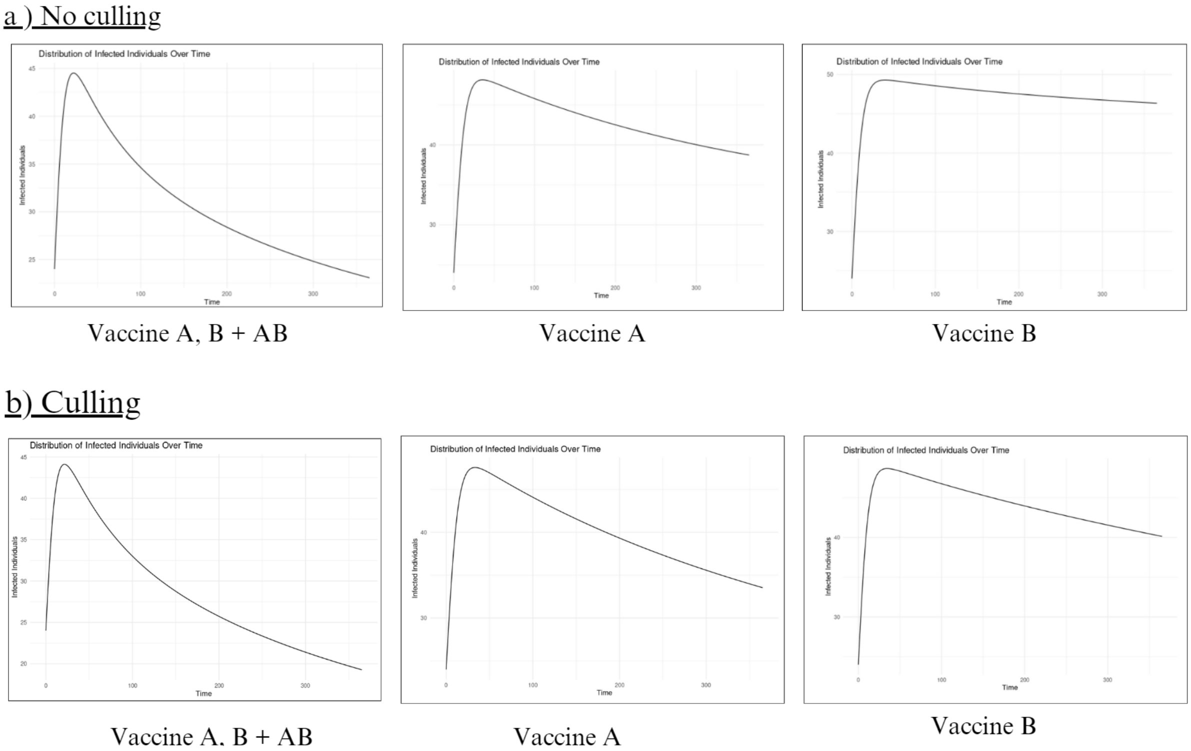


Figure 1. Number of infected individuals over time (365 days) where no vaccines are used. (a) Shows when culling was implemented. (b) Shows when culling was not implemented

Vaccination Combined with Culling and No Culling

Subsequent analysis focused on vaccination strategies integrated with culling and no culling. The implementation of Vaccines A, B alone and their combination alongside culling exhibited a uniform pattern of infection progression where the number of infected individuals would increase as susceptibles got infected and then decreased as individuals recovered and the susceptible population decreased. The integration of culling notably increased the decline in infection density no matter the vaccination strategy, as shown by a steeper decrease in infected individuals in a shorter amount of time as illustrated in Figure 2.



a) No culling

Vaccine A, B + AB

Vaccine A

Vaccine B

b) Culling

Vaccine A, B + AB

Vaccine A

Vaccine B

Figure 2. Number of infected individuals over time (365 days) for different vaccine strategies when there is no culling (a) and culling (b). Strategies are labeled on the graphs.

Comparative Efficacy of Vaccines Strategies

In our study, Vaccine A proved more effective than Vaccine B in controlling brucellosis. As shown in Figure 2, Vaccine A resulted in a quicker and steeper decrease in infection cases. This suggests that reducing the transmission rate, as Vaccine A does, is more effective in lowering infection cases than merely shortening the recovery period, which is Vaccine B's approach. When both vaccines are used and combined (strategy A, B and both) we see the greatest improvement in infection cases in the shortest amount of time as the peak, whether or not there is culling, is the lowest compared to using only vaccine A or B. Additionally, with this strategy, the decrease in infectant is quickest over time as indicated by the steep decrease in infected individuals after the peak.

Epidemic Peak Size and Maximum Epidemic Sizes

Table 1 highlights the peak sizes of the epidemic for various vaccination and culling strategies. Vaccine A generally led to a lower peak than Vaccine B, and using Vaccines A, B and a combination of both with culling showed the earliest and lowest peak on day 21. Similar patterns were seen when no culling was implemented (Table 2.), where populations using vaccine A,B and both saw the lowest peak (44 individuals) at the earliest times (22 days) and Vaccine A performed better than Vaccine B.

Between the vaccine strategies, culling or no culling did not significantly lower the peak size but rather accelerated the decline of infections, effectively flattening the curve more swiftly post-peak when culling was implemented vs. when it is not. This is shown through the lower Time of Maximum epidemic Size.

Table 1. Final Epidemic size, Maximum epidemic size and time of maximum epidemic size for the different vaccine strategies when there is culling

Condition	**Final Epidemic Size**	**Maximum Epidemic Size**	**Time of Maximum epidemic Size (Days)**
Control: No vaccine + culling	1,588,152.63	49.04	41.77
Vaccine A + culling	3,744,333.75	47.56	33.26
Vaccine B + culling	2,215,861.36	48.65	34.56
Vaccine A+B+AB + culling	2,027,703.44	44.11	21.51

Table 2. Final Epidemic size, Maximum epidemic size and time of maximum epidemic size for the different vaccine strategies when there is no culling.

Condition	**Final Epidemic Size**	**Maximum Epidemic Size**	**Time of Maximum epidemic Size (Days)**
Control: No vaccine no culling	1,588,152.63	49.88	58.72
Vaccine A + no culling	3,837,852.11	48.14	35.63
Vaccine B + no culling	2,245,682.64	49.29	39.74
Vaccine A+B+AB + no culling	2,138,178.56	44.53	22.08

Discussion

To study the transfer of infectious diseases like brucellosis and the benefits of different interventions, we created a mathematical model which expanded on the variables of the SIR model to consider the effects of vaccination and culling. Our results show that vaccinations that focus on limiting transmission compared to recovery are more effective in reducing the spread of infectious disease. Additionally, a vaccination strategy that incorporates a vaccine that reduces transmission rates, promotes recovery and one that does both was the most effective at decreasing the number of infected individuals. We also conclude that culling was more effective at removing infected individuals from the population compared to no culling.

The homogeneous mixing assumption does not account for the varied contact rates among cattle or groupings based on age or sex. Additionally, environmental factors which can significantly influence brucellosis dynamics, were not considered. The large final epidemic size indicated by the model suggests the possibility of an overestimation of the infection's spread due to duplicate counting as infected individuals already counted in the final epidemic size of one strategy (ex. Infected + Vaccine A) are recounted when vaccinated and transferred to another strategy (ex. Infected + Vaccine A gets vaccinated with Vaccine B), highlighting a need for further validation of the method by which final epidemic size is calculated.

Conclusion

In conclusion, the model's insights aligned with our expectations, highlighting the most effective method against brucellosis transmission as a combination of vaccine A, vaccine B, culling, and the simultaneous use of both vaccine A and B. Our model is especially suitable for stable herds with no new entries or exits. Therefore, it is crucial to incorporate factors such as differential susceptibility among cows, the potential for vaccine failure, and alternative transmission pathways to improve accuracy. It is important to consider that in regions of high prevalence, the vaccination of susceptible hosts and culling of infected animals may be the only viable way to control and eradicate brucellosis (Seleem et al., 2010).

References

CFIA (Canadian Food Inspection Agency). 2018. Bovine Brucellosis Surveillance Program.

OIE (World Organisation for Animal Health). 2009. Brucella abortus RB51 vaccine. Manual of Diagnostic Tests and Vaccines for Terrestrial Animals.

Schelling, E., Diguimbaye, C., Daoud, S., Nicolet, J., Boerlin, P., Tanner, M., & Zinsstag, J. 2007. Brucellosis and Q-fever seroprevalences of nomadic pastoralists and their livestock in Chad.

Seleem, M. N., Boyle, S. M., & Sriranganathan, N. (2010). Brucellosis: A re-emerging zoonosis. *Veterinary Microbiology*, *140*(3-4), 392–398. https://doi.org/10.1016/j.vetmic.2009.06.021

Brucellosis Roundup

Group F

Parameters

beta_0 - Transmission rate of unvaccinated
beta_A - Transmission rate of A vaccine
beta_B - Transmission rate of B vaccine
beta_AB - Transmission rate of AB vaccine

gamma_0 - Recovery rate of unvaccinated
gamma_A - Recovery rate for vaccine A
gamma_B - Recovery rate for vaccine B
gamma_AB - Recovery rate for both vaccines

c - Rate of culling

V_A - vaccine rate of vaccine A
V_B - vaccine rate of vaccine B
V_AB - rate of getting vaccinated by both A and B

Differential Equations

S - Unvaccinated and susceptible S_A - vaccinated with A and susceptible S_B - vaccinated with B and susceptible S_AB - vaccinated with AB and susceptible

I - Infected and unvaccianted I_A - Infected and vaccinated with A I_B - Infected and vaccinated with B I_AB - Infected and vaccinated with AB

R - Recovery of unvaccinated individuals R_A - Recovery of vaccinated A individuals R_B - Recovery of vaccinated A individuals R_AB - Recovery of vaccinated A individuals

Equations

Susceptible

$$dS/dt = -S * (\beta_A * I_A + \beta_B * I_B + \beta_A B * I_{AB} + \beta_0 * I + V_A + V_B + V_{AB})$$

Breakdown:
$-S\beta_A * I_A$ -> Infection by individual with vaccine A
$-S\beta_B * I_B$ -> Infection by individual with vaccine B
$-S\beta_A B * I_A B$ -> Infection by individual with vaccine AB
$-S\beta_0 * I$ -> Infection by individual that's unvaccinated
$-S * V_A$ -> Remove individuals that get vaccinated with vaccine A

$-S * V_A$ -> Remove individuals that get vaccinated with vaccine B
$-S * V_A B$ -> Remove individuals that get vaccinated with both vaccine A & B

This logic is similar to dS_A/dt, dS_B/dt, $dS_A B$ Only differences will be explained

$$dS_A/dt = -S_A * (\beta_A * I_A + \beta_B * I_B + \beta_A B * I_{AB} + \beta_0 * I + V_B) + S * V_A$$

$-S_A * V_B$ -> Remove individuals that get vaccine B (they move into the S_AB population)
$S * V_A$ -> Add unvaccinated individuals that get vaccinated with vaccine A

$$dS_B/dt = -S_B * (\beta_A * I_A + \beta_B * I_B + \beta_A B * I_{AB} + \beta_0 * I + V_A) + S * V_B$$

$-S_B * V_A$ -> Remove individuals that get vaccine A (they move into the S_AB population)
$S * V_B$ -> Add unvaccinated individuals that get vaccinated with vaccine B

$$dS_{AB}/dt = -S_{AB} * (\beta_A * I_A + \beta_B * I_B + \beta_{AB} * I_{AB} + \beta_0 * I) + S * V_{AB} + S_A * V_B + S_B * V_A$$

$S * V_{AB}$ -> Add unvaccinated individuals that get both vaccines at the same time
$S_A * V_B$ -> Add individuals already with vaccine A that get vaccine B
$S_B * V_A$ -> Add individuals already with vaccine B that get vaccine B

Infected

$$dI = S * (\beta_A * I_A + \beta_B * I_B + \beta_A B * I_{AB} + \beta_0 * I) - I * (c + \gamma_0 + V_A + V_B + V_{AB})$$

Breakdown
$S * \beta_A * I_A$ - Add susceptible individuals that get infected by individuals with vaccine A
$S * \beta_B * I_B$ - Add susceptible individuals that get infected by individuals with vaccine B
$S * \beta_B * I_{AB}$ - Add susceptible individuals that get infected by individuals with vaccine A&B
$S * \beta_0 * I$ - Add susceptible individuals that get infected by unvaccinated individuals

$-I * c$ - remove individuals that are culled
$-I * \gamma_0$ - remove individuals that recover
$-I * V_A$ - remove unvax individuals that are infected and get vaccine A
$-I * V_B$ - remove unvax individuals that are infected and get vaccine B
$-I * V_{AB}$ - remove individuals that are infected and get vaccine A&B

$$dI_A < -S_A * (\beta_A * I_A + \beta_B * I_B + \beta_A B * I_{AB} + \beta_0 * I) - I_A * (c + \gamma_A + V_B) + I * V_A$$

$I * V_A$ - Add uninfected, infected individuals that get vaccine A

$$dI_B < -S_B * (\beta_A * I_A + \beta_B * I_B + \beta_{AB} * I_{AB} + \beta_0 * I) - I_B * (\gamma_B + c + V_A) + I * V_B$$

$I * V_A$ - Add uninfected, infected individuals that get vaccine B

$$dI_{AB} < -S_{AB} * (\beta_A * I_A + \beta_B * I_B + \beta_{AB} * I_{AB} + \beta_0 * I) - I_{AB} * (\gamma_{AB} + c) + I * V_{AB} + I_A * V_B + I_B * V_A$$

$I * V_{AB}$ - Add uninfected, infected individuals that get vaccine A&B
$I_A * V_B$ - Add vaccine A, infected individuals that get vaccine B
$I_B * V_A$ - Add vaccine B, infected individuals that get vaccine A

Recovery

$$\begin{aligned} dR &= I * \gamma_0 \\ dR_A &= I_A * \gamma_A \\ dR_B &= I_B * \gamma_B \\ dR_{AB} &= I_{AB} * \gamma_{AB} \end{aligned}$$

$I * \gamma_0$ - unvaccinated individuals that recover
$I_A * \gamma_A$ - vaccinated A individuals that recover
$I_B * \gamma_B$ - vaccianted B individuals that recover
$I_A B * \gamma_A B$ - vaccinated AB individuals that recover

###Code

```r
library(shiny)
library(deSolve)
library(ggplot2)

# define parameters
beta_0 <- 0.0006  # Unvaccinated transmission rate
beta_A <- 0.0002 # Vaccinated A transmission rate
beta_B <- 0.0003  # Vaccinated B transmission rate
beta_AB <- 0.0001 # Transmission rate of individual with AB vaccine

gamma_0 <- 0.0001 # Unvaccinated recovery rate
gamma_A <- 0.0003 # Recovery rate for vaccine A
gamma_B <- 0.0002   # Recovery rate for vaccine B
gamma_AB <- 0.0005  # Recovery rate for both vaccines

c <- 0.0004 # Rate of culling

V_A <- 0.0001 # vaccine rate of vaccine A
V_B <- 0.0001 # vaccine rate of vaccine B
V_AB <- 0.0001 # rate of getting vaccinated by both A and B


#### Model ####

# put parameter values into vector params
params <- c(beta_0 = beta_0, beta_A = beta_A, beta_B = beta_B, beta_AB = beta_AB,
            gamma_0 = gamma_0, gamma_A = gamma_A, gamma_B = gamma_B, gamma_AB = gamma_AB,
            c = c, V_A = V_A, V_B = V_B, V_AB = V_AB)
```

```
# define initial conditions
initial_S <- 10
initial_I <- 5
initial_R <- 1
initial_S_A <- 10
initial_S_B <- 10
initial_S_AB <- 10
initial_I_A <- 2
initial_I_B <- 2
initial_I_AB <- 1
initial_R_A <- 1
initial_R_B <- 1
initial_R_AB <-1

state <- c(S = initial_S, S_A = initial_S_A, S_B = initial_S_B, S_AB = initial_S_AB,
           I = initial_I, I_A = initial_I_A, I_B = initial_I_B, I_AB = initial_I_AB,
           R = initial_R, R_A = initial_R_A, R_B = initial_R_B, R_AB = initial_R_AB)

# define time
times <- seq(0, 365, 0.001)

model <- function(time, state, params) {
  with(as.list(c(state, params)), {
    # Model equations

    # Suseceptible
    dS <- -S * (beta_A * I_A + beta_B * I_B + beta_AB * I_AB + beta_0 * I + V_A + V_B + V_AB)
    dS_A <- -S_A * (beta_A * I_A + beta_B * I_B + beta_AB * I_AB + beta_0 * I + V_B) + S * V_A
    dS_B <- -S_B * (beta_A * I_A + beta_B * I_B + beta_AB * I_AB + beta_0 * I + V_A) + S * V_B
    dS_AB <- -S_AB * (beta_A * I_A + beta_B * I_B + beta_AB * I_AB + beta_0 * I) + S * V_AB + S_A * V_B

    # Infected
    dI <- S * (beta_A * I_A + beta_B * I_B + beta_AB * I_AB + beta_0 * I) - I * (c + gamma_0 + V_A + V_
    dI_A <- S_A * (beta_A * I_A + beta_B * I_B + beta_AB * I_AB + beta_0 * I) - I_A * (gamma_A + c + V_
    dI_B <- S_B * (beta_A * I_A + beta_B * I_B + beta_AB * I_AB + beta_0 * I) - I_B * (gamma_B + c + V_
    dI_AB <- S_AB * (beta_A * I_A + beta_B * I_B + beta_AB * I_AB + beta_0 * I) - I_AB * (gamma_AB + c)

    # Recovery
    dR <- I * gamma_0 - (V_A + V_B + V_AB) * R
    dR_A <- I_A * gamma_A - (V_B + V_AB) * R_A + V_A * R
    dR_B <- I_B * gamma_B - V_A * R_B + V_B * R
    dR_AB <- I_AB * gamma_AB + V_AB * R + V_A * R_B + V_B * R_A

    return(list(c(dS, dS_A, dS_B, dS_AB, dI, dI_A, dI_B, dI_AB, dR, dR_A, dR_B, dR_AB)))
  })
}

# UI setup
ui <- navbarPage("Brucellosis Mechanistic Model",
  tabPanel("Model Dashboard",
    fluidPage(
      fluidRow(
        sidebarLayout(
          sidebarPanel(
            wellPanel(
              tags$small(
                actionButton("runButton", "RUN", size = "large"),
                br(),
                br(),
                sliderInput("time", "Time", min = 0, max = 365, value = 50),
                selectInput("strategy", "Strategy",
                            choices = c(
                              "no_culling_no_vaccine",
                              "culling_no_vaccine",
                              "vaccine_A_no_culling",
                              "vaccine_A_culling",
                              "vaccine_B_no_culling",
                              "vaccine_B_culling",
                              "vaccine_AB_no_culling",
                              "vaccine_AB_culling"
                            ),
                            selected = "no_culling_no_vaccine"
                ),
                sliderInput("beta_0", "Unvaccinated Transmission Rate", min = 0, max = 1, value = 0.0000
                sliderInput("beta_A", "Vaccinated A Transmission Rate", min = 0, max = 1, value = 0.0002
                sliderInput("beta_B", "Vaccinated B Transmission Rate", min = 0, max = 1, value = 0.0003
                sliderInput("beta_AB", "Transmission Rate for AB Vaccine", min = 0, max = 1, value = 0.0
                sliderInput("gamma_0", "Unvaccinated Recovery Rate", min = 0, max = 1, value = 0.0001, s
                sliderInput("gamma_A", "Recovery Rate for Vaccine A", min = 0, max = 1, value = 0.0003,
                sliderInput("gamma_B", "Recovery Rate for Vaccine B", min = 0, max = 1, value = 0.0002,
                sliderInput("gamma_AB", "Recovery Rate for Both Vaccines", min = 0, max = 1, value = 0.0
                sliderInput("c", "Rate of Culling", min = 0, max = 1, value = 0.0004, step = .0001),
                sliderInput("V_A", "Vaccine Rate for A", min = 0, max = 1, value = 0.0001, step = .0001)
                sliderInput("V_B", "Vaccine Rate for B", min = 0, max = 1, value = 0.0001, step = .0001)
                sliderInput("V_AB", "Rate of Getting Vaccinated by Both A and B", min = 0, max = 1, valu
              )
            )
          ),
          mainPanel(
            tabsetPanel(
              tabPanel("Distribution Over Time", plotOutput("plot")),
              tabPanel("Final Epidemic Size", tableOutput("finalEpidemicSize")),
              tabPanel("Maximum Epidemic Size", tableOutput("maxEpidemicSize")),
              tabPanel("Time of Maximum Epidemic Size", tableOutput("timeMaxEpidemicSize"))
            )
          )
        )
      )
    )
  )
)


# Server setup
server <- function(input, output) {
  # Reactive function to update the distribution plot and summary tables
```

```
reactive_data <- reactive({
  params <- c(
    beta_0 = input$beta_0,
    beta_A = input$beta_A,
    beta_B = input$beta_B,
    beta_AB = input$beta_AB,
    gamma_0 = input$gamma_0,
    gamma_A = input$gamma_A,
    gamma_B = input$gamma_B,
    gamma_AB = input$gamma_AB,
    c = input$c,
    V_A = input$V_A,
    V_B = input$V_B,
    V_AB = input$V_AB
  )

  state <- c(
    S = 10,
    S_A = 5,
    S_B = 1,
    S_AB = 10,
    I = 10,
    I_A = 10,
    I_B = 2,
    I_AB = 2,
    R = 1,
    R_A = 1,
    R_B = 1,
    R_AB = 1
  )

  times <- seq(0, input$time, 0.001)

  # Model simulation
  simulation_result <- ode(y = state, times = times, func = model, parms = params)

  # Create a data frame for the plot
  plot_data <- data.frame(Time = simulation_result[, "time"],
                          Infected = rowSums(simulation_result[, grep("^I", colnames(simulation_resul

  # Calculate summary statistics for final epidemic size
  final_epidemic_size <- sum(simulation_result[, grep("^R", colnames(simulation_result))])

  return(list(
    plot_data = plot_data,
    final_epidemic_size = final_epidemic_size,
    max_epidemic_size = max(plot_data$Infected),
    time_max_epidemic_size = plot_data$Time[which.max(plot_data$Infected)]
  ))
})

observeEvent(input$runButton, {
  # Trigger the simulation when the "Run" button is clicked
```

```
    output$plot <- renderPlot({
      plot_data <- reactive_data()$plot_data

      ggplot(plot_data, aes(x = Time, y = Infected)) +
        geom_line() +
        labs(title = "Distribution of Infected Individuals Over Time",
             x = "Time",
             y = "Infected Individuals") +
        theme_minimal()
    })

    output$finalEpidemicSize <- renderTable({
      final_epidemic_size <- reactive_data()$final_epidemic_size
      data.frame(Statistic = "Final Epidemic Size", Value = round(final_epidemic_size, 5))
    })

    output$maxEpidemicSize <- renderTable({
      max_epidemic_size <- reactive_data()$max_epidemic_size
      data.frame(Statistic = "Maximum Epidemic Size", Value = round(max_epidemic_size, 5))
    })

    output$timeMaxEpidemicSize <- renderTable({
      time_max_epidemic_size <- reactive_data()$time_max_epidemic_size
      data.frame(Statistic = "Time of Maximum Epidemic Size", Value = round(time_max_epidemic_size, 5))
    })

  })
}

# Run the Shiny app
#shinyApp(ui = ui, server = server)
```