

2. Gün

1. Uygulama: Bölmeli modellere giriş

Önceki oturumlarda bulaşıcı hastalıkları tanımlarken bölmeli modellerin nasıl kullanılabileceğini inceledik. Ayrıca geçiş oranlarının nasıl belirleneceğini, bunun zaman dağılımlarıyla olan ilişkisini ve basit modelleme çıktısını nasıl yorumlayacağımızı da inceledik. İlk uygulamada öncelikle basit bir kohort modelinin nasıl kodlandığını göreceğiz ve ayrıca modelin temel varsayımlarının test edilmesini de inceleyeceğiz.

1. Basit bir kohort modeli

Enfekte olmuş kişilerden oluşan bir başlangıç popülasyonu ile başladığımız ve iyileşmeye geçişe izin verdiğimiz kohort modelimizi hatırlayın. Bu modelin temel yapı taşlarının ne olduğuna bakalım.

```
# Bazı faydalı paketler yükleyin:
library(deSolve) # Diferansiyel denklemleri çözmek için
library(reshape2) # Model çıktımızı değiştirmek için
library(ggplot2) # Düzgün grafikler oluşturmak için!
library(here) # Her zaman çalışma dizinimizde olduğumuzdan emin olmak için

## here() C:/Users/juano/Dropbox/Work/Consultancy/WHO/Romania/Epi-
course/Contents/WHO ID modelling/R'de başlar

# Bu basit satır Ortamı temizler böylece en başından başlarız
rm(list = ls())

# Sisteme bunun bizim çalışma dizinimiz olduğunu söyleyelim
setwd(here())

#####
# Şimdi kohort modelimizi kodlayalım

# Toplam popülasyonumuza N diyelim. N için bir parametre oluşturalım
N <- 1000

### Bazı model girdileri

# Başlangıç koşulları: Kodlamak istediğimiz kohort modelinin iki aşaması vardır I (Enfekte) ve R (iyileşmiş)

# Modelin durum değişkenlerinin bulunduğu bir vektör oluşturalım

initial_state_values <- c(
  I = N,
```

Bulaşıcı hastalık dinamiklerinin R'de modellenmesi üzerine kısa kurs

```
R = 0)

# I ve R model bölmelerine başlangıç değerleri atadığımızza dikkat edin

# Günlük parametre değerleri:

# Enfeksiyon döneminin ortalama 2 gün olduğunu artık biliyoruz
# Buna göre iyileşme oranını (gamma) şu şekilde ifade edebiliriz:

parameters <- c(gamma = 1/2)

# ZAMAN ADIMLARI:

# Bu, 0'dan 50 güne kadar günlük aralıklarla
times <- seq(from = 0, to = 50, by = 1) # zamanında modeli çözmek için
zaman adımları dizisini saklayan bir Vektördür

### Artık modeli kodlayın !

## ODE'ler
# Aşağıdaki model fonksiyonu şu girdi argümanları alır
# (yazılan sıraya göre): zaman, durum ve parametreler
# Diferansiyel denklemlerin nasıl kodlandığına dikkat edin ve notlarınızla
karşılaştırın!

cohort_model <- function(time, state, parameters) {

  with(as.list(c(state, parameters)), {

    # N toplam popülasyon boyutunun (her bir bölmedeki insan sayısının
    toplamı) hesaplanması
    N <- I+R

    # Diferansiyel denklemler
    dI <- -gamma * I
    dR <- gamma * I

    return(list(c(dI, dR)))
  })
}
```

1.1. Kohort modelini çalıştırmak

I ve R bölmeleri, bu durumlar için başlangıç koşullarıyla basit bir kohort modeli kodladık ve gama geçiş parametresini iyileşme oranını yansıtacak şekilde tanımladık.

Bulaşıcı hastalık dinamiklerinin R'de modellenmesi üzerine kısa kurs

Şimdi modeli çalıştıralım ve çıktıların bazılarını bakalım:

```
## MODEL ÇIKTISI (Ode entegrasyon algoritmasını kullanarak diferansiyel
denklemleri çözmek)

# Ode fonksiyonunu (deSolve paketinden) kullanırız ve yukarıda oluşturduğumuz
# sistemi ve parametreleri geçiririz

# Model çıktısı veri çerçevesi çıktısı1'de saklanacaktır

output1 <- as.data.frame(ode(y = initial_state_values,
                             times = times,
                             func = cohort_model,
                             parms = parameters))

# Konsola output1 yazarak çıktıya bir bakalım

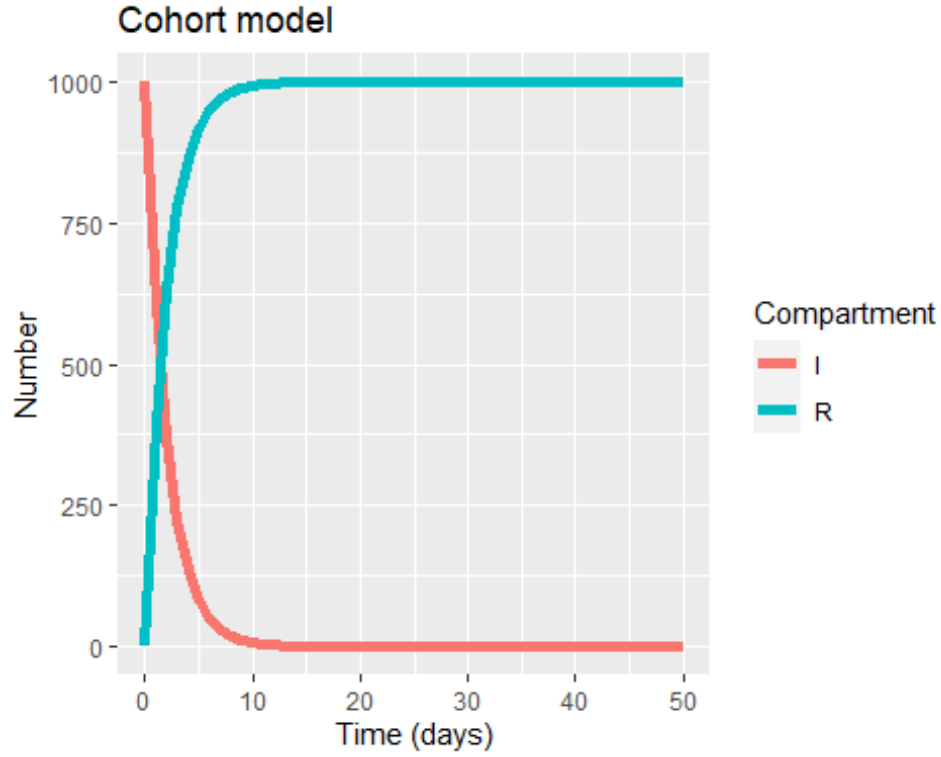
# Her bir durum değişkeninin (I ve R) bir sütuna sahip olduğunu görüyorsunuz
# Uzun format olarak adlandırdığımız sonuçları grafik haline getirmek için
yeniden şekillendirme özelliğini kullanalım
# Çıktımızı bu formatta almak için "melt" fonksiyonu ve paketi

# çıktı veri setini uzun formata dönüştür
output1_long <- melt(as.data.frame(output1), id = "time")

# Konsola output1_long yazarak çıktıya göz atalım

## Model çıktısını grafik haline getirin
# I ve R bölmelerindeki insan sayısının zamana bağlı grafiğini çizin
ggplot(data = output1_long, # grafik çizmek için verilerin bulunduğu
        aes(x = time, y = value, colour = variable, group = variable)) +
geom_line(size=2) + # verileri satır olarak gösterin
  xlab("Time (days)") + # x eksenini için etiket ekleyin
  ylab("Number") + # y eksenini için etiket ekleyin
  labs(colour = "Compartment", # açıklama başlığı ekleyin
        title = "Cohort model")
```

Bulaşıcı hastalık dinamiklerinin R'de modellenmesi üzerine kısa kurs



Görev: Yukarıdaki kodu kullanarak aynı modeli çalıştırın ancak şimdi kohortumuzdaki ortalama enfeksiyon süresinin 2 gün yerine 10 gün olduğu bir senaryo hayal edin. Grafiğin neden farklı görüldüğünü açıklayabilir misiniz?