Comparing post-sampling correction methods for Approximate Bayesian Computation

Zenabu Suboi (zenabu@aims.ac.za)
African Institute for Mathematical Sciences (AIMS)

Supervised by: Professor Wim Delva and Doctor Marijn C. Hazelbag
South African Centre for Epidemiological Modelling and Analysis (SACEMA)-University of Stellenbosch, South Africa

24 May 2018

Submitted in partial fulfillment of a structured masters degree at AIMS South Africa

Abstract

Calibrating Agent Based Models (ABMs) to empirical data through likelihood estimation is often impossible. Therefore simulation-based Approximate Bayesian Computation (ABC) is used to calibrate ABMs. Improvements on ABC can either be in the algorithm for obtaining the posterior without correction or as a post-sampling correction method. It is unclear which of these post-sampling correction methods performs best. We set out to compare performance of different post-sampling correction methods (Linear and Neural network adjustments) for ABC in a simple ABM simulation setting (e.g. SIR), where likelihood estimation is used as the gold standard. The posterior densities of these methods for three scenarios were compared to that of the likelihood-based posterior with respect to percentage overlap. The within tolerance computation times for these post-sampling correction methods were also compared. The results showed that the post-sampling correction methods are effective in improving the posterior at bigger tolerance but fail when it comes to sufficiently small tolerance. Linear adjustment performed better in improving the quality of the rejection ABC posterior and was less computationally expensive compared to the neural network adjustment.

Declaration

I, the undersigned, hereby declare that the work contained in this research project is my original work, and that any work done by others or by myself previously has been acknowledged and referenced accordingly.

Zenabu Suboi, 24 May 2018

Contents

1. Introduction

Agent based models (ABMs) are models used to capture the dynamics of complex systems. Such systems often self-organize themselves. ABMs include models of agent behaviour and are used to observe the actions and interactions of such agents. In observing complex systems, ABMs offer an approach to capture social systems that comprise agents who interact with and influence each other (Macal and North, 2010). Results from these models are intended to inform decision making and policies. ABMs have gained popularity in a wide range of disciplines. Their applications include forecasting emerging infectious diseases, where an Agent based model was developed to forecast the 2014-2015 Ebola epidemic in Liberia and was subsequently used during the 2015-2016 Ebola forecasting challenge (Venkatramanan et al., 2017).

Calibrating agent based models to empirical data through likelihood estimation is often impossible. Approximate Bayesian computation (ABC) allows likelihood-free calibration of ABMs to empirical data. Due to the complex nature of ABMs, they tend to be a limitation to statistical inference since their likelihoods cannot be calculated explicitly as in simple models. ABC offers an alternative such that, explicit calculation of the likelihood is not required (Hartig et al., 2011). However, ABC is computationally demanding in realistic settings with many parameters, since large numbers of simulations are required for estimation of the posterior. In order to reduce the computational burden, previous research has focused on developing ABC methods obtaining a good approximation to the posterior using a minimal number of simulations.

The most basic form of ABC is the rejection ABC. This algorithm samples parameter points from the prior distribution and with each parameter point, a dataset is simulated. It discards a sampled parameter point along the simulation if the simulated data is too different from the observed (Sunnåker et al., 2013). Improvements on ABC can either be in the algorithm for obtaining the posterior without correction or as a post-sampling correction method that accounts for the the mismatches between simulated and observed datasets. Previous research on ABC includes proposing different post-sampling correction methods. What is unknown to users of these algorithms to our knowledge is the performance of the correction methods as compared under various scenarios.

In this research work, we perform a simulation study to find out which of the following two post-sampling correction methods (linear and neural network adjustments) for ABC performs best in terms of percentage overlap and computation time, using likelihood estimation as the gold standard. This study presents a brief introduction to ABMs, ABC and SIR (Susceptible-Infected-Recovered) model, provides the methods used in the study, explores and discusses the results obtained and finally identifies the best performing post-sampling correction method.

2. Methods

This chapter highlights the methods used in performing the simulation study. It explains in detail rejection ABC and regression ABC. It states and explains the model under which data sets are generated. It also touches on the general concept of likelihood estimation and poisson likelihood. Finally, it elaborates on the methods used in the comparison of the posterior densities (percentage overlap and time used).

2.1 Rejection ABC

ABC consists of computational methods and techniques that make use of Bayesian statistics. These ABC techniques are relevant for calibrating stochastic models to empirical data because they are easy to implement and can be applied to any model (Sunnåker et al., 2013). ABC techniques operate by sampling parameter values $(\theta_i), i = 1, ..., N$ from the prior distribution $\pi(\theta)$ and given these parameter values, data (y) is generated under a statistical model. A summary statistic $(\hat{t}\ )$ of the generated data (y) must satisfy a proximity criterion with the target statistic (t^*) such that $d(t^*, \hat{t}\ ) \leq \epsilon$, where d expresses the metric between the data sets x and y, and ϵ represents a tolerance level. Corresponding parameter values are retained for generated summaries $(\hat{t}\ )$ that are closer to the target statistic (t^*) than the tolerance (ϵ). Figure 2.1 gives an illustration of how the rejection ABC algorithm works. The simulator (M) is run each time with a newly sampled parameter value (θ) from the prior distribution obtaining a simulated summary statistic (red and green dots). Based on the tolerance level (ϵ), a decision is made whether to retain the particular parameter value if the simulated summary statistic is further from the target statistic (y_0) than (ϵ).

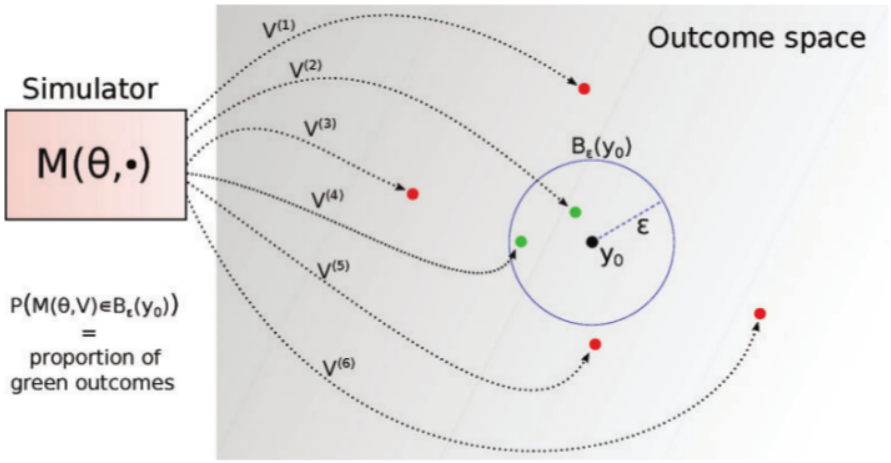


Figure 2.1: An illustration of how the Rejection ABC algorithm works (Lintusaari et al., 2017).

The distribution of these retained parameter values is expected to assume that of the desired posterior without the exact calculation of the likelihood. From Bayesian inference, estimation of the posterior distribution depends on the prior distribution and the likelihood. The posterior is defined as

$$\pi(\theta|x) \propto p(x|\theta)\pi(\theta) \tag{2.1.1}$$

where θ is a vector of parameters, x is the observed data, $\pi(\theta|x)$ is the posterior distribution, $p(x|\theta)$ is the likelihood and $\pi(\theta)$ is the prior distribution. ABC techniques use this same knowledge in the approximation of the posterior using summary statistics instead of the observed data since the likelihood is intractable. Equation 2.1.1 is modified to approximate the posterior as follows

$$\pi(\theta|\hat{t}) \propto p(\hat{t}|\theta)\pi(\theta). \tag{2.1.2}$$

2.2 Regression adjustments

Having obtained a posterior from the ABC methods for posterior approximation, the algorithms for improving the quality of the posterior (post-sampling correction methods) are used as a second step. Post sampling correction methods are applied to parameter combinations accepted by rejection ABC with the aim of improving the quality of the posterior. Regression ABC (local-linear and Neural network adjustments) adjusts these parameter values in order to account for the mismatches between the simulated data and the target summary statistics by fitting a regression model with the parameter values as the response and the simulated summary statistics as the predictors. Consequently, the values of the parameters are shifted along the regression line, based on the distance of their summary statistics to the target (Blum, 2017). From simple linear regression, we have that

$$\theta_i = \beta_0 + \beta_1\hat{t}_i + \delta_i, \quad i = 1, ..., m \tag{2.2.1}$$

where θ_i is the response variable, $\hat{t}_i$ are the independent variables, δ_i are the residuals, β_0 and β_1 are the parameters of the model and m is the number of retained parameters. Just as in simple linear regression, β_0 and β_1 are obtained using the method of least squares to minimize the sum of squares of the residuals. Thus

$$\sum_{i=1}^{m}\delta_i^2 = \sum_{i=1}^{m}(\theta_i - \beta_0 - \beta_1\hat{t}_i)^2 \tag{2.2.2}$$

where we obtain the estimates of the parameters of the model by differentiating equation 2.2.2 once with respect to β_0 and β_1 and set to zero as follows

$$\frac{\partial\sum_{i=1}^{m}\delta_i^2}{\partial\beta_0} = -2\sum_{i=1}^{m}(\theta_i - \beta_0 - \beta_1\hat{t}_i) = 0$$
$$\hat{\beta}_0 = \bar{\theta}_i - \hat{\beta}_1\bar{\hat{t}}_i$$

and

$$\frac{\partial\sum_{i=1}^{m}\delta_i^2}{\partial\beta_1} = -2\sum_{i=1}^{m}(\theta_i - \beta_0 - \beta_1\hat{t}_i)\hat{t}_i = 0$$
$$\hat{\beta}_1 = \frac{\sum_{i=1}^{m}(\theta_i - \bar{\theta}_i)(\hat{t}_i - \bar{\hat{t}}_i)}{\sum_{i=1}^{m}(\hat{t}_i - \bar{\hat{t}}_i)^2}$$

we then obtain the adjusted parameters as

$$\theta_i^* = \hat{\beta}_0 + \hat{\beta}_1 t^* + \delta_i, \quad i = 1, ..., m \tag{2.2.3}$$

where θ_i^* are the adjusted parameter values, $\hat{\beta}_0$ and $\hat{\beta}_1$ the estimates of the parameters of the model, t^* is the target statistic and δ_i is the i^{th} residual.

For linear adjustment, the assumption here is that there exists a linear relationship between the retained parameters and their summary statistics. Neural network adjustment uses a similar approach, but allows for non-linearity in the model (equation 2.2.1). Figure 2.2 gives an illustration of how regression ABC adjusts parameters with the hope of obtaining a posterior.

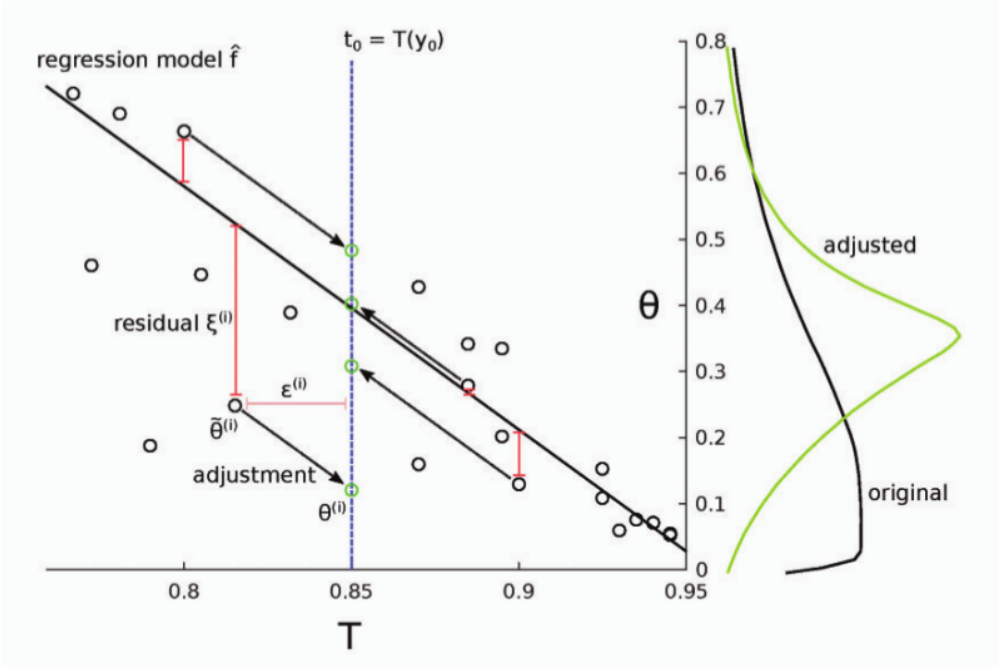


Figure 2.2: Illustration of Regression adjustment (Lintusaari et al., 2017)

The black points obtained from plotting the parameters against the simulated data are shifted along the fitted regression line towards the target statistic (T= 0.85) with preserved residuals and their corresponding parameter values are recorded (green points). These adjusted parameter values are the new posterior. Here, we observe that an adjusted posterior (green curve) was obtained after the adjustment.

2.3 Simulation model

In this study, the ABC rejection algorithm was applied to a simple SIR (Susceptible - Infected - Recovered) model which was used to generate datasets. "The SIR model is an epidemiological model that computes the theoretical number of infected individuals with a contagious disease in a closed population over time. This model involves a system of three non-linear ordinary differential equations (ODEs) relating the number of susceptible $S(t)$, number of infected $I(t)$, and number of recovered $R(t)$ individuals " (Weiss, 2013). The SIR model works under a number of assumptions such as: the mode of transmission of the disease from infected to susceptible is through direct contact between infected and susceptible individuals. The closed population (e.g. a school or a city) is usually divided into compartments denoted

by $S(t)$, $I(t)$ and $R(t)$. The following system of ODEs represents the SIR model:

$$\frac{dS}{dt} = -\beta SI \tag{2.3.1}$$

$$\frac{dI}{dt} = \beta SI - \gamma I \tag{2.3.2}$$

$$\frac{dR}{dt} = \gamma I \tag{2.3.3}$$

where $\beta > 0$ is the disease transmission rate, $\gamma > 0$ is the recovery rate, $D_{inf} = 1/\gamma$ is the duration of infection and $R_0 = \beta D_{inf}$ is the basic reproductive number (Weiss, 2013; Sebastian Funk and Johnson, 2017 (accessed April 24, 2018). Figure 2.3 illustrates the plot of a stochastic SIR model run in the R software over a time period of 75 days for a population of 1000 individuals. The blue curve indicates susceptible, the red curve indicates Infected and the green curve indicates Recovered individuals.

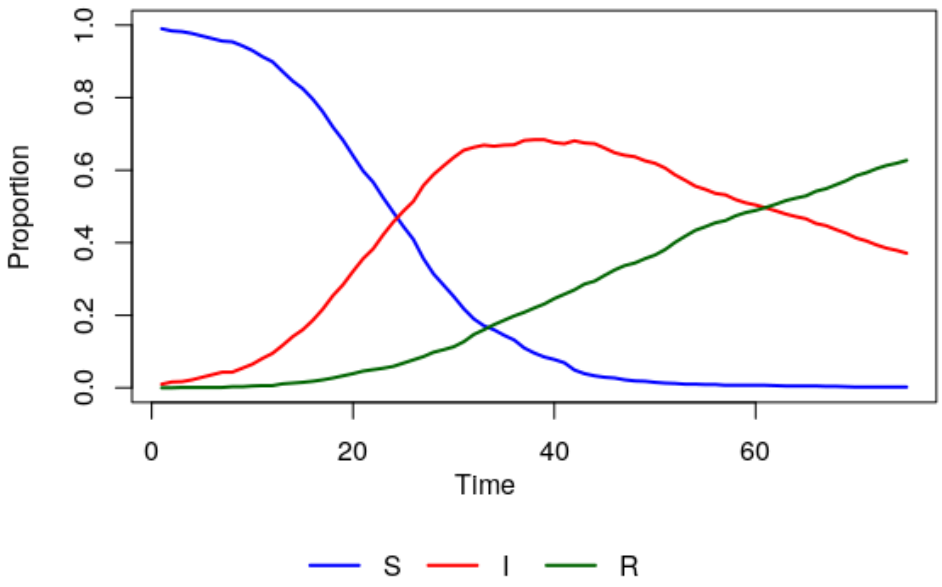


Figure 2.3: Plot of a run of a stochastic SIR model

We used the prevalence at time = 50 and time = 75 as our target statistics.

2.4 Creating and applying a raster

"A raster comprises a matrix of cells or pixels arranged into rows and columns to form a grid where each cell contains a value which represents information" (Esri, 2017 (accessed May 12, 2018). In order to compare the posterior densities of the various methods to the likelihood posterior density, we created a raster using the raster function in the raster library where we applied a grid to each posterior in order to quantify the density of each cell or pixel. The raster was created by considering the minimum and maximum values of beta and gamma produced by all of the methods. This parameter space was divided into 200x200 equally sized bins with beta values on the x-axis and gamma values on the y-axis (see Figure 2.4). This then formed a grid in which the posterior densities lie.

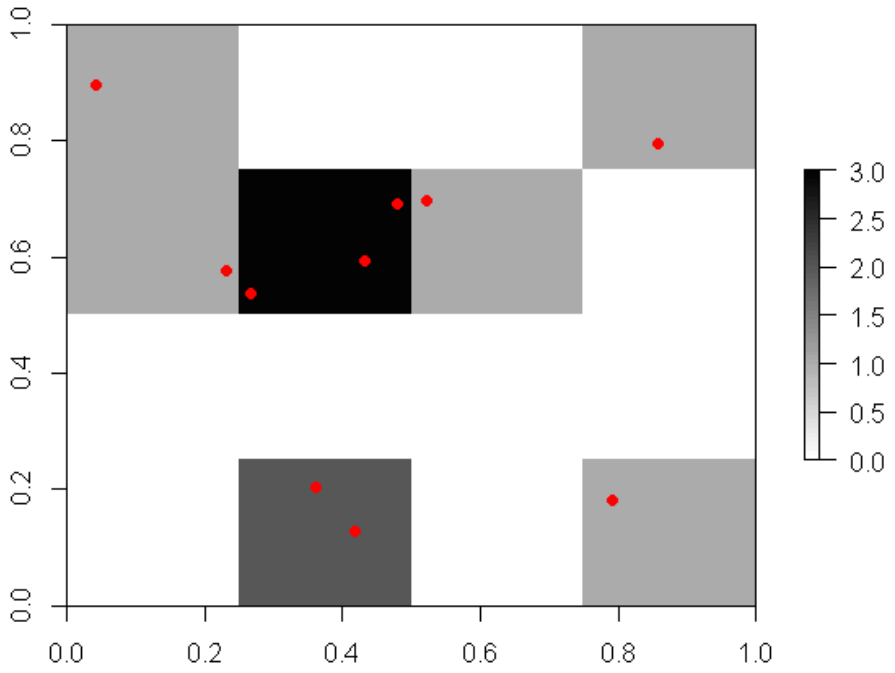


Figure 2.4: Example of a 4x4 raster applied to a posterior

2.5 Poisson likelihood

Likelihood estimation is used to find an optimal way to fit a distribution to a dataset. Given observed values $x_1, x_2, ..., x_n$, the likelihood of a parameter θ is the function

$$L(\theta) = \prod_{i=1}^{n} f(x_i|\theta) \tag{2.5.1}$$

where the random variables are independent and identically distributed variables.

Our target statistic is prevalence at time=50 and time=75, where prevalence is the number of infected individuals divided by the total number of individuals at a particular time. In our simulation we used a closed population of a fixed size, therefore using the prevalence is the same as using the number of infected individuals as a count. This allows us to use the number of infected individuals as a count to calculate a Poisson likelihood.

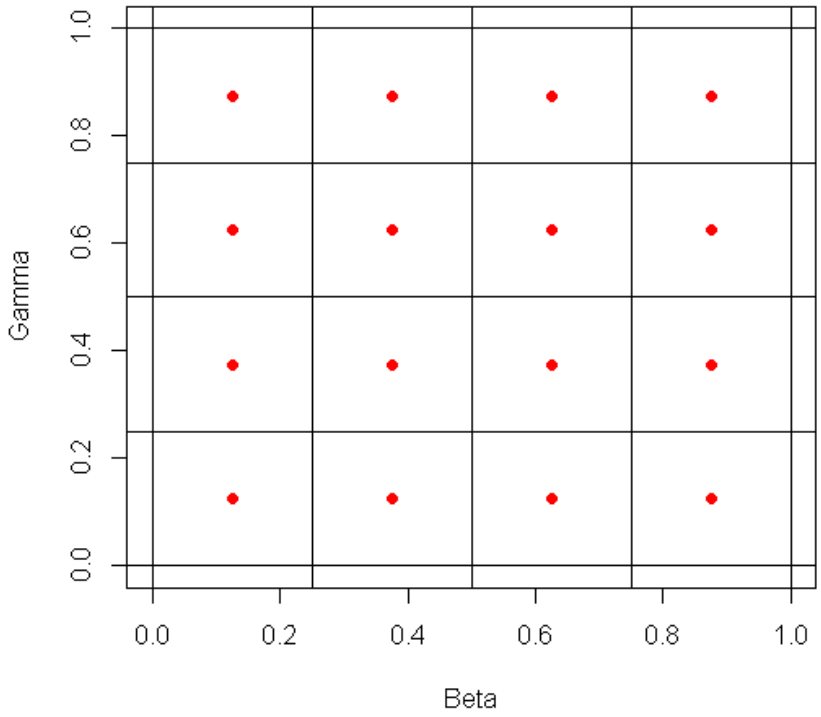


Figure 2.5: Example of calculating the Likelihood for the centre of each grid square

The probability density function for the poisson distribution is given by

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!} \tag{2.5.2}$$

The likelihood function for the joint probability of observations becomes

$$L(\lambda) = p(x|\lambda) = \prod_{i=1}^{n} \frac{\lambda^{x_i} e^{-\lambda}}{x_i!} \tag{2.5.3}$$

2.6 Comparison of methods

We based our comparison on computation time and percentage overlap. For computation time, we recorded the time it took each post-sampling correction method to perform adjustment and obtain new posterior under each scenario. For percentage overlap, we wanted to find out how each posterior can overlap the likelihood posterior. We computed it as follows

$$P_{ij} = \left[1 - \frac{\sum |A_{ij} - L_i|}{2n\epsilon_i}\right] \times 100 \tag{2.6.1}$$

where P_{ij} is the percentage overlap for method j and tolerance (scenario) i, A_{ij} represents the matrix form of the raster applied on method j and tolerance (scenario) i, L_i represents the likelihood matrix of tolerance (scenario) i, n is the number of simulations run by the rejection ABC algorithm.

3. Simulation procedure

This chapter describes in detail the simulation procedure. It shows how realistic targets were obtained using a stochastic SIR model with fixed parameter values. It also describes how the rejection ABC algorithm was applied to the stochastic SIR model in order to generate data set at each model run. This chapter presents the application of post-sampling correction methods after rejection ABC for all three scenarios investigated. It describes in detail also how a grid was created to be applied to the estimated posterior densities. Finally, it explains how the likelihood for the centre of each grid square was obtained using a poisson likelihood.

3.1 Performing Simulations

R version 3.3.3 (www.r-project.org) was used to perform the statistical analyses. We obtained the data sets from a stochastic SIR model using the SIR function in the SimInf library (Widgren et al., 2016). A closed population of size 1000 was used with initial compartmental population sizes as 990 susceptible, 10 infected and no recovered individuals. We defined the time period over which we wanted to simulate the disease spread as 75 days. To obtain realistic target statistics (mean prevalence at time = 50 and time = 75), we ran the model 10000 times with fixed parameter values $\beta = 0.2$ and $\gamma = 0.02$. The rejection ABC algorithm was applied to the SIR model for the simulation using the function ABC_rejection in the EasyABC library. During the simulation, this function launched a series of 1.000.000 model simulations with model parameters beta and gamma drawn from a uniform prior distribution where $\beta \sim U(0, 1)$ and $\gamma \sim U(0, 0.5)$ respectively. For each scenario, we used the abc function, also in the EasyABC library to apply the post-sampling correction methods and decreasing levels of tolerance to the result of rejection ABC.

In all, we examined three different simulation scenarios for the two post-sampling correction methods investigated. The scenarios were to run each method for three different tolerance levels ($\epsilon = 0.1, \epsilon = 0.01$ and $\epsilon = 0.001$), indicating the proportion of samples closest to the target statistics retained. For each simulation scenario, we applied the post-sampling correction methods after rejection ABC. We recorded their percentage overlap as compared to the posterior of the likelihood estimation and computation times . We ran $1.000.000$ model simulations using the rejection ABC algorithm obtaining datasets including all the 1.000.000 parameter combinations used. The scenarios were later introduced and desired parameter combinations were retained ($100000, 10000$ and 1000 for tolerance levels $0.1, 0.01$ and 0.001 respectively) using R software version 3.3.3 (www.r-project.org). For each scenario, sampling continued until the predefined criterion for inclusion was met and parameter combinations of beta and gamma were stored.

We applied the raster to each posterior in order to obtain posterior densities. To calculate the likelihood, we computed the midpoints of every two successive parameter values along the sequence of parameter values (beta and gamma) used to create the raster. This was to ensure that the same raster can be applied to the likelihood as well in order to allow comparison. Having obtained these new parameter values, we computed the likelihood for all possible parameter combinations using the SIR$PointObs function in the fitR library. The likelihood computed here is the probability of observing the prevalences produced by running the deterministic SIR model with these parameter values given our target prevalence values. This is done under a poisson distribution with the computed target prevalences as a mean. The SIR$PointObs function implements the poisson likelihood and hence computes the poisson likelihood

for the centre of each grid square. The likelihood was computed in each case and stored in a 200x200 matrix which is of the same dimension as the grid applied to the posterior densities. We standardized the likelihoods by dividing each likelihood by the sum of the likelihoods so that the sum of the standardized likelihoods equal one. For each scenario, we scaled up the likelihood matrix so that we can compare to the raster matrices for the methods in that scenario. This was done to ensure that the likelihood sum up to the number of parameter estimates obtained by applying scenario. Thus for tolerances 0.1, 0.01 and 0.001, we multiplied the likelihood matrix by 100000, 10000 and 1000 respectively. Figure 3.1 below represents the flow chart of the simulation procedure.

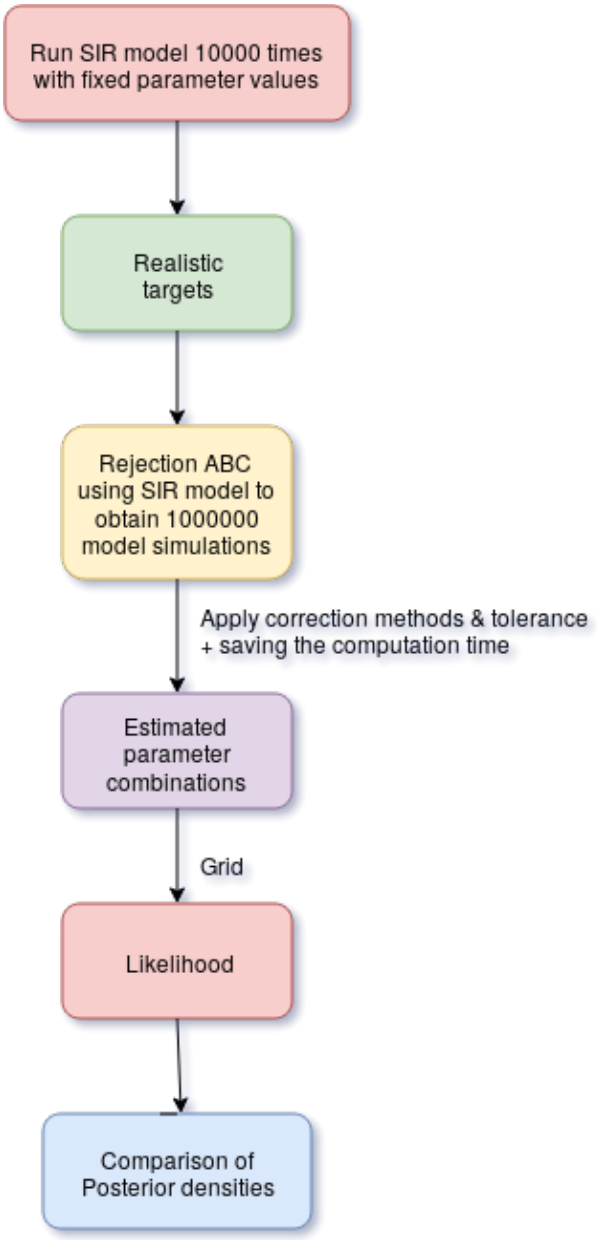


Figure 3.1: Flow chart of the simulation procedure

4. Results and Discussion

This chapter presents the results obtained after applying the post-sampling correction methods to the rejection ABC posterior. It presents plots of the posterior densities of the various post-sampling correction methods including the posterior densities for rejection ABC and likelihood estimation. It also presents the tabulated results obtained from the percentage overlap and time used.

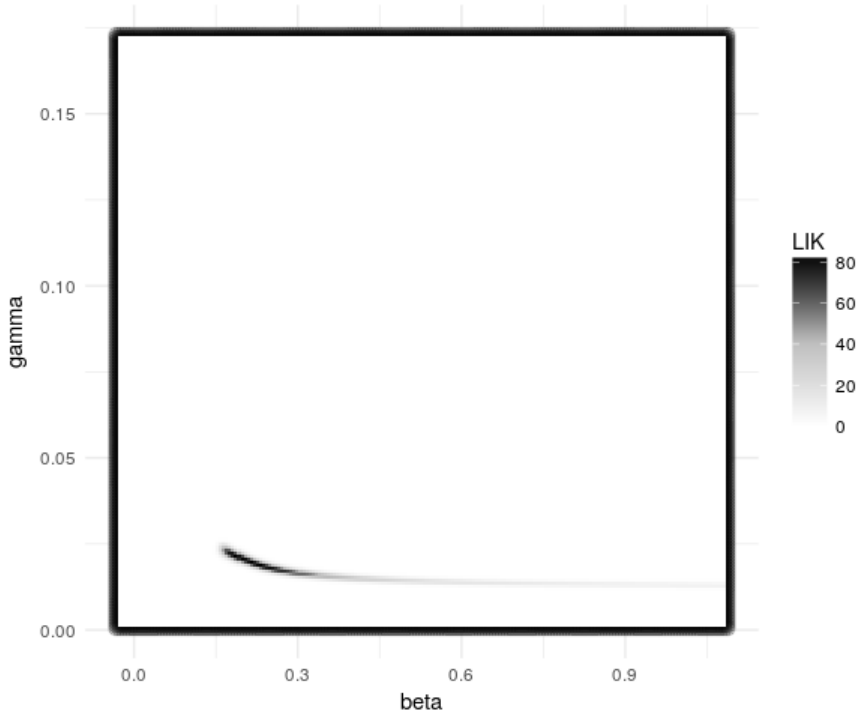


Figure 4.1: raster plot of the likelihood-based posterior

Figure 4.1 represents the raster plot of the likelihood-based posterior. The plots obtained from the linear and neural network adjustments are compared to the likelihood-based posterior and their performances in terms of percentage overlap are recorded.

The plots on the first row of Figure 4.2 represent the raster plots for rejection ABC, linear adjustment and neural network adjustment for tolerance 0.1. Plots on the second row of Figure 4.2 represent the raster plots for rejection ABC, linear adjustment and neural network adjustment for tolerance 0.01. Plots on the third row of Figure 4.2 represent the raster plots for rejection ABC, linear adjustment and neural network adjustment for tolerance 0.001.

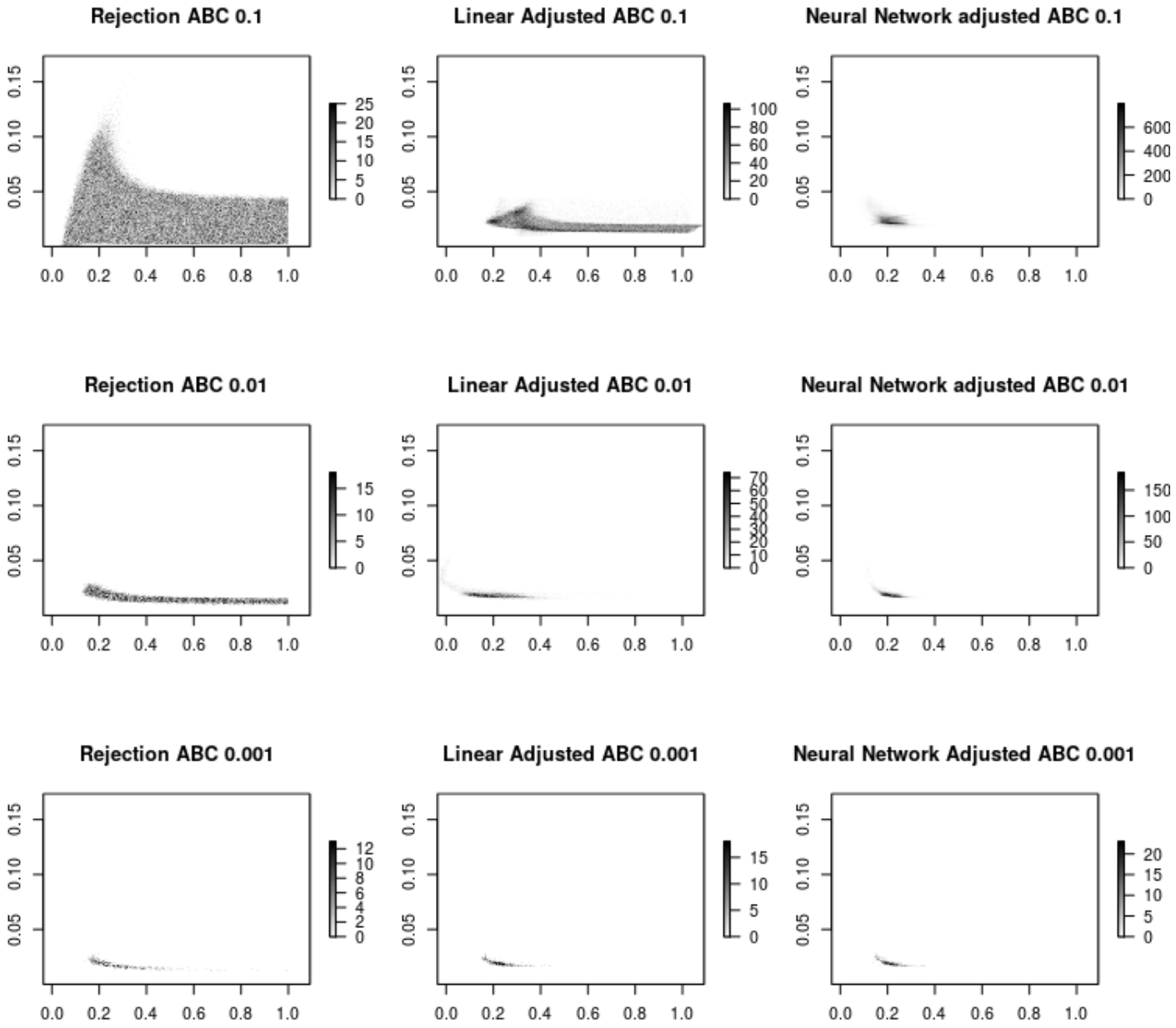
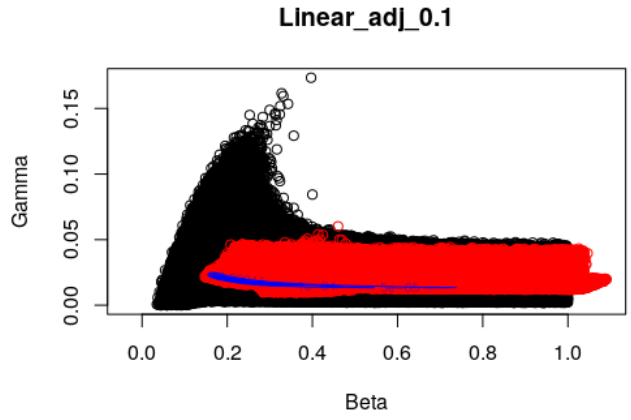
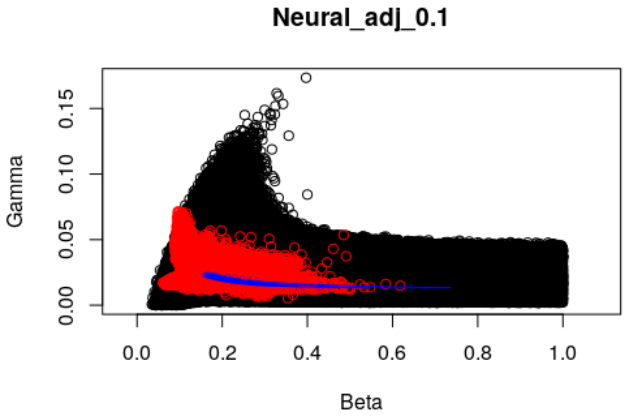


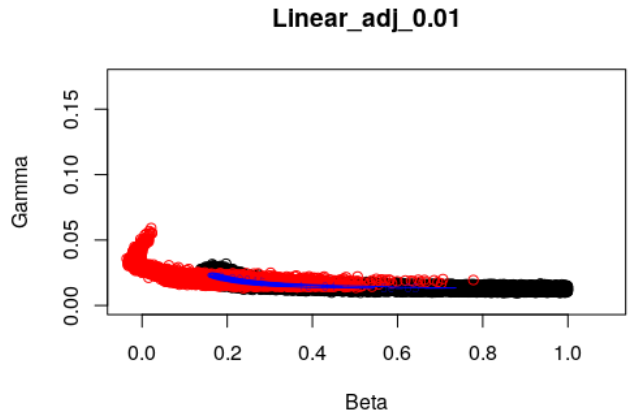
Figure 4.2: Raster plots for all three scenarios



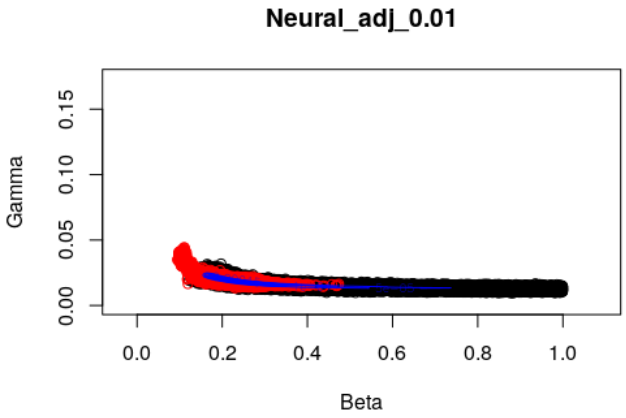
(a) Posterior for Tolerance=0.1



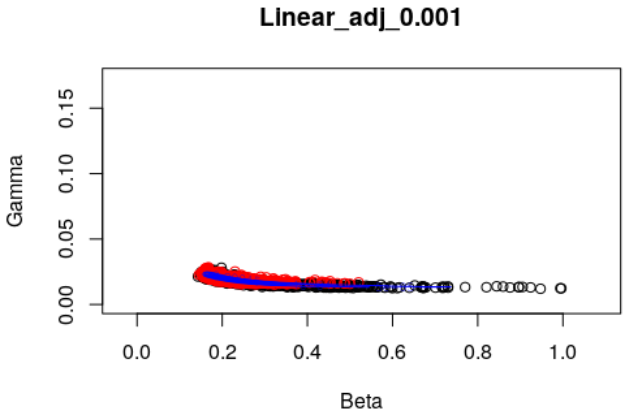
(b) Posterior for Tolerance=0.1



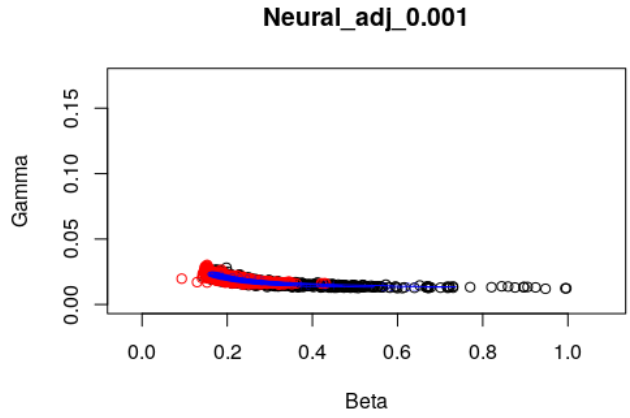
(c) Posterior for Tolerance=0.01



(d) Posterior for Tolerance=0.01



(e) Posterior for Tolerance=0.001



(f) Posterior for Tolerance=0.001

Figure 4.3: This figure represents the plots of the posterior densities. The plots obtained from the linear and neural network adjustments are compared to the likelihood-based posterior and their performances in terms of percentage overlap are recorded. For all plots in this Figure, the black represents the posterior estimated by the unadjusted rejection ABC, the red represents the plot of posterior estimated after adjustment and the blue represents the plot of the posterior estimated by likelihood estimation.

4.1 Visual comparison of plots

Figure 4.3 represents the plots obtained after applying regression adjustments and the unadjusted rejection ABC posterior densities. Figures 4.3a, 4.3c and 4.3e represent the combined plots of the three estimated posterior densities obtained from the likelihood, unadjusted rejection ABC and linear adjustment. From Figure 4.3a where tolerance $(\epsilon) = 0.1$, we observe that local-linear adjustment improved the unadjusted rejection ABC posterior. The adjusted parameter values are closer to the likelihood-based posterior than those of the unadjusted rejection ABC. From Figure 4.3c, we observe that most of the adjusted parameter values from local-linear adjustment did not overlap the likelihood-based posterior estimation hence not improving the unadjusted rejection ABC posterior. We therefore see that rejection ABC performed better without adjustment for tolerance $(\epsilon) = 0.01$. Also from Figure 4.3e, we observe that the linear adjusted posterior is shifted to one side of the likelihood-based posterior while the posterior estimated by the unadjusted rejection ABC performed very well for tolerance $(\epsilon) = 0.001$.

From Figure 4.3, Figures 4.3b, 4.3d and 4.3f represent the combined plots of the three estimated posterior densities obtained from the likelihood, unadjusted rejection ABC and neural network adjustment. From Figure 4.3b where tolerance $(\epsilon = 0.1)$, we observe that the posterior estimated by the neural network adjustment performed well in improving the unadjusted rejection ABC. The adjusted parameter values are closer to the likelihood-based posterior than those of the unadjusted rejection ABC. From Figure 4.3d, we observe that most of the adjusted parameter values from neural network adjustment did not overlap the likelihood-based posterior hence not improving the unadjusted rejection ABC posterior. We therefore see that rejection ABC performed better without adjustment for tolerance $(\epsilon) = 0.01$. Also from Figure 4.3f, we observe that the neural network adjusted posterior is shifted to one side of the likelihood-based posterior but the posterior estimated by the unadjusted rejection ABC performed very well for tolerance $(\epsilon) = 0.001$.

4.2 Percentage overlap

Table 4.1 represents the percentage overlap between the likelihood-based posterior density and the ABC posterior densities compared under the three scenarios. For the first scenario $(\epsilon = 0.1)$, we observe that linear adjustment performed best (22.85 percent) as compared to the other two methods (see Figure 4.3a and Figure 4.3b). For the second scenario $(\epsilon = 0.01)$, we observe that rejection ABC performed best (40.44 percent) as compared to the other two methods (see Figure 4.3c and Figure 4.3d). Lastly, for the third scenario $(\epsilon = 0.001)$, we observe that rejection ABC performed best (61.33 percent) as compared to the two post-sampling correction methods (see Figure 4.3e and Figure 4.3f).

Table 4.1: Percentage overlap

Tolerance	Rejection ABC	Linear reg ABC	Neural net ABC
0.1	6.46	22.85	19.08
0.01	40.44	30.30	39.10
0.001	61.33	47.51	43.21

Figure 4.4 represents the plot of the percentage overlap of each method as compared to the likelihood-based posterior under the various scenarios investigated. We observe that percentage overlap increases for all three methods as tolerance decreases. As tolerance decreases, neural network adjustment seems

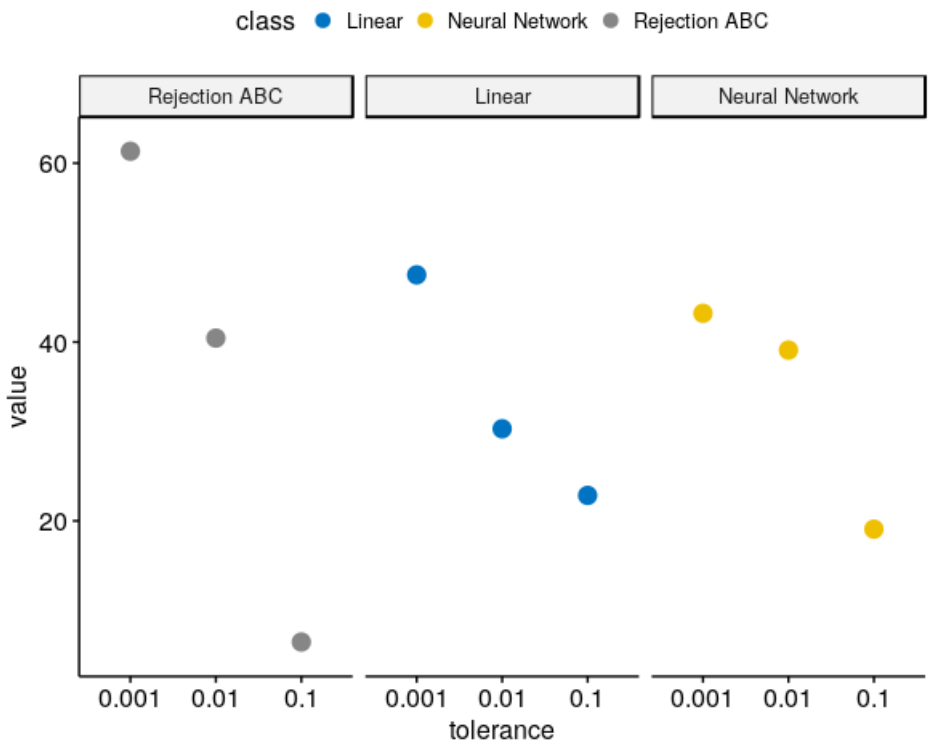


Figure 4.4: Plot of percentage overlap for the three methods

to be insensitive to the change in tolerance hence producing almost equal percentage overlap. We see also that percentage overlap increases rapidly for rejection ABC as tolerance decreases hence performing better than the two post-sampling correction methods .

4.3 Time used

Table 4.2 represents the time used by each post-sampling correction method for the three scenarios. It took 48046.10 seconds (13.34 hours) for rejection ABC to run the 1000000 model simulations. The computation times recorded in Table 4.2 for the post-sampling correction methods represent the additional time taken to perform adjustments. We observe that as tolerance decreases, computation times for both correction methods decrease as well. We also observe that neural network adjustment is more computationally expensive than local-linear across all tolerance levels.

Table 4.2: Time used (seconds)

Tolerance	Number of parameters	Linear reg ABC	Neural net ABC
0.1	100000	1.41	229.01
0.01	10000	1.28	15.68
0.001	1000	1.12	3.70

4.4 Discussion

In this research work, we performed a simulation study to find out which of the post- sampling correction methods for ABC performs best in terms of percentage overlap and computation time, using likelihood estimation as the gold standard.

Percentage overlap increased for all three methods as tolerance decreased. However, as tolerance decreased, the post-sampling correction methods were unable to improve the posterior obtained by rejection ABC.

Neural network adjusted posterior tends to be less sensitive to change in tolerance due to the additional flexibility it adds to the adjustment unlike linear adjustment (Blum, 2017). This was confirmed as we see in Table 4.1 that there is a small difference between the neural network percentage overlaps for tolerance 0.01 and 0.001 (39.10 and 43,21 respectively). Linear adjustment saw a substantial difference in percentage overlap each time tolerance was decreased. This then implies that linear adjusted posterior is sensitive to change in tolerance (Blum, 2017).

We noticed that linear adjustment had less computation time as we compared within tolerance computation times. As tolerance decreased, computation times for both correction methods decrease as well. Neural network adjustment was more computationally expensive than local-linear across all tolerance levels. So, neural network adjustment is sensitive to number of parameters while linear adjustment is not.

The SIR model used for the simulations is a simple ABM whose likelihood can be calculated explicitly. This happened to be a strength, it allowed the comparison of ABC methods to the likelihood approach. Some ABMs have many parameters and summary statistics but this SIR model only used two parameters and two summary statistics, this may be a limitation of the current study because our results may not be applicable to complex ABM's with many parameter values and summary statistics.

5. Conclusion

In this study, we performed a simulation study to find out which post-sampling correction method (linear and neural network adjustments) for ABC performs best in terms of percentage overlap and computation time, using likelihood estimation as the gold standard. Data used was obtained by performing simulations from a stochastic SIR model. Two post-sampling correction methods were investigated under three scenarios. Parameter combinations were retained as the posterior densities. Computing percentage overlaps between the likelihood posterior and the ABC posterior densities,we observed that at a bigger tolerance, the correction methods are effective in improving rejection ABC posterior with linear adjustment performing better than neural network.

However, when tolerance was reduced, both correction methods reduced the quality of the rejection ABC posterior instead. We also noticed that as tolerance decreased, computation times for both correction methods decreased as well. Neural network adjustment was computationally expensive compared to local-linear across all tolerance levels. We can conclude based on the results of this study that when the tolerance is chosen to be sufficiently small, rejection ABC performs well in the approximation of the posterior. Also, at a bigger tolerance, we conclude that linear adjustment should be used since it is computationally efficient and performs better at such tolerance.

Suggestions for future research are to consider basing the comparison also on posterior distance where we will compute the earth mover's distance between each posterior density and that of the likelihood. Earth mover's distance computes the distance between distributions where it assumes each distribution is a heap of earth. The earth mover's distance is the computed as the amount of work required to pile up one distribution on top of the other. Also, comparing other ABC methods (e.g. Sequential ABC, Adaptive ABC, etc.).

Acknowledgements

To GOD be the glory, great things HE has done. I thank GOD for seeing me through this journey successfully. I thank my supervisors Prof Wim and Dr Marijn for their support, help and patience towards the completion of this research work. I thank my family for their love and prayers. I also thank Mr. Adams Prince Amoah for his love, prayers and encouragement that have gotten me this far. Not forgetting Mr Nasiru Suleman for his constant motivation.

Lastly, I thank all my colleagues who made my stay at AIMS memorable. Most especially Samuel, Emmanuel, Grace, Louisah, Bathata and Lorain. God bless you all.

References

Bagni, R., Berchi, R., and Cariello, P. A comparison of simulation models applied to epidemics. *Journal of Artificial Societies and Social Simulation*, 5(3), 2002.

Beaumont, M. A., Zhang, W., and Balding, D. J. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

Blum, M. G. Regression approaches for approximate bayesian computation. *arXiv preprint arXiv:1707.01254*, 2017.

Burton, A., Altman, D. G., Royston, P., and Holder, R. L. The design of simulation studies in medical statistics. *Statistics in medicine*, 25(24):4279–4292, 2006.

Esri. *Manage Data*, 2017 (accessed May 12, 2018). URL http://desktop.arcgis.com/en/arcmap/10.3/manage-data/raster-and-images/what-is-raster-data.htm.

Hartig, F., Calabrese, J. M., Reineking, B., Wiegand, T., and Huth, A. Statistical inference for stochastic simulation models–theory and application. *Ecology letters*, 14(8):816–827, 2011.

Lenormand, M., Jabot, F., and Deffuant, G. Adaptive approximate bayesian computation for complex models. *Computational Statistics*, 28(6):2777–2796, 2013.

Leykum, L., Kumar, P., Parchman, M., McDaniel, R. R., Lanham, H., and Agar, M. Use of an agent-based model to understand clinical systems. *Journal of Artificial Societies and Social Simulation*, 15 (3):2, 2012.

Lintusaari, J., Gutmann, M. U., Dutta, R., Kaski, S., and Corander, J. Fundamentals and recent developments in approximate bayesian computation. *Systematic biology*, 66(1):e66–e82, 2017.

Macal, C. M. and North, M. J. Tutorial on agent-based modelling and simulation. *Journal of simulation*, 4(3):151–162, 2010.

Sebastian Funk, A. C. and Johnson, H. *Model fitting and inference for infectious disease dynamics*, 2017 (accessed April 24, 2018). URL https://sbfnk.github.io/mfiidd/introduction.html.

Sunnåker, M., Busetto, A. G., Numminen, E., Corander, J., Foll, M., and Dessimoz, C. Approximate bayesian computation. *PLoS computational biology*, 9(1):e1002803, 2013.

Venkatramanan, S., Lewis, B., Chen, J., Higdon, D., Vullikanti, A., and Marathe, M. Using data-driven agent-based models for forecasting emerging infectious diseases. *Epidemics*, 2017.

Weiss, H. H. The sir model and the foundations of public health. *Materials matematics*, pages 0001–17, 2013.

Widgren, S., Bauer, P., and Engblom, S. Siminf: An r package for data-driven stochastic disease spread simulations. *arXiv preprint arXiv:1605.01421*, 2016.