

serosim: an R package for simulating serological survey data arising from vaccination, epidemiological and antibody kinetics processes

Arthur Menezes^{*1}, Saki Takahashi², Isobel Routledge³, C. Jessica E. Metcalf^{1,4}, Andrea L. Graham^{1,5}, James A. Hay^{*6,7}

¹ Department of Ecology and Evolutionary Biology, Princeton University, Princeton, NJ, USA.

² Department of Epidemiology, Johns Hopkins Bloomberg School of Public Health, Baltimore, MD, USA.

³ Department of Medicine, University of California San Francisco, San Francisco, CA, USA.

⁴ Princeton School of Public and International Affairs, Princeton University, Princeton, NJ, USA.

⁵ Santa Fe Institute, Santa Fe, NM, USA.

⁶ Big Data Institute, University of Oxford, Oxford, United Kingdom.

⁷ Center for Communicable Disease Dynamics, Harvard T.H. Chan School of Public Health, Boston, MA, USA.

* Corresponding authors

E-mail: am83@princeton.edu ; james.hay@ndm.ox.ac.uk

Supporting information

Supplementary Text

S1 Text. Helpful functions used to generate inputs for **runserosim**

S2 Text. Optional **runserosim** arguments

Supplementary Figures

S1 Fig. Example **biomarker_map** before and after reformatting.

S2 Fig. Example **exposure_histories** array

Supplementary Tables

S1 Table. Names and descriptions of the main arguments required in the **runserosim** function.

S2 Table. Description of **runserosim** outputs.

S3 Table. Description of functions to plot **runserosim** outputs.

S4 Table. Names and descriptions of the ready-to-use exposure models included in **serosim**.

S5 Table. Names and descriptions of the ready-to-use immunity models included in **serosim**.

S6 Table. Names and descriptions of the ready-to-use antibody models and **model_pars** files included in **serosim**.

S7 Table. Names and descriptions of the ready-to-use **draw_parameters** functions included in **serosim**.

S8 Table. Names and descriptions of the ready-to-use observation models included in **serosim**.

43 Supplementary Text

44

45 S1 Text. Helpful functions used to generate inputs for runserosim

46

47 **S1.1 Text. generate_pop_demography:** This function helps users build their demography tibble
 48 with birth times, removal times, and specific demographic elements of interest. This function calls
 49 both the **simulate_birth_times** and **simulate_removal_times** functions which simulate random
 50 birth times and removal times for each individual. See the **generate_pop_demography** help file
 51 for more information on how to specify the limits and probabilities of birth and removal times. With
 52 the `aux` in **generate_pop_demography**, users can specify other demographic elements of
 53 interest by passing through a list of the column name, options and distributions for each option
 54 type. The returned demography tibble can be modified post-hoc to use user-specified distributions
 55 and values.

56

57 **S1.2 Text. reformat_biomarker_map:** This function will reformat the `biomarker_map` or
 58 `model_pars` objects so that `exposure_ID` and `biomarker_ID` are either both numeric (if passed
 59 as characters) or characters (if passed as numeric) (See S1 Fig). **runserosim** requires that both
 60 `exposure_ID` and `biomarker_ID` are numeric entries.

61

62 **S1.3 Text. plot_biomarker_mediated_protection:** If the user has specified an immunity model
 63 with biomarker mediated-protection, they can use this function to produce a plot of the probability
 64 of infection given an individual's biomarker quantity at exposure conditional on
 65 `biomarker_prot_midpoint` and `biomarker_prot_width` which are specified within
 66 `model_pars`. Ready-to-use immunity models which incorporate biomarker mediated protection
 67 are `immunity_model_ifxn_biomarker_prot` and
 68 `immunity_model_vacc_ifxn_biomarker_prot` (S5 Table). Users can use this function to
 69 determine the best values for these two inputs within `model_pars`.

70

71 **S1.4 Text. plot_biomarker_dependent_boosting:** If the user has specified a function to draw
 72 parameters with biomarker quantity dependent boosting effects, they can use this function to
 73 produce a plot displaying the proportion of full boost received at each starting biomarker quantity
 74 given the biomarker quantity ceiling threshold and the biomarker quantity ceiling gradient which
 75 are specified within `model_pars`. Ready-to-use `draw_parameters` functions which incorporate
 76 biomarker dependent boosting are `draw_parameters_fixed_fx_biomarker_dep` and
 77 `draw_parameters_random_fx_biomarker_dep` (S7 Table). Users can use this function to
 78 determine the best values for these two inputs within `model_pars`.

79 S2 Text. Optional runserosim arguments

80

81 **S2.1 Text. exposure_histories_fixed:** The `exposure_histories` array, specified by
 82 `exposure_histories`, is a 3-dimensional array indicating the exposure history (1 = exposed;
 83 0 = not exposed) for each individual (dimension 1) at each time (dimension 2) for each exposure
 84 event (aka `exposure_ID`) (dimension 3) (S2 Fig). This array normally contains NA's but can also
 85 include pre-specified information if exposure histories are known which can be passed to

86 **runerosim** with `exposure_histories_fixed`. Wherever an entry is provided, **runerosim**
87 assumes that exposure time and type are known and fixed (e.g., specifying known vaccination
88 times or specifying known infections from a transmission model). If the entry is left as NA, then
89 **runerosim** simulates an exposure event for that entry using the specified models.

90

91 **S2.2 Text.** `verbose`: If an integer is specified then a progress update will be printed once the
92 simulation reaches that individual and every multiple thereafter.

93

94 **S2.3 Text.** `attempt_precomputation`: If TRUE, the simulation attempts to perform as much
95 pre-computation as possible for the exposure model to speed up the main simulation code. If
96 FALSE, the simulation skips this step.

97

98 **S2.4 Text.** Model specific arguments: The user may need to specify additional arguments
99 required for the ready-to-use functions that they select. Refer to the function help file for more
100 information on required arguments.

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

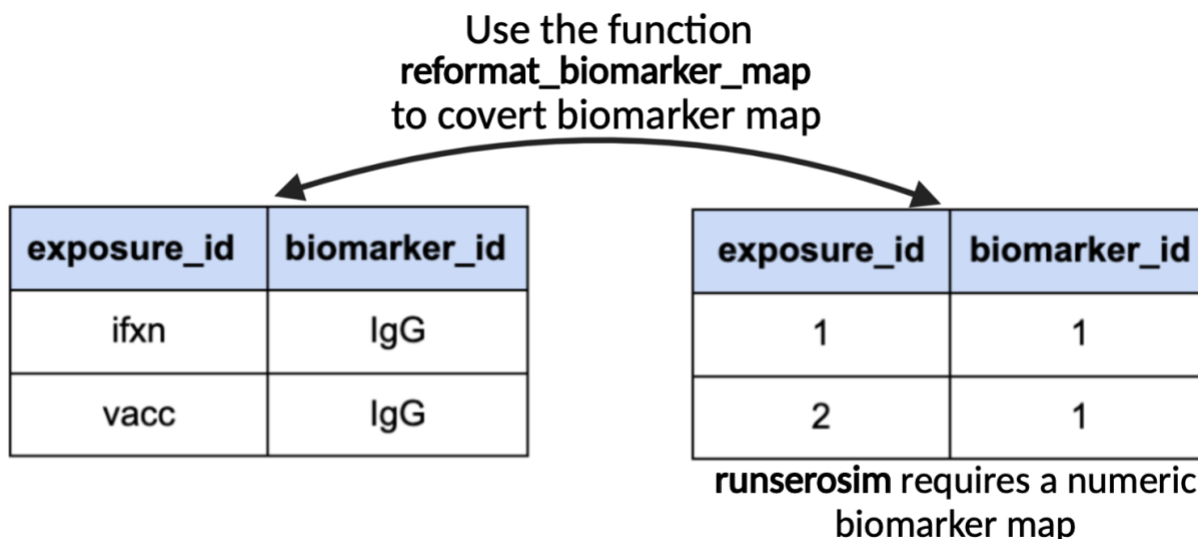
127

128

129

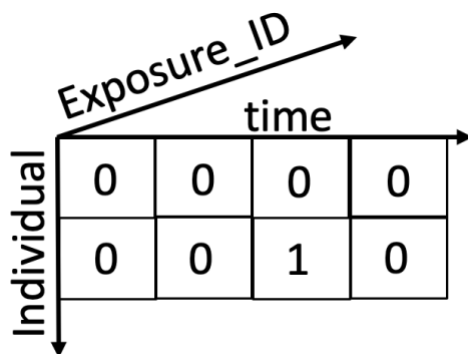
130
131
132

Supplementary Figures



133
134
135
136
137
138
139
140

S1 Fig. Example biomarker_map before and after reformatting. The runserosim function requires a numeric biomarker map input as seen on the right. In this example, we have two exposure events (*exposure_id = ifxn and vacc*) and we are interested in tracking one biomarker (*biomarker_id=IgG*) produced by both exposure events.



141
142
143
144
145
146
147
148
149
150

S2 Fig. Example exposure_histories array. The exposure_histories argument is a 3-dimensional array indicating the individual (dimension 1) at each time (dimension 2) for each exposure event (dimension 3). Here, individual 1 never had a successful exposure while individual 2 was exposed at time 3. (See S2.1 Text)

151 **S1 Table. Names and descriptions of the main arguments required in the runserosim function.** Note:
 152 additional arguments may be needed depending on models selected within the functions section.

Argument	Name	Description
Data objects		
<code>simulation_settings</code>	Simulation settings	A list of parameters governing the simulation time step settings
<code>demography</code>	Demography information	A tibble of demographic information for each individual (e.g., sex, socioeconomic status ...)
<code>observation_times</code>	Serological survey settings	A tibble of observation times and biomarker for each individual
<code>foe_pars</code>	Force of exposure parameters	A 3D array providing the force of infection or vaccination for each exposure event, group and time
<code>biomarker_map</code>	Exposure to biomarker map	A table specifying the relationship between exposure types (x) and biomarker (b)
<code>model_pars</code>	Model parameters	A tibble of parameters needed for the antibody kinetics model, immunity model, observation model and the draw_parameters function
Functions		
<code>exposure_model</code>	Exposure model	A function which will determine the probability that an individual is exposed to an exposure event which can be any event which leads to biomarker production.
<code>immunity_model</code>	Immunity model	A function which determines whether an exposure event is successful conditional on what the actual exposure is (vaccination, infection or re-infection) and the relevant factors.
<code>antibody_model</code>	Antibody model	A function that tracks antibody kinetics, or more broadly biomarker kinetics for each biomarker produced from successful exposure events.
<code>observation_model</code>	Observation model	A function which indicates how observed biomarker quantity Y are generated as a probabilistic function of the true, latent biomarker quantity A .
<code>draw_parameters</code>	Draw model parameters	A function to simulate parameters needed for the antibody model from <code>model_pars</code>

154 **S2 Table. Description of runserosim outputs.**

Output	Name	Description
exposure_histories	Exposure histories	An array of exposure histories across all individuals, time steps and exposure events
exposure_force	probability of an exposure event	An array of exposure probabilities across all individuals, time steps and exposure events. The exposure probability is the output of exposure_model .
exposure_probabilities	Probability of a successful exposure event	An array of the probability of a successful exposure event across all individuals, time steps and exposure events. The probability of a successful exposure event is the probability of exposure (output of exposure_model) multiplied by the probability of success (output of immunity_model)
biomarker_states	Biomarker states	True biomarker quantities for all individuals across all time steps and biomarker(s)
observed_biomarker_states	Observed biomarker states	True and observed biomarker quantities at the observation time for the observed individuals and biomarker(s)
kinetics_parameters	Kinetics parameters	The simulated antibody kinetics parameters for all individual's successful exposure events.

155
156
157
158
159
160
161
162
163
164
165
166
167
168

169 **S3 Table. Description of functions to plot runserosim outputs.**

Name	Description
plot_subset_individuals_history	Plots biomarker quantities and exposure histories for a subset of individuals
plot_exposure_histories	Plots exposure histories across time for all individuals and exposure events; requires <code>exposure_histories_long</code>
plot_exposure_force	Plots the probability of an exposure event across time for all individuals and exposure events; requires <code>exposure_force_long</code>
plot_exposure_probabilities	Plots the probability of a successful exposure event across time for all individuals and exposure events; requires <code>exposure_probabilities_long</code>
plot_biomarker_quantity	Plots true biomarker quantities across all time steps for all individuals and biomarkers; requires <code>antibody_states</code>
plot_obs_biomarkers_one_sample	Plots observed biomarker quantities for all individuals and biomarkers; requires <code>observed_antibody_states</code>
plot_obs_biomarkers_paired_sample	Plots observed paired biomarker quantities for all individuals and biomarkers; requires <code>observed_antibody_states</code>
plot_exposure_model	Plots the probability of exposure over time for the provided exposure model
plot_antibody_model	Plots example trajectories of the provided antibody kinetics model

170
171
172
173
174
175
176
177
178
179
180
181
182
183

184
185

S4 Table. Names and descriptions of the ready-to-use exposure models included in serosim.

Name	Description
Exposure Model	
exposure_model_fixed	Probability of exposure for each time point is drawn directly from the <code>foe_pars</code> array.
exposure_model_simple_FOE	Probability of exposure ($1-e^{-\lambda}$) depends on the force of exposure (λ) at that time for that group.
exposure_model_dem_mod	Probability of exposure ($1-e^{-\lambda}$) depends on the force of exposure (λ) at the current time t for group g modulated by relevant demographic elements specified within the <code>dem_mod</code> . Within <code>dem_mod</code> , users can specify which demographic elements (including age) and how they affect the probability of exposure and by how much.
exposure_model_sir	Finds the probability of exposure governed by an SIR model with specified parameters for each exposure type and group combination.
exposure_model_gaussian_process	Generates an incidence curve (probability of infection per unit time) and associated parameters from a Gaussian Process model assuming that the covariance function (kernel) on time follows the squared exponential covariance function.

186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207

208
209**S5 Table. Names and descriptions of the ready-to-use immunity models included in serosim.**

Name	Description
Immunity Model	
immunity_model_all_successful	Probability of success is 1 so all exposure events are successful immunologically.
immunity_model_vacc_only	This immunity model should only be used if all exposures are vaccination events. The probability of successful exposure(vaccination event) depends on the number of vaccines an individual has received prior to time t and whether an individual is eligible for vaccination. If the individual is under the maximum vaccinations allotted and eligible for vaccination then the probability of a successful exposure event is 1.
immunity_model_vacc_ifxn_simple	This immunity model should only be used with vaccines and natural infection. The probability of successful exposure for vaccination events depends on the number of vaccines an individual has received prior to time t and their current age. If the individual is under the maximum vaccinations allotted and is of an age eligible for vaccination then the probability of successful exposure is 1. The probability of a successful natural infection is dependent on the total number of infections that individual has experienced thus far. If the individual is under the maximum number of infections allotted then the probability of successful exposure event is 1.
immunity_model_ifxn_biomarker_prot	This immunity model should only be used for natural infection events. The probability of a successful exposure event is dependent on the individual's biomarker quantity at the time of exposure. User specified <code>biomarker_prot_midpoint</code> and <code>biomarker_prot_width</code> within <code>model_pars</code> is used to calculate biomarker-mediated protection.
immunity_model_vacc_ifxn_biomarker_prot	This immunity model should be used if exposures represent vaccination and natural infection events. The probability of a successful vaccination exposure event depends on the number of vaccines received prior to time t while the probability of successful infection is dependent on the biomarker quantity at the time of exposure and the total number of successful infections prior to that point. This immunity model combines immunity_model_vacc_only and immunity_model_ifxn_biomarker_prot .

210
211
212
213
214
215
216
217

218 **S6 Table. Names and descriptions of the ready-to-use antibody models and model_pars files**
 219 **included in serosim.**

Name	Description
Antibody Model	
antibody_model_monophasic	This monophasic antibody boosting-waning model assumes that for each exposure there is a boost and boost waning parameter.
antibody_model_biphasic	Biphasic antibody boosting-waning model with titer-dependent boosting. This model assumes that for each exposure there is a set of long-term boost, long-term boost waning, short-term boost, and short-term boost waning parameters. This model incorporates titer-ceiling effects by taking into account an individual's preexisting titer at the time of a new exposure event.
antibody_model_typhoid	Power law antibody boosting and waning model
Model parameters	
model_pars_README.csv	This file was used for the README file and the example within the paper. This model parameter tibble provides a template for simulations with 2 exposure events each containing the same biomarker, <code>antibody_model_monophasic</code> , <code>immunity_model_vacc_ifxn_simple</code> , <code>draw_parameters_random_fx</code> and <code>observation_model_continuous_noise</code> .
model_pars_cs1.csv	This file was used for case study 1 and provides a template for simulations with 2 exposure events each containing the same biomarker. Exposure 1 represents natural infection and exposure 2 represents vaccination. Users wishing to simulate a similar system simply need to input their desired parameter values into the mean, sd and distribution columns. This file includes all parameters needed for <code>antibody_model_biphasic</code> , <code>draw_parameters_random_fx_biomarker_dep</code> , <code>immunity_model_vacc_ifxn_biomarker_prot</code> and <code>observation_model_continuous_bounded_noise</code> .
model_pars_cs2.csv	This model_pars file was used for case study 2 and provides a template for simulations with 3 exposure events. Exposure 1 represents vaccination with 2 biomarkers, while exposures 2 and 3 represent natural infection boosting each biomarker individually. Users wishing to simulate a similar system simply need to input their desired parameter values into the mean, sd and distribution columns. This file includes all parameters needed for <code>antibody_model_biphasic</code> , <code>draw_parameters_random_fx_biomarker_dep</code> , <code>immunity_model_vacc_ifxn_biomarker_prot</code> and <code>observation_model_continuous_bounded_noise</code> .

221
 222 **S7 Table. Names and descriptions of the ready-to-use `draw_parameters` functions included in**
 223 **`serosim`.**

Name	Description
draw_parameters_ fixed_fx	This function draws parameters directly from the distributions specified by <code>model_pars</code> for the antibody model with fixed effects. This function ensures that all individuals have the same parameters.
draw_parameters_ random_fx	This function draws parameters directly from <code>model_pars</code> for the antibody model with random effects. Parameters are drawn randomly from a distribution with mean and standard deviation specified within <code>model_pars</code> .
draw_parameters_ fixed_fx_biomarker_de p	This function adds biomarker quantity ceiling effects to the draw_parameters_fixed_fx function. Here, an individual's realized biomarker boost is dependent on their biomarker quantity at the time of the exposure event.
draw_parameters_ random_fx_biomarker_ dep	This function adds biomarker quantity ceiling effects to the previous draw_parameters_random_fx function. Here an individual's realized biomarker boost is dependent on their biomarker quantity at the time of the exposure event.

224
 225
 226
 227
 228
 229
 230
 231
 232
 233
 234
 235
 236
 237
 238
 239
 240
 241
 242
 243
 244
 245
 246
 247

248
249**S8 Table. Names and descriptions of the ready-to-use observation models included in serosim.**

Name	Description
Observation Model	
observation_model_continuous	This observation model observes the latent biomarker quantities given a continuous assay with no added noise. Therefore the observed biomarker quantity is simply given by the true latent biomarker quantity.
observation_model_continuous_bounded	This observation model observes the latent biomarker quantities given a continuous assay with user-specified lower and upper limits and no added noise.
observation_model_discrete	This observation model observes the latent biomarker quantities given a discrete assay with user-specified ranges within the <code>discrete</code> argument and no added noise.
observation_model_continuous_noise	This observation model observes the latent biomarker quantities given a continuous assay with added noise. The added noise represents assay variability and is done by sampling from a distribution with the latent biomarker quantity as the mean and the measurement error as the standard deviation. The observation standard deviation and distribution are defined within <code>model_pars</code> as the “ <code>obs_sd</code> ” parameter. The user can also use the optional <code>sensitivity</code> and <code>specificity</code> arguments to account for assay sensitivity and specificity. False negatives are simulated by setting an observed quantity to 0 with probability <code>sensitivity</code> . False positives are simulated by drawing a random quantity from the bounded range for a true 0 biomarker quantity with probability <code>1-specificity</code> .
observation_model_continuous_bounded_noise	This observation model observes the latent biomarker quantities given a continuous assay with user-specified lower and upper limits and added noise. The added noise represents assay variability and is done by sampling from a distribution with the latent biomarker quantity as the mean and the measurement error as the standard deviation. The observation standard deviation and distribution are defined within <code>model_pars</code> as the “ <code>obs_sd</code> ” parameter. The user can also use the optional <code>sensitivity</code> and <code>specificity</code> arguments to account for assay sensitivity and specificity. False negatives are simulated by setting an observed quantity to the assay's lower bound with probability <code>sensitivity</code> . False positives are simulated by drawing a random quantity from the bounded range for a true 0 biomarker quantity with probability <code>1-specificity</code> .
observation_model_discrete_noise	This observation model observes the latent biomarker quantities given a discrete assay with user-specified ranges within <code>discrete</code> and added noise. The added noise represents assay variability and is done by sampling from a distribution with the latent biomarker quantity as the mean and the measurement error as the standard deviation. The observation standard deviation and distribution are

	<p>defined within <code>model_pars</code> as the “<code>obs_sd</code>” parameter. The user can also use the optional <code>sensitivity</code> and <code>specificity</code> arguments to account for assay sensitivity and specificity. False negatives are simulated by setting an observed quantity to the assay's lower bound with probability <code>sensitivity</code>. False positives are simulated by drawing a random quantity from the bounded range for a true 0 biomarker quantity with probability <code>1-specificity</code>.</p>
--	---

250
251
252