EpiMod: Pertussis

Beccuti Marco, Castagno Paolo, Pernice Simone

Contents

Introduction

In this document we describe how to use the R library *epimod*. In details, *epimod* implements a new general modeling framework to study epidemiological systems, whose novelties and strengths are:

1. the use of a graphical formalism to simplify the model creation phase;
2. the automatic generation of the deterministic and stochastic process underlying the system under study;
3. the implementation of an R package providing a friendly interface to access the analysis techniques implemented in the framework;
4. a high level of portability and reproducibility granted by the containerization (Veiga Leprevost et al. 2017) of all analysis techniques implemented in the framework;
5. a well-defined schema and related infrastructure to allow users to easily integrate their own analysis workflow in the framework.

The effectiveness of this framework is showed through two case studies, the wellknown and simple SIR model, and much more complex model related to the pertussis epidemiology in Italy.

How to start

Install *epimod*:

```
install.packages("devtools")
library(devtools)
install_github("qBioTurin/epimod", dependencies=TRUE)
```

```
library(epimod)
```

Then, the following function must be used to download all the docker images used by *epimod*:

```
downloadContainers()
```

Something to know

All the *epimod* functions print the following information:

- *Docker ID*, that is the CONTAINER ID which is executed by the functon;
- *Docker exit status*, if 0 then the execution completed with success, otherwise an error log file is saved in the working directory.

Cases of study

Pertussis Model

We now describe how the framework functions can be combined to obtain an analysis workflow for the Pertussis model introduced in the main paper: *A computational framework for modeling andstudying pertussis epidemiology and vaccination*.

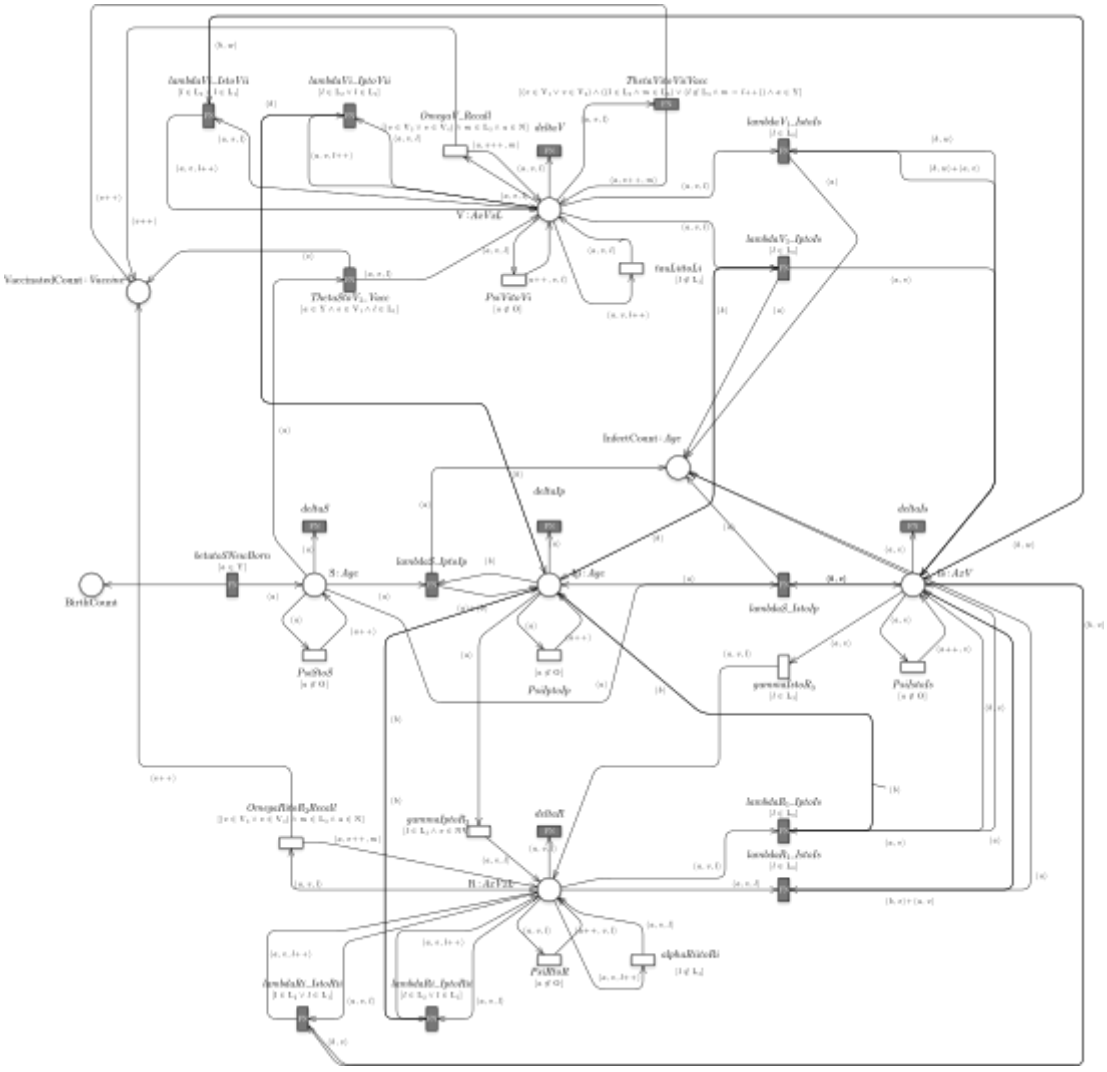

Figure 1: Petri Net representation of the Pertussis model.

We first introduce all the functions, the constants, and the numerical values associated with the general transitions rate (the black boxes in figure 1). Then, we show the proposed framework and how it can be successfully used to study and analyze pertussis infection and the relative vaccination cycle in Italy.

General Transitions

In this section we describe in details the firing rate function associated with the generic transitions in our Pertussis model. Let us recall that:

1. $f_{\langle 1 \rangle}(\hat{x}(\nu), \nu)$ is the speed of the transition $t \in T_g$ and $\hat{x}(\nu)$ represents the vector of the average number of tokens for all the input places of t. For brevity when the function does not depend on the color instance c then we omit it reporting only the transition t, i.e. $f_t(\hat{x}(\nu), \nu)$.
2. given a transition with a specific color domain, the unfolding procedure generates automatically the transition name combined with all the possible combinations of the color classes associated. For instance, the transition $ContactS\_IpToIp$ is unfolded into $ContactS\_IpToIp\_a_1\_a_1$, $ContactS\_IpToIp\_a_1\_a_2$, $ContactS\_IpToIp\_a_1\_a_3$, etc.

All the general transitions of the model are now explained in details and all the constants are summarized in Table 1.

Transitions modeling the contacts. All the transitions representing the contact between a person belonging to the age class a_i with one from a_j, which are grouped in the following set:

$$\begin{aligned}T_{contact} = \{&ContactVi\_IpToRii\_a_i\_a_j, ContactVi\_IsToRii\_a_i\_a_j,\\ &ContactV\_IpToIs\_a_i\_a_j, ContactV\_IsToIs\_a_i\_a_j, ContactR\_IpToIs\_a_i\_a_j,\\ &ContactR\_IsToIs\_a_i\_a_j, ContactRi\_IpToRii\_a_i\_a_j, ContactRi\_IsToRii\_a_i\_a_j,\\ &ContactS\_IpToIp\_a_i\_a_j, ContactS\_IsToIp\_a_i\_a_j\}_{i,j=1,2,3}\end{aligned}$$

are defined by the following function:

$$f_t(\hat{x}(\nu), \nu) = prob(t) * \frac{\lambda_{a_i,a_j}}{x^{tot}_{a_j}(\nu)} \prod_i \hat{x}_i(\nu) \tag{1}$$

where $t \in T_{contact}$, $x^{tot}_{a_j}(\nu)$ represents the number of people in the a_j class at time ν, λ_{a_i,a_j} the contact rate between the a_i and a_j classes corresponds to the i-row and j-column of the contact matrix 2. $prob(t)$ is a function which returns depending on the transition t if its rate has to be multiplied by a specific probability and it is defined as follows

$$prob(t) = \begin{cases} prob\_infectionS & t \in \{ContactS\_IpToIp\_a_i\_a_j, ContactS\_IsToIp\_a_i\_a_j\} \\ prob\_boost & t \in \{ContactRi\_IpToRii\_a_i\_a_j, ContactRi\_IsToRii\_a_i\_a_j, \\ & \quad ContactVi\_IpToRii\_a_i\_a_j, ContactVi\_IsToRii\_a_i\_a_j\} \\ prob\_infectionR\_l1 & t \in \{ContactV\_IpToIs\_a_i\_a_j, ContactV\_IsToIs\_a_i\_a_j, \\ & \quad ContactR\_IpToIs\_a_i\_a_j, ContactR\_IsToIs\_a_i\_a_j\} \end{cases}$$

Finally, $\prod_i \hat{x}_i(\nu)$ is the product of the average numbers of individuals in the input places of the transition t.

Transitions modeling the deaths. Let us define the set of the transitions modeling the death of a person in age class a_i as

$$\begin{aligned}T_{death} = \{&DeathS\_a_i, DeathIp\_a_i, DeathIp\_a_i,\\ &DeathR\_a_i, DeathV\_a_i\}_{i=1,2,3}.\end{aligned}$$

Then we can define the function providing the speed of a transitions $t \in T_{death}$ as

$$f_t(\hat{x}(\nu), \nu) = \delta_{a_i}(\nu)\hat{x}(\nu)$$

where $\delta_{a_i}(\nu)$ is the death rate with respect the age class a_i, and it changes its value depending on the current year:

$$\delta_{a_i}(\nu) = death[i, \nu],$$

where $death[i, \nu]$ is the death rate referred to the year given by ν and the age class a_i considering the rates matrix defined from 1974 to 2016 (columns) for each of three age classes (rows), reported in github.com/qBioTurin/epimod.

Transitions modeling the birth. The birth events are modeled by the transition *Birth*, which is characterized by the following function:

$$f_t(\hat{x}(\nu), \nu) = \mu(\nu)x^{tot}(\nu),$$

where $\mu(\nu)$ is the birth rate per person estimated from 1974 to 2016, obtained from the ISTAT. Since this rate represents the mean number of Italian newborns (in the year ν) per person, it must be multiplied by the total number of Italians (in the year ν), i.e. $x^{tot}(\nu)$.

Transitions modeling the vaccination. The vaccination process starts with the transition *FirstVaccination*, and continues (for the administrations of two further doses) by the *Vaccination* transition. Both of them are characterized by the following function:

$$f_t(\hat{x}(\nu), \nu) = \chi(\nu)\hat{x}(\nu), \tag{2}$$

where $\chi(\nu)$ is administration rate of one vaccine dose, computed as explained in Sec. Vaccination rates. This rate is then multiplied by $\hat{x}(\nu)$, which represents the number of newborns without vaccination, so in the color class NV, (referring to the *FirstVaccination* transition), or the number of newborns already vaccinated one or two times, (referring to the *Vaccination* transition). When the possibility of vaccination failure is also considered, than the function 2 is multiplied to the vaccination failure probability p_v, obtaining the following rates for the *FirstVaccination* and *Vaccination* transitions instead of eq.2:

$$f_t(\hat{x}(\nu), \nu) = \chi(\nu)\hat{x}(\nu) * (1 - p_v).$$

To model the vaccination failure further transitions are drawn in the model for simulating the vaccine administrations without an increasing of the resistance level, their rates are defined as follows:

$$f_t(\hat{x}(\nu), \nu) = \chi(\nu)\hat{x}(\nu) * (p_v). \tag{3}$$

Parameters

All the parameters file reporting their values can be found at the following link github.com/qBioTurin/epimod.

Contact rates. We are considering the contact matrix provided by (Mossong 2008), in which the Italian contact rates depending on the age are reported.

Initial marking. The initial marking is a vector defined by 179 variables, that are all the places of our ESSN associated with all the corresponding color classes combinations (given by the color domain **A**, **V**, and **L**). Since the simulations start from the 1974, from when we are considering no vaccination, all the places with colors different to *NV* (i.e., no vaccination) are settled to zero. For obvious reasons, all the places representing counters (birth, vaccination, and infection counting) are settled to zero. From (Gonfiantini et al. 2014) and the surveillance data, we were able to estimate the number of infects in each age class in the 1974. In details, during the whole year there were reported 7'400 cases distributed as follow: 15% in N, 80% in Y, and 5% in O. Supposing for simplicity: (1) to split as equals the number of infects between the I_p and I_s places, and (2) to obtain the mean initial number of infects to scale their values with a factor of 21/365, since the healing mean time is 21 days. This is necessary because the time scale of our simulations is *days*.

Finally, the remaining places to estimate their initial markings are given by : the susceptible in each age class (*S_a1, S_a2, S_a3*), and the recovered without vaccination in each age and resistance level class (*R_a1_nv_l4, R_a2_nv_l2, R_a2_nv_l3, R_a2_nv_l4, R_a3_nv_l2, R_a3_nv_l3, R_a3_nv_l4*). For simplicity we consider *R_a1_nv_l1, R_a1_nv_l2, R_a1_nv_l3* equal to zero since the time necessary to decrease the immunity level is greater than 1 year.

Since it is known the size during the 1974 of the Italian populations and how they are distributed among the age classes, removing the infects, we have to estimate through the sensitivity and calibration analysis their distribution among the susceptible and recovered, and the resistance levels as well.

Vaccination rate. The vaccination rate is defined through the vaccination policy and the properties characterizing the Exponential Negative Distribution. So to estimate the vaccination rate χ let us introduce three i.i.d. random variables :

1. death in the first year: $D \sim Exp(\delta_{a_1})$,

Symbol	*Parameter*	*Value*
γ	Healing rate	21 days
θ^{vacc}	Decay of the resistance derived from vaccination	7 years (21 months per level)
θ^{inf}	Decay of the resistance derived from infection	14 years (42 months per level)
X_0	Initial population distribution	see Sec. *Initial marking*
$\chi(\nu)$	Vaccinations rate (calculated imposing a fixed vaccine coverage)	Minimum of exponential distributions see Sec. *Vaccination rate.*
$\mu(\nu)$	Birth rate depending on the time	Obtained from ISTAT see Sec. *Birth rate.*
$\delta_{a_i}(\nu)$	Death rates depending on time and on the age class	Obtained from ISTAT see Sec. *Death rates.*
$\lambda_{a,b}$	Contact rates between subjects in the age classes a and b	see Table 2
$prob\_boost$	prob. of a natural boost occurrence	To be estimated
$prob\_infetionS$	prob. of infection success of a susceptible	To be estimated
$prob\_infetionR\_l1$	prob. of infection success of a recovered with minimum resistance	To be estimated

Table 1: Parameters of the ESSN model.

	a_1	a_2	a_3
a_1	0.2136752137	0.5586592179	0.1438848921
a_2	0.5586592179	0.0205212395	0.0364033491
a_3	0.1438848921	0.0364033491	0.0063742988

Table 2: Contact rates per age class.

2. growth from the first age class to the second one: $G \sim Exp(\lambda_{a_1,a_2})$,
3. vaccination: $V \sim Exp(\chi)$, (equal for each vaccination dose),

and the events (i) $V_i = \{i^{th}\text{vaccination dose is submitted}\}$, $i = 1, 2, 3$, and (ii) $A_1 = \{\text{belonging to the first age class}\}$. Therefore, requiring that the probability to complete the vaccination cycle in absence of disease during first year is equal to p, i.e.

$$\mathbb{P}\{(3 vaccination\ doses\ are\ submitted)|(before\ the\ first\ year)\} = \mathbb{P}\{V_1 \cap V_2 \cap V_3 | A_1\} = p, \tag{4}$$

Initial condition	*Value*
Ip_a1	3.198904e+01
Ip_a2	1.748466e+02
Ip_a3	6.415068e+00
Is_a1_nv	3.198904e+01
Is_a2_nv	1.748466e+02
Is_a3_nv	6.415068e+00
S_a1 + R_a1_nv_l4	866703
S_a2 + R_a2_nv_l1+ R_a2_nv_l2 + R_a2_nv_l3 + R_a2_nv_l4	15685693
S_a3 + R_a3_nv_l1+ R_a3_nv_l2 + R_a3_nv_l3 + R_a3_nv_l4	37837299

Table 3: Initial conditions.

we are able to deduce that

$$\begin{aligned}
\mathbb{P}\{V_1 \cap V_2 \cap V_3 | A_1\} = & \\
&= \prod_{i=1}^{3} \mathbb{P}\{V_i | A_1\} \\
&= (\mathbb{P}\{V_1 | A_1\})^3 \\
&= (\mathbb{P}\{V = \min(D, G, V)\})^3 \qquad (5) \\
&= (\frac{\chi}{\chi + \delta_{a_1} + \lambda_{a_1,a_2}})^3 = p. \qquad (6)
\end{aligned}$$

Where we suppose that the random variables $D,\ G, V$ are i.i.d., the three events $V_i,\ i = 1, 2, 3$ as well, and that one vaccination dose is submitted iif the transition modeling the vaccination fires before the transitions modeling the death or the growth (and so the exit form the first age class). Then, the properties characterizing the exponential distributions [1] are exploited to obtain an analytic formula depending on the unknown parameter χ, and the known parameters $\delta_{a_1},\ \lambda_{a_1,a_2},\ p$. Indeed solving the following equation

$$(\frac{\chi}{\chi(\nu) + \delta_{a_1}(\nu) + \lambda_{a_1,a_2}})^3 - p(\nu) = 0,$$

we are able to estimate the vaccinate rate χ depending on the vaccine policy defined a priori. The probabilities p were calculated from the percentage of newborns who completed the vaccination cycle. These percentages were reported every year from 1994 to 2016. For this reason p is time dependent, and they were obtained from (Gonfiantini et al. 2014,@IstatCopVacc).

Birth rate. The birth rate has been computed directly from data provided by ISTAT for the period $1974 \sim 2016$. In details, it is defined as the average births per day in the reference period with a dependence on the year.

Death rates. Similarly to the birth rate, the death rates are defined as the average deaths per day from 1974 to 2016, with a dependence on the age class and the year.

[1] Let $X_1, \ldots, X_n$ be independent random variables, with X_i having an $Exp(\lambda_i)$ distribution, respectively. Then the distribution of $\min(X_1, \ldots, X_n)$ is $Exp(\lambda_1 + \cdots + \lambda_n)$, and the probability that the minimum is X_i is $\frac{\lambda_i}{\lambda_1 + \cdots + \lambda_n}$.

Model Generation

The starting point is the derivation from the Pertussis model the corresponding underlying stochastic and deterministic processes by using the function *model_generation*. Then the derived deterministic process is represented by a system of 179 ODEs, while the derived stochastic process is characterized by 1965 possible events. Let us observe that the functions associated with the general transitions (the black boxes in figure 1) are implemented in the file *transitions.cpp*, which is passed as input parameter of the function.

```
generation <- model_generation(net_fname = "./Net/Pertussis.PNPRO",
                               functions_fname = "./Cpp/transitions.cpp")
```

Sensitivity analysis

Since the model is characterized by 15 unknown parameters, three of the represent the probabilities of having (i) the *susceptible infection success*, i.e., the infection of a susceptible individual due to a contact with an infected individual, namely *prob_infectionS*, (ii) the *resistant infection success*, i.e., the infection of a vaccinated or recovered individual with the minimum resistance level due to a contact with an infected individual, namely *prob_infectionR_l1*, and finally (iii) *the natural boosts*, i.e., the restoring of the resistance level to the maximum when a person with resistance level different from the minimum level comes into contact with an infected individual, namely *prob_boost*. The remaining 12 unknown parameters are the initial marking of the susceptible and recovered places. We can apply the function *sensitivity_analysis()* on the deterministic process previously generated and considering the data of the period from 1974 to 1994 as reference targets, in order to identify which parameters are most sensitive w.r.t. the counts of infects. Therefore, the following function input parameters are passed:

1. **solver_fname**: *Pertussis.solver*;
2. **n_config**: the model is run *2^14*;
3. **f_time**: since all the rates were calculated daily and we want to simulate 21 years, then the *f_time* has to be *365*21*;
4. **s_time**: the step time is set to 1 year, *365* days;
5. **parameters_fname**: in *Functions_list.csv* the parameters which have to vary and also the parameters that have to be passed to the general functions stored in *transition.cpp* are reported.

```
#>   Tag          Name        Function
#> 1   g       b_rates          b_rate
#> 2   g       c_rates          c_rate
#> 3   g       d_rates          d_rate
#> 4   g       v_rates          v_rate
#> 5   g probabilities     probability
#> 6   i          init initial_marking
#>                                      Parameter1
#> 1 file='/home/docker/data/input/init_conf.RData'
#> 2 file='/home/docker/data/input/init_conf.RData'
#> 3 file='/home/docker/data/input/init_conf.RData'
#> 4 file='/home/docker/data/input/init_conf.RData'
#> 5 file='/home/docker/data/input/init_conf.RData'
#> 6 file='/home/docker/data/input/init_conf.RData'
```

6. **functions_fname**: in *Functions.R* the functions reported in the third column of *Functions_list.csv* are implemented. For instance, the function associated to the generation of the three probabilities is defined as follows:

```
# if x is null then it means that we have to sample the values by
# exploiting the uniform distribution between 0 and 0.25. Let us note
# that three values corresponding to the three probabilities are
# generated since the uniform intervals are all identical to [0,0.25].
# Differtly when x is not null, then it means the we are using the
# optimization algorithm and the prob. values are already sampled,
# for this reason we take just the first three values of x (the vector
# with size equals to the number of parameter which are varing),
# 0 is given to the probability of vaccination failure that will be used
# in model_analyis.
# The three values obtained are automatically saved in a file called as
# the corresponding name in the Function_list.csv (second column),
# and then read and exploited by the functions in transitions.cpp .

probability <- function(file, x = NULL)
{
    load(file)
    if( is.null(x) ){
        x <- runif(n = length(probabilities), min=0, max=0.25)
        x[length(x)] = 0
    }
    else{
        x <- c(x[c(1:3)],0)
    }
    return(matrix(x, ncol = 1))
}
```

Let us note that the probabilities values generated are used by the functions modeling all the transitions representing the contact between two individuals (implemented in *transitions.cpp*). For this reason the tag associated with this parameter is g and not p (where p has to be used when a transition rate or a single place is under analysis). Similarly, the function *initial_marking* characterizing the generation of the unknown initial markings of certain places is defined in order to satisfy the following constraints:

$$\begin{aligned} &S\_a1 + R\_a1\_nv\_l4 = 866703 \\ &S\_a2 + R\_a2\_nv\_l1 + R\_a2\_nv\_l2 + R\_a2\_nv\_l3 + R\_a2\_nv\_l4 = 15685693 \\ &S\_a3 + R\_a3\_nv\_l1 + R\_a3\_nv\_l2 + R\_a3\_nv\_l3 + R\_a3\_nv\_l4 = 37837299 \end{aligned} \tag{7}$$

Hence, the values estimated during the sensitivity analysis and model calibration steps are the proportions of the total number in each age class. For instance, the values .4 and .8 might be associated to S_a1 and $R\_a1\_nv\_l4$, meaning that the 0.3333333% (i.e., .4/(.4+.8)) of the total 866703 individuals are in S_a1 and the remaining are in R_a1_nv_l4. For more details regarding the constraints in eqs.7 and the functions associated to the general transitions (in specific how the probabilities are used), see Sec. General transitions, eq. 1. 7. **target_value_fname**: since we are interested to calculate the PRCCs over the time w.r.t. the count of infects per year, the *Select.R* file provides the function to obtain the vector storing the total number of infections per year.

```
Select<-function(output)
{
  ynames <- names(output)
  col_names <- "(InfectCount_a){1}[0-9]{1}"
  col_idxs <- c( which(ynames %in% grep(col_names, ynames, value=T)) )
  # Reshape the vector to a row vector
  ret <- rowSums(output[,col_idxs])
```

```
  return(as.data.frame(ret))
}
```

8. **reference_data**: the Pertussis surveillance data from the 1974;

```
#>         1974  1975  1976 1977  1978  1979  1980
#> Infects 7413 10786 18354 8076 12582 18142 14170
```

9. **distance_measure_fname**: the squared error estimator via trajectory matching on the number of cases per year is implemented through the *msqd()* function.

```
msqd<-function(reference, output)
{
    ynames <- names(output)
    # InfectCount is the place that counts how many new infects occurs during the whole
    # period, for this reason we have to do the difference to obtain the number of cases
    # per year. Given this difference the squared error is calculated w.r.t. the reference
    # data.
    col_names <- "(InfectCount_a){1}[0-9]{1}"
    col_idxs <- c( which(ynames %in% grep(col_names, ynames, value=T)) )
    infects <- rowSums(output[,col_idxs])
    infects <- infects[-1]
    diff<-c(infects[1],diff(infects,differences = 1))
    ret <- sum(( diff - reference )^2 )
    return(ret)
}
```

Finally,

```
sensitivity_analysis(n_config = 2^14,
                     parameters_fname = "./input/Functions_list.csv",
                     functions_fname = "./Rfunction/Functions.R",
                     solver_fname = "./Net/Pertussis.solver",
                     f_time = 365*21,
                     s_time = 365,
                     timeout = "1d",
                     parallel_processors=40,
                     reference_data ="./input/reference_data.csv",
                     distance_measure_fname="./Rfunction/msqd.R",
                     target_value_fname="./Rfunction/Select.R"
                     )
```

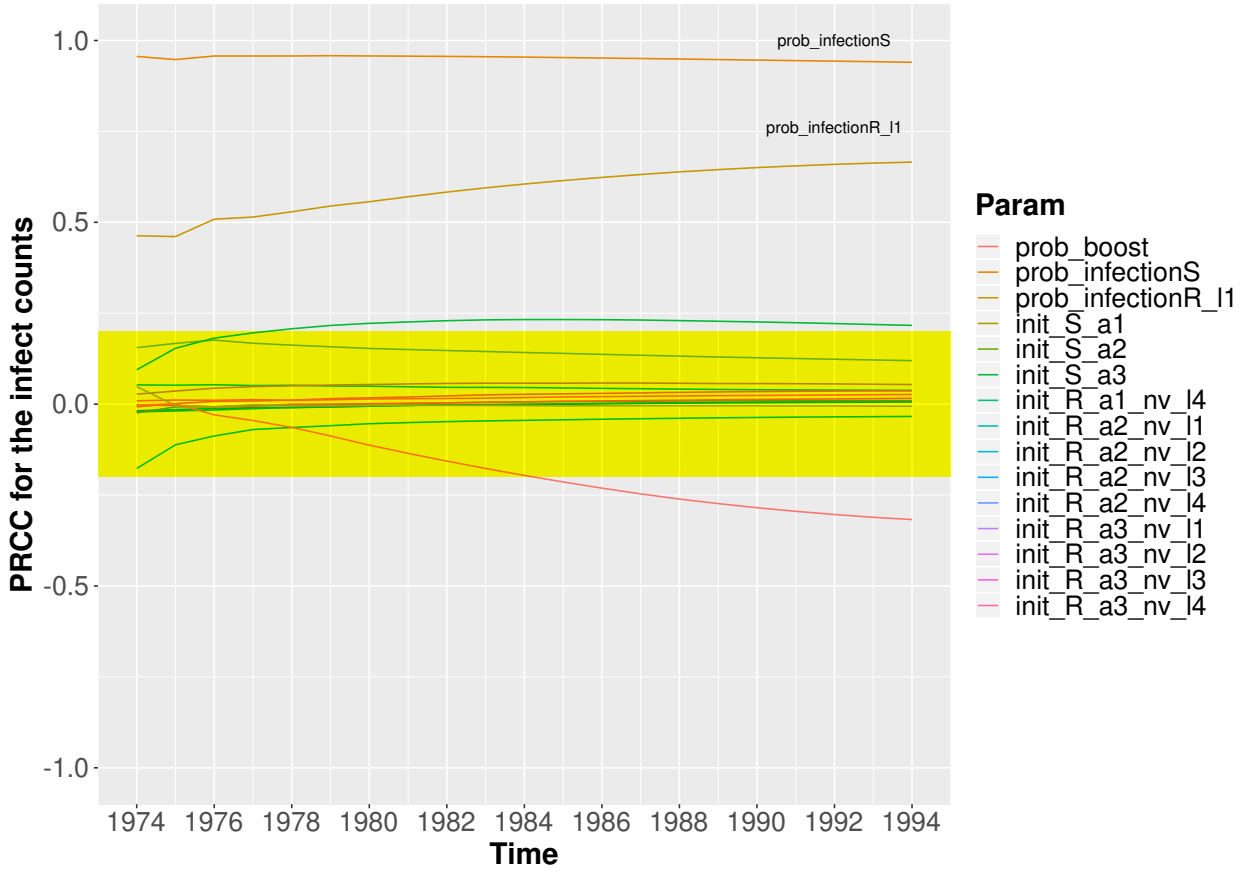

Figure 2: PRCCs values for the selected input parameters with respect the number of infections.

From figure 2 it is straightforward to argue that the *prob_infectionS* is the most important parameter affecting the *infects* behavior, followed by *prob_infectionR_l1*. Differently the *prob_boost* probability and the initial number of susceptible and recovered individuals in each age class are irrelevant with respect to the infection behavior.

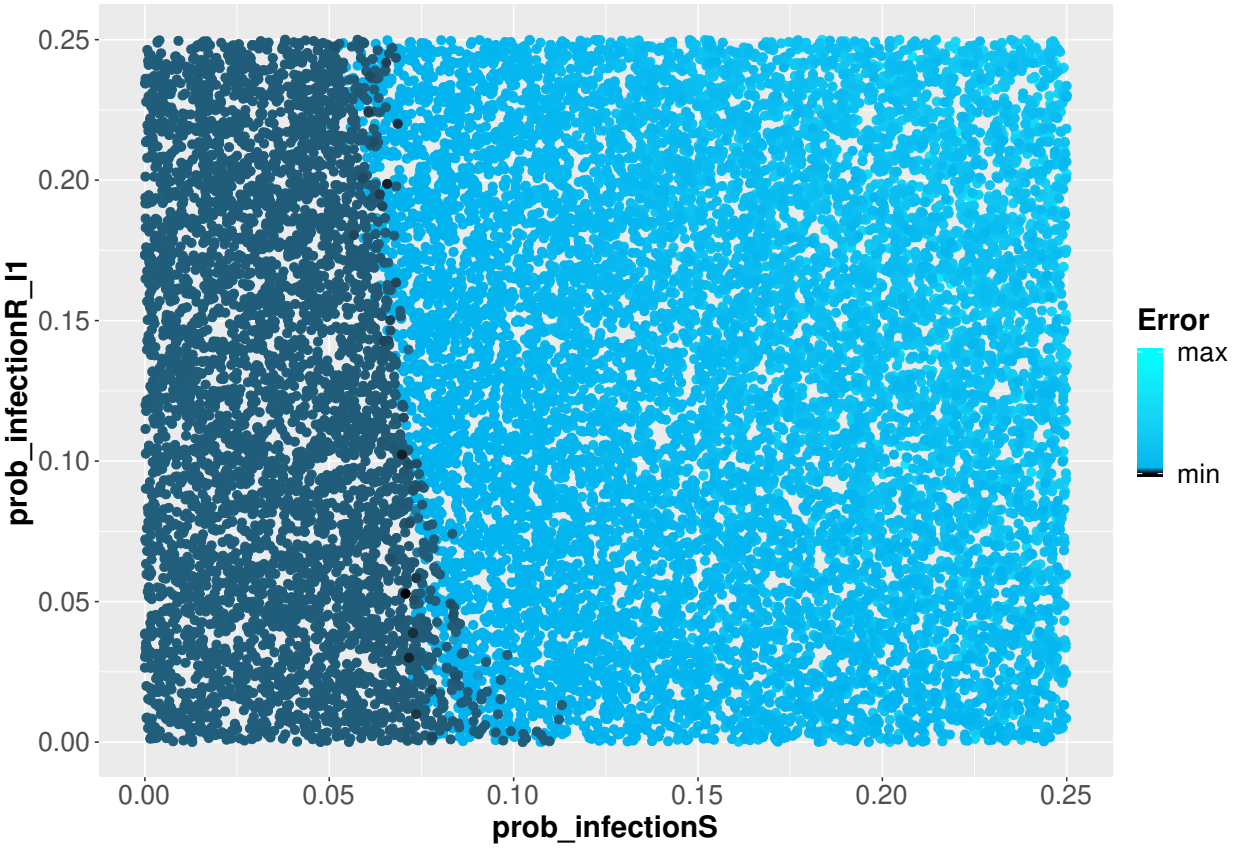

Figure 3: Scatter Plot showing the squared error between the real and simulated infection cases.

In figure 3, the squared error between the real and simulated infection cases from 1974 to 1994 are plotted varying the *prob_infectionS* parameter (on the x-axis) and *prob_infectionR_l1* parameter (on the y-axis). Each point is then colored according to a linear gradient function starting from color dark blue (i.e., lower value) and moving to color light blue (i.e., higher values). From this plot we can observe that higher squared errors are obtained when *prob_infectionS* assumes values greater than 0.09 (see light blue) or smaller than 0.06 (see the dark blue points, representing the parameters configuration with minimum error w.r.t. the real data, when *prob_infectionS* values are between $[0; 0.1]$). Therefore, according to this we shrunk the search space associated with the *prob_infectionS* parameter from $[0; 0.25]$ to $[0; 0.1]$ in the calibration phase.

Model Calibration

The aim of this phase is to adjust the model unknown parameters to have the best fit of simulated behaviors to the real data, i.e. the reference data. Firstly, the function *model_calibration()* is applied on the generated deterministic process to fit its behavior to the real infection data (from 1974 to 1994) using squared error estimator via trajectory matching implemented in the function *msqd()* passed as an input parameter. Note that the information derived by the sensitivity analysis is exploited to reduce, where it is possible, the number of parameters to be estimated and/or their search space.

```
model_calibration(parameters_fname = "./input/Functions_list.csv",
                  functions_fname = "./Rfunction/Functions.R",
                  solver_fname = "./Net/Pertussis.solver",
                  f_time = 365*21,
```

```
                  s_time = 365,
                  reference_data = "./input/reference_data.csv",
                  distance_measure_fname = "./Rfunction/msqd.R",
                  # Vectors to control the optimization
                  # init_V taken from trace id 13521
                  ini_v = c(0.05, 0.07, 0.1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0),
                  ub_v = c(0.25, 0.1, 0.25, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1),
                  lb_v = c(0, 0, 0, 1e-7, 1e-7, 1e-7, 1e-7, 1e-7, 1e-7, 1e-7,
                           1e-7, 1e-7, 1e-7, 1e-7, 1e-7),
                  ini_vector_mod = TRUE
                  )
```

```
#> Optimal configuration:
#>       prob_boost prob_infectionS prob_infectionR_l1 init_S_a1  init_S_a2
#> 87160 0.09328468      0.06742504         0.09999983 0.3362519 0.04485452
#>       init_S_a3 init_R_a1_nv_l4 init_R_a2_nv_l1 init_R_a2_nv_l2 init_R_a2_nv_l3
#> 87160  0.968471       0.3454245      0.02239408      0.01690299           1e-07
#>       init_R_a2_nv_l4 init_R_a3_nv_l1 init_R_a3_nv_l2 init_R_a3_nv_l3
#> 87160       0.9765406       0.3134586      0.06156321    0.0007210937
#>       init_R_a3_nv_l4
#> 87160    0.0001294989
```

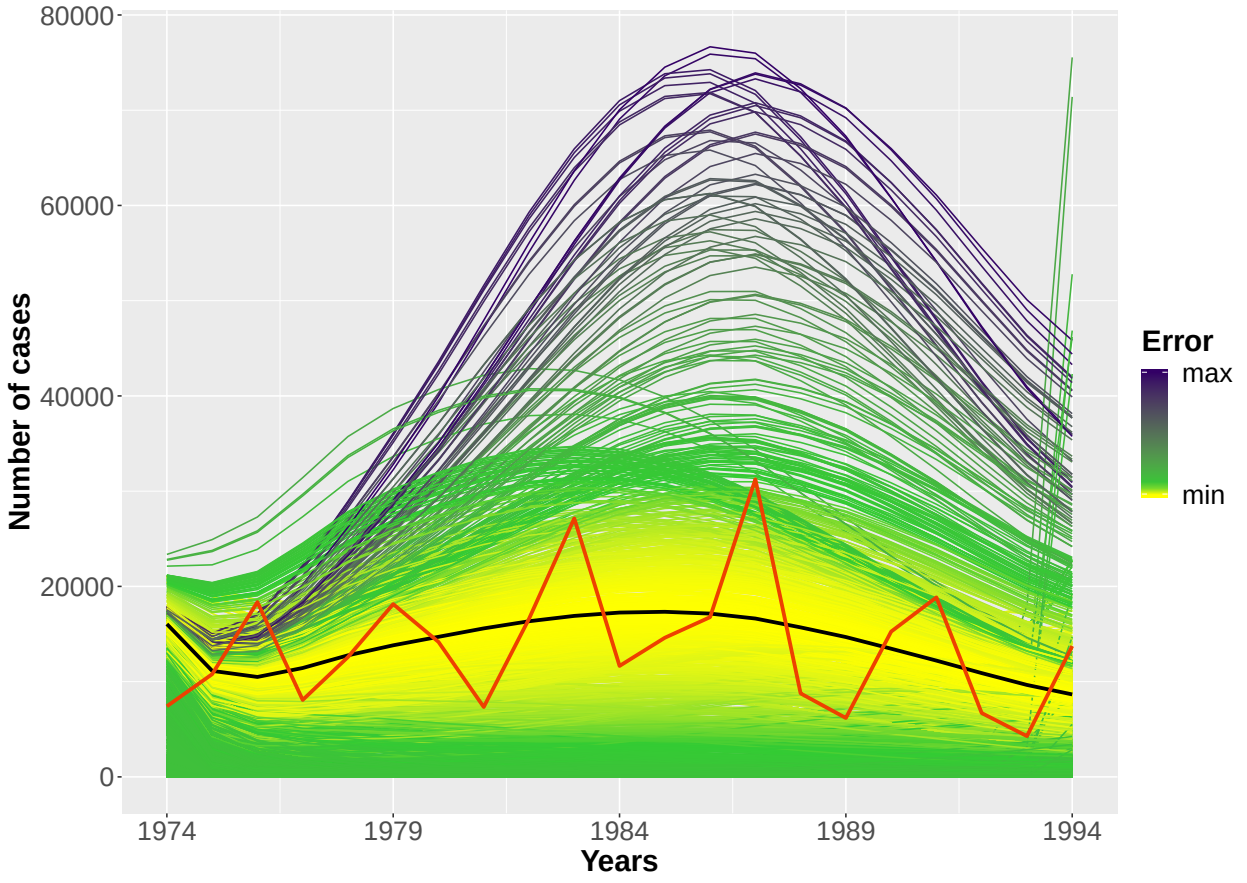

Figure 4: Model Calibration considering the deterministic model. Here a subset of the trajectories obtained from the optimization phase. The color of each trajectory depends on the squared error w.r.t. the Pertussis surveillance trend (red line). The black line is the optimal one.

Then, starting from the parameters configuration obtained from the deterministic model calibration, that is

`0.09328468, 0.06742504, 0.09999983, 0.3362519, 0.04485452, 0.968471, 0.3454245, 0.02239408, 0.01690299, 1e-07, 0.9765406, 0.3134586, 0.06156321, 0.0007210937, 0.0001294989`

the function *model_calibration()* is applied on the generated stochastic process to fit its behavior to the real infection data using Akaike Information Criterion (AIC) via trajectory matching, implemented in the R script *aic.R*. Let us note that the parameter search space of this second optimization step is computed from the result obtained from the previous step considering a 20% confidence interval around each parameter value.

```
model_calibration(parameters_fname = "./input/Functions_list.csv",
                  functions_fname = "./Rfunction/Functions.R",
                  solver_fname = "./Net/Pertussis.solver",
                  solver_type = "TAUG",
                  f_time = 365*21,
                  s_time = 365,
                  n_run = 250,
                  parallel_processors=40,
                  reference_data = "./input/reference_data.csv",
                  distance_measure_fname = "./Rfunction/aic.R",
                  ini_v = best_run,
                  ub_v = 1.2*best_run,
                  lb_v = .8*best_run,
                  ini_vector_mod = TRUE
                  )
```

```
#> [1] "Optimal configuration="
#>     prob_boost prob_infectionS prob_infectionR_l1 init_S_a1  init_S_a2
#> 726  0.1028314      0.06422971          0.1173022 0.3996977 0.04611963
#>     init_S_a3 init_R_a1_nv_l4 init_R_a2_nv_l1 init_R_a2_nv_l2 init_R_a2_nv_l3
#> 726  1.082384       0.2943042      0.02526472      0.01475493    8.009888e-08
#>     init_R_a2_nv_l4 init_R_a3_nv_l1 init_R_a3_nv_l2 init_R_a3_nv_l3
#> 726       0.9688963       0.3208456      0.06788778     0.000676801
#>     init_R_a3_nv_l4
#> 726    0.0001424256
```

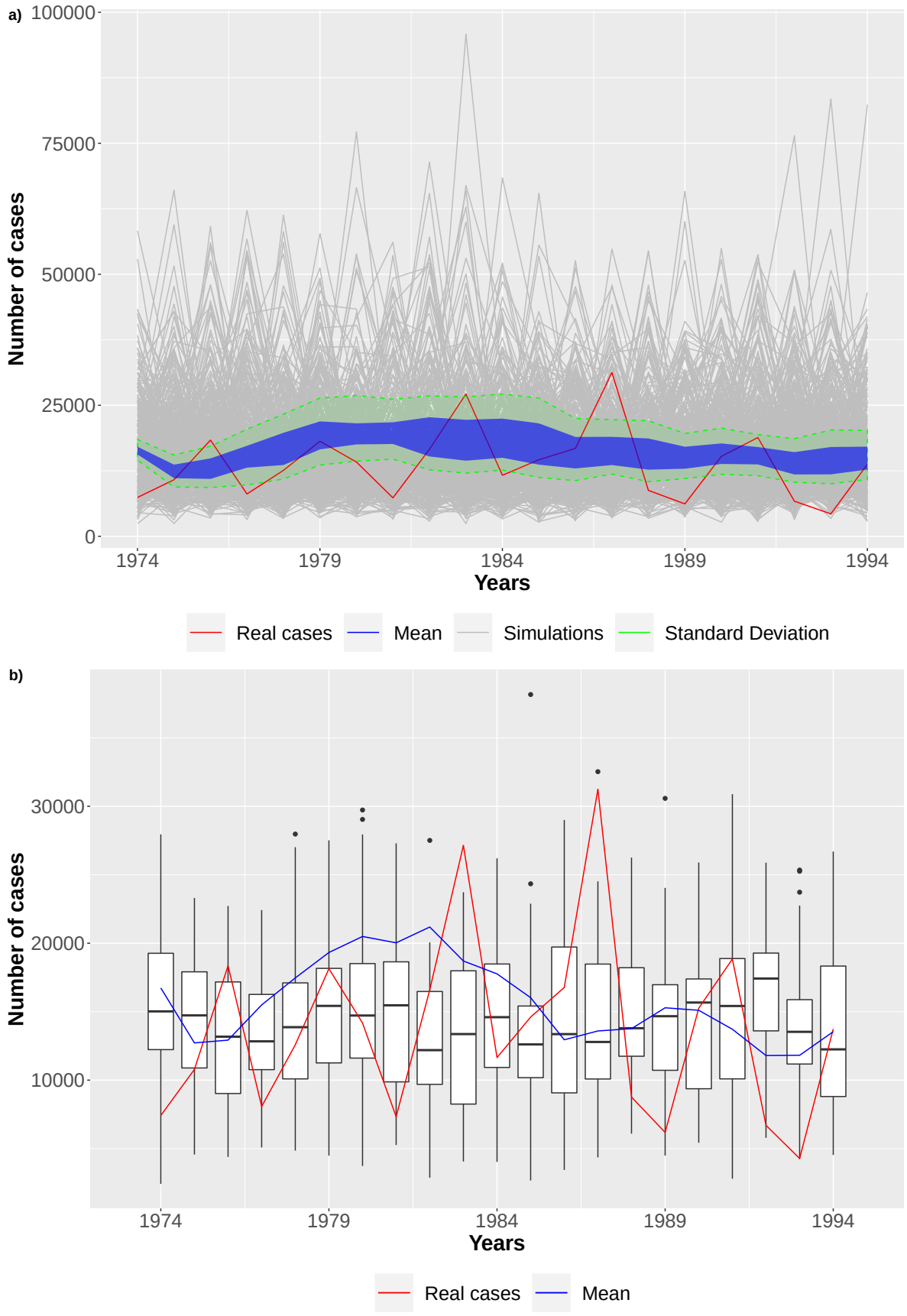

Figure 5: a) 25000 trajectories (grey) over the whole time interval are reported. b) Boxplots considering the best configuration.

Figure 5 shows trajectories (grey lines) for the 15 best parameters configurations discovered. The blue area contains the average trajectories derived for the first ten best parameter configurations, while the two green lines provide the associated confidence interval. We can observe that a good approximation of the surveillance data (red line) from the 1974 to 1994 is obtained.

Model Analysis

In this last phase of our workflow the user can analyse the calibrated model to answer specific questions and to derive new insights. In our case study we show a simple what-if analysis. In particular we investigate the impact of different vaccination failure probabilities with respect to the number of infection cases. The simulated time period is from 1974 to 2016, and the pertussis vaccination program is started in 1995, with an average vaccination coverage starts from 50% and transitions linearly to 95% in 8 years, ("Ministero Della Salute. Coperture Vaccinali." n.d., @Tozzi2014).

In details, the *model_analysis()* function is applied by exploiting the optimal parameters configuration derived from the calibration analysis on the stochastic model, namely *optim.stoch*.

```
model_analysis(solver_fname =  "./Net/Pertussis.solver",
               f_time = 365*43,
               s_time = 365,
               n_config = 1,
               n_run = 250,
               parallel_processors = 40,
               solver_type = "TAUG",
               parameters_fname = "./input/Functions_list.csv",
               functions_fname = "./input/Functions.R",
               ini_v = optim.stoch ,
               ini_vector_mod = TRUE)
```

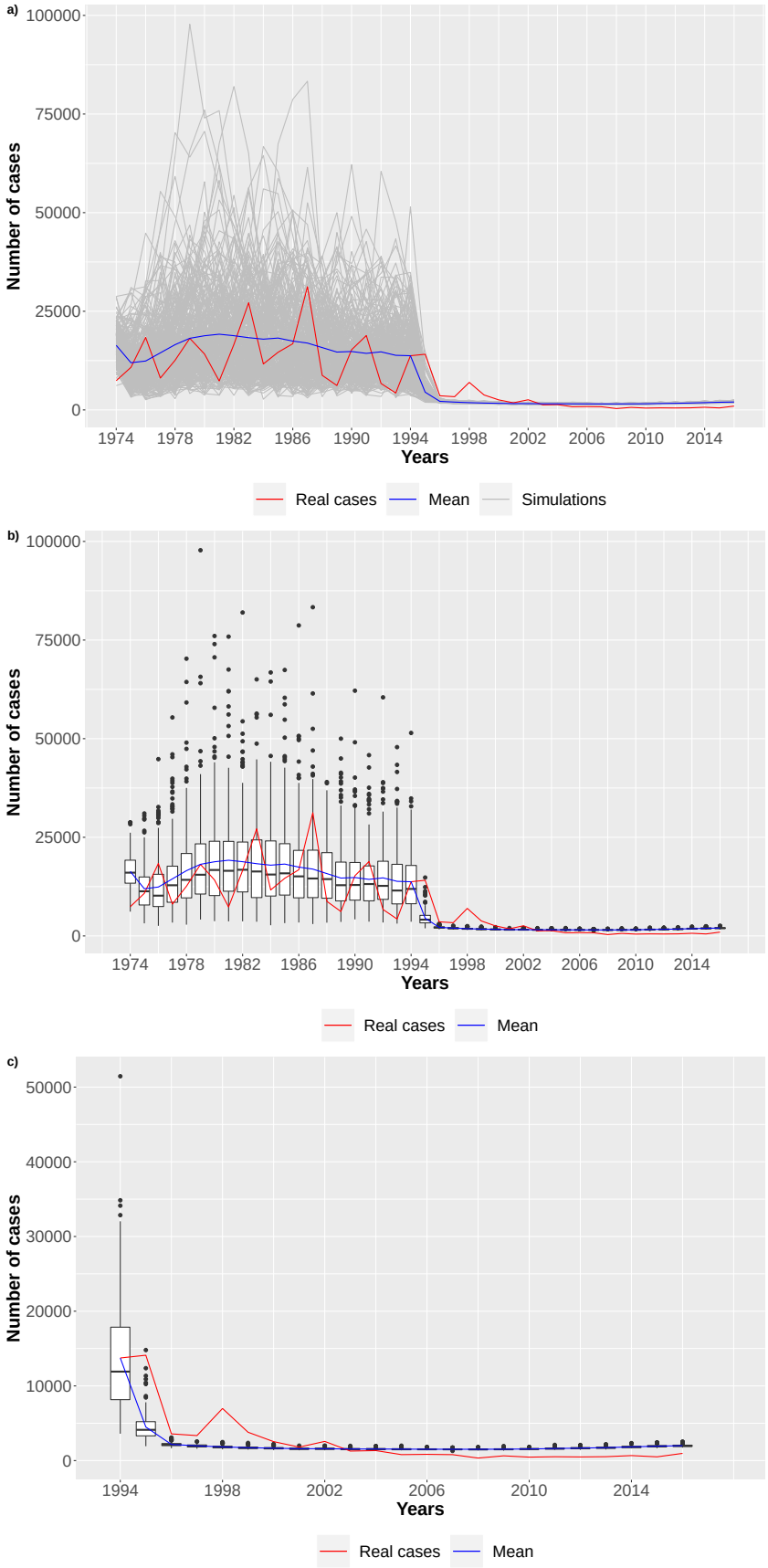

Figure 6: 250 trajectories (grey) considering the stochastic model. The blue dashed line is the mean trend and the red one the Pertussis surveillance.

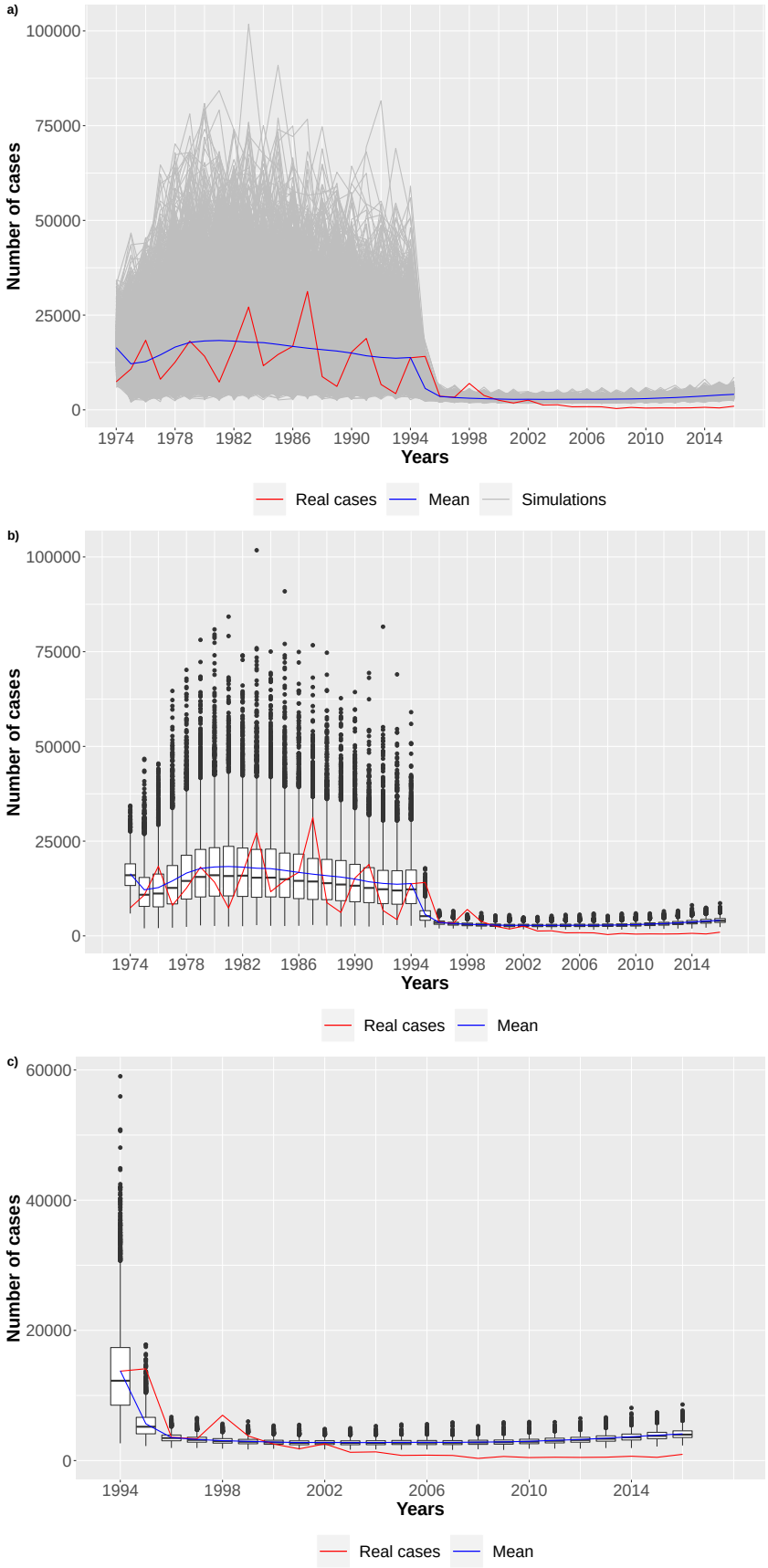

Figure 7: Probability of vaccine failure settled to .1.

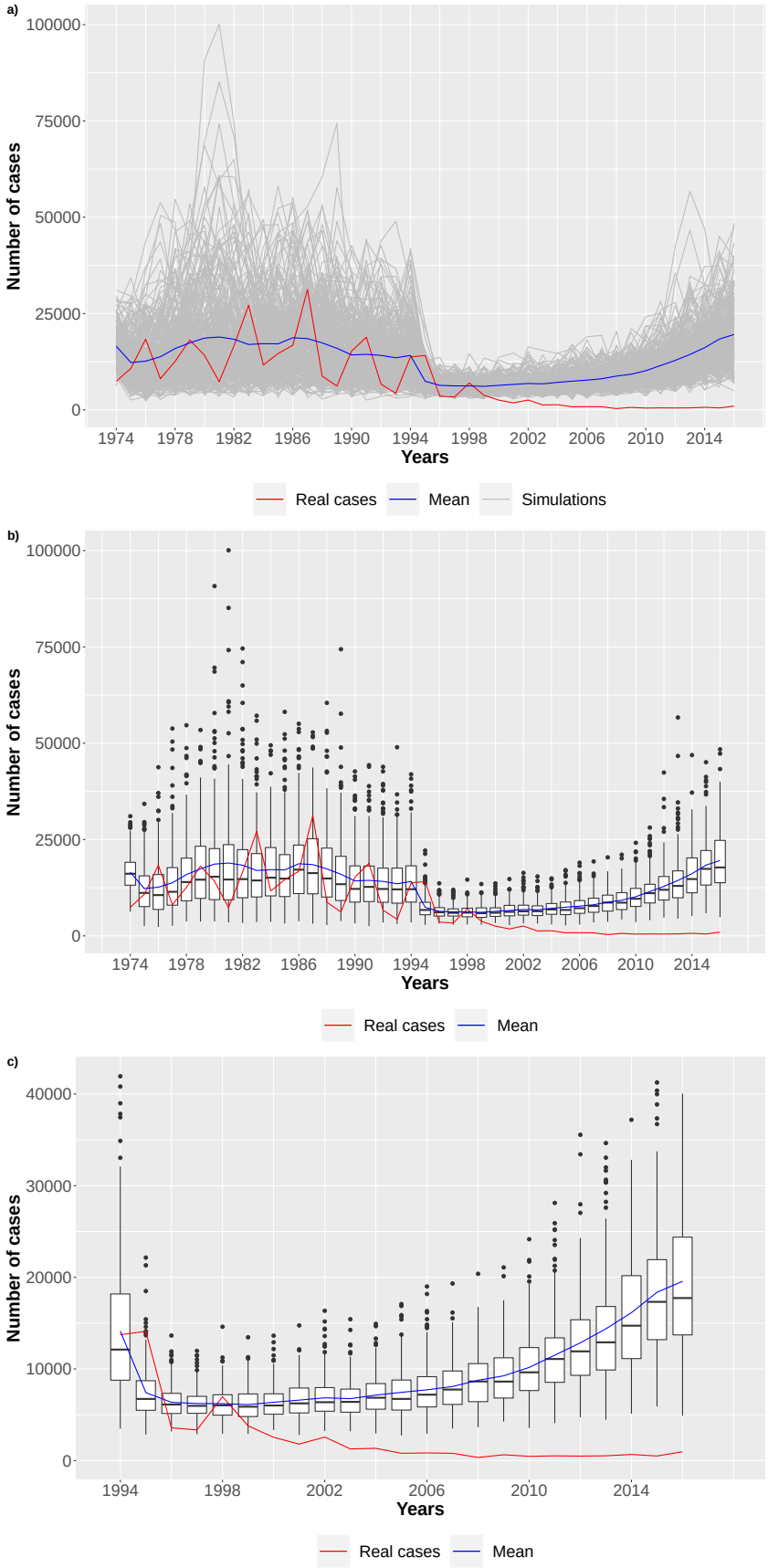

Figure 8: Probability of vaccine failure settled to .2.

Fig.6 shows the decreasing of number of infection cases after the starting of the vaccination policy, with dynamics comparable with the reference data. If we add to the model a vaccination failure probability, i.e. we add a fourth probability given by p_v (see Sec.General transitions, eq. 3), then we have to modify the vector returned by the function *probability()* implemented in *Functions.R* as follows:

```
probability <- function(file, x = NULL)
{
    load(file)
    if( is.null(x) ){
        x <- runif(n = length(probabilities), min=0, max=0.25)
        x[length(x)] = 0
    }
    else{
    ##### Here we add the vaccination failure probability: p_v
        x <- c(x[c(1:3)],p_v)
    }
    return(matrix(x, ncol = 1))
}
```

In figures 7 and 8 we show how the number of infection cases is affected by the increasing vaccination failure probabilities from 0.1 to 0.20. We can observe that only probabilities greater than 0.10 have an effect on the number of infection cases.

Figures 6, 7, and 8 show a) 250 trajectories (grey) considering the stochastic model over the whole time interval. The blue dashed line represents the mean trend; the red line represents the Pertussis surveillance trend. In the picture b) the boxplots over the time period are plotted, and in c) the zoom considering the last 21 years is reported.

References

Gonfiantini, M V, E Carloni, F Gesualdo, E Pandolfi, E Agricola, E Rizzuto, S Iannazzo, M L Ciofi Degli Atti, A Villani, and A E Tozzi. 2014. "Epidemiology of Pertussis in Italy: Disease Trends over the Last Century." *Eurosurveillance* 19 (40).

"Ministero Della Salute. Coperture Vaccinali." n.d.

Mossong, Niel AND Jit, Joël AND Hens. 2008. "Social Contacts and Mixing Patterns Relevant to the Spread of Infectious Diseases." *PLOS Medicine* 5 (3): 1–1. https://doi.org/10.1371/journal.pmed.0050074.

Veiga Leprevost, Felipe da, Björn A Grüning, Saulo Alves Aflitos, Hannes L Röst, Julian Uszkoreit, Harald Barsnes, Marc Vaudel, et al. 2017. "BioContainers: an open-source and community-driven framework for software standardization." *Bioinformatics* 33 (16): 2580–2.