deSolve with Ellie's parameter

2025-06-09

```r
CDCActiveTotal <- c(25102, 24206, 22726, 21212, 19751, 18285, 17492, 16305, 15937, 15048, 14828, 14494,
ActiveCasesTotal <- CDCActiveTotal/1000000 #Calculates total active infections from the CDC data

CDCTBDeaths <- c(1631, 1631+1478, 1631+1478+1336, 1631+1478+1336+1202, 1631+1478+1336+1202+1166, 1631+14
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493,
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493+470,
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493+470+528,
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493+470+528+5
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493+470+528+5
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493+470+528+5
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493+470+528+5
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493+470+528+5
1631+1478+1336+1202+1166+1112+930+776+764+784+711+657+648+652+554+585+529+569+539+510+555+493+470+528+5
TotalDeaths <- CDCTBDeaths/1000000

CDCTBDeathsNC <- c(1631, 1478, 1336, 1202, 1166, 1112, 930, 776, 764, 784, 711, 657, 648, 652, 554, 585
DeathsNC <- CDCTBDeathsNC/1000000

CDCActiveHR <- c(1534, 1543, 1351, 1284, 1195, 1120, 1000, 981, 897, 914, 904, 872, 842, 845, 797, 835,
ActiveCasesHR <- CDCActiveHR/1000000

CDCActiveMDR <- c(484, 431, 327, 250, 201, 155, 157, 146, 151, 158, 119, 128, 125, 124, 124, 107, 115,
ActiveCasesMDR <- CDCActiveMDR/1000000

# Function
tb_model <- function(t, state, parms) {
  with(as.list(c(state, parms)), {

    dS  <- (rho * N -
           ql*t1 * S*I1/N -
           ql*t2 * S*I2/N -
           ql*t3 * S*I3/N -
           ql*t4 * S*I4/N +
           z1*phi1*I1 + z2*phi2*I2 + z3*phi3*I3 + z4*phi4*I4 +
           (1 - l)*alpha*N -
           mu0*S)

    dE1 <- ((1 - p)*ql*t1 * S*I1/N -
           vL * E1 +
           (1 - y1)*(1 - z1)*phi1 * I1 +
           l*alpha*(1 - r2 - r3 - r4)*N -
           mu0*E1)

    dI1 <- (p*ql*t1 * S*I1/N +
           vL*E1 -
```

```r
            phi1*I1 -
            mu0*I1 -
            mu*I1)

  dE2 <- ((1 - p)*ql*t2 * S*I2/N -
            vL * E2 +
            (1 - y2)*(1 - z2)*phi2 * I2 +
            gamma*(1 - z1)*y1*phi1 * I1 +
            l*alpha*r2*N -
            mu0*E2)

  dI2 <- (p*ql*t2 * S*I2/N +
            vL*E2 -
            phi2*I2 -
            mu0*I2 -
            mu*I2)

  dE3 <- ((1 - p)*ql*t3 * S*I3/N -
            vL * E3 +
            (1 - y2)*(1 - z3)*phi3 * I3 +
            (1 - gamma)*(1 - z1)*y1*phi1 * I1 +
            l*alpha*r3*N -
            mu0*E3)

  dI3 <- (p*ql*t3 * S*I3/N +
            vL*E3 -
            phi3*I3 -
            mu0*I3 -
            mu*I3)

  dE4 <- ((1 - p)*ql*t4 * S*I4/N -
            vL * E4 +
            (1 - z2)*y2*phi2 * I2 +
            (1 - z3)*y2*phi3 * I3 +
            (1 - z4)*phi4 * I4 +
            l*alpha*r4*N -
            mu0*E4)

  dI4 <- (p*ql*t4 * S*I4/N +
            vL*E4 -
            phi4*I4 -
            mu0*I4 -
            mu*I4)

  dD  <- (mu * (I1 + I2 + I3 + I4))

  dN  <- (rho*N +
            alpha*N -
            mu*(I1 + I2 + I3 + I4) -
            mu0*N)

  list(c(dS, dE1, dI1, dE2, dI2, dE3, dI3, dE4, dI4, dD, dN))
})
```

```r
}

# Time steps
deltaT <- 0.1
finalYr <- 57
cutoffYr <- 8/deltaT
totT <- finalYr / deltaT

# simulation
 hill <- function(initial=cutoffYr+1, final=totT+1, dataSet=P) {
  # recursive=TRUE collapses dataframe to labeled vector
  initv <- c(dataSet[initial,], recursive=TRUE)
  # times = data points to be calculated
  times <- initial:final*deltaT
  # compute master results
  mres <- lsoda(initv, times, tb_model, parms)
  # mres[,-1] = mres without 1st column
  dataSet[initial:final,] <- c(mres[,-1])
  return(dataSet)
 }


# Create composite metrics (from Ellie's code)
 generateResults <- function(mres) {
  with(as.list(parms), {
    Susceptible <- mres$S
    Exposed1 <- mres$E1
    Infectious1 <- mres$I1
    Exposed2 <- mres$E2
    Infectious2 <- mres$I2
    Exposed3 <- mres$E3
    Infectious3 <- mres$I3
    Exposed4 <- mres$E4
    Infectious4 <- mres$I4
    Dead <- mres$D
    Total <- mres$S + mres$E1 + mres$I1 + mres$E2 + mres$I2 + mres$E3 + mres$I3 + mres$E4 + mres$I4
    InfectiousTotal <- mres$I1 + mres$I2 + mres$I3 + mres$I4
    return(data.frame(Susceptible, Exposed1, Infectious1, Exposed2, Infectious2, Exposed3, Infectious3,
  })
}

# Parameters
parms <- c(
  a2 = 0.0667044593952596,      # Proportion of initial latent TB cases that are H-resistant
  a3 = 0.036740641715005,       # Proportion of initial latent TB cases that are R-resistant
  a4 = 0.00657318516168743,      # Proportion of initial latent TB cases that are MDR

  alpha = 0.00425,    # Immigration rate into the US (fixed)

  b = 0.57151103077922,       # Proportion of initial active TB cases that are drug-susceptible

  gamma = 0.595633820397779,   # Proportion of H-resistance acquisition cases

  l = 0.0303478768619243,       # Proportion of immigrants that have LTBI
```

3

```
    # lambda = .5,   # Effective contact rate

  mu = 0.0495773325092159,     # TB-specific mortality rate
  mu0 = 0.013,   # Background (non-TB) mortality rate (fixed)

  p = 0.0835297864163294,       # Proportion of exogenous infections that are acute

  phi1 = 0.787279893178493,   # Rate of end of treatment for DS TB
  phi2 = 0.562803372205235,    # Rate of end of treatment for H-resistant TB
  phi3 = 0.326452181814238,    # Rate of end of treatment for R-resistant TB
  phi4 = 0.313439025112893,     # Rate of end of treatment for MDR TB

  # q = 0.1,       # Proportion of active TB cases that are infectious

  ql = 1.95014937780797, # combined q with lambda

  r2 = 0.000974651146680117,      # Proportion of immigrant H-resistant LTBI cases
  r3 = 0.0648224649368785,      # Proportion of immigrant R-resistant LTBI cases
  r4 = 0.0276833213865757,      # Proportion of immigrant MDR LTBI cases

  rho = 0.0179, # US birth rate (fixed)

  t1 = 0.044127013348043,     # Time in treatment for DS
  t2 = 0.0336318691493943,     # Time in treatment for H-resistant
  t3 = 0.0244579056947259,     # Time in treatment for R-resistant
  t4 = 0.0591143056983128,     # Time in treatment for MDR

  vL = 0.00126206192770042,    # Progression rate from latent to active TB

  y1 = 0.318704390781932,     # Failed treatment leading to H/R resistance from DS
  y2 = 0.896799218840897,     # Failed treatment leading to MDR from H- or R-resistant TB


  z1 = 0.877329782699235,     # Success rate of DS TB treatment
  z2 = 0.760776009969413,      # Success rate of H-resistant TB treatment
  z3 = 0.636800507875159,      # Success rate of R-resistant TB treatment
  z4 = 0.41258505214937     # Success rate of MDR TB treatment
)
```

```
# Initial conditions (Ellie's code)
    S <- E1 <- I1 <- E2 <- I2 <- E3 <- I3 <- E4 <- I4 <- D <- N <- rep(0,totT) #Sets compartment values
    P <- data.frame(S, E1, I1, E2, I2, E3, I3, E4, I4, D, N) #Creates a matrix of compartment values

    list2env(as.list(parms), envir = .GlobalEnv)
```

```
## <environment: R_GlobalEnv>
```

```
    #Total Population
    P$N[1] <- 280.726081 #From census data
    #LTBI
    P$E1[1] <- 11.213*(1-a2-a3-a4) #Data from Hill
    P$E2[1] <- 11.213*a2
    P$E3[1] <- 11.213*a3
    P$E4[1] <- 11.213*a4
```

```r
    #Active TB
    P$I1[1] <- (b*(CDCActiveTotal[1] - CDCActiveHR[1] - CDCActiveMDR[1])/(mu0 + mu + phi1))/1000000
    #Method from Hill; The CDC tracks H-Resistant and MDR cases.
    #Those leftover are either drug-susceptible or R-resistant (scaled by b and (b-1), respectively)
    P$I2[1] <- (CDCActiveHR[1]/(mu0 + mu + phi2))/1000000
    P$I3[1] <- ((1-b)*(CDCActiveTotal[1] - CDCActiveHR[1] - CDCActiveMDR[1])/(mu0 + mu + phi3))/1000000
    P$I4[1] <- (CDCActiveMDR[1]/(mu0 + mu + phi4))/1000000
    #Susceptible Population
    P$S[1] <- P$N[1] - P$E1[1] - P$I1[1] - P$E2[1] - P$I2[1] - P$E3[1] - P$I3[1] - P$E4[1] - P$I4[1]

    yrs <- seq(1993, 1993+finalYr, deltaT)

    P <- hill(1, totT+1, P)
    Results <- generateResults(P)
```

```r
#Plot model and CDC data for total active cases, HR cases, MDR
#cases, and TB deaths on the same plot, using two sets of axes
years = 1993:2023 #For CDC data where we have through 2013
plot(yrs, (Results$Infectious1*(mu0+mu+phi1)) +  (Results$Infectious2*(mu0+mu+phi2)) + (Results$Infecti
points(years,ActiveCasesTotal,col='red')
years=1993:2022 #For CDC data on TB deaths, where we have through #2022 only
lines(yrs, Results$Dead, col='black')
points(years, TotalDeaths,col='black')
par(new = TRUE) #This uses the right-hand side axis, since HR and
#MDR cases exist on a much smaller scale
years = 1993:2023
plot(yrs, Results$Infectious2*(mu0+mu+phi2), axes = FALSE,  bty = "n", xlab = "", ylab = "", col='blue'
points(years,ActiveCasesHR, col='blue')
lines(yrs,Results$Infectious4*(mu0+mu+phi4),col='green')
points(years,ActiveCasesMDR,col='green')
mtext("Drug-resistant Cases in millions",side=4)
axis(side=4,at=c(0,0.001))
legend('top', legend=c(' Total active cases',
                                    'H-resistant active cases',
                    'MDR active cases',
                    'Cumulative TB deaths'),
    col=c('red', 'blue','green', 'black'), lty=c(1,1, 1, 1),cex = 0.6)
```

Fitting Model to Data

Legend:
- Total active cases
- H–resistant active cases
- MDR active cases
- Cumulative TB deaths

Total active cases or TB deaths in millions

Drug–resistant Cases in millions

Year