Imperial College London

Stochastic modelling of bacterial interactions in the gut microbiota

MSc Epidemiology

CID: 00846124

Word count: 9860

Date of submission: 24/08/2020

Acknowledgements

My primary thanks go to my supervisors. I am very grateful for the continuous help, support and motivation given by both Dr [REDACTED] and Dr [REDACTED]. The patient and friendly manner in which they advised me throughout the dissertation period made this work possible. In addition, it has been a pleasure and a privilege being part of their team over the summer and being able to work on such an exciting project. Also thank you to my family for the support throughout the academic year.

Reflection statement

Throughout this project I have gained and improved on all the tools a successful junior researcher should have. The support my supervisors provided me was instrumental in this academic growth; there is a significant difference the knowledge I have of this field now compared to what I had at the beginning of the summer. I improved on my writing skills, which has always been an academic weakness of mine, especially my ability to write long pieces of work whilst maintaining a clear and succinct style of writing. I also vastly improved on my programming skills, both in the logical approach to solving a problem, as well as the use of concise syntax. In undertaking this project, I also strengthened my interest in mathematics by being able to apply it to a real world and public health problem, something which I had not had the opportunity to do in my previous higher education experience. Most importantly, this project has piqued my interest in microbiology and its impact on epidemiology, and has given me a much more robust idea of the areas in which I wish to work in the future.

Contents

List of tables

List of figures

List of abbreviations

ABC	Approximate Bayesian Computation
C. albicans	*Candida albicans*
C. difficile	*Clostridioides difficile*
CDI	*Clostridioides difficile* infection
CTMC	Continuous-time Markov chain
CI	Credible interval
E. coli	*Escherichia coli*
ELFI	Engine for Likelihood-Free Inference
ODE	Ordinary differential equation
SDE	Stochastic differential equation
S. pneumoniae	*Streptococcus pneumoniae*
rRNA	Ribosomal ribonucleic acid

Abstract

Introduction

Competition occurs in all ecosystems, including environments in which bacteria reside. The primary motivation to compete is to obtain resources limited in availability. The human gut is a diverse microbial jungle, home to 100 trillion microorganisms from several different species. Competition is an important force shaping their dynamics, and if understood properly would give greater ability to understand and manipulate our microbiome. *Clostridioides difficile* is an anaerobic, spore-forming bacteria, dormant within the mammalian gut. Usually kept at bay through competition with other bacteria, large scale disruption of the microbiota can promote growth of *C. difficile*. It is broadly unknown which species effectively compete with *C. difficile*, and would be most effective in treating this infection. Competition between species can be modelled through the Lotka-Volterra equations, a system of ordinary differential equations, and have recently been applied to competition between gut bacteria.

Aims/objectives

The aim of this study was to compute stochastic solutions for an N-species Lotka-Volterra model, and infer competition values using Approximate Bayesian Computation (ABC). I applied this to data from a murine microbiome model, and compared my methodology and results with a previous deterministic approach.

Methods

I solved the deterministic and stochastic forms of the Lotka-Volterra equations for two species numerically. We formulated the equations as a continuous-time Markov chain (CTMC) by using the probabilities of either species' cells dividing or dying at discrete time steps, which was solved using a variant of the Gillespie algorithm. We used this model in conjunction with ABC, a method of making likelihood-free inferences, to estimate values of parameters. I then tested this inference framework using various combinations of parameters to gauge its accuracy. The experimental data was collected as microbiome readings measured from the guts of mice infected with *C. difficile*, which I fitted using the simulator. I ran simulations between *C. difficile* and other bacterial species within the mice guts, in order to estimate the values of the competition parameters (denoted by α).

Results

Stochastic models describe the finer changes in population dynamics compared to the deterministic model, where state is assumed constant. I found that denser sampling at early timepoints was optimal for inferring parameters because it gave the greatest resolution in detecting strong competitive

interactions – rapid loss of species due to strong competitive effects results in a lack of information for the simulator to draw inferences from. At low competition terms ($\alpha < 1$) the simulator provided accurate and precise estimations for known parameters.

Discussion

Through methods of solving CTMCs I captured the stochastic nature of competition, giving reasonable variance in the model's parameters. The use of Bayesian inference also gave credible intervals for parameters and allowed the use of custom priors in the model, unlike previous inference algorithms. In an initial data, run the estimates I obtained were relatively poor, with large credible intervals. While the properties of this inference approach are promising, more work is needed to adapt it to scaled data. Making plausible inferences may be easier when observed data is more familiar to the mathematical modeller handling the simulator.

Introduction

1.1. Competition as a force within ecological communities

Competition amongst species is a natural phenomenon that exists in all ecosystems and habitats. The primary reason for this centres on the scarcity of resources[1]. Historically, competition has been studied using plants and animals. A classic example of competition is between the snowshoe hare (*Lepus americanus*) and the Canadian lynx (*Lynx canadensis*), a scenario in which the hare is predated upon by the lynx[2]. Where there is an abundance of a hare population, the lynx population will also thrive due there being an existence of the required resource, in this case food. Conversely, if the hare population begins to decline, or relocate habitat (which is in fact, constricted as a consequence of being predated upon[3]), over a period of time the lynx population will die out due to the depletion of food. Though the interactions between the two species can be described linearly, the resulting dynamics are generally complex and non-linear, due to the coupling of the two populations.

1.2. Intraspecific and interspecific competition in higher organisms

Intraspecific competition signifies members of the same species competing with one another for a resource or resources. Demonstrations of posture and vocal conflict are sometimes used to establish dominance, a determining factor in how a resource will be utilised and by who[4]. An example is a wasp, the German yellowjacket (*Vespula germanica*), which is an opportunistic predator that scavenges in order to satisfy its ranged diet of honey, brood, and honeybees. Intraspecific competition has been observed during the wasps' feeding times. Experimental work concluded that there exists a positive correlation between how densely populated a wasp population is and the frequency of competition[5]. In general, this effect leads to logistic growth, which is the decrease of a population's growth rate as it tends towards the carrying capacity. The carrying capacity is the maximum number of individuals within a population that can be supported by the resources available in the local environment.

In contrast, interspecific or *interspecies* competition involves two different species in contention for the same resource. In Italian bodies of water situated within mountainous regions, larvae of the common house mosquito (*Culex pipiens*) and the tiger mosquito (*Aedes albopictus*) are at competition with each other for a resource. Namely, the availability of a suitable breeding site; which is imperative

to each species' survival[6]. However, only one species can occupy these conditions and make use of the resources, thus resulting in interspecific competition.

1.3. Bacterial interactions

Competition is also observed between bacteria and can also be intraspecific or interspecific. For example, glucose is a nutrient needed for growth. When there is a limited supply of glucose, bacterial populations risk becoming limited in their growth, and so interspecific competition will occur to prevent this from happening. Similar to any population of a higher organism where there is a limited resource, bacterial populations will experience logistic growth. There is agreement from experimental work and simulations conducted in recent studies that bacterial strains already residing in a host are preferentially placed when faced with challenging strains of the same species. *Shen et al.* (2019) concluded that a residing *Streptococcus pneumoniae* strain, referred to as the 'resident' prevails in asymmetric competition with an intruding *S. pneumoniae* strain (the challenger)[7]. Though the two strains are of the same species, the gene expression in each strain varies, with the resident producing a toxin-antitoxin system which kills the challenger. What is noteworthy about intraspecific competition in bacteria is that due to their unicellular nature and lack of autonomy, these competitive reactions occur as a result of single proteins. This reinforces the already proven hypothesis that competition exists due to evolution, and is a mechanism for success observed across the biological tree of life[8].

Research also suggests that there are two distinct categories of competition between bacterial species[9]. Firstly, exploitative competition is a scenario in which the competitors consume the resource directly, rather than any direct interactions taking place amongst competitors. The second category involves a direct and hostile contest between the species, after which the species who 'won' the interaction will consume the resource(s) that was initially being contested for. This is otherwise known as 'contest' competition[9]. Interestingly, as an extension of exploitative competition methods, bacteria also compete indirectly through immunity. An example is bacteria within a species but with different genotypes ('strains'), that utilise a particular antigen it contains in order to trigger an immune response, which recognizes all strains carrying the same antigen, but rejects the others[10]. Thus, only the strains with shared antigens will be in competition with one another. This means that genotypes with less common antigens have higher fitness, known as negative frequency-dependent selection[11].

Similar to how one observes scenes in the wild of species competing for territorial possession, the goal for a bacterial species is to secure ownership of the host. The human gut boasts a rich and diverse

microbiota that consists of an over 100 trillion microorganisms[12], making it amongst the most densely populated microbial environments on earth. Naturally, in such a microbe and bacteria rich setting, interspecific competition is inevitable. An established instance of this is the ability of *Escherichia coli* (*E. coli*), a Gram-negative anaerobic bacterium to kill off *Candida albicans* (*C. albicans*), a pathogenic yeast found within the flora of the human gut. The presence of *E. coli* acts as an inhibitor to the growth of the *C. albicans*[13]; the former essentially outcompeting the latter and causing drastic declines in its population[13]. This is just one example of the vast number of contests that take place amongst bacterial strains in the gut, most of which we do not fully understand.

1.4. Application of metagenomics to the study of bacterial competition

Metagenomics can be simply defined as the analysis of all genetic material collected from a given sample. Modern methods on sequencing, and a typical study comprises of sequencing, bioinformatics, data analysis, followed by an interpretation in the context of its appropriate field. It is important in the context of this thesis to understand the purpose of metagenomics to study microbial communities within living organisms. Estimates show that less than 2% of bacteria are able to be cultured in laboratories[14]. Interestingly, isolated growth under laboratory conditions is dependent on the accurate replication of the natural environments of these bacteria. In addition to their sensitivity to oxygen, other conditions necessary include the correct nutrients, pH and temperature[15]. This is hard to replicate, which strengthens our need for metagenomics, so that such bacteria can be analysed without having to be cultured repeatedly.

Naturally, computational methods of research play a significant role in the study of genetic sequences, and these are continuously evolving. When assessing competition generally one has the choice between correlative and mechanistic models. Correlative models are only qualitative and do not assume a particular model of competition[16], whereas mechanistic models are quantitative, and allow us to compute numerical values of specific parameters and undertake counterfactuals, such as simulating dynamics if a certain species was increased in prevalence[17].

The findings that result from metagenomics are an instrumental aspect of epidemiology. Understanding how bacteria compete amongst each other can provide a solid foundation for further research into methods of combatting disease, whether it is prevention, diagnosis or treatment.

1.5. Clostridioides difficile

Clostridioides difficile, often referred to as *C. difficile*, is a bacterial species highly prevalent in soil, though its habitats are widespread throughout nature and exists within the gut of mammals[18]. A *C. difficile* strain is Gram-positive and anaerobic, meaning it is capable of growth in the absence of oxygen, and it is also highly motile due to its flagella[20][21]. As a spore-forming bacteria, *C. difficile* produces highly resistant spores which are able to survive in oxygen. Thus, these spores can contaminate surfaces such as those within hospital environments, and their resistance to alcohol-based disinfectants can result in a higher rate of transmission[22].

C. difficile can invade the colon after contact with contaminated surfaces, though there are background traces of the bacteria present within the healthy human gut[18]. The two most significant pathways to infection are contact with contaminated surfaces, as well as through antibiotic treatment[20]. Antibiotics indiscriminately kill all species of bacteria in the gut, those which cause unwanted infections as well as harmless commensals. *C. difficile* is an extremely resistant bacteria, so it is left unaffected. This leaves the intestinal microbiota at risk of *C. difficile* strains which would have been dormant, but can now utilise the energy stored from dormancy in order to grow and colonize this niche. Once the *C. difficile* strains are active and have grown in population size, they then release toxins which target the intestines, usually causing symptoms of colitis and diarrhoea[20]. There are two such types of toxins: enterotoxin A and cytotoxin B, otherwise referred to as TcdA and TcdB respectively, although it is TcdB that leads to severe inflammation of the colon[20]. Ultimately, these toxins cause host cell death after cell entry and subsequent disruption of cytoskeletal structure. This encompasses shrinkage and rounding of cells, rendering them unable to communicate with each other. Thus, gut epithelial cells are left unable to carry out essential functions such as division and movement[20].

Hamm et al. (2006) measured the effects TcdB had on the embryo of a zebrafish. The experiment compared a control embryo, to one that had been given a 24-hour treatment of 37 nM TcdB (the case). After having been exposed to the TcdB, the heart of the case was found to have been localized by the toxin, and to a lesser degree, the yolk sac and eye[22].

The role health care facilities play in the incidence of *Clostridioides difficile* infection (CDI) must not be overlooked. In 2011, *C. difficile* was the cause of 453,000 infections in the United

States[23],[24],[25], two thirds of which had associations with patient care facilities. Given that transmission can occur easily through contact with contaminated surfaces, this raises concern over the state of hygiene within these facilities. However, this issue extends beyond hospitals. Of the 453,000 cases, 76% of them occurred in patients who had not had an overnight stay at a hospital facility. A further 82% of these cases reported that they had visited settings such as doctors' and dentists' offices within a 12-week period prior to providing *C. difficile* positive stool samples[25]. The implication here is that exposure to the bacteria can arise from general practice surgeries, dental practices and other non-hospital facilities, not just from nosocomial transmission.

Methods of treating CDI vary depending on the severity of the disease. Antibiotics that specifically target *C. difficile* can be used. For mild-to-moderate disease onset, metronidazole or vancomycin, both antibiotics to treat CDI induced diarrhoea are initial options. Vancomycin is also used to treat severe CDI, although at a higher dosage (125 mg for mild or moderate infection against 500 mg for severe infection)[26]. Once there is a recurrence of the infection, the chances of further recurrences range anywhere from 40 to 65%[26]. In this case, a faecal transplant can be used to treat recurrences of CDI[26]. The operation involves transplanting the faeces of a healthy donor into an infected individual so that the diversity within the microbiome is re-established, preventing further recurrences. While this method of treatment is not commonly performed, interest in the procedure is increasing[27], and there is no evidence to suggest this treatment should be contraindicated for any type of patient, regardless of how immunosuppressed they are[27].

The research conducted into *C. difficile* has provided us with a useful understanding of the significance of interspecies competition in supressing pathogens. Continuing forward, the public health impact of *C. difficile* depends on its handling in health care facility settings, which includes prevention, diagnosis and treatment.

1.6. Lotka-Volterra system of equations

Modelling in ecology often involves describing the population dynamics of species, which can be altered as a result of interspecies competition. We can characterize these dynamics as rates of change, best described by differential equations.

Within an arbitrary ordinary differential equation (ODE), the term ordinary refers to ordinary derivatives, i.e. of functions of a single variable. Consider an independent variable t and a function $f = f(t)$; together with n of its derivatives $f, f', \ldots, f^{(n)}$. Where

$$f^{(n)} \equiv \frac{d^n f}{dt^n};\ n \geq 1.$$

Then, the equation

$$f^{(n)} = F\left(t, f, f', f'', \ldots, f^{(n-1)}\right).$$

is an n^{th} order ordinary differential equation. This study uses a system of first order coupled differential equations, i.e. $n = 1$:

$$f' = F(t, f).$$

This is a relationship between t, f, f', and our task in solving differential equations is to obtain the unknown function of f.

The Lotka-Volterra equations, sometimes referred to as the predator-prey equations, are a system of two first order time dependent differential equations.

The two functions represent the prey and predator in turn. Traditionally, the set of equations was derived by Alfred Lotka in 1910 in the theory of chemical reactions[28]. It was later used by Vito Volterra in 1926 to investigate population changes of animal species residing in overlapping geographical ranges[29]. The system can be described by the coupled equations:

$$\frac{dx}{dt} = \alpha x - \beta xy \ \textcircled{1}$$

$$\frac{dy}{dt} = \delta xy - \gamma y \ \textcircled{2}$$

The two parameters x and y signify the number of prey and the number of predators respectively. Then, each equation is representative of the growth rate of the corresponding species. The growth rate of the prey is $\frac{dx}{dt}$, while $\frac{dy}{dt}$ is the growth rate of the predator. The parameter t represents time. We are then left with α, β, γ and δ. These are parameters that signify the interactions between both species. The prey in this system of equations is assumed to grow exponentially, which is signified in

equation ① (the prey) by the term αx. The second term in the equation, βxy, represents the rate at which the prey is predated upon by the predator. The rate of predation and the rate at which the predator and prey meet are assumed to be proportional to one another, known as the law of mass-action. In other words, if the prey exists in greater numbers than the predator, fewer prey will be predated upon, and vice versa. If either x or y holds a value of 0, then there is no predation, because an absence of a prey population means the predator has nothing to predate upon, and the absolute lack of the predator means there is nothing to predate upon the prey. For example, if we assume the value of y is 0, the equation for the prey then becomes:

$$\frac{dx}{dt} = \alpha x \ ③$$

signifying continuous exponential growth of the prey. Equation ① can be interpreted simply; the rate of change of a prey's population is the prey's own growth rated subtracted by the rate at which it is predated upon.

The population dynamics of the predator is represented by equation ②. The predator's rate of growth is denoted as δxy. Loss rate of the population can be due to naturally occurring death, or the predator expanding the ranges of its geographical distribution. This is represented by the term γx. Conversely to what we derived in equation ③, absence of a prey population can be shown by:

$$\frac{dy}{dt} = -\gamma y \ ④$$

This results in exponential decay of the predator population, the opposite of the exponential growth experienced by the prey in the predator's absence. Furthermore, general variants of these equations are possible which include both birth and death of predator and prey, and allow competition in either direction between two interacting species.

As with any set of differential equations, the Lotka-Volterra model has equilibrium points when both equations are set equal to zero:

$$\frac{dx}{dt} = 0 \text{ and } \frac{dy}{dt} = 0$$

$$\alpha x - \beta xy = 0 \ and \ \delta xy - \gamma y = 0$$

Solving these provide analytic solutions to the equations. If interspecific competition is greater than intraspecific competition, one species will die out as $t \to \infty$. Whereas if intraspecific competition is greater than interspecific competition, coexistence between both species occurs. Both species having high values of interspecific competition means the final state of each species is dependent on the starting population sizes for both species. It should be noted that within this system, the growth rate of each species does not have a bearing on the final state of either species, rather, competition is the most significant parameter which determines the system.

This deterministic model's dynamics are set entirely by values of the parameters and initial conditions. In other words, absolute certainty is assumed within the model, there is no randomness involved. Thus, when initial conditions are defined, the model will produce the same output each time it is run. A deterministic approach is useful when there is not much randomness in the system, for example when there are large population sizes of each species. Another issue arises in trying to model a small population size with a deterministic approach, as the populations are continuous, and a species with a non-integer sized population could grow to a large size in this model. Stochastic models are more realistic and better reflect the inherent randomness of the systems they model, and as a result they model small population sizes more accurately. The prefix stochastic implies that we are modelling an entity which is both random and changes through time. In the context of the Lotka-Volterra model, there is randomness in the growth of the populations. The presence of a random term in stochastic differential equations means each time they are solved, there will be a different output.

To define the Lotka-Volterra equations as a stochastic model, the deterministic equation can be perturbed with a suitably scaled random component. Consider a first order ordinary differential equation:

$$\frac{dq}{dt} = f\big(t, y(t)\big), \qquad y(t_0) = y_0 \ \textcircled{5}$$

where $y(t_0) = y_0$ is the initial condition. Then, let $\{X_t : t \in \mathbb{R}^+\}$ represent a continuous time (stochastic) process that satisfies the stochastic differential equation (SDE):

$$dX_t = a(X_t, t)dt + b(X, t)dW_t, \qquad X_0 = x \ \textcircled{6}$$

Then ⑥ is the differential equation in ⑤, with $b(X,t)dW_t$ being an additional random term where dW_t is an increment in a Brownian motion.

Computational approaches of integrating this model may utilise techniques such as the Euler-Maruyama method of solving SDEs.

1.7. Parameter inference in complex models

The methods discussed so far allow simulation of populations through time, given a known set of parameters. However, in experimental data typically it is these trajectories that will be observed at certain time points, and the parameters will be unknown. Inferring the most likely parameters from the data requires statistical inference.

For stochastic models, methods such as particle filtering[30] or Approximate Bayesian Computation (ABC)[31] allow for the inferences of these parameters. When applied to parameters within the Lotka-Volterra model, these techniques make use of prior probability distributions as a way of defining a range of parameter values that would be plausible given the observed data. Traditionally, Bayesian inference involves prior beliefs being updated through the likelihood function. Due to the existence of latent variables (unobservable quantities which must be inferred) within such models, or analytic intractability, it is not always possible to generate a likelihood function for traditional Bayesian inference. This is where likelihood-free methods such as ABC are used. The basic idea behind ABC is to guess values of parameters, run the simulator, and calculate a discrepancy between the output and the observed data. Using this as an approximation for the likelihood, an approximate posterior distribution can be inferred, which can then be sampled from, to extract marginal posterior estimates for each of the model parameters.

1.8. Aims and objectives

The aim of this study is to program a simulator that allows for the estimation and inference of the parameters of the Lotka-Volterra model; to allow us to understand the strengths of interactions between strains of *C. difficile* and other bacterial species in the gut microbiota. Furthermore, this simulator will aid in identifying additional conditions and parameters that may affect the competition amongst these bacteria.

A two species stochastic Lotka-Volterra model has been recently developed[7], which I will then extend to allow an n number of species. This model will then be solved via the appropriate numerical techniques and used in conjunction with likelihood-free inference methods in order to approximate parameters. The key parameter of interest in this study is the alpha term, which represents the strength of competition between bacterial species.

Once publicly available data has been fitted using the simulator, I will compare the accuracy of the methods I used against existing approaches to this problem.

Methods

2.1. The research problem

Experimental data has been collected previously that contains population sizes of bacteria at differing time points[17]. This was collected from the guts of ten mice infected with *C. difficile*. Whilst the population dynamics of various bacterial species can be seen over a time scale, we need a way of estimating the numerical values of the competitive interactions that take place between them. Thus, this project involves creating a simulator that estimates parameters of the Lotka-Volterra model when applied to competition between bacterial strains of various species. Such a simulator will require the application of the Lotka-Volterra model (in the form of stochastic differential equations), and methods of Bayesian statistics in conjunction with one another.

2.2. Extension of the Lotka-Volterra model

The deterministic form of the Lotka-Volterra systems applied to bacterial competition was defined as a pair of nonlinear first-order ordinary differential equations[7]:

$$\frac{dN_1}{dt} = r_{N_1} N_1 \left(\frac{K_1 - N_1 - \alpha_{N_2 N_1} N_2}{K_1} \right) \quad ⑦$$

$$\frac{dN_2}{dt} = r_{N_2} N_2 \left(\frac{K_2 - N_2 - \alpha_{N_1 N_2} N_1}{K_2} \right) \quad ⑧$$

Compared to equations ① and ②, equations ⑦ and ⑧ include both inter and intra specific competition, and allow inter-specific competition to be in either direction. The two species in this system are denoted by N_1 and N_2. K_1 and K_2 signify the carrying capacity for each species, which is the maximum size of all the combined bacteria populations that can live in a specific area. This was assumed equal for both populations. The two competition terms, $\alpha_{N_1 N_2}$ and $\alpha_{N_2 N_1}$ represent the competition of species 1 upon species 2 and species 2 upon species 1 respectively. These are the main parameters that are usually unknown and needed to be inferred.

The stochastic form of this system was then described by the authors as the following:

$$dN_1 = \left(B_{N_1} - D_{N_1}\right)dt + \left(\sqrt{B_{N_1} + D_{N_1}}\right)dW(t) \quad ⑨$$

$$dN_2 = \left(B_{N_2} - D_{N_2}\right)dt + \left(\sqrt{B_{N_2} + D_{N_2}}\right)dW(t) \quad ⑩$$

In equation ⑨, the terms B_{N_1} and D_{N_1} are the cell growth and death rates of species 1 respectively, while in equation ⑩, B_{N_2} is the species 2 growth and D_{N_2} is the species 2 death.

Table 1: Transition probabilities of events for each species' change of state. Generalised version of table from Shen et al. (2019)[7]

Event	**Change**	**Notation**	**Probability**
Species 1 cell growth	$\Delta N_1 = +1$	B_{N_1}	$r_{N_1} N_1$
Species 1 cell death	$\Delta N_1 = -1$	D_{N_1}	$r_{N_1} N_1 \left(\frac{N_1 + \alpha_{N_2 N_1} N_2}{K}\right)$
Species 2 cell growth	$\Delta N_2 = +1$	B_{N_2}	$r_{N_2} N_2$
Species 2 cell death	$\Delta N_2 = -1$	D_{N_2}	$r_{N_2} N_2 \left(\frac{N_2 + \alpha_{N_1 N_2} N_1}{K}\right)$

Each of these events were described to have a transition probability[7], as seen in Table 1. The simulations calculated population sizes as continuous-time Markov chains or CTMCs (a stochastic process which will experience changes of state due to the random variable). As a result, events that change discrete population sizes such as births and deaths, occurred according to these transition probabilities.

I used these two sets of equations as a base model to test out different values of parameters and plot trajectories of the population dynamics over time. Firstly, I programmed an extended form of the deterministic model to handle n species. The parameters this model took were the number of species, a time range, the size of each species' population, the carrying capacity for each species, the growth rate of each species, and the alpha terms for each species. The time range I specified was between 0 and 100; the units which were arbitrary and scale linearly in relation to the growth rate *r*, and were not of significance at this step. For the sake of simplicity of syntax, in my model I referred to each species as species 1 (N_1) and species 2 (N_2). I selected a population size of $N_1 = 100$ and $N_2 = 500$. As the carrying capacity is assumed equal for all species, I defined the value for this parameter as $K = 1000$ for each. Growth rates were set to equal values of $r_{N_1} = 1.5$ and $r_{N_2} = 1.5$. For the two alpha terms, I defined a 2x2 matrix which takes a value of 1 across the diagonal to give the correct intra-specific term in equations ⑦ and ⑧. The reason for the use of a matrix is due to the

relationship between the number of species and number of competition (alpha) terms: for any number of n species, there will be an $n^2 - n$ number of alpha terms:

$$A = \begin{pmatrix} 1 & \alpha_{12} & \alpha_{13} & \cdots & \alpha_{1n} \\ \alpha_{21} & \ddots & & & \alpha_{2n} \\ \alpha_{31} & & \ddots & & \vdots \\ \vdots & & & \ddots & \alpha_{n-1n} \\ \alpha_{n1} & \alpha_{n2} & \cdots & \alpha_{nn-1} & 1 \end{pmatrix}$$

This is because each species is assumed to compete against every other species, and $\alpha_{N_1N_2} \neq \alpha_{N_2N_1}$. In other words, the effect of N_1 on N_2 cannot be assumed to be the same as the effect of N_2 on N_1. Furthermore, the significance of a value of 1 along the diagonal is so that all the carrying capacities are equal. The values of alpha I chose were $\alpha_{N_1N_2} = 1.5$ and $\alpha_{N_2N_1} = 0.5$. I then plotted the trajectories of these populations to compare with the stochastic solutions.

To solve the ODEs, I used the solve_ivp function within the SciPy[32] module and to plot the solutions, matplotlib[33] was used.

2.3. Gillespie algorithm

There are a number of ways to integrate the stochastic form of the differential equation, for instance previous studies have made use of the Euler-Maruyama method[7]. The Gillespie algorithm generates solutions of the CTMC form of stochastic equations and their trajectories. This method was utilised in solving the CTMC primarily due to its accuracy of solutions produced at small population sizes, as it respects discrete population sizes.

For the first step of the algorithm, the system requires the initialization of population sizes and parameters including competition terms and growth rates. Each species has a population that can be in a discrete state i.e. they are real positive values such as 0, 1, 2, up until infinity. The algorithm then moves forwards in a particular time step, and at each stage the current state of one of the species' populations changes either by increasing or decreasing by 1. These changes are represented by the transition probabilities outlined in Table 1; either the growth rate or death rate of either species, so there are four different possibilities from which the state can change at each interval. A randomly generated number between zero and one is drawn and assigned to one of the four events based on their probabilities depending on how they are weighted by the probability. For example, if the growth

rate for N_1 was higher than the other rates, that particular event would likely be chosen. The changes can be defined as:

$$Change\ of\ state = \begin{cases} N_1 = N_1 + 1, & if\ Y < \dfrac{B_{N_1}}{Probability_{sum}} \\ N_1 = N_1 - 1, & if\ Y > \dfrac{B_{N_1}}{Probability_{sum}}\ and\ Y < \dfrac{B_{N_1} + D_{N_1}}{Probability_{sum}} \\ N_2 = N_2 + 1, & if\ Y > \dfrac{B_{N_1} + D_{N_1}}{Probability_{sum}}\ and\ Y < \dfrac{B_{N_1} + D_{N_1} + B_{N_2}}{Probability_{sum}} \\ N_2 = N_2 - 1, & otherwise. \end{cases}$$

where Y is the random value drawn, and the sum of the transition probabilities is defined as:

$$Probability_{sum} = B_{N_1} + B_{N_2} + D_{N_1} + D_{N_2}.$$

Next, the time interval T in which this update takes place must be calculated. This is dependent on the value of $Probability_{sum}$. For example, if there are many changes in the state of a species' population, the time steps will need to be closer together for a change of ±1 to either population. Whereas if the changes in a population are slow, the time between changes is going to be long. The distribution of the number of events can be described by a Poisson distribution with rate equal to $Probability_{sum}$. A Poisson distribution has an exponentially distributed time between events. To increment the time interval therefore, a random draw from this exponential distribution with rate $\frac{1}{Probability_{sum}}$ can be made. The algorithm then makes a stop at each point and evaluates which type of transition of population state takes place. Once this is done, each time value is increased by the randomly generated value in the previous step and the population count for either species is updated. The algorithm repeats these time steps until T has been exceeded.

2.4. Tau-leaping

Tau-leaping is another method of solving the CTMC problem, and is an extension of the Gillespie algorithm. The key difference between the two methods is that in tau-leaping all changes in population sizes are recorded over an interval of a constant specified length tau (τ) before the system is updated and moves onto the next time step. In other words, in each 'leap', the expected rate of

change of each population is computed. Rather than a single individual being born or dying in the time window, multiple individuals are added or subtracted. Each of the four changes are Poisson distributed with rate scaled by the time step tau, and are updated by making a random draw from each of these distributions, and adding or subtracting from the current population size. As this allows for the change of state for multiple individuals, as opposed to the movement of one individual at a time that the Gillespie algorithm allowed for, tau-leaping is computationally much faster than the Gillespie algorithm.

The rates are then recalculated for each species' populations after a 'leap' of τ, until the time interval has been exceeded. Whilst there exists literature on effectively selecting a value of τ[34], for this study I chose a value of $\tau = 0.01$, which was small enough to capture changes in the population, and still fast to run. This is further explored in the results section of this study.

2.5. Data

For this study, I used microbiome data which had been previously sampled from the guts of ten mice infected with *C. difficile*. The experiment was originally carried out by *Buffie et al.* (2012), and the data is publicly available and was published as part of *Stein et al. (2013)*[17]. The set contains the population sizes of multiple bacterial species at differing time points for each of the ten mice. The mice were orally administered *C. difficile* spores of the VPI 10463 strain. Each bacterial population was identified by the authors through 16S rRNA sequencing. *Stein et al.* (2013) ran their own parameter estimation algorithm and inferred values for the competition terms. This was based on the ODE form of the Lotka-Volterra equations, and parameters were inferred using a penalised regression. This acted as a benchmark for when I conducted my own inferences using this set of data.

2.6. Bayesian optimisation

Bayesian inference entails the use of Bayes' theorem (⑪) to continuously update the probability of a hypothesis being true, given previous evidence as it becomes established. However, for this model, it is not possible to derive a likelihood function which is where likelihood-free methods such as ABC can be used. This is a technique I utilised to estimate parameters, in conjunction with the previous methods of solving the CTMC.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad ⑪$$

$$where\ P(B|A)\ is\ the\ likelihood$$
$$P(A)\ is\ the\ prior\ probability$$
$$P(B)\ is\ the\ posterior\ probability$$

Firstly, I set prior distributions for each of the parameters to be inferred. This is so that the simulator initially has a distribution from which estimated parameters would be plausible. I defined priors from a uniform distribution with a lower bound of 0 and an upper bound of 5. The initial choice of these priors was due to an alpha value of 5 being very high for a competition term, and this value also cannot be below 0, so I knew that alpha was likely to be within this range.

We then made use of the Bayesian optimization for likelihood-free inference method, which requires the definition of a distance function between the simulator and the data. Then, the simulator generates another set of data, and calculates the discrepancies between this generated data and the observed data. For this distance, I chose the Euclidean distance, which measures linear distance between two points. This was summed over all time points, and the logarithm was taken. All ABC methods were implemented through the Engine for Likelihood Free-Inference (ELFI)[35] package in Python.

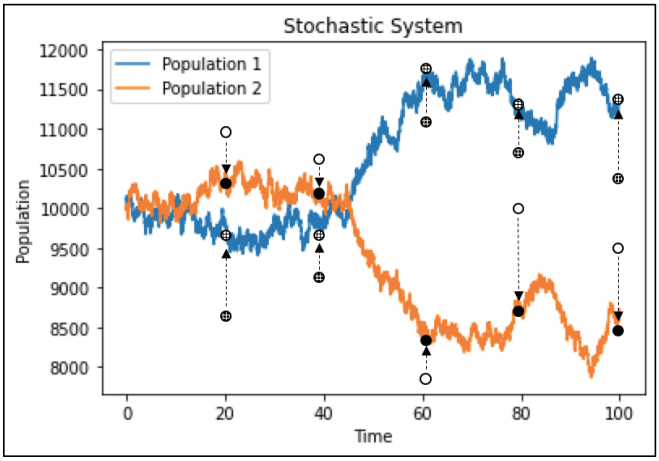

Figure 1: A demonstration of how the distance function calculates the discrepancies between values in the simulated data, and those in the observed data. The white grid patterned points and filled black points are the actual values of the observed data for N_1 and N_2 respectively. The hollow white points and black grid patterned points are the values generated by the simulator for N_1 and N_2 respectively. The dotted line signifies the discrepancy between the two points as calculated by the Euclidean distance function. Population 1 = N_1 and Population 2 = N_2.

Prior to fitting the data, I ran a number of simulations on artificially generated data generated from the model itself, with known parameters. The rationale behind this was to inspect which parameters

affect the accuracy of the estimations. Additionally, I did this to investigate which measurement times provide the most accurate inferences. That is, at which time interval and which time points data collected from the subject allowed the simulator to infer accurate alpha values. I simulated a set of 'observed' data by running the tau-leaping method on the stochastic form of the Lotka-Volterra equations. I did this for different combinations of values for the parameters and a number of varying measurement times. Firstly, I chose two different time intervals: from 0-100 and 0-20. The reason for experimenting with the latter was because the majority of population changes often occur within a short time interval at the beginning of an experimental run. I then simulated each of these intervals linearly, and also exponentially by distributing them in log space and then taking exponentials. The measurement points within these time intervals were uniformly distributed. Next, I created observed data with differing values of alpha, and then ran the simulator to test how accurately it could infer these values. Finally, I varied other parameters such as the carrying capacity and species growth rates to test whether these had any involvement in the simulator's ability to infer the unknown parameters. When testing the simulator, the only parameter that needed to be determined was alpha; all other parameters held assumed or known values.

When experimenting with the time intervals and measurement points, I tested the accuracy with observed alpha values of $\alpha_{N_1N_2} = 0.3$ and $\alpha_{N_2N_1} = 0.6$. Whichever time range would result in the simulator returning the most accurate values of alpha would indicate an optimal method for time points to collect data at.

Once I had tested the simulator to ensure the alpha term could be inferred for differing time steps and varying population sizes, carrying capacities and growth rates, I used it to fit the real observed data from *Stein et al. (2013)*[17].

A number of mice in this data did not have *C. difficile*, which narrowed down the quantity of observations. I first passed data into the simulator for one mouse which had *C. difficile*, and compared the competition it had with the bacteria *Akkermansia*. For the starting population sizes, I selected the values for the populations measured at time t=0. However, the authors of *Stein et al*. (2013) scaled the population sizes and measurement points by $10^{11}\ rRNAcopies/cm^3$[17]. As the gut of a mouse is only $1.34 \pm 0.8\text{cm}^3$ in volume[36], this means there were large population sizes for each species. Due to the long computational times and memory usage this scale would result in, I applied a scale of $10^5\ rRNAcopies/cm^3$. I was able to do this because this formulation of the equations is invariant with respect to this scaling[17]. I then estimated the competition parameters between *C. difficile* and

Coprobacillus, *and C. difficile* and *Enterococcus*. Finally, I compared my inferences to those made via a penalised regression in *Stein et al*. (2013) [17].

2.7. Computational resources utilised

The ODE model and stochastic CTMC model, and the parameter inference simulator were computed and programmed in Python version 3.7[37]. The versions of each module used were: elfi (0.7.6), gillespy2 (1.5.3), graphviz (2.42.3), matplotlib (3.3.0), networkx (1.11), numba (0.50.1), numpy (1.19.1), scipy (1.3.1), time (1.8), toolz (0.10.0). The Python IDE used was Spyder (4.1.4). The data used from *Stein et al*. (2013) [17] was downloaded as a Microsoft Excel spreadsheet and can be found at https://doi.org/10.1371/journal.pcbi.1003388.s001. All simulations were run on an Apple MacBook Air (2017). All the code we wrote for this study can be found at https://github.com/isa-a/mscproject.

Results

3.1. Comparison of deterministic and stochastic models

I started by comparing the deterministic and stochastic models across a range of parameter values. To begin, I used alpha values of $\alpha_{N_1N_2} = 1.5$ and $\alpha_{N_2N_1} = 0.5$. This was to test the effect a competitive species has on a less competitive species.

As N_1 was set to have a much stronger competitive effect on N_2 than N_2 did on N_1 (1.5 against 0.5), I expected there to be growth in N_1's population, and a decline of N_2. Once I produced trajectories for both my deterministic and stochastic model, I then compared the two models visually. Figure 2A displays solutions produced by the deterministic Lotka-Volterra system. From the plot it is clear that N_1 starts to grow from its initial size of 100 at a sharp incline and continues until it reaches the carrying capacity of 1000, at which there is no more room for growth so the population plateaus at this value. N_2 on the other hand, sees a small growth in its size, however it begins to decline shortly thereafter and dies off fairly fast (at around time t=10). This is due to N_2 having a larger starting population size (500) than N_1 (100) – as N_2 grows, it exhibits a stronger negative effect over time as a result of the competition upon it by N_1.

I then contrasted this output to that produced by Gillespie's algorithm to solve the CTMC (Fig. 2B). The two plots have a similar general shape. However, the presence of noise here is observable at this scale. I observed that the state of the populations is in fact not constant, and there are fluctuations in the sizes occurring throughout the time interval. This allowed for a clearer view of the population dynamics. Firstly, I found that there is an improved degree of accuracy when visually inspecting the times at which population sizes change. For example, N_2 appears to increase and decrease in miniscule sizes towards the end of its decline at around time 10. Whereas the deterministic model plot does not reflect this as there is no noise, so changes in population size at specific time points cannot be inferred. Secondly, N_1 peaked above the carrying capacity often.

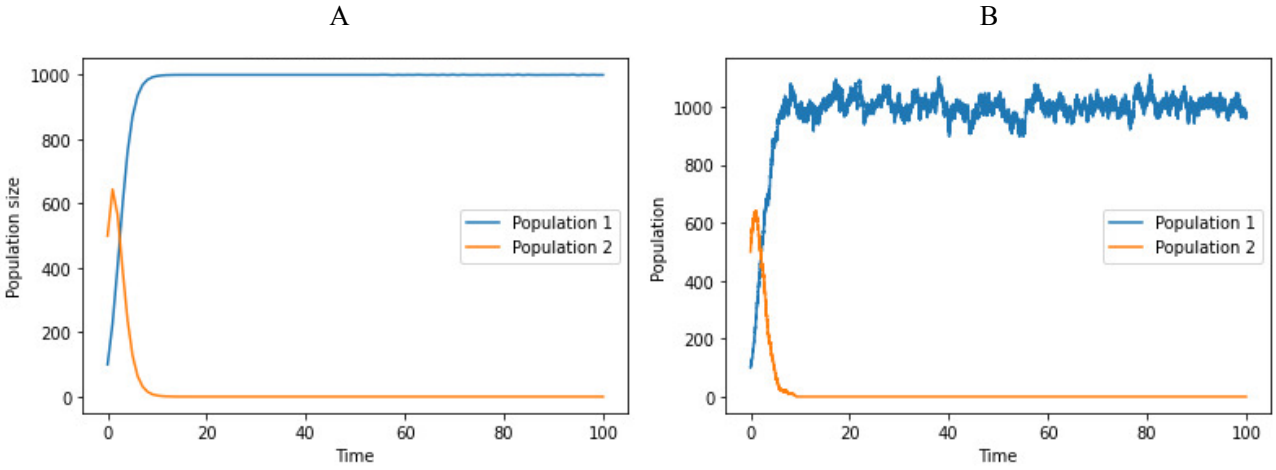

Figure 2: Plotted solutions of the deterministic Lotka-Volterra model (2A) with a carrying capacity of 1000, population sizes of 100 and 500 for N_1 and N_2 respectively. The out produced by Gillespie's algorithm (2B) assumes the same parameters as the deterministic model. Population 1 = N_1 and Population 2 = N_2. $\alpha_{N_1N_2} = 1.5$ and $\alpha_{N_2N_1} = 0.5$

I then compared the solutions produced by Gillespie's algorithm with the solutions based on the tau-leaping method. I found that the main difference in the methods, the use of tau, had an effect on the amount of noise produced in each trajectory. When I tested small values for tau, such as 0.01, there was a larger frequency of noise. Whereas high values of tau, for instance 1, produced fewer bits of noise, however there is a greater deviation represented through the larger peaks and dips. The trajectories also appear to have very low resolution, almost taking a polygon shape which prevents us from examining the finer changes in population sizes. This was to be expected however, as a smaller value of tau means there are a greater number of points to sample interactions from as opposed to larger values of tau. The value of 0.01 I initially chose for tau was done arbitrarily. Whilst there are specific procedures for selecting an appropriate value of tau[34], I selected 0.01 in order to maintain uniformity throughout the study. Figure 3 reflects the differences in noise as a result of the tau value chosen.

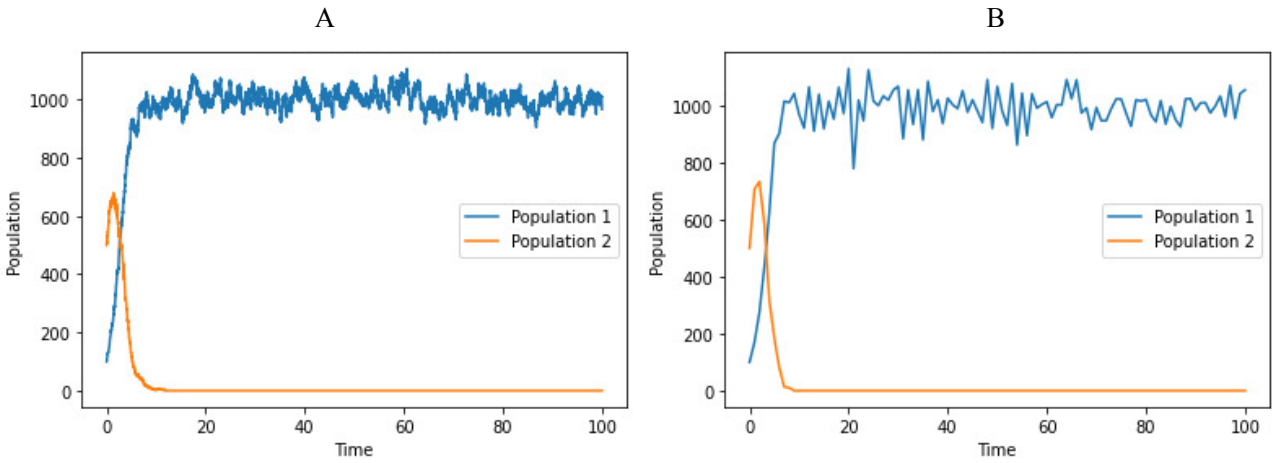

Figure 3: Solutions to the stochastic model computed via the tau-leaping method. Plot B was produced with a tau of 0.01, plot A with a tau of 1. Population 1 = N_1 and Population 2 = N_2.

Both the deterministic and stochastic versions of the model allowed for coexistence of species at large populations (Figure 4A, B). For arbitrary large starting population sizes, both species' populations sum to the shared carrying capacity at all times, and neither population dies off within a finite time range. As the time tends towards infinity, one species dying off would be expected due to randomness. Figure 4 shows a comparison of trajectories when all alpha values are set to 1, for different population sizes. For small starting population sizes (4C, D), coexistence does not occur across the time range, with one species dying off quickly and the other's population continuing to fluctuate. Smaller populations are therefore more strongly affected by the use of the stochastic model, as expected.

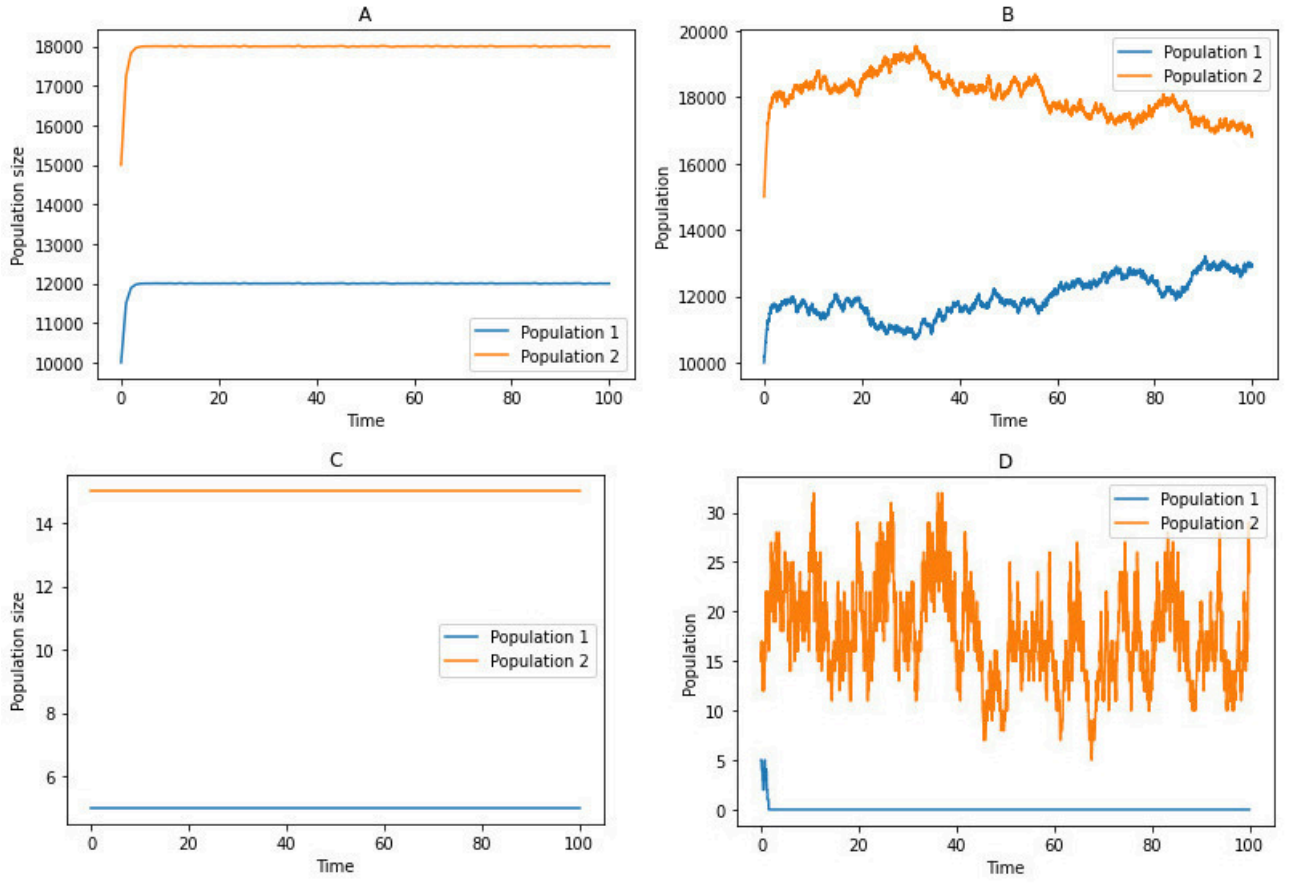

Figure 4: Coexistence can occur between species at large population sizes, as shown by the deterministic (A) and stochastic (B) models. At low population sizes (C), (D), one population dies off, However, this is due to randomness as opposed to competition. The deterministic model at low population sizes (C) appears as two horizontal. This is because the finer fluctuations in state are not considered, as they are in the stochastic model (D). Population 1 = N_1 was and Population 2 = N_2. $N_1 = 10{,}000$ and $N_2 = 15{,}000$ in A,B. $N_1 = 5$ and $N_2 = 15$ in C,D. Both competition terms were set equal ($\alpha_{ij} = 1$ and $\alpha_{ii} = 1$). The stochastic trajectories were computed by the tau-leaping method where $\tau = 0.01$.

I observed contrasting computation times when using both methods of solving the CTMC. For large population sizes upwards of 10,000, the Gillespie algorithm took a relatively long time to produce solutions compared to the tau-leaping method. I used the 'jit' function (just-in-time compiler) from Python's numba module, which ran the two models as a compiled language would, allowing for faster computation times. Thus, using jit with the tau-leaping method to solve the CTMC is much faster than just the Gillespie algorithm. Table 2 outlines the average times each method took to produce solutions over five different runs of the model. Computational speed is important for parameter inference as the simulator will be run thousands of times or more.

Table 2: Average time of computation for each method over five different runs of the Gillespie algorithm tau-leaping, with and without jit

Method	**Using jit (mm:ss:ms)**	**Without jit (mm:ss:ms)**
Gillespie algorithm	00:24:06	00:46:79
Tau-leaping	00:00:60	00:01:27

Having extended the tau-leaping method of solving the CTMC to be able to solve an n number of species, I was able to plot trajectories for 3 (Figure 5) and 4 different species (Figure 6).

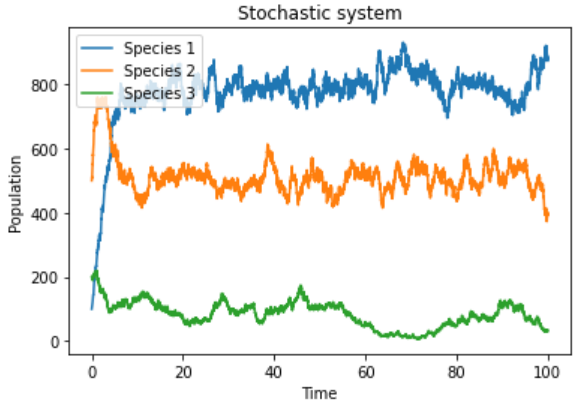

Figure 5: stochastic trajectories showing dynamics of 3 species with various competitive strengths on each other. Species 1 starts at a much smaller population size than species 2 and species 3, however its strength of competition means the species will grow while the other species decline in population size.

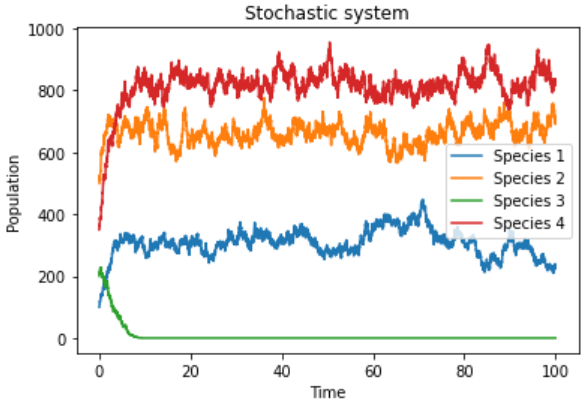

Figure 6: stochastic trajectories showing dynamics of 4 species with various competitive strengths on each other. In comparison to Figure 5, the newly introduced species 4 is more competitive than species 1, which inhibits its growth and species 4 is able to grow towards the carrying capacity. species 3 dies off in this scenario due to the increase in competition that results from species 4.

3.2. Estimating parameters with simulated data

To test the simulator's ability to estimate the competition parameter alpha (α), I used different combinations of parameter values and measurement points through the use of ABC. A time range of

0-100 distributed exponentially with measurement points after every 10 units of time produced the most accurate estimations of alpha values below 1 (Table 3). For simulated data where I set the alpha values to $\alpha_{N_1N_2} = 0.3$ and $\alpha_{N_2N_1} = 0.6$, the simulator returned mean estimations for alpha of 0.302 and 0.607 respectively (95% Credible Intervals 0.198-0.415 and 0.547-0.666 respectively, errors of 0.67% and 1.17% respectively). The accuracy of the estimations was exact to one significant figure. The plot in Figure 7 illustrates the discrepancy between the observed and simulated data; each point represents the difference between an inference from the two sets of data. Thus, the higher the discrepancy for each estimation, the greater the inaccuracy for that inference of alpha. There was a high concentration of points around values of 0.3 and 0.6 for each alpha respectively, indicating values very close to the observed alphas. The posterior marginals generated for both α estimations can be seen plotted in Figure 8. These simply outline the estimates for the α parameter as a distribution.

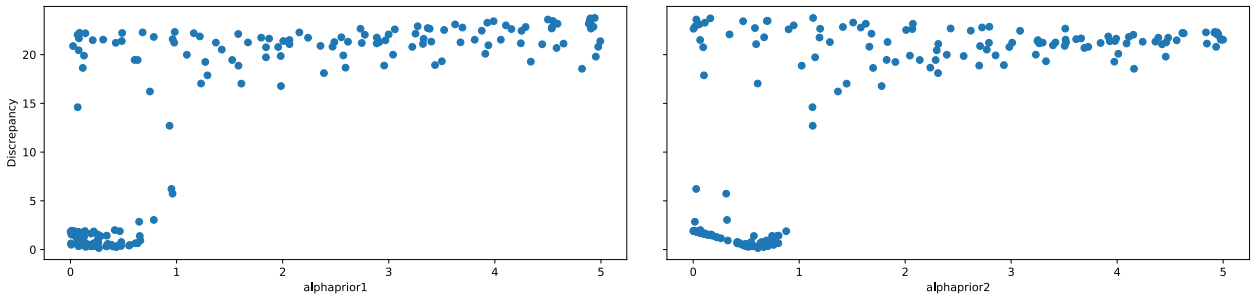

Figure 7: Discrepancies for each alpha value inferred, where the observed values were $\alpha_{N_1N_2} = 0.3$ and $\alpha_{N_2N_1} = 0.6$

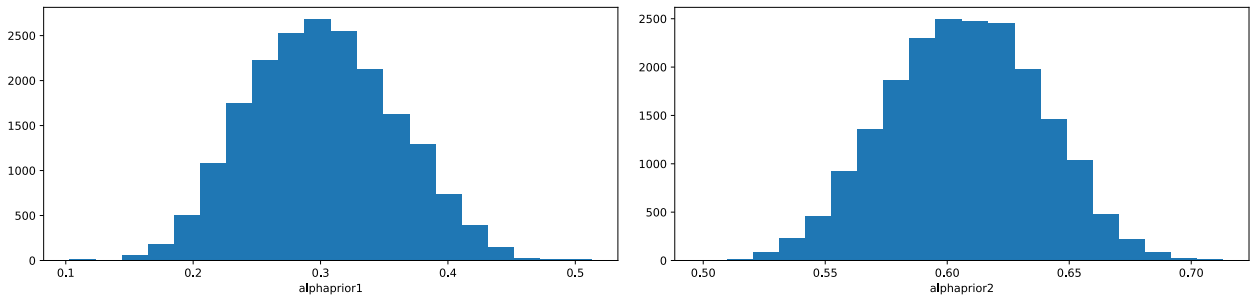

Figure 8: Posterior marginals of each alpha value inferred, where the observed values were $\alpha_{N_1N_2} = 0.3$ and $\alpha_{N_2N_1} = 0.6$

The other time intervals and measurement points I analysed varied in the accuracy of estimations the simulator returned. As well as testing the exponentially distributed 0-100-time range, I ran the simulator with linearly distributed measurement points in the 0-100-time range. For alphas of $\alpha_{N_1N_2} = 0.3$ and $\alpha_{N_2N_1} = 0.6$, the simulator returned estimations of 0.295 and 0.621 respectively (95% CI 0.064-0.505 and 0.486-0.736 respectively). In comparison to the exponentially distributed interval of 0-100 this linear range was slightly less accurate to a greater number of significant figures but still returned exact values to one significant figure (errors of 1.67% for $\alpha_{N_1N_2}$ and 3.5% for $\alpha_{N_2N_1}$).

The errors of estimations for this time scale were larger than the errors when times were exponentially distributed. For a linearly distributed time scale of 0-20, the simulator returned an estimation of 0.52 for $\alpha_{N_1N_2}$ and 0.795 for $\alpha_{N_2N_1}$ (95% CI 0.053-0.968 and 0.557-1.040 respectively). This same time scale distributed exponentially resulted in the simulator inferring values more accurately however, with estimations for $\alpha_{N_1N_2}$ and $\alpha_{N_2N_1}$ being 0.293 and 0.602 respectively. The reason for experimenting with time ranges of 0-20 was due to my observation that this is the range within which the majority of the interactions were occurring, and as a result the population sizes for both species were changing the most (Figure 9). After time 20, both populations were in a relatively steady state.

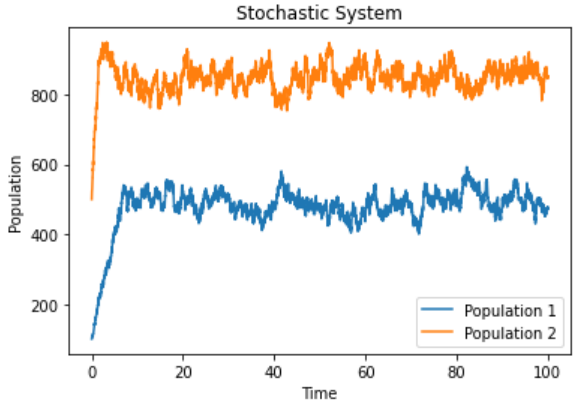

Figure 9: The dynamics of each species when $\alpha_{N_1N_2} = 0.3$ and $\alpha_{N_2N_1} = 0.6$. The changes in each population as a result of interactions take place before time t=20. It is important to note that even if the two trajectories do not intersect, it does not necessarily mean there is no interspecies competition. Population 1 = N_1 and Population 2 = N_2.

Table 3: Inferences for alpha terms made by the simulator for varying time points at which microbiome data is collected

Time range	**No. of measurement points**	**Mean of estimations ($\alpha_{N_1N_2}$, $\alpha_{N_2N_1}$)**	**95% credible intervals ($\alpha_{N_1N_2}$, $\alpha_{N_2N_1}$)**
Linear 10-100	10	0.295, 0.621	(0.064-0.505), (0.486-0.736)
Exponential 10-100	10	0.302, 0.607	(0.198-0.415), (0.547-0.666)
Linear 0-20	10	0.520, 0.795	(0.053-0.968), (0.557-1.040)
Exponential 0-20	10	0.293, 0.602	(0.194-0.394), (0.543-0.661)

For different alpha values below 1, the simulator provided a sensible degree of accuracy in its estimations. However, I found that when I set at least one of the alpha values to 1 or above, the inferences it made had large errors. For example, a set of data I generated had alpha terms of $\alpha_{N_1N_2} = 0.6$ and $\alpha_{N_2N_1} = 1$. The simulator estimated these values to be 0.211 and 0.883 respectively. While

the estimation for $\alpha_{N_2N_1}$ was not as far off its observed value, there was a relatively significant error in the estimation for $\alpha_{N_1N_2}$ (approximately 0.389). As a result of this finding, I conducted the same experiment with different observed values for one of the alpha terms above 1 (Table 4) to further investigate this loss of accuracy. The rationale behind testing this choice of values was to investigate how sensitive the simulator was to values above 1. I kept $\alpha_{N_1N_2}$ fixed at a value of 0.6 and varied values of $\alpha_{N_2N_1}$ as 1.5 and 2. Table 4 outlines the estimations returned for both alpha term estimations when at least one observed value was above 1.

Table 4: Estimates of alpha values returned by the simulator when at least one observed value of alpha was 1 or above

Alpha values tested ($\alpha_{N_1N_2}$, $\alpha_{N_2N_1}$)	**Mean of estimations ($\alpha_{N_1N_2}$, $\alpha_{N_2N_1}$)**	**95% CI of estimations ($\alpha_{N_1N_2}$, $\alpha_{N_2N_1}$)**
0.6 and 1	0.211, 0.883	(0.007-0.636), (0.739-1.011)
0.6 and 1.5	1.124, 1.693	(0.533-1.822), (1.428-2.141)
0.6 and 2	1.165, 2.218	(0.051-1.662), (2.777-3.003)

On the other hand, in situations where there was coexistence between two species with large populations, the simulator was able to provide slightly more accurate inferences of α. For the scenario depicted in Figure 4A and 4B, all competition terms were 1 across the α matrix to permit coexistence. The simulator inferred values for $\alpha_{N_1N_2}$ and $\alpha_{N_2N_1}$ to be 0.724 and 0.722 respectively. The discrepancy plot for this estimation (Figure 10) shows a preference for values generally near to the observed values of 1, however there was still an error of 27.6% for $\alpha_{N_1N_2}$ and 27.8% for $\alpha_{N_2N_1}$.

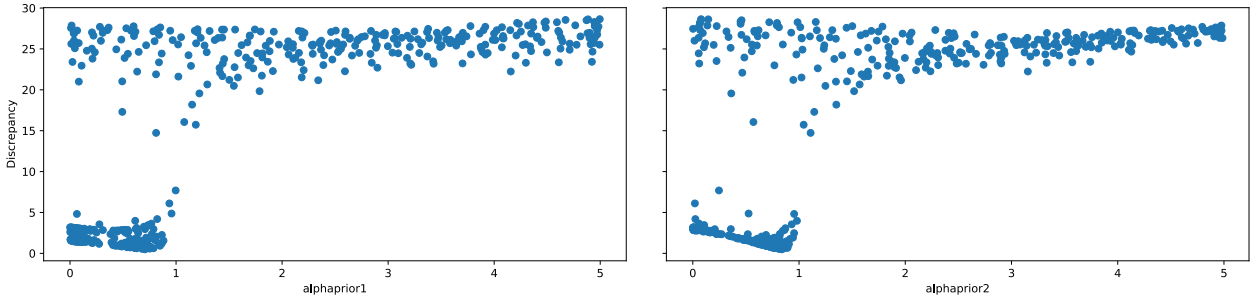

Figure 10: Discrepancies for competition terms estimated when their observed values were 1. Both species here coexisted and neither died off.

The carrying capacity did not appear to have a large effect on the estimations returned. For an increased carrying capacity of 10,000 for both populations, the simulator estimated $\alpha_{N_1N_2}$ as 0.224 and $\alpha_{N_2N_1}$ as 0.557, for observed values of 0.3 and 0.6 respectively (95% CI 0.122-0.332 and 0.509-0.607 respectively, error of 7.6% for $\alpha_{N_1N_2}$ and 4.3% for $\alpha_{N_2N_1}$).

3.3. Estimating parameters with experimental observed data

The inferences I made on the experimental data from *Stein et al.* (2013) [17] estimated *C. difficile* and *Akkermansia* to be extremely competitive upon each other. The simulator returned values of $\alpha_{C.diff/Akkermansia} = 2.68$ and $\alpha_{Akkermansia/C.diff} = 2.97$ (95% CI 0.112-4.912 and 0.384-4.907 respectively). These values are a considerable contrast to the estimates of $\alpha_{C.diff/Akkermansia} = 0.18$ and $\alpha_{Akkermansia/C.diff} = 0.38$ inferred in *Stein et al.* (2013) who applied a scale of $10^{11}\ rRNAcopies/cm^3$ compared to my scale of $10^5\ rRNAcopies/cm^3$. In comparison to Stein's estimations, there is an error of 1374% for $\alpha_{N_1N_2}$ and 683% for $\alpha_{N_2N_1}$. The wide credible intervals also tell us that these estimations are not precise. However, the greater competitive effect that *Akkermansia* has on *C. difficile* can still be seen. The discrepancy plots produced for these estimations (Figure 11) were difficult to interpret, there was a high concentration of points at each of the bounds defined for the priors (0 and 5) and a somewhat random scatter of points in between.

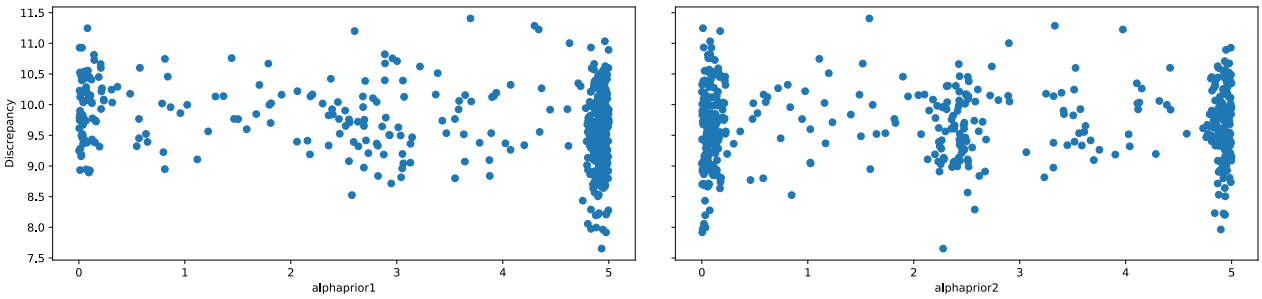

Figure 11: Discrepancies for the estimations of competition parameters between C. difficile and Akkermansia. The mean estimations returned were $\alpha_{N_1N_2} = 2.68$ and $\alpha_{N_2N_1} = 2.97$.

For competition between *C. difficile* and *Coprobacillus*, the simulator inferred parameter values of $\alpha_{C.diff/Coprobacillus} = 3.35$ and $\alpha_{Coprobacillus/C.diff} = 0.166$ (95% CI 0.422-4.961 and 0.005-0.525 respectively). Stein's estimation for these parameters were $\alpha_{C.diff/Coprobacillus} = 0.303$ and $\alpha_{Coprobacillus/C.diff} = 0.443$. The error of my estimations was therefore 1005.6% for $\alpha_{C.diff/Coprobacillus}$ and 62.5% for $\alpha_{Coprobacillus/C.diff}$.

I also inferred parameters for competition between *C. difficile* and *Enterococcus*. I computed estimations of $\alpha_{C.diff/Enterococcus} = 0.22$ and $\alpha_{Enterococcus/C.diff} = 2.89$ (95% CI 0.007-0.683 and 0.25-4.914), in comparison to the inferences of $\alpha_{C.diff/Enterococcus} = 0.01$ and

$\alpha_{Enterococcus/C.diff} = 0.11$ from Stein et al (2013) [17]. My estimations' errors were 2100% for $\alpha_{C.diff/Enterococcus}$ and 2527% for $\alpha_{Enterococcus/C.diff}$.

Table 5: A comparison of the competition parameters estimated by my simulator, and those estimated in Stein et al. (2013)

Competition scenario	**Parameters estimated**	**Parameters estimated by *Stein et al.***
C. difficile and *Akkermansia*	$\alpha_{C.diff/Akkermansia} = 2.68$ $\alpha_{Akkermansia/C.diff} = 2.97$	$\alpha_{C.diff/Akkermansia} = 0.18$ $\alpha_{Akkermansia/C.diff} = 0.38$
C. difficile and *Coprobacillus*	$\alpha_{C.diff/Coprobacillus} = 3.35$ $\alpha_{Coprobacillus/C.diff} = 0.166$	$\alpha_{C.diff/Coprobacillus} = 0.303$ $\alpha_{Coprobacillus/C.diff} = 0.443$
C. difficile and *Enterococcus*	$\alpha_{C.diff/Enterococcus} = 0.22$ $\alpha_{Enterococcus/C.diff} = 2.89$	$\alpha_{C.diff/Enterococcus} = 0.01$ $\alpha_{Enterococcus/C.diff} = 0.11$

While ABC is a computationally fast approach to making inferences, each of them utilised a considerable amount of computer memory and the simulator took a relatively long amount of time to complete one set of inferences, even with the tau-leaping method and jit. Table 6 outlines the length of time each set of species' inferences took through the simulator.

Table 6: Time taken for the simulator to make inferences for each of the bacterial species combinations I tested

Species run through simulator	**Inference computation time (hh:mm:ss)**
C. difficile and *Akkermansia*	00:34:45
C. difficile and *Coprobacillus*	00:30:56
C. difficile and *Enterococcus*	00:31:08

Discussion

4.1. Deterministic and stochastic systems

The deterministic form of the Lotka-Volterra model is ideal for initially getting an idea of how different species behave at different population sizes, carrying capacities, growth rates and amount of competition between species. Numerical integration tells us, how fast a species grows, how fast the other(s) dies off, or whether coexistence occurs. It is chiefly beneficial for examining a species' general population dynamics over a time interval, but cannot be used to investigate the finer fluctuations in state, due the equilibrium state of each species' population being constant. While it can be used to infer parameters of the model, as done in *Stein et al*. (2013) [17], the inferences returned are to a poor degree of accuracy at small population sizes.

The stochastic form of the model on the other hand, considers as many population changes as possible given the time step that has been defined. The tau-leaping outperforms the Gillespie algorithm computationally, and allows for a constant time step as opposed to a randomly generated value for each step. While it can be thought of as an extension of the Gillespie algorithm, the latter becomes computationally slow for large population sizes. There was a considerable improvement in computational speed for the tau-leaping method compared to the Gillespie algorithm. This can be further improved with the use of Python functions like jit, which translate interpreted code into compiled code.

Cao et al. (2006) discussed methods to select an efficient step size value for τ, which are in fact improvements on previous step size selection procedures to give more accurate results of population changes at each time step[34]. Due to the time constraints this study fell within, I utilised neither of these methods. Rather, I chose an arbitrary, small value of τ for my calculations to ensure as many population changes could be recorded as possible. However, this was not a hurdle in obtaining my results.

4.2. Accuracy of the simulator in estimating parameters

The measurement points at which data is collected certainly has an effect on how accurately the simulator is able to return parameter estimations. Specifically, those measurement points which are exponentially distributed across the time interval generally provided a greater degree of accuracy in

estimations compared to when they are linearly distributed. Furthermore, the number of measurement points taken within the time interval also had an impact on the accuracy of inferences. For example, the exponentially distributed time scale of 0-100 with five equally spaced measurement points returned an underestimation of $\alpha_{N_2N_1}$ where the observed value was 0.6, whereas data collected at 10 different time points provided estimations exact to three significant figures. Additionally, for an exponentiated time interval of 0-20, with four measurement points the estimations were slightly off to three significant figures, a degree of accuracy which was observed when this same time interval had ten measurement points within it. It should be noted that due to the stochastic nature of the model used each simulation is going to return a different output, and whilst there were differing degrees of accuracy amongst outputs, the variation in errors amongst them was not too great. While a greater number of measurement points appeared to improve accuracy of estimations, it remains to be seen what effect uniformly distributed the points has. For example, in the original experiment conducted by *Buffie et al*. (2012) using the murine model, data was collected at non-uniformly distributed measurement points. The inferences made in *Stein et al*. (2013) appeared to give plausible values for the parameters in comparison to my estimations; perhaps measurement points which were not equally spaced out could be a factor in improving estimations.

Another key finding through the various simulations I ran was the simulator's considerable loss of accuracy when at least one of the competition terms had observed values of 1 or above. Considering that a greater value of α means a greater competitive effect on a species, it makes sense that the species with the lower α value will die off faster. As a result of the less competitive species' population size reaching zero, the measurement points towards the end of the time interval will hold a value of zero. Thus, there is not sufficient information on the population size for the simulator to give a precise estimate (competition cannot occur between two species after one of them has died off). So, the problem here is not the numerical value of the competition term, rather that the simulator does not have information to work with. Within all instances of my experiments where only one observed competition term was 1 or above and the other was below 1, estimations for both parameters were inaccurate. A potential way to correct this would be to sample data from more measurement points at the beginning of the time interval, to compensate for the lack of information towards the end of the interval. In spite of erroneous values returned by the simulator, this still allows qualitative inferences that a competition parameter is greater than one, which may be all that is needed to find a species which kills *C. difficile*.

It is also important to consider the distinction between the accuracy and the precision of estimations. For example, the 95% credible intervals returned for each estimation are useful in gauging precision.

Even if the simulator returned an estimation that had a large error compared to the observed value, as long as the observed value lies within a narrow 95% CI, we can say the inference was precise. The computation of credible intervals was an improvement on previous inference methods, which did not allow accuracy or precision to be gauged.

The scaling of parameter values as done in *Stein et al.* (2013) [17] was also difficult to work out; there was ambiguity as to which parameters I should have scaled when fitting the observed data. While it makes sense that the authors of applied scales considering the size of the mouse guts from which they collected the data, computational issues such as speed and lack of data for inferences to be made from were present in my study. As a result, I had to use smaller scales, and omit scaling of certain parameters such as the growth rate and time interval. Perhaps the correct combination of scaled parameters would be able to provide accurate estimations, but it is important to establish such combinations and the rationality behind them prior to making inferences, and especially so as the stochastic model is not scaling invariant.

4.3. Extension of the simulator and future work

There is a great diversity of the bacteria within the gut microbiota, and in reality there are several different species competing against each other, not just two. Thus, the inference step of the simulator needs to be extended to estimate several parameters. As each species competes against every other species and itself, for an n number of species, there will be n^2 terms ($n^2 - n$ competition terms, plus n carrying capacity terms). This means the simulator will be taking in many more parameters to return estimations for, resulting in longer computational times and inevitably greater memory usage. Increased accuracy of stochastic solutions and faster computational speeds would assist in inferring a greater number of parameters, which is where compiled code and an optimal τ would be useful. However, while ABCs are the fastest inference approach we have available, the current methods I used in this study are not capable of these computations; there would be too many parameters and not enough information to calculate discrepancies from. One possible way of circumventing this problem would be to use more informative priors, from distributions other than the uniform distribution. For example, a half-normal distribution centred at zero, which would move the possible parameter values closer to 0. Additionally, previous inferences of parameters made from experimental data, such as in *Stein et al*. (2013) [17] could be integrated into the selection procedure of priors. There could still be inaccurate inferences of large values, but the simulator would require more evidence from the population trajectories to support such values being plausible. Extending ELFI's capability of

handling high numbers of parameters would be another way of making a greater number of inferences at a time.

The results I obtained from estimating parameters also provide a foundation for shaping the setups of future experimental work. The finding that measurement points affect estimation accuracy bears significance on this. In the original experiment from which the data was collected, the gut of each mouse was measured within a time scale of 0-23 days. The measurement points within this scale were not equally spaced out, whereas when I initially tested different time scales, I distributed points equally. Doing so returned both accurate and precise estimations for generated data I tested the simulator with. However, when it came to fitting the actual data from the experiment with the mice, my estimations were largely inaccurate in comparison to those made by previous methods. In theory, the more points data is measured at, the more information the simulator has to make inferences from, so denser sampling, whether uniformly distributed or not may be optimal. However, in reality, each data collection point would require the sacrifice of a mouse, and in accordance with the relevant set of ethics, using fewer mice used would be desirable. Furthermore, metagenomics can assist in the optimal procedures to take in future experiments. For example, plotting trajectories for species at differing parameter values gives an idea of population dynamics over time, so if one species is dying off before a certain time point, it would make sense to only collect data up until that time.

4.4. Application to the epidemiology of *C. difficile*

Clostridioides difficile is dormant within the gut microbiota and kept at bay through bacterial competition with other species[18]. Growth of *C. difficile* occurs when antibiotics wipe out the other species of bacteria and there are no microorganisms left to inhibit its growth. Thus, if we know which species show strong competitive effects against *C. difficile*, it would be possible to further investigate these in relation to *C. difficile*. For example, a bacterial species that already resides in the gut and keeps *C. difficile* at bay, or a species taken from elsewhere that has proven to outcompete *C. difficile* could be examined for a potential new treatment option. The idea would be to first identify a particular species of bacteria that has a strong competitive effect on *C. difficile* (i.e. an alpha term above 1). Then, after examining this particular species to ensure it does not cause further infections or serious side effects, it could be included in a treatment pill. Conducting further experiments similar in nature to *Buffie et al.* (2012) [16], for example reintroducing this bacteria in varying amounts to the gut through the pill, would allow us to investigate if *C. difficile* is killed. Such experiments could then help determine dosage and schedules for this option of treatment.

The ability to design counterfactual runs of the model would also play a key role in the investigation of *C. difficile* treatment methods; after fitting experimental data to the simulator, it can then be run with parameter values different to those that were actually observed in the experiment. This allows us to predict what the outcome would be if we made an intervention, for example, if we were to add a dose of *Akkermansia* half way through, which is more competitive on *C. difficile* than *C. difficile* is on *Akkermansia*, examine the population dynamics after administering antibiotics, or if we varied the starting population sizes.

C. difficile is not the only application of epidemiology this project should be limited to. The Lotka-Volterra equations are analogous to the SIR compartmental models and their extensions, with differential equations representing entries and exits of each stage of an infectious disease. If these rates can be stochastically modelled similarly to how a species' population would be modelled, ABC could also be used to make inferences of various unknown parameters. Complex compartmental models for which it is difficult to derive a likelihood would especially benefit from likelihood-free methods.

4.5. Conclusion

There is a wide scope for future work on this project and extending it to the application of different fields, including epidemiology, but also microbiology and ecology. When studied in conjunction with one another, the findings and results drawn from these studies allow us to improve current methods in conducting experiments and treating and preventing disease. Mathematical and statistical methods continue to improve, and their real-world applications greatly aid in the study of disease.

References

1. Smolla M, Gilman RT, Galla T, Shultz S. Competition for resources can explain patterns of social and individual learning in nature. Proceedings.Biological sciences. 2015; 282 (1815): 20151405. Available from: doi: 10.1098/rspb.2015.1405 Available from: https://pubmed.ncbi.nlm.nih.gov/26354936 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4614750/ .

2. Gilpin ME. Do Hares Eat Lynx? The American Naturalist. 1973; 107 (957): 727-730. Available from: http://www.jstor.org/stable/2459670 .

3. Pigot AL, Tobias JA. Species interactions constrain geographic range expansion over evolutionary time. Ecology Letters. 2013; 16 (3): 330-338. Available from: doi: 10.1111/ele.12043 Available from: https://doi.org/10.1111/ele.12043 .

4. Clutton-Brock TH, Albon SD. The Roaring of Red Deer and the Evolution of Honest Advertisement. Behaviour. [Online] Brill; 1979;69(3–4): 145–170. Available from: doi:10.1163/156853979x00449

5. Pusceddu M, Mura A, Floris I, Satta A. Feeding strategies and intraspecific competition in German yellowjacket (Vespula germanica). PloS one. 2018; 13 (10): e0206301. Available from: doi: 10.1371/journal.pone.0206301 Available from: https://pubmed.ncbi.nlm.nih.gov/30365519 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6203408/ .

6. Marini G, Guzzetta G, Baldacchino F, Arnoldi D, Montarsi F, Capelli G, et al. The effect of interspecific competition on the temporal dynamics of Aedes albopictus and Culex pipiens. Parasites & vectors. 2017; 10 (1): 102. Available from: doi: 10.1186/s13071-017-2041-8 Available from: https://pubmed.ncbi.nlm.nih.gov/28228159 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5322594/ .

7. Shen P, Lees JA, Bee GCW, Brown SP, Weiser JN. Pneumococcal quorum sensing drives an asymmetric owner–intruder competitive strategy during carriage via the competence

regulon. Nature Microbiology. 2019; 4 (1): 198-208. Available from: doi: 10.1038/s41564-018-0314-4 Available from: https://doi.org/10.1038/s41564-018-0314-4 .

8. Alberts B, Johnson A, Lewis J, et al. Molecular Biology of the Cell. 4th edition. New York: Garland Science; 2002. The Diversity of Genomes and the Tree of Life. Available from: https://www.ncbi.nlm.nih.gov/books/NBK26866/

9. Hibbing ME, Fuqua C, Parsek MR, Peterson SB. Bacterial competition: surviving and thriving in the microbial jungle. Nature reviews.Microbiology. 2010; 8 (1): 15-25. Available from: doi: 10.1038/nrmicro2259 Available from: https://pubmed.ncbi.nlm.nih.gov/19946288 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2879262/ .

10. Corander J, Fraser C, Gutmann MU, Arnold B, Hanage WP, Bentley SD, et al. Frequency-dependent selection in vaccine-associated pneumococcal population dynamics. Nature Ecology & Evolution. 2017; 1 (12): 1950-1960. Available from: doi: 10.1038/s41559-017-0337-x Available from: https://doi.org/10.1038/s41559-017-0337-x .

11. Brisson D. Negative Frequency-Dependent Selection Is Frequently Confounding. Frontiers in Ecology and Evolution. 2018; 6 10. Available from: https://www.frontiersin.org/article/10.3389/fevo.2018.00010 .

12. Guinane CM, Cotter PD. Role of the gut microbiota in health and chronic gastrointestinal disease: understanding a hidden metabolic organ. Therapeutic advances in gastroenterology. 2013; 6 (4): 295-308. Available from: doi: 10.1177/1756283X13482996 Available from: https://pubmed.ncbi.nlm.nih.gov/23814609 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3667473/ .

13. Cabral DJ, Penumutchu S, Norris C, Morones-Ramirez J, Belenky P. Microbial competition between Escherichia coli and Candida albicans reveals a soluble fungicidal factor. Microbial cell (Graz, Austria). 2018; 5 (5): 249-255. Available from: doi: 10.15698/mic2018.05.631 Available from: https://pubmed.ncbi.nlm.nih.gov/29796389 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5961918/ .

14. Wade W. Unculturable bacteria--the uncharacterized organisms that cause oral infections. Journal of the Royal Society of Medicine. 2002; 95 (2): 81-83. Available from: doi: 10.1258/jrsm.95.2.81 Available from: https://pubmed.ncbi.nlm.nih.gov/11823550 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1279316/ .

15. Stewart EJ. Growing Unculturable Bacteria. Journal of Bacteriology. 2012; 194 (16): 4151. Available from: doi: 10.1128/JB.00345-12 Available from: http://jb.asm.org/content/194/16/4151.abstract .

16. Buffie CG, Jarchum I, Equinda M, Lipuma L, Gobourne A, Viale A, et al. Profound Alterations of Intestinal Microbiota following a Single Dose of Clindamycin Results in Sustained Susceptibility to Clostridium difficile-Induced Colitis. Infection and immunity. 2012; 80 (1): 62. Available from: doi: 10.1128/IAI.05496-11 Available from: http://iai.asm.org/content/80/1/62.abstract .

17. Stein RR, Bucci V, Toussaint NC, Buffie CG, Rätsch G, Pamer EG, et al. Ecological modeling from time-series inference: insight into dynamics and stability of intestinal microbiota. PLoS computational biology. 2013; 9 (12): e1003388. Available from: doi: 10.1371/journal.pcbi.1003388 Available from: https://pubmed.ncbi.nlm.nih.gov/24348232 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3861043/ .

18. Bien J, Palagani V, Bozko P. The intestinal microbiota dysbiosis and Clostridium difficile infection: is there a relationship with inflammatory bowel disease? Therapeutic advances in gastroenterology. 2013; 6 (1): 53-68. Available from: doi: 10.1177/1756283X12454590 Available from: https://pubmed.ncbi.nlm.nih.gov/23320050 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3539291/ .

19. Baban ST, Kuehne SA, Barketi-Klai A, Cartman ST, Kelly ML, Hardie KR, et al. The role of flagella in Clostridium difficile pathogenesis: comparison between a non-epidemic and an epidemic strain. PloS one. 2013; 8 (9): e73026. Available from: doi: 10.1371/journal.pone.0073026 Available from: https://pubmed.ncbi.nlm.nih.gov/24086268 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3781105/ .

20. Di Bella S, Ascenzi P, Siarakas S, Petrosillo N, di Masi A. Clostridium difficile Toxins A and B: Insights into Pathogenic Properties and Extraintestinal Effects. Toxins. 2016; 8 (5):

134. Available from: doi: 10.3390/toxins8050134 Available from: https://pubmed.ncbi.nlm.nih.gov/27153087 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4885049/ .

21. Lawley TD, Croucher NJ, Yu L, Clare S, Sebaihia M, Goulding D, et al. Proteomic and Genomic Characterization of Highly Infectious Clostridium difficile 630 Spores. Journal of Bacteriology. 2009; 191 (17): 5377. Available from: doi: 10.1128/JB.00597-09 Available from: http://jb.asm.org/content/191/17/5377.abstract .

22. Hamm EE, Voth DE, Ballard JD. Identification of Clostridium difficile toxin B cardiotoxicity using a zebrafish embryo model of intoxication. Proceedings of the National Academy of Sciences of the United States of America. 2006; 103 (38): 14176-14181. Available from: doi: 10.1073/pnas.0604725103 Available from: https://pubmed.ncbi.nlm.nih.gov/16966605 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1599930/ .

23. Lessa FC, Mu Y, Bamberg WM, Beldavs ZG, Dumyati GK, Dunn JR, et al. Burden of Clostridium difficile Infection in the United States. N Engl J Med. 2015; 372 (9): 825-834. Available from: doi: 10.1056/NEJMoa1408913 Available from: https://doi.org/10.1056/NEJMoa1408913 .

24. Lessa FC, Gould CV, McDonald LC. Current status of Clostridium difficile infection epidemiology. Clinical infectious diseases : an official publication of the Infectious Diseases Society of America. 2012; 55 Suppl 2 S65-S70. Available from: doi: 10.1093/cid/cis319 Available from: https://pubmed.ncbi.nlm.nih.gov/22752867 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3388017/ .

25. Guh AY, Mu Y, Winston LG, Johnston H, Olson D, Farley MM, et al. Trends in U.S. Burden of Clostridioides difficile Infection and Outcomes. New England Journal of Medicine. [Online] Massachusetts Medical Society; 2020;382(14): 1320–1330. Available from: doi:10.1056/nejmoa1910215

26. Ofosu A. Clostridium difficile infection: a review of current and emerging therapies. Annals of gastroenterology. 2016; 29 (2): 147-154. Available from: doi: 10.20524/aog.2016.0006

Available from: https://pubmed.ncbi.nlm.nih.gov/27065726 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4805733/ .

27. Brandt LJ. Fecal transplantation for the treatment of Clostridium difficile infection. Gastroenterology & hepatology. 2012; 8 (3): 191-194. Available from: https://pubmed.ncbi.nlm.nih.gov/22675283 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3365524/ .

28. Lotka AJ. Contribution to the Theory of Periodic Reactions. The Journal of physical chemistry. 1910; 14 (3): 271-274. Available from: doi: 10.1021/j150111a004 Available from: https://doi.org/10.1021/j150111a004 .

29. Volterra V. Variations and Fluctuations of the Number of Individuals in Animal Species living together. ICES Journal of Marine Science. 1928; 3 (1): 3-51. Available from: doi: 10.1093/icesjms/3.1.3 Available from: https://doi.org/10.1093/icesjms/3.1.3 .

30. Hu S, Zhang Q, Wang J, Chen Z. Real-time particle filtering and smoothing algorithms for detecting abrupt changes in neural ensemble spike activity. Journal of neurophysiology. 2018; 119 (4): 1394-1410. Available from: doi: 10.1152/jn.00684.2017 Available from: https://pubmed.ncbi.nlm.nih.gov/29357468 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5966736/ .

31. Lintusaari J, Gutmann MU, Dutta R, Kaski S, Corander J. Fundamentals and Recent Developments in Approximate Bayesian Computation. Systematic Biology. 2017; 66 (1): e66-e82. Available from: doi: 10.1093/sysbio/syw077 Available from: https://pubmed.ncbi.nlm.nih.gov/28175922 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5837704/ .

32. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. Nature Methods. 2020; 17 (3): 261-272. Available from: doi: 10.1038/s41592-019-0686-2 Available from: https://doi.org/10.1038/s41592-019-0686-2 .

33. Hunter JD. Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering. 2007; 9 (3): 90-95. Available from: doi: 10.1109/MCSE.2007.55 .

34. Cao Y, Gillespie DT, Petzold LR. Efficient step size selection for the tau-leaping simulation method. The Journal of chemical physics. 2006; 124 (4): 044109. Available from: doi: 10.1063/1.2159468 Available from: https://doi.org/10.1063/1.2159468 .

35. Lintusaari J, Vuollekoski H, Kangasrääsiö A, Skytén K, Järvenpää M, Gutmann M, et al. ELFI: Engine for likelihood-free inference. Journal of Machine Learning Research. 2017; 19 .

36. Casteleyn C, Rekecki A, Van Der Aa A, Simoens P, Van Den Broeck W. Surface area assessment of the murine intestinal tract as a prerequisite for oral dose translation from mouse to man. Lab Anim. 2010; 44 (3): 176-183. Available from: doi: 10.1258/la.2009.009112 Available from: https://doi.org/10.1258/la.2009.009112 .

37. Python Software Foundation. Python Language Reference, version 3.7. Available at http://www.python.org