

# EEB313 - Project Trial 2

**ctrl/cmd+F “CODE” to go straight to the code** It's a mess I recommend going to “Visual” mode for non-code stuff

## 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 unvaccinated 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 B individuals R\_AB - Recovery of vaccinated AB individuals

## Equations

Let us know if we missed any factors or need to remove some!

## Susceptible

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

Breakdown:

$-S\beta_A * I_A$  -> Infection by individual with vaccine A

- $-S\beta_B * I_B \rightarrow$  Infection by individual with vaccine B
- $-S\beta_{AB} * I_{AB} \rightarrow$  Infection by individual with vaccine AB
- $-S\beta_0 * I \rightarrow$  Infection by individual that's unvaccinated
- $-S * V_A \rightarrow$  Remove individuals that get vaccinated with vaccine A
- $-S * V_B \rightarrow$  Remove individuals that get vaccinated with vaccine B
- $-S * V_{AB} \rightarrow$  Remove individuals that get vaccinated with both vaccine A & B

*This logic is similar to  $dS_A/dt$ ,  $dS_B/dt$ ,  $dS_{AB}$  Only differences will be explained*

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

- $-S_A * V_B \rightarrow$  Remove individuals that get vaccine B (they move into the  $S_{AB}$  population)
- $S * V_A \rightarrow$  Add unvaccinated individuals that get vaccinated with vaccine A

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

- $-S_B * V_A \rightarrow$  Remove individuals that get vaccine A (they move into the  $S_{AB}$  population)
- $S * V_B \rightarrow$  Add unvaccinated individuals that get vaccinated with vaccine B

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

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

## Infected

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

### 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
- $S * \beta_{AB} * I_{AB}$  - 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

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

- $I * V_A$  - 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

$$dR = I * \gamma_0$$

$$dR_A = I_A * \gamma_A$$

$$dR_B = I_B * \gamma_B$$

$$dR_{AB} = I_{AB} * \gamma_{AB}$$

$I * \gamma_0$  - unvaccinated individuals that recover

$I_A * \gamma_A$  - vaccinated A individuals that recover

$I_B * \gamma_B$  - vaccinated B individuals that recover

$I_{AB} * \gamma_{AB}$  - vaccinated AB individuals that recover

## CODE

```
## Loading required package: deSolve
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
```

```
## v dplyr      1.1.3      v readr      2.1.4
```

```
## v forcats    1.0.0      v stringr    1.5.0
```

```
## v ggplot2    3.4.3      v tibble     3.2.1
```

```
## v lubridate  1.9.2      v tidyr      1.3.0
```

```
## v purrr      1.0.2
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
```

```
## x dplyr::lag()     masks stats::lag()
```

```
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
#### Set up ####
```

```
# define parameters
```

```
beta_0 <- 0.2 # Unvaccinated transmission rate
```

```
beta_A <- 0.1 # Vaccinated A transmission rate
```

```
beta_B <- 0.2 # Vaccinated B transmission rate
```

```
beta_AB <- 0.1 # Transmission rate of individual with AB vaccine
```

```
gamma_0 <- 0.015 # Baseline recovery rate
```

```
gamma_A <- 0.015 # Recovery rate for vaccine A
```

```
gamma_B <- 0.3   # Recovery rate for vaccine B
```

```
gamma_AB <- 0.3  # Recovery rate for both vaccines
```

```

p_culling <- 0.1 # Probability of culling infected individuals
c <- 0.1 # Rate of culling

V_A <- 0.2 # vaccine rate of vaccine A
V_B <- 0.2 # vaccine rate of vaccine B
V_AB <- 0.04 # 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)

initial_S <- 90
initial_I <- 10
initial_R <- 0
initial_S_A <- 0
initial_S_B <- 0
initial_S_AB <- 0
initial_I_A <- 0
initial_I_B <- 0
initial_I_AB <- 0
initial_R_A <- 0
initial_R_B <- 0
initial_R_AB <- 0

# define initial conditions
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 times to save
times <- seq(0, 15, 1)

# define the model!
modified_sir <- function(time, state, params){
  with(as.list(c(state,params)),{

    # 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

    # 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_B + V_AB)
    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_B) +
      I*V_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
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

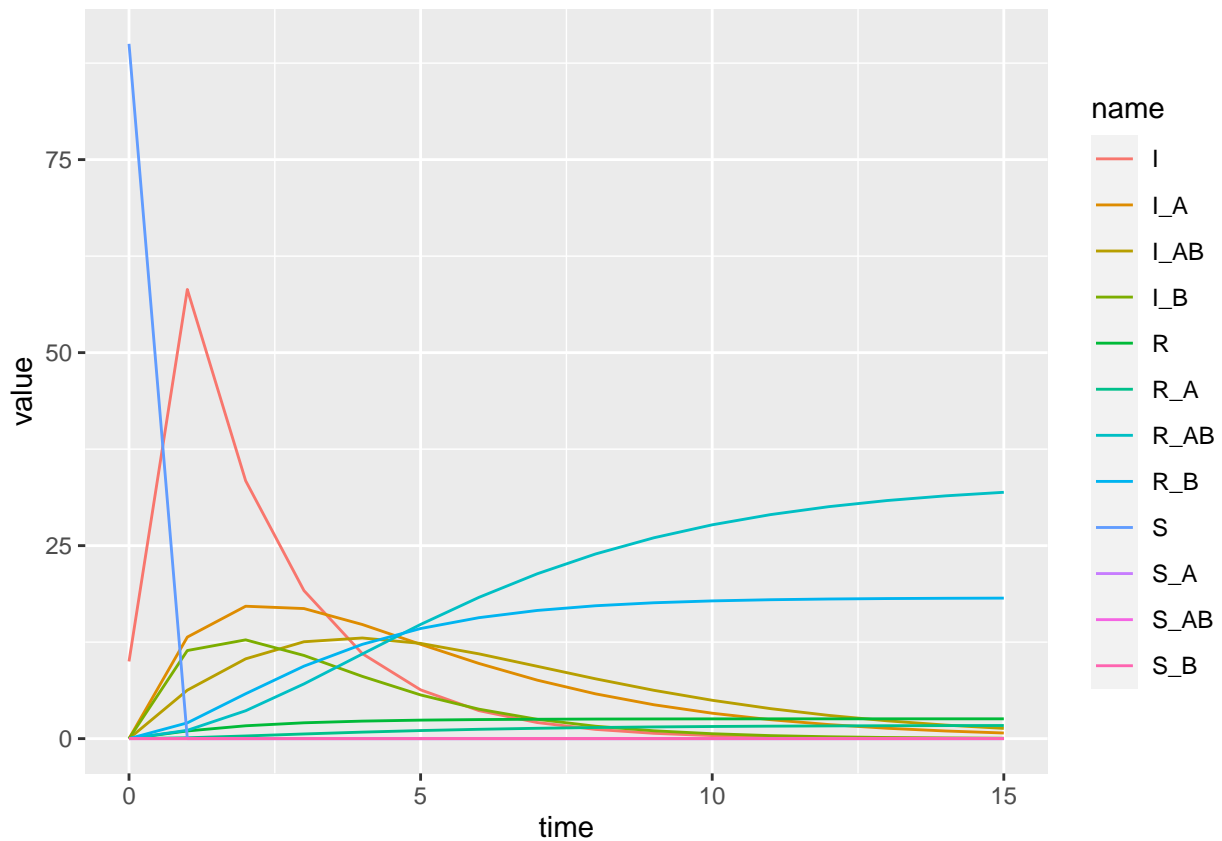
# Recovery
dR <- I*gamma_0
dR_A <- I_A*gamma_A
dR_B <- I_B*gamma_B
dR_AB <- I_AB*gamma_AB

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

# numerically integrate equations!
out <- as.data.frame(ode(state, times, modified_sir, params))

out %>% pivot_longer(! time) %>%
  ggplot(aes(x = time, y = value, color = name)) +
  geom_line()

```



Next Steps:

- verify the code is working

- Verify the accuracy of the code
- Make it proportional to whole population somehow
- Get interpretable results