epidemix—An interactive multi-model application for teaching and visualizing infectious disease transmission

Ulrich Muellner[a],[1], Guillaume Fournié[b],[*],[1], Petra Muellner[a], Christina Ahlstrom[a], Dirk U. Pfeiffer[b],[c]

[a] Epi-interactive, P.O. Box 15327, Miramar, Wellington, 6243, New Zealand
[b] Veterinary Epidemiology, Economics and Public Health Group, Pathobiology and Population Sciences Department, Royal Veterinary College, Hatfield, AL9 7TA, UK
[c] School of Veterinary Medicine, To Yuen Building, 31 To Yuen Street, City University of Hong Kong, Kowloon, Hong Kong

ARTICLE INFO

ABSTRACT

Mathematical models of disease transmission are used to improve our understanding of patterns of infection and to identify factors influencing them. During recent public and animal health crises, such as pandemic influenza, Ebola, Zika, foot-and-mouth disease, models have made important contributions in addressing policy questions, especially through the assessment of the trajectory and scale of outbreaks, and the evaluation of control interventions. However, their mathematical formulation means that they may appear as a "black box" to those without the appropriate mathematical background. This may lead to a negative perception of their utility for guiding policy, and generate expectations, which are not in line with what these models can deliver. It is therefore important for policymakers, as well as public health and animal health professionals and researchers who collaborate with modelers and use results generated by these models for policy development or research purpose, to understand the key concepts and assumptions underlying these models.

The software application epidemix (http://shinyapps.rvc.ac.uk) presented here aims to make mathematical models of disease transmission accessible to a wider audience of users. By developing a visual interface for a suite of eight models, users can develop an understanding of the impact of various modelling assumptions – especially mixing patterns – on the trajectory of an epidemic and the impact of control interventions, without having to directly deal with the complexity of mathematical equations and programming languages. Models are compartmental or individual-based, deterministic or stochastic, and assume homogeneous or heterogeneous-mixing patterns (with the probability of transmission depending on the underlying structure of contact networks, or the spatial distribution of hosts). This application is intended to be used by scientists teaching mathematical modelling short courses to non-specialists – including policy makers, public and animal health professionals and students – and wishing to develop hands-on practicals illustrating key concepts of disease dynamics and control.

1. Introduction

Mathematical models of disease transmission conceptualise the spread of infectious agents within single or multiple host populations using mathematical language (Keeling and Rohani, 2007; Vynnycky and White, 2010). They are used to generate insights into the factors influencing disease dynamics, to identify knowledge gaps and to guide subsequent data collection. They are also used to assess the trajectory and scale of disease outbreaks, the impact of mitigation strategies and the uncertainty surrounding these outputs. By improving our understanding of how host, pathogen and environmental factors drive transmission, they have been transformative in our ability to control human and animal diseases (Heesterbeek et al., 2015), as well as plant

diseases (Cunniffe et al., 2015), and have had, within the last decades, an increasingly important role in guiding the design of policies and outbreak responses in both the public and animal health sectors (Cauchemez et al., 2016; Fraser et al., 2009; Houben et al., 2016; Keeling et al., 2001; WHO Ebola Response Team, 2014).

It has been highlighted that effective communication between epidemiologists and decision-makers is particularly critical during an outbreak, as is transparency to facilitate the acceptance of model outputs (Pfeiffer, 2010). However, as new data and computational resources become available, models are becoming increasingly complex, and require specific mathematical and computational expertise for implementation and interpretation. Their mathematical formulation makes their assumptions explicit, which requires an understanding of

---

the relevant mathematical methodologies. As a result, a model and its assumptions might be difficult to comprehend by those without the specialist mathematical knowledge. This seriously impedes efforts in communicating model results and their associated uncertainty and limits public and animal health professionals and policymakers' capacity to use such models.

It is therefore important for policymakers, as well as public and animal health professionals and researchers who collaborate with modelers and use results generated by these models for policy development or research purpose, to understand the underlying key concepts and assumptions. Several user-friendly software packages are available to non-specialists for developing mathematical models without requiring users to specify mathematical equations or be proficient in any programming language (e.g. Berkeley Madonna). Mastering these software packages may nonetheless require several hours to become familiar with their interface, which represents an obstacle to the development of short courses. Also, their user-friendly interface is often limited to the development of compartmental models; the development of individual-based models with more complex inter-host mixing patterns generally requires the user to specify mathematical equations. Further, visualization and analytic tools available for infectious disease epidemiology were recently summarized (Carroll et al., 2014), with the authors concluding that interactive graphics with dynamic features facilitate the exploration of complex data and are highly desired by the user.

In this paper, we present a mathematical modelling application, *epidemix*. Our objective was to develop a teaching tool allowing users to learn about key concepts of infectious disease dynamics, and to obtain an understanding of the impact of various modelling assumptions – especially mixing patterns – on the trajectory of an epidemic and the impact of control interventions. Knowledge in mathematics is not a prerequisite, and the step-by-step specification of model structure and parameters do not necessitate learning how to use the application or the reading of a user manual. Graphics and animations capturing changing outputs are produced as input variables and model settings are modified. This application is intended to be used by scientists teaching mathematical modelling short courses to non-specialists – including policy makers, public and animal health professionals and undergraduate students – and wishing to develop hands-on practicals illustrating key concepts of infectious disease dynamics and control.

2. Methods and results

2.1. Software

For easy access, the tool was developed to be available via the internet with a standard browser (http://shinyapps.rvc.ac.uk) without the installation of specialist software. RStudio Shiny (Chang et al., 2015) was selected as the base development framework which enabled publishing to the web as well as offline capability by using RStudio. The models were coded as R scripts. Each model code was wrapped into an R function which was called by the server-side script (server.R) of the RStudio Shiny application. This enabled a separation between the interface and the computation of the models. For example, the code of a model could be easily changed without affecting the application. To enable some of the functionality and visual elements of the core RStudio Shiny, functionality was extended by using JavaScript and CSS. In particular, NVD3 line charts were integrated to create dynamic graph outputs for the infection states over time.

The following R packages were utilised: Shiny (Chang et al., 2015), network (Butts, 2008), networkD3 (Allaire et al., 2017), ggplot2 (Wickham, 2009), GGally (Schloerke et al., 2016), sna (Butts, 2010), spatstat (Baddeley et al., 2015), nvd3, deSolve (Soetaert et al., 2010) and adaptivetau (Johnson, 2014).

The following functional elements were utilised:

- Reactivity: this core concept of RStudio Shiny enabled a dynamic adjustment of the model parameters triggering a re-calculation of the model data and outputs. Where possible, outputs were cached for faster computation.
- Isolation: The graphical outputs for the deterministic models could be displayed instantly after a parameter was changed by the user. As the stochastic models run through several simulations the time to compute the underlying data is significantly longer, therefore an "Update graph" button was introduced which just triggered a re-calculation of the model when a parameter changed. These input values were "isolated" from being reactive in this context. Certain input controls, e.g. time sliders or controls to select a certain simulation, were still reactive as they didn't require a re-calculation of the model.
- Progress indicators: For the stochastic models a progress bar was added providing the user with a visual indicator that a calculation is in progress. The progress bar was linked to the simulations to be processed.
- Download data: Model outputs were available for download in CSV format.
- Default values: On initial launch of the tool default parameter values were applied which represent a starting point for further adjustments.

Model formulation and R scripts were reviewed. The interface was tested by users of different background. This pilot testing was conducted to identify programming errors, inconsistency or application components that were unclear to the user. Feedback was accounted for to create the first version of *epidemix*, described here.

3.1. Model and parameter options

The application was designed such that users can choose from a suite of eight different model types. Additional details about the models and parameters are described in the Supplementary Material. In brief, all models simulate disease spread through a population of hosts. After having selected the type of model to be used, the user can define the characteristics of the host population, the infection states to be taken into account, the control strategies to be tested, and the associated model parameters. Parameters only become available after the appropriate model and model structure have been selected. Information buttons provide explanations of each model option and parameter.

3.2. Models

Models are either compartmental or individual-based, deterministic or stochastic, and assume homogeneous or heterogeneous-mixing patterns (Table 1). In the following, "unit" refers to the unit modelled for the host population of interest. For instance, this unit can be an individual if the host population of interest is a human population, or a cow or a cattle farm if the spread of a disease is simulated within a cattle farm or a group of cattle farms, respectively. In compartmental

Table 1
The eight model types included in the epidemix application.

Model	Deterministic/ Stochastic	Compartmental/Individual-based	Mixing
1	Deterministic	Compartmental	Homogeneous
2	Deterministic	Compartmental	Two-group
3	Stochastic	Compartmental	Homogeneous
4	Stochastic	Compartmental	Two-group
5	Stochastic	Individual-based	Homogeneous
6	Stochastic	Individual-based	Two-group
7	Stochastic	Individual-based	Network
8	Stochastic	Individual-based	Spatial

models, units are grouped within "compartments" within which all units have the same characteristics, and the number of units within each compartment is tracked over time. In contrast, each unit is explicitly modelled and its individual infection status followed over time in an individual-based model. While deterministic models describe the expected average trajectory of an epidemic and do not account for the effects of chance, stochastic models allow taking into account the random nature of transmission events. In contrast to a deterministic model, a stochastic model will not produce a single outcome for a given set of parameter, but a distribution. Stochastic and deterministic compartmental models are implemented in continuous time, and individual-based models in discrete time. Several contact mixing patterns can be implemented.

(1) Under "homogeneous mixing", all units have the same average number of effective contacts with other units per unit of time.
(2) Under "heterogeneous mixing (or two-groups)", the host population is divided into two groups (e.g. according to their social status, age, species, geographical location). The average number of effective contacts made within and between each group can differ.
(3) Each unit can be treated as a node in a network of contacts. Disease transmission between two given units is, therefore, constrained by the requirement of an existing link between them.
(4) Each unit can be given a fixed spatial location. The probability of transmission is a function of the Euclidian distance between units.

3.3. Infection states

A unit can pass through an ordered sequence of infection states: Susceptible (S), Infected (latent, E), Infectious and asymptomatic (Ia), Infectious and symptomatic (Is), Recovered (R). The user can specify the particular sequence of infection states to be considered in the model so that the characteristics of different infectious diseases can be represented, with the restriction that the infection state "Susceptible" has to be included, and one of the two infectious states (asymptomatic and/or symptomatic). Susceptible units are free of the disease, but can become infected. Infected units are incapable of transmitting the infection, as distinct from infectious units, which are infected and capable of transmitting the infection. A distinction is made between asymptomatic and symptomatic units as, in the case of the spread of animal diseases, only symptomatic units (animals or farms) are assumed to be detectable and therefore available for potential culling as part of a control strategy. At the end of the infectious period, units can either recover, or be removed from the population (e.g. an individual or an animal dying from an infection, a farm being depopulated following infection). If all or some infectious units are removed from the population at the end of their infectious period, removed units can be replaced with susceptible units immediately, after a period of time (e.g. depopulated farms are authorized to repopulate after a downtime period), or not at all. In the latter case, the population size would decrease. In a compartmental model, the rates of transition between infected states (E, Ia, Is and R) are constant, meaning that the time spent by units within these compartments (e.g. latent period, symptomatic infectious period) follow an exponential distribution. In contrast, individual-based models are implemented such that the risk of transitioning between infected states depends on the time since infection: units transit between these infected states (from E to Ia, from Ia to Is, and from Is to R) after a fixed period of time.

3.4. Characteristics of the host population

The host population can be closed or open. Units may leave the population for reasons other than those associated with infection (e.g. deaths unrelated to infection, emigration, sale of animal(s), end of production cycle on a farm), and be immediately replaced by a new unit (e.g. birth, immigration, purchase of animal(s), start of a new

production cycle in a farm). As mentioned above, the mixing pattern between units within the modelled host population can be homogeneous or heterogeneous. In the case of network-mixing, the network structure is assumed to be stable over time, and the distribution of contacts is fixed by the user. Random networks are generated using the Erdos Renyi model (Erdös and Rényi, 1960), and heterogeneous networks are generated by the Barabasi-Albert model (Barabasi and Albert, 1999; Csardi and Nepusz, 2006), resulting in a Poisson or a right-skewed distribution of contacts among the units, respectively. In the spatial-mixing model, each unit is given a fixed spatial location within a square area. Units can be distributed randomly within the area, or clustered spatial distribution can be generated by the user (Baddeley et al., 2015).

3.5. Control strategies

Two control strategies can be implemented: vaccination or culling. In the network-mixing model, units can be either (1) randomly chosen for vaccination, or (2) vaccination can be targeted towards units with the highest number of contacts. To control infection spreading between farms (or animals), symptomatic infected farms can be actively depopulated, and, therefore, removed from the population. In the spatial model, it is possible to simulate the implementation of ring culling. Once a unit (e.g. farm) is found to be infected and removed, all other farms located within a given radius of the case are also removed, regardless of their actual infection state.

3.6. Outputs

The core functionality of "*epidemix*" is described on the application's home page (Fig. 1), illustrating step-by-step parameter selection for eight different model types and data visualization and export options. Data are visualized as line charts, data matrices, network diagrams and tables, all of which can be exported in pdf and/or csv file format.

Data outputs are immediately updated once input parameters have been modified, with both inputs and outputs visualized on the same screen (Fig. 2). Each infection state is color-coded for easy differentiation in line charts, networks, and matrices and simulations can be manually adjusted or automatically played to explore infection state changes over time.

For stochastic models, optional variability graphs show the mean number of units per infection state including 10th and 90th percentiles over all simulations for a given set of parameters. Network specific visualization enables users to monitor infection state changes of individual units over time based on their connections to neighboring units (Fig. 3A). Similarly, spatial specific visualization options encourage users to explore individual time steps and simulations according to the geographical position of the units (Fig. 3B).

3.7. User guide

User guidance is a key element of "*epidemix*" and context-sensitive information is provided on all screens to support use of the tool. Information buttons provide definitions of input parameters and dependencies, which aids user understanding of each parameter and what options are available (Fig. 4). This help text is also available in the Supplementary Material.

4. Discussion

An easy-to-use interactive interface was developed that allows users to select specific infectious disease models and carry out calculations based on a variety of input parameters. The unified access to different modelling approaches under the same visualization platform allows users to compare assumptions, parameters, and differences in outputs given different conditions, allowing users to understand key concepts of

infectious disease dynamics, and to appreciate the impact of various modelling assumptions as well as the impact of control interventions. An important feature of "*epidemix*" is that, whilst easy to use, the epidemiologically essential complexities of the model remain intact. Simplifications and abstractions of data have the potential to improve communication and minimize cognitive overload (Carroll et al., 2014), but it can also oversimplify data leading to uncertainty (Karlsson et al., 2013). The transparency of the models presented allows users with a range of levels of expertise to explore how transmission events are affected by changing model structure and parameters. The results of

multiple stochastic model simulations can be visualized as independent outputs, allowing the users to observe the inherent variability in simulation outputs.

For deterministic models, data outputs are instantaneously updated as input parameters are modified, accentuating differences in input parameters and control measures. This is a significant improvement over applications that require submission of each new set of parameters, as dynamic visualizations can improve users' pattern recognition and accelerate comprehension of complex data that would traditionally be presented in a static format (Folorunso and Shawn
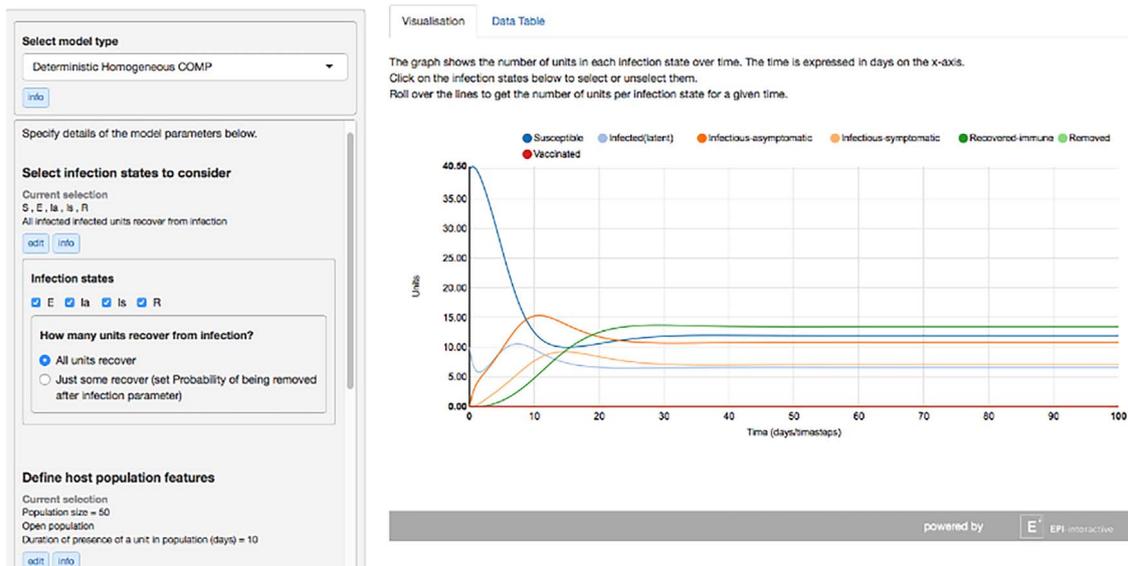
Fig. 2. *epidemix* screenshot displaying input parameter settings (left) and line graph showing the number of units in each infection state over time (right).
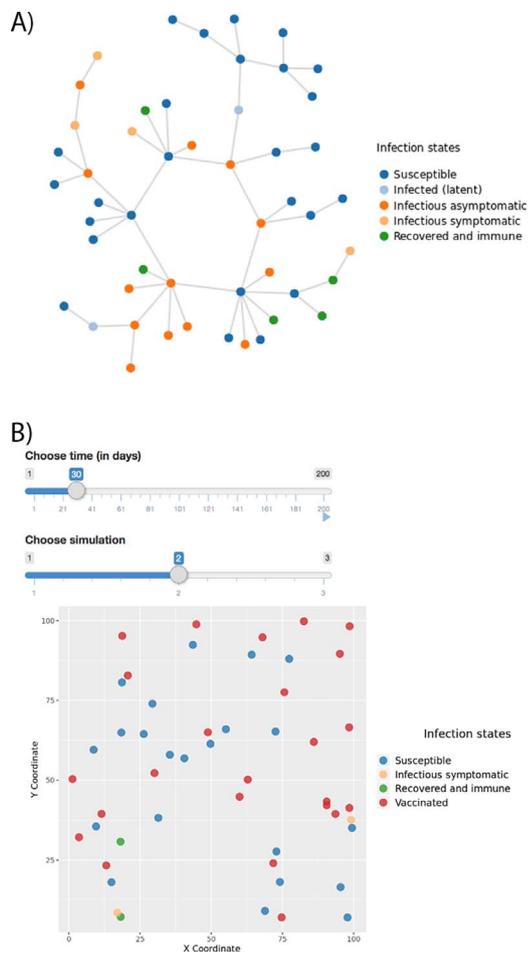
Fig. 3. Example visualization of a stochastic network IBM (A) and spatial IBM (B) model simulation in *epidemix*.
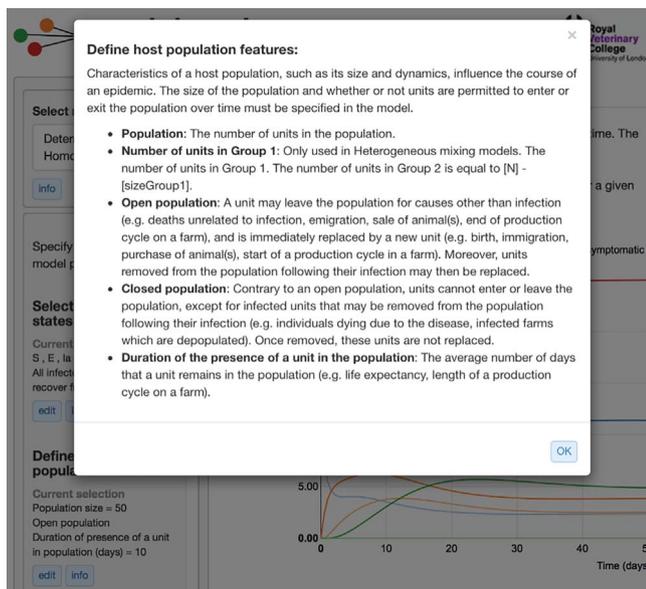
Fig. 4. Example *epidemix* info button description of host population features.

Ogunseye, 2008).

In *epidemix*, the complex interactions and dependencies of model parameters are clearly explained, enabling users with limited mathematical and computational backgrounds to assess how, for example, vaccination affects the spread of disease under different mixing patterns

of the population at risk. Different models are visualized using the same interface, unifying different approaches and making it easier to use. *epidemix* provides context-sensitive support (Aleven et al., 2003), where relevant information is provided on-screen while users are choosing parameters and testing models. This is preferable to providing information in a separate guide book or manual that may interrupt the flow of thinking.

A similar approach has been used by others for interactive tools in animal health developed for similar purposes, which highlights the need for tools like *epidemix*. For example, the communication and visual exploration of technical animal health information were demonstrated using animal health and surveillance data in Switzerland (Muellner et al., 2015) and influenza surveillance data in Hong Kong (Cheng et al., 2011). Tools are available to analyze and visualize disease transmission in both human and animal populations; however, these are typically developed for expert users and research purpose. For instance, OutbreakTools, a platform implemented in R, was developed to analyze and visualize outbreak data (Jombart et al., 2014), including a simulation function that generates outbreak data using a standard SIR model. *epidemix* aims to provide a platform for non-specialists to develop a better understanding of the assumptions on which mathematical models rely.

Funding

Acknowledgements

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.epidem.2017.12.003.

References

Aleven, V., Stahl, E., Schworm, S., Fischer, F., Wallace, R., 2003. Help seeking and help design in interactive learning environments. Rev. Educ. Res. 73, 277–320.

Allaire, J.J., Gandrud, C., Russell, K., Yetman, C.J., 2017. NetworkD3: D3 JavaScript Network Graphs from R. R Package Version 0.3.

Baddeley, A., Rubak, E., Turner, R., 2015. Spatial Point Patterns: Methodology and Applications with R. Chapman and Hall/CRC Press, London, pp. 2015. http://www.crcpress.com/Spatial-Point-Patterns-Methodology-and-Applications-with-R/Baddeley-Rubak-Turner/9781482210200/.

Barabasi, A.L., Albert, R., 1999. Emergence of scaling in random networks. Science 286, 509–512.

Butts, C.T., 2008. Network: a package for managing relational data in r. J. Stat. Softw. 24, 1–36.

Butts, C.T., 2010. Sna: Tools for Social Network Analysis.

Carroll, L.N., Au, A.P., Detwiler, L.T., Fu, T.C., Painter, I.S., Abernethy, N.F., 2014. Visualization and analytics tools for infectious disease epidemiology: a systematic review. J. Biomed. Inf. 51, 287–298.

Cauchemez, S., Nouvellet, P., Cori, A., Jombart, T., Garske, T., Clapham, H., Moore, S., Mills, H.L., Salje, H., Collins, C., Rodriquez-Barraquer, I., Riley, S., Truelove, S., Algarni, H., Alhakeem, R., AlHarbi, K., Turkistani, A., Aguas, R.J., Cummings, D.A., Van Kerkhove, M.D., Donnelly, C.A., Lessler, J., Fraser, C., Al-Barrak, A., Ferguson, N.M., 2016. Unraveling the drivers of MERS-CoV transmission. Proc. Natl. Acad. Sci. U. S. A. 113, 9081–9086.

Chang, W., Cheng, J., Allaire, J., Xie, Y., McPherson, J., 2015. Shiny: Web Application Framework for R. R Packag Version 011.

Cheng, C.K., Ip, D.K., Cowling, B.J., Ho, L.M., Leung, G.M., Lau, E.H., 2011. Digital

dashboard design using multiple data streams for disease surveillance with influenza surveillance as an example. J. Med. Internet Res. 13, e85.

Csardi, G., Nepusz, T., 2006. The igraph software package for complex network research. InterJournal. Complex Syst. 1695, 1–9.

Cunniffe, N.J., Stutt, R.O., DeSimone, R.E., Gottwald, T.R., Gilligan, C.A., 2015. Optimising and communicating options for the control of invasive plant disease when there is epidemiological uncertainty. PLoS Comput. Biol. 11, e1004211.

Erdös, P., Rényi, A., 1960. On the evolution of random graphs. Publ. Math. Inst. Hung. Acad. Sci. 5, 17–61.

Folorunso, O., Shawn Ogunseye, O., 2008. Challenges in the adoption of visualization system: a survey. Kybernetes 37, 1530–1541.

Fraser, C., Donnelly, C.A., Cauchemez, S., Hanage, W.P., Van Kerkhove, M.D., Hollingsworth, T.D., Griffin, J., Baggaley, R.F., Jenkins, H.E., Lyons, E.J., Jombart, T., Hinsley, W.R., Grassly, N.C., Balloux, F., Ghani, A.C., Ferguson, N.M., Rambaut, A., Pybus, O.G., Lopez-Gatell, H., Alpuche-Aranda, C.M., Chapela, I.B., Zavala, E.P., Guevara, D.M.E., Checchi, F., Garcia, E., Hugonnet, S., Roth, C., Coll, W.R.P.A., 2009. Pandemic potential of a strain of influenza a (H1N1): early findings. Science 324, 1557–1561.

Heesterbeek, H., Anderson, R.M., Andreasen, V., Bansal, S., De Angelis, D., Dye, C., Eames, K.T., Edmunds, W.J., Frost, S.D., Funk, S., Hollingsworth, T.D., House, T., Isham, V., Klepac, P., Lessler, J., Lloyd-Smith, J.O., Metcalf, C.J., Mollison, D., Pellis, L., Pulliam, J.R., Roberts, M.G., Viboud, C., Isaac Newton Institute, I.D.D.C., 2015. Modeling infectious disease dynamics in the complex landscape of global health. Science 347, aaa4339.

Houben, R.M., Lalli, M., Sumner, T., Hamilton, M., Pedrazzoli, D., Bonsu, F., Hippner, P., Pillay, Y., Kimerling, M., Ahmedov, S., Pretorius, C., White, R.G., 2016. TIME impact – a new user-friendly tuberculosis (TB) model to inform TB policy decisions. BMC Med. 14, 56.

Johnson, P., 2014. Adaptivetau: Tau-leaping Stochastic Simulation. R Package

Version 2.2.

Jombart, T., Aanensen, D.M., Baguelin, M., Birrell, P., Cauchemez, S., Camacho, A., Colijn, C., Collins, C., Cori, A., Didelot, X., Fraser, C., Frost, S., Hens, N., Hugues, J., Hohle, M., Opatowski, L., Rambaut, A., Ratmann, O., Soubeyrand, S., Suchard, M.A., Wallinga, J., Ypma, R., Ferguson, N., 2014. OutbreakTools: a new platform for disease outbreak analysis using the R software. Epidemics 7, 28–34.

Karlsson, D., Ekberg, J., Spreco, A., Eriksson, H., Timpka, T., 2013. Visualization of infectious disease outbreaks in routine practice. Stud. Health Technol. Inf. 192, 697–701.

Keeling, M.J., Rohani, P., 2007. Modeling Infectious Diseases in Humans and Animals. Princeton University Press, UK.

Keeling, M.J., Woolhouse, M.E., Shaw, D.J., Matthews, L., Chase-Topping, M., Haydon, D.T., Cornell, S.J., Kappey, J., Wilesmith, J., Grenfell, B.T., 2001. Dynamics of the 2001 UK foot and mouth epidemic: stochastic dispersal in a heterogeneous landscape. Science 294, 813–817.

Muellner, U.J., Vial, F., Wohlfender, F., Hadorn, D., Reist, M., Muellner, P., 2015. Timely reporting and interactive visualization of animal health and slaughterhouse surveillance data in Switzerland. Front. Vet. Sci. 2, 47.

Pfeiffer, D.U., 2010. Veterinary Epidemiology: An Introduction. John Wiley & Sons.

Schloerke, B., Crowley, J., Cook, D., Briatte, F., Marbach, M., Thoen, E., Elberg, A., Larmarange, J., 2016. GGally: Extension to 'ggplot2'. R Package Version 1.3.0.

Soetaert, K., Petzoldt, T., Setzer, R.W., 2010. Solving differential equations in R: package deSolve. J. Stat. Softw. 33, 1–25.

Vynnycky, E., White, R., 2010. An Introduction to Infectious Disease Modelling. Oxford University Press, UK.

WHO Ebola Response Team, 2014. Ebola virus disease in West Africa–the first 9 months of the epidemic and forward projections. N. Engl. J. Med. 371, 1481–1495.

Wickham, H., 2009. Ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag, New York.