EMD 538

Hanyu Wang

```
load("/Users/macbook/Desktop/finalexamitis2025.RData")
```

Q1

1A. What is R0 for this epidemic, as measured from the growth rate over the first 7 days of the epidemic? (8 pts)

```
# Let's set f and v
D <- 1.5   # days
L <- 0     # onset of infectiousness is instantaneous
v <- L + D # duration of the serial interval
f <- L / v # ratio of the latent period to the serial interval
```

```
# Fit a Poisson regression to the log of the cumulative number of cases
t <- 1:7
CumCasesE <- cumsum(finalexamitis2025$ObsCases[1:7])
mod <- glm(CumCasesE ~ t, family = poisson)
# The default link of poisson is log link

# Extract the growth rate, r
r <- coef(mod)["t"]
r # 0.1569798
```

```
##         t
## 0.1569798
```

```
# Calculate R0
R0 <- (r^2) * (1 - f) * f * (v^2) + r * v + 1
R0 # 1.23547
```

```
##       t
## 1.23547
```

Answer:

R0 for this epidemic, as measured from the growth rate over the first 7 days of the epidemic, is 1.23547.

1B. Plot the case reproductive number Rt for each day of the epidemic. What is the last day for which Rt >1? (8 pts)

```
obsV <- finalexamitis2025$serint / 24
```

```
# R has an efficient function to do so for gamma distribution!
# install.packages("MASS")
library(MASS) # Load this library: MASS
```

```r
gpars <- fitdistr(obsV, densfun = "gamma",
                  start = list("shape" = 1, "rate" = 1))[[1]]
```

```
## Warning in densfun(x, parm[1], parm[2], ...): NaNs produced
```

```
## Warning in densfun(x, parm[1], parm[2], ...): NaNs produced
```

```r
# fitdistr finds MLE for you, start = initial guesses,
# [[1]] extracts the first parameter for your estimates;
# [[2]] would give you standard errors
gpars
```

```
##     shape      rate
## 1.1615429 0.9498136
```

```r
# This way, we don't have to create nLLgamma
```

```r
# Probability density of serial interval
# dgamma() calculates the probability of infection after 1, 2, ..., 100 days
# shape and rate come from MLE of your serial interval data
g <- dgamma(1:100, shape = gpars[1], rate = gpars[2])

# Create a 100x100 matrix to store infection probabilities
# Rows = day when new case occurs (i = infected day)
# Columns = day of potential infector (j = source day)
# p[i,j] = probability that a case on day i was infected by a case on day j
p <- matrix(0, nrow = 100, ncol = 99)

# Loop over days starting from day 2 (day 1 has no previous cases)
for (i in 2:length(finalexamitis2025$ObsCases)) {
  if (finalexamitis2025$ObsCases[i] > 0) {
    # Loop over all previous days (potential infectors)
    for (j in 1:(i - 1)) {
      # Only consider previous days with actual cases
      if (finalexamitis2025$ObsCases[j] > 0) {
        # Compute p[i,j] = probability a case on day i was infected by day j
        # Numerator: g[i-j] = probability of infection after (i-j) days
        # Denominator: sum of "infectious pressure" from all previous days
        #   seq(i-1, 1, -1) generates a sequence of numbers: (i-1, i-2, ..., 1)
        #   to pick the correct serial interval probabilities in reverse
        #   REMEMBER: %*% = matrix multiplication operator, multiplies each
        #   serial interval probability by the number of cases on that day
        p[i, j] <- g[i - j] / (g[
          seq(i - 1, 1, -1)] %*% finalexamitis2025$ObsCases[1:(i - 1)])
      }
    }
  }
}
```

```r
R0 # value we will compare Rj to
```

```
##       t
## 1.23547
```

```r
Rj <- rep(NA, 30)
```

```r
# Loop over each day t
for (t in 1:length(finalexamitis2025$ObsCases)) {
  # Rj[t] = expected number of secondary cases caused by cases on day t
  Rj[t] <- p[t:length(finalexamitis2025$ObsCases), t] %*%finalexamitis2025$ObsCases[t:length(finalexamit

## the code above is
## Rj[t] <- p[t:length(finalexamitis2025$ObsCases), t] %*%
## finalexamitis2025$ObsCases[t:length(finalexamitis2025$ObsCases)]}

Rj
```

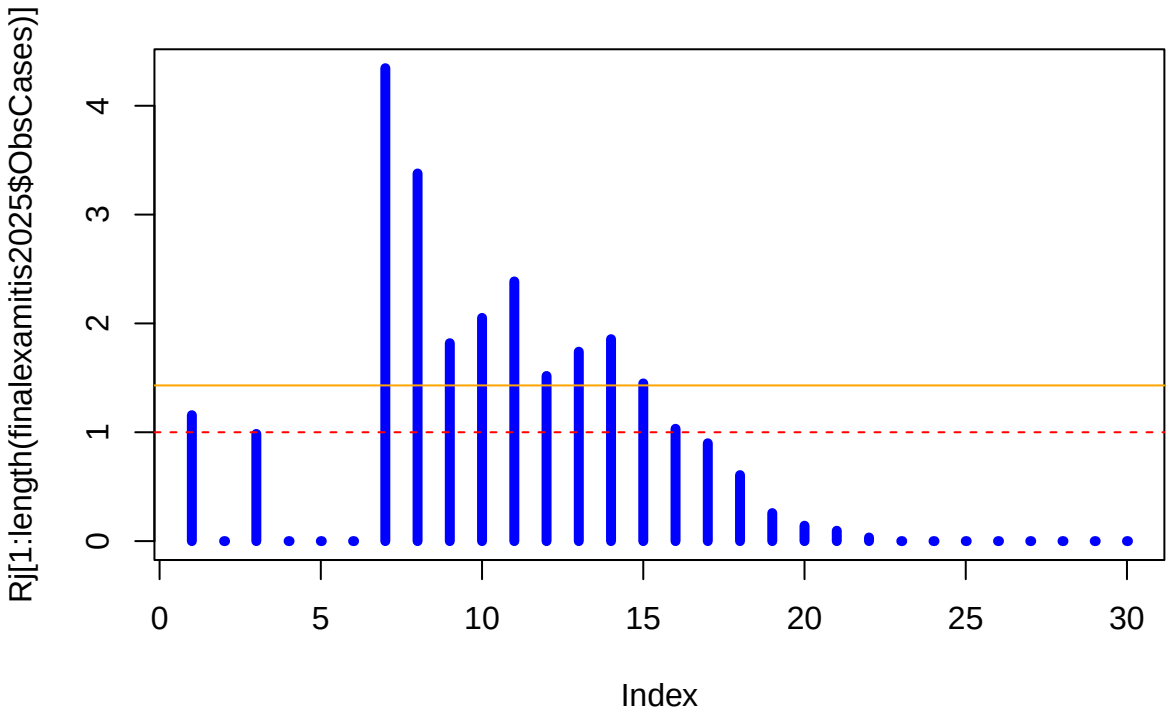
```
##  [1] 1.15702330 0.00000000 0.98463461 0.00000000 0.00000000 0.00000000
##  [7] 4.34507363 3.37765743 1.81858896 2.05076631 2.38572308 1.51721663
## [13] 1.74028610 1.85559344 1.44866043 1.03193833 0.89891825 0.60634202
## [19] 0.25755006 0.14154424 0.09533390 0.03293442 0.00000000 0.00000000
## [25] 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
```

```r
# The number of secondary cases that the people on that day caused
# (effective reproduction number on day j)

# Compute mean Rj across days with non-zero values
# (we exclude zero values, e.g., last day where no further infections can occur)
mean(Rj[1:25][Rj[1:25] > 0])
```

```
## [1] 1.430321
```

```r
# Make a plot of Rj
plot(Rj[1:length(finalexamitis2025$ObsCases)], type = "h", col = "blue", lwd = 5)
abline(h = 1, col = "red", lty = 2) # R0=1
abline(h = mean(Rj[Rj > 0]), col = "orange") # Mean
```




```r
# How does the mean value compare to the value of R0 you calculated above?
```

Answer:

The last day for which $R_t > 1$ is Day 16.

1C. What is the interpretation of the Rt value on day 12?

```
Rj[12]
```

```
## [1] 1.517217
```

Answer:

The R_t value on day 12 is 1.517217.

It means that the number of secondary infections generated by one infectious individual in day 12 in a fully susceptible population is about 1.52.

Since $R_t > 1$, it means that disease spread, epidemic can spread, but not guaranteed.

Q2

2A. Based on this information and working back from the total number of confirmed cases (sum of "ObsCases"=795), how many of the N=6740 Yale undergraduates would you estimate to have been infected with FeV (with or without symptoms)? Give a point estimate and 95% confidence interval. (Assume the probabilities of having symptoms, seeking care, being tested, and testing positive are uniformly distributed across the corresponding 95% confidence intervals, as in Reed C et al, Emerg Infect Dis, 2009 and lab 5.)

```
set.seed(123)

RepCases <- sum(finalexamitis2025$ObsCases)  # = 795

TotCases <- rep(NA, 10000)

for (i in 1:length(TotCases)){

  # Randomly sample one value from each univariate distribution
  p_symp <- runif(n=1,
                  min=0.630,
                  max=0.728)
  p_care <- runif(n=1,
                  min=0.152,
                  max=0.254)
  p_test <- runif(n=1,
                  min=0.757,
                  max=0.959)
  p_pos <- runif(n=1,
                  min=0.783,
                  max=0.975)

  # Total reporting probability
  p_report <- p_symp * p_care * p_test * p_pos
```

```
  # Work backward from confirmed cases
  TotCases[i] <- RepCases / p_report
}
```

```
quantile(TotCases, c(0.5,0.025,0.975))
```

```
##       50%      2.5%     97.5%
##  7788.120  5641.339 11018.319
```

```
#   50%      2.5%     97.5%
# 7788.120  5641.339 11018.319
```

Answer:

I estimate that approximately 7,788 Yale undergraduates have been infected with FeV (with or without symptoms)?

The 95% confidence interval is (5641.339, 11018.319).

Note that this estimate exceeds the total undergraduate population (N = 6,740), reflecting uncertainty in the reporting probabilities.

2B. Should the fact that we did not observe all infectious individuals bias our estimate of R0 from Question 1A? Why or why not?

Answer:

No, incomplete observation of all infectious individuals should not bias our estimate of R0 from Question 1A, since the reporting rate is approximately constant during the early exponential growth phase.

Q3

3A. What is the estimated crude secondary attack rate (SAR), assuming there was (at most) one index case in each dorm room and NO on-going transmission? (HINT: The index case, if there is one, is included in the "dorm.outbreak" table. Hence, the number of secondary cases is equal to the number infected minus 1, and the number of roommates is equal to the number living in the dorm room minus 1.)

```
print(finalexamitis2025$dorm.outbreak)
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##  [1,]    0    5    6    0    1    1    0    0    0
##  [2,]    2   12   13    7    3    1    0    0    0
##  [3,]    0    4   12    5    5    2    1    1    0
##  [4,]    0    0    4    3    7    5    1    0    0
##  [5,]    0    0    0    2    4    2    1    0    0
##  [6,]    0    0    0    0    2    1    2    0    0
##  [7,]    0    0    0    0    0    2    1    0    1
##  [8,]    0    0    0    0    0    0    1    0    0
##  [9,]    0    0    0    0    0    0    0    0    0
## [10,]    0    0    0    0    0    0    0    0    0
```

```
# Numerator of SAR (How many individuals were infected?)
# the number of secondary cases is equal to the number infected minus 1
```

```r
# secondary cases = infected - 1
sum(finalexamitis2025$dorm.outbreak[1,])*0 +
  sum(finalexamitis2025$dorm.outbreak[2,])*(1-1) +
  sum(finalexamitis2025$dorm.outbreak[3,])*(2-1) +
  sum(finalexamitis2025$dorm.outbreak[4,])*(3-1) +
  sum(finalexamitis2025$dorm.outbreak[5,])*(4-1) +
  sum(finalexamitis2025$dorm.outbreak[6,])*(5-1) +
  sum(finalexamitis2025$dorm.outbreak[7,])*(6-1) +
  sum(finalexamitis2025$dorm.outbreak[8,])*(7-1) +
  sum(finalexamitis2025$dorm.outbreak[9,])*(8-1) +
  sum(finalexamitis2025$dorm.outbreak[10,])*(9-1) # 143
```

```
## [1] 143
```

```r
# Denominator of SAR (How many individuals were at risk (susceptible)?)
# the number of roommates is equal to the number living in the dorm room minus 1

# susceptible contacts = room size - 1
sum(finalexamitis2025$dorm.outbreak[,1])*0 +
  sum(finalexamitis2025$dorm.outbreak[,2])*1 +
  sum(finalexamitis2025$dorm.outbreak[,3])*2 +
  sum(finalexamitis2025$dorm.outbreak[,4])*3 +
  sum(finalexamitis2025$dorm.outbreak[,5])*4 +
  sum(finalexamitis2025$dorm.outbreak[,6])*5 +
  sum(finalexamitis2025$dorm.outbreak[,7])*6 +
  sum(finalexamitis2025$dorm.outbreak[,8])*7 +
  sum(finalexamitis2025$dorm.outbreak[,9])*8 # 357
```

```
## [1] 357
```

```r
# Thus, an estimated SAR is:
143 / 357 # 0.7002801
```

```
## [1] 0.4005602
```

Answer:

The estimated crude secondary attack rate (SAR) is 0.7002801.

3B. What is the campus probability of infection (CPI) and estimated secondary attack rate (SAR) allowing for on-going transmission within dorm rooms and on campus? (HINT: You can estimate these via maximum likelihood using one of the files from lab.)

```r
longiniLL <- function(pars, data) {
  B <- pars[1] # escape probability from community
  Q <- pars[2] # escape probability from one infected HH member

  K <- dim(data)[2] # K = Maximum household size = # of columns
  m <- matrix(0, K + 1, K)
  # Rows = # of infected individuals (0 to K)
  # Columns = household size (1 to K)

  m[1, 1] <- B # Probability of 0 of 1 HH member infected has to be B
  m[2, 1] <- 1 - B # Probability of 1 of 1 HH member infected
```

```
  for (k in 2:K) { # Loop over household sizes (columns)
    m[1, k] <- B^k
    # Probability everyone in HH escapes infection from the community
    for (j in 1:k) { # Loop over possible infections (rows)
      m[j + 1, k] <- choose(k, j) * m[j + 1, j] * (B^(k - j)) * Q^(j * (k - j))
      # Probability j out of k HH members infected
    }
    m[k + 1, k] <- 1 - sum(m[, k]) # Probability everyone in HH infected
  }

  llikl <- 0
  for (k in 1:K) { # Loop over household sizes
    for (j in 0:k) { # Loop over everyone infected
      llikl <- llikl + data[j + 1, k] * log(m[j + 1, k])
      # Add to log likelihood at previous step
      # data = a_jk * log(m_jk)
    }
  }
  return(-llikl)
}
```

```
# Find optimal values of B and Q by minimizing the negative log-likelihood

BQest <- optim(c(0.8, 0.2), longiniLL, data = finalexamitis2025$dorm.outbreak)$par
BQest # 0.5472099 0.9252978
```

```
## [1] 0.5472099 0.9252978
```

```
# the estimated community probability of infection (CPI) and
#    secondary attack rate (SAR)?

CPI <- 1 - BQest[1]
SAR <- 1 - BQest[2]

CPI # 0.4527901
```

```
## [1] 0.4527901
```

```
SAR # 0.07470218
```

```
## [1] 0.07470218
```

Answer:

The campus probability of infection (CPI) is 0.4527901.

The estimated secondary attack rate (SAR) is 0.07470218.

3C. Why is crude SAR estimated in 3A greater or less than the SAR calculated in 3B? Explain. (6 pts)

Because the crude SAR estimated in 3A does not distinguish between people who were infected within dorm rooms and those got infected on campus, while the SAR calculated in 3B does.

Therefore, fewer infections are counted to secondary transmission within dorm rooms for the SAR calculated in 3B – The crude SAR estimated in 3A is biased upward, while the SAR estimated in 3B is lower.

Q4

4A. What is the average degree of this network? (5 pts)

```
# degree = number of links emanating from a single individual
d1 <- 2
d2 <- 2
d3 <- 4
d4 <- 2
d5 <- 2
d6 <- 4
```

```
average_degree <- (d1+d2+d3+d4+d5+d6)/6
average_degree
```

```
## [1] 2.666667
```

Answer:

The average degree of this network is approximately 2.67.

4B. What is the clustering coefficient for this network? (5 pts)

```
n_triangle <- 2
n_triples <- 16

# clustering coefficient = ratio of triangles to triples
clustering_coef <- (n_triangle)/n_triples
clustering_coef
```

```
## [1] 0.125
```

The clustering coefficient for this network is 0.125.

Q5

5A. What is the total effect of the AutomaticA vaccine against confirmed disease, as measured by the cumulative incidence (DE_{CI})? (5 pts)

(a) 88.8%

(b) 86.0%

(c) 85.5%

(d) 88.2%

```
ARVv_CI <- 7/465
ARVu_CI <- 48/464

DE <- 1 - ARVv_CI/ARVu_CI
DE
```

```
## [1] 0.8544803
```

Answer: (a)

5B. What is the indirect effect of the AutomaticA vaccine, as measured by the incidence rate (IE_{IR})? (5 pts)

(a) 18.7%

(b) 20.2%

(c) 31.3%

(d) 55.3%

```
ARVu_IR <- 48/13304
ARUu_IR <- 739/163350

IE <- 1- ARVu_IR/ARUu_IR
IE
```

```
## [1] 0.2024945
```

Answer: (b)

5C. What is the unbiased measure of the overall effect (OE) of the AutomaticA vaccine if protection from the vaccine is all-or-nothing? (5 pts)

(a) 85.5%

(b) 88.2%

(c) 53.5%

(d) 55.3%

Answer: (a)

$OE = 1 - \frac{ARV_{u+v}}{ARU_u}$

$DE = 1 - \frac{ARV_u}{ARU_u}$

Since the vaccine is all-or-nothing, vaccination does not modify susceptibility among those not protected. Therefore, the attack rate among vaccinated individuals is independent of exposure status.

$ARV_u = ARV_{u+v}$.

```
OE <- DE
OE # 0.8544803
```

```
## [1] 0.8544803
```

I used AI (ChatGPT) for this question. I asked it to give me a hint, and it said I should consider the assumptions of the all-or-nothing vaccine and link OE to DE.

5D. What groups would you compare to estimate the direct effect of the AutomaticA vaccine? Do you expect this to be a biased estimate? Explain. (6 pts)

Answer:

I would compare the vaccinated and unvaccinated students in Franklin and Murray Colleges.

This estimate is expected to be biased, because unvaccinated students in Franklin and Murray Colleges may benefit from indirect protection due to vaccination of others. Therefore, the risk among unvaccinated students

in Franklin and Murray Colleges is reduced, the number of confirmed cases among them is reduced, and the direct effect of vaccination is biased toward the null.

Q6

6A. What is/are the dominant period(s) of oscillation in the FLI time series, and has it changed over time? (8 pts)

```
logfe <- log(finalexamitis2025$FLI)

tmax <- length(finalexamitis2025$FLI)

# dt: time step in years
dt <- 1/52 # weekly case reports

# Calculate the absolute value of the power from FFT using a function fft()
ym <- abs(fft(logfe))
# What is abs()? Try this:
abs(-0.5); abs(-2); abs(-5000)
```

```
## [1] 0.5
```

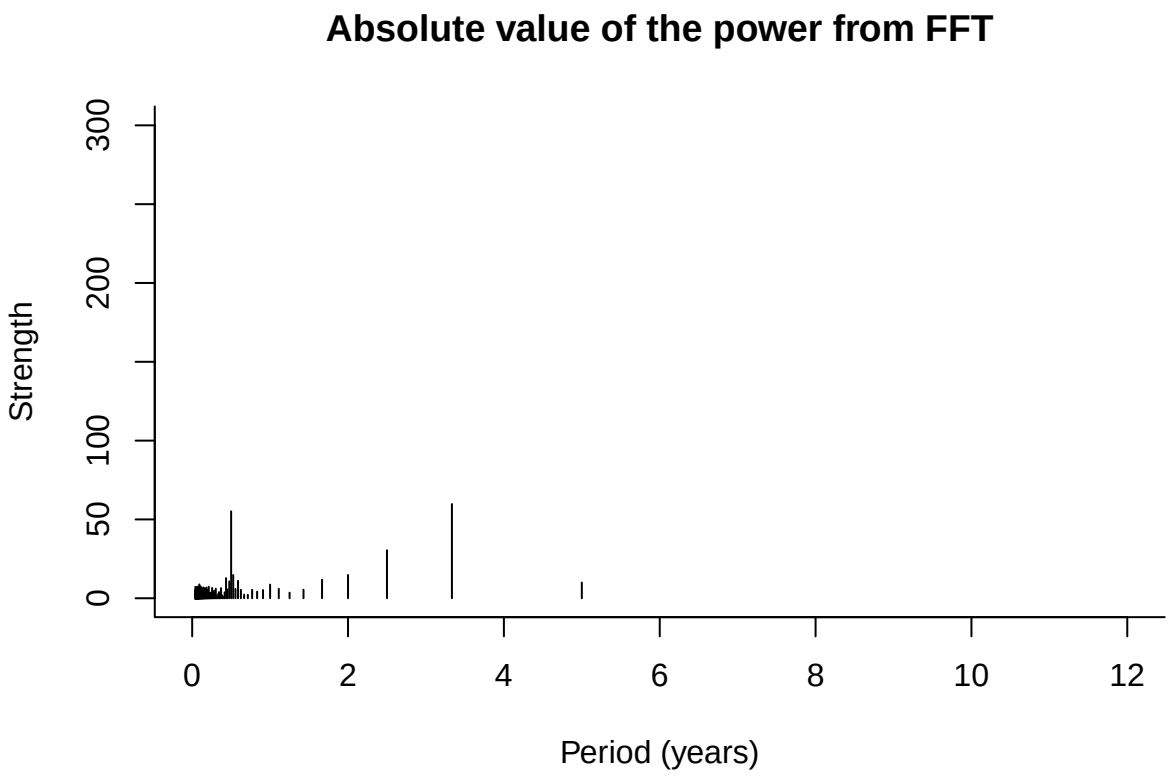
```
## [1] 2
```

```
## [1] 5000
```

```
# Frequency of oscillations (in bi-weeks)
f <- (0:(tmax-1))/tmax

# Period of oscillations (in years) = 1 / frequency
t <- dt/f

# Make a plot
plot(x=t[2:round(tmax/2)+1],
     y=ym[2:round(tmax/2)+1],
     type='h', xlim=c(0,12),
     ylim=c(0,300), bty="l",
     xlab='Period (years)', ylab='Strength',
     main="Absolute value of the power from FFT")
```



Answer:

According to the figure provided in Q6, from summer 2025 to approximately spring 2030, the dominant periods of oscillation is 6 months (biannual). The peaks occur around February and September, suggesting that outbreaks tend to happen at the start of school semesters (because of more contact).

The dominant periods do not change over time until approximately spring 2030 – the time series of the number of cases after 2030 do not show any clear periodic patterns.

Although the figure "Absolute value of the power from FFT" shows two peaks at approximately 6 months and 3.25 year, the result of 3.25 year might be due to the fact that the number of cases was higher in the autumn of 2026 and the autumn of 2029, making the three-year periods of oscillation more apparent. Since the three-year periods of oscillation was only observed once, this result only shows a small number of larger outbreaks rather than a stable oscillatory pattern.

Therefore, the dominant periods of oscillation in the FLI time series are approximately 6 months.

There is evidence that it changes over time – there is a high peak at 3.25 year.

6B. Assuming the AutomaticA vaccine was introduced in April 2030 (week 248), what was the amplitude of biannual (i.e. twice per year) peaks in FLI cases before and after vaccine introduction? (7 pts)

```
time_before <- finalexamitis2025$time[1:248]
time_after <- finalexamitis2025$time[248:length(finalexamitis2025$time)]
```

before vaccine introduction

```
# Create harmonic terms as follows:
# 6-month periods
cos6 <- cos( 4 * pi * time_before ) # same as cos( 2* pi * time/0.5)
sin6 <- sin( 4 * pi * time_before ) # same as sin( 2* pi * time/0.5)
```

```r
df <- data.frame(
  time  = time_before,
  sin6 = sin6
)

# Fit harmonic regression
mod_before <- glm(logfe[1:248] ~ cos6 + sin6 + time_before)

## my original code was:

## mod_before <- glm(logfe ~ cos6 + sin6 + time_before)

## I used AI(ChatGPT), input all my previous code and asked why this error occurs

## Error in model.frame.default(formula = logfe ~ cos6 + sin6 + time_before,  :
##  variable lengths differ (found for 'cos6')

## It reminded me that the length(logfe) does not equal to 248
```

```r
beta_sin6<-coef(mod_before)['sin6']
beta_cos6<-coef(mod_before)['cos6']

##### 6-month period #####

# Amplitude
amp6_before   <- sqrt(beta_sin6^2 + beta_cos6^2)
print(paste0("6 months period amplitude :", round(amp6_before,3)))
```

```
## [1] "6 months period amplitude :0.28"
```

```r
# Phase angle
phase6_before <- -atan(beta_sin6/beta_cos6)
print(paste0("6 months period phase angle :", round(phase6_before,3)))
```

```
## [1] "6 months period phase angle :1.454"
```

after vaccine introduction

```r
# Create harmonic terms as follows:
# 6-month periods
cos6 <- cos( 4 * pi * time_after ) # same as cos( 2* pi * time/0.5)
sin6 <- sin( 4 * pi * time_after ) # same as sin( 2* pi * time/0.5)


df <- data.frame(
  time  = time_after,
  sin6 = sin6
)

# Fit harmonic regression
mod_after <- glm(logfe[248:length(finalexamitis2025$time)] ~ cos6 + sin6 + time_after)
```

```r
beta_sin6<-coef(mod_after)['sin6']
beta_cos6<-coef(mod_after)['cos6']
```

```r
##### 6-month period #####

# Amplitude
amp6_after   <- sqrt(beta_sin6^2 + beta_cos6^2)
print(paste0("6 months period amplitude :", round(amp6_after,3)))
```

```
## [1] "6 months period amplitude :0.117"
```

```r
# Phase angle
phase6_after <- -atan(beta_sin6/beta_cos6)
print(paste0("6 months period phase angle :", round(phase6_after,3)))
```

```
## [1] "6 months period phase angle :1.339"
```

Answer:

The amplitude of biannual peaks in FLI cases before and after vaccine introduction are 0.28 and 0.117, respectively.