Model Selection for Influenza Outbreaks

Shibi He

Introduction

This project uses Approximate Bayesian Computation and sequential Monte Carlo sampling (ABC-SMC) to replicate Figure 3(a) and 3(c) in the paper "Simulation-based model selection for dynamical systems in systems and population biology (Tony and Stumpf, Bioinformatics (2010)". The research question is to investigate whether the spread of different strains of the influenza virus can be described by the same model. The authors considered two scenarios: (i) different outbreaks of the same strain, and (ii) outbreaks of different modlecular strains of the influenza virus.

The are two parameters we need to infer, one is the probability that a susceptible individual does not get infected from the community q_c, and the other one is the probability that a susceptible indivdual escapes infection within their own household q_h. The probability that j out of the s susceptibles in a household become infected is given by:

$$w_{js} = \binom{s}{j} w_{jj} (q_c q_h^j)^{s-j},$$

where $w_{0s} = q_c^s$, $s = 0, 1, 2, ...$ and $w_{jj} = 1 - \sum_{i=0}^{j-1} w_{ij}$.

Data

The data we use are obtained from the supplement of Tony and Stumpf's paper. The first dataset contains information from two separate outbreaks of the same strain of influenza virus. The second dataset contains information about outbreaks of two different influenza infection. We compute the corresponding probability of infection w_{js} for each dataset.

```r
A1 = matrix(c(66,87,25,22,4,
              13,14,15,9, 4,
              0, 4, 4, 9, 1,
              0, 0, 4, 3, 1,
              0, 0, 0, 1, 1,
              0, 0, 0, 0, 0), nrow=6, ncol=5, byrow=TRUE)

A2 = matrix(c(44,62,47,38,9,
              10,13,8,11, 5,
              0, 9, 2, 7, 3,
              0, 0, 3, 5, 1,
              0, 0, 0, 1, 0,
              0, 0, 0, 0, 1), nrow=6, ncol=5, byrow=TRUE)

B1 = matrix(c(9,12,18, 9,4,
              1, 6, 6, 4,3,
              0, 2, 3, 4,0,
              0, 0, 1, 3,2,
              0, 0, 0, 0,0,
              0, 0, 0, 0,0), nrow=6, ncol=5, byrow=TRUE)

B2 = matrix(c(15,12,4,11,17,4,0,21,4,0,0,5), nrow=4, ncol=3, byrow=TRUE)
```

```
A1 = sweep(A1, 2, colSums(A1), "/")
A2 = sweep(A2, 2, colSums(A2), "/")
B1 = sweep(B1, 2, colSums(B1), "/")
B2 = sweep(B2, 2, colSums(B2), "/")
```

Methodology

As discussed in the paper, the ABC-SMC procedure defines a prior distribution and sequentially constructs intermediate distributions, which converge to the posterior distribution. We use ABC-SMC to sample 5000 pairs of q_c and q_h.

ABC-SMC Step 1: sample parameters from a prior distribution (population 1)

```
set.seed(610)
```

```
generate_abc_sample = function(observed_data,
                               prior_distribution,
                               data_generating_function,
                               distance,
                               epsilon) {
    while(TRUE) {
        theta = prior_distribution()
        y = data_generating_function(observed_data, theta[1], theta[2])
        if(distance(y, observed_data) < epsilon) {
            return(theta)
        }
    }
}
```

```
# prior: uniform distribution
prior_distribution = function() runif(2)
```

```
data_generating_function = function(observed_data, qc, qh){
    Nrow = nrow(observed_data)
    Ncol = ncol(observed_data)
    w = matrix(rep(0, Nrow*Ncol), nrow = Nrow, ncol=Ncol)
    for (s in 1:Ncol){
        for (i in 1:Nrow){
            j = i-1
            if (j > s){
                w[i,s] = 0
            } else if (i == 1){
                w[i,s] = qc^s
            } else if (j < s) {
                w[i,s] = choose(s,j) * w[i,j] * (qc * qh^j)^(s-j)
            } else {
                #w[i,s] = 1- (colSums(w)[s]-w[i,s])
                w[i,s] = 1 - sum(w[1:i-1, s])
            }
        }

    }
    return(w)
}
```

```
distance = function(matrix, data){
    dist = sqrt(sum((matrix - data)^2))
    return(dist)
}
```

ABC SMC Step 2: write function to generate intermediate distributions (population 2 to T-1)

```
generate_abc_sample2 = function(observed_data,
                                population,
                                data_generating_function,
                                distance,
                                epsilon) {
    while(TRUE) {
        theta = population[, sample(ncol(population), 1, replace=TRUE)]
        theta[1] = rnorm(1, mean=theta[1], sd=0.01)
        theta[2] = rnorm(1, mean=theta[2], sd=0.01)
        y = data_generating_function(observed_data, theta[1], theta[2])
        if(distance(y, observed_data) < epsilon) {
            return(theta)
        }
    }
}
```

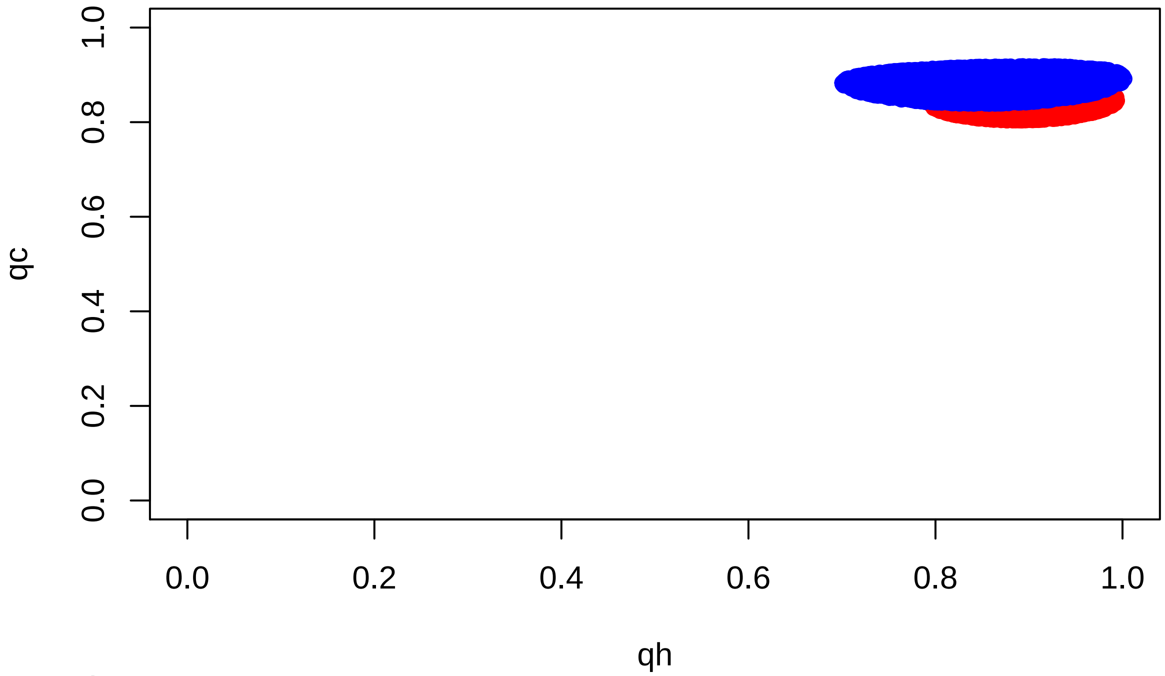
ABC SMC Step 3: generate the final population T

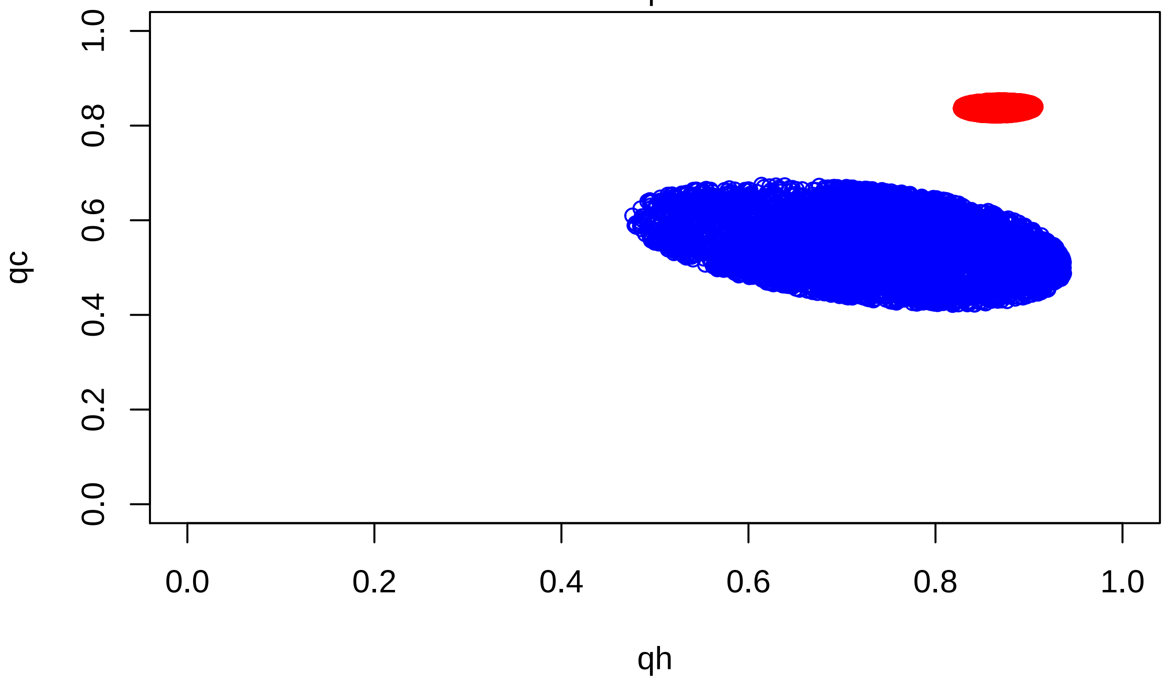
```
ABC_SMC_sample = function(observed_data){
    epsilon = seq(1, 0.32, -0.04)
    N = 5000
    for (i in 1: length(epsilon)){
        if (i == 1){
             posterior_samples = replicate(n=N, generate_abc_sample(
                 observed_data,
                 prior_distribution,
                 data_generating_function,
                 distance, epsilon[i]))

        } else{
            posterior_samples = replicate(n=N, generate_abc_sample2(
                 observed_data,
                 population = posterior_samples,
                 data_generating_function,
                 distance, epsilon[i]))
        }

    }
    return(posterior_samples)
}
```

```
q_A1 =ABC_SMC_sample(A1)
q_A2 =ABC_SMC_sample(A2)
```

```
q_B1 =ABC_SMC_sample(B1)
q_B2 =ABC_SMC_sample(B2)
```

Plot the distribution





Conclusion

The above graph shows the posterior distribution of q_c and q_h. The first graph corresponds to the scenario of the two separate outbreaks of the same strain of influenza virus while the second graph corresponds to the case of two outbreaks of two different strains of influenza infection. Clearly, we see q_c and q_h are largely overlapped in the first graph and separated in the second graph. These graphs suggest that we can not describe the two scenarios of influenza infection outbreaks using the same model.