DEGREE PROJECT IN MATHEMATICS,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2020

Calibration of Breast Cancer Natural History Models Using Approximate Bayesian Computation

OSCAR BERGQVIST

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ENGINEERING SCIENCES

Calibration of Breast Cancer Natural History Models Using Approximate Bayesian Computation

OSCAR BERGQVIST

Degree Projects in Mathematical Statistics (30 ECTS credits)
Master*s Programme in Applied and Computational Mathematics
KTH Royal Institute of Technology year 2020
Supervisor at Karolinska Institutet, MEB: Keith Humphreys
Supervisor at KTH: Henrik Hult
Examiner at KTH: Henrik Hult

TRITA-SCI-GRU 2020:089
MAT-E 2020:051

Royal Institute of Technology
School of Engineering Sciences
KTH SCI
SE-100 44 Stockholm, Sweden
URL: www.kth.se/sci

Abstract

Natural history models for breast cancer describe the unobservable disease progression. These models can either be fitted using likelihood-based estimation to data on individual tumour characteristics, or calibrated to fit statistics at a population level. Likelihood-based inference using individual level data has the advantage of ensuring model parameter identifiability. However, the likelihood function can be computationally heavy to evaluate or even intractable.

In this thesis likelihood-free estimation using Approximate Bayesian Computation (ABC) will be explored. The main objective is to investigate whether ABC can be used to fit models to data collected in the presence of mammography screening [2]. As a background, a literature review of ABC is provided.

As a first step an ABC-MCMC algorithm is constructed for two simple models both describing populations in absence of mammography screening, but assuming different functional forms of tumour growth. The algorithm is evaluated for these models in a simulation study using synthetic data, and compared with results obtained using likelihood-based inference.

Later, it is investigated whether ABC can be used for the models in presence of screening [2]. The findings of this thesis indicate that ABC is not directly applicable to these models. However, by including a sub-model for tumour onset and assuming that all individuals in the population have the same screening attendance it was possible to develop an ABC-MCMC algorithm that carefully takes individual level data into consideration in the estimation procedure. Finally, the algorithm was tested in a simple simulation study using synthetic data.

Future research is still needed to evaluate the statistical properties of the algorithm (using extended simulation) and to test it on observational data where previous estimates are available for reference.

Sammanfattning

Kalibrering av natural history models för bröstcancer med approximate bayesian computation

Natural history models för bröstcancer är statistiska modeller som beskriver det dolda sjukdomsförloppet. Dessa modeller brukar antingen anpassas till data på individnivå med likelihood-baserade metoder, eller kalibreras mot statistik för hela populationen. Fördelen med att använda data på individnivå är att identifierbarhet hos modellparametrarna kan garanteras. För dessa modeller händer det dock att det är beräkningsintensivt eller rent utav omöjligt att evaluera likelihood-funktionen.

Huvudsyftet med denna uppsats är att utforska huruvida metoden *Approximate Bayesian Computation* (ABC), som används för skattning av statistiska modeller där likelihood-funktionen inte är tillgänglig, kan implementeras för en modell som beskriver bröstcancer hos individer som genomgår mammografiscreening [2]. Som en del av bakgrunden presenteras en sammanfattning av modern ABC-forskning.

Metoden består av två delar. I den första delen implementeras en ABC-MCMC algoritm för två enklare modeller. Båda dessa modeller beskriver tumörtillväxten hos individer som ej genomgår mammografiscreening, men modellerna antar olika typer av tumörtillväxt. Algoritmen testades i en simulationsstudie med syntetisk data genom att jämföra resultaten med motsvarande från likelihood-baserade metoder.

I den andra delen av metoden undersöks huruvida ABC är kompatibelt med modeller för bröstcancer hos individer som genomgår screening [2]. Genom att lägga till en modell för uppkomst av tumörer och göra det förenklande antagandet att alla individer i populationen genomgår screening vid samma ålder, kunde en ABC-MCMC algoritm utvecklas med hänsyn till data på individnivå. Algoritmen testades sedan i en simulationsstudie nyttjande syntetisk data.

Framtida studier behövs för att undersöka algoritmens statistiska egenskaper (genom upprepad simulering av flera dataset) och för att testa den mot observationell data där tidigare parameterskattningar finns tillgängliga.

Acknowledgements

First of all, I would like to sincerely thank my supervisor at Karolinska Institutet, Keith Humphreys, for his guidance and support throughout the project. Keith has always been taking his time for our meetings, from which I have learned very much about statistics and research in general.

I would also like to thank my supervisor at KTH, Henrik Hult, for his thesis supervision and feedback.

Moreover, I would like to express my gratitude to Gabriel Isheden, Rickard Strandberg and Andreas Karlsson at Karolinska Institutet for explaining their highly interesting research, which has been of great relevance for this thesis.

Lastly, I would like to thank my dear sister Linnéa Bergqvist for her great support, and for taking her time to read through my thesis.

Stockholm
May 19, 2020
Oscar Bergqvist

Contents

1

Introduction

Breast cancer is the most common form of cancer for women, both in Sweden [14] and globally [25]. The vast majority of women diagnosed with breast cancer are postmenopausal [2]. Mammography screening aims to detect breast cancer tumours in an early stage using a low dose of X-rays [1]. In screening programs, women of certain ages are invited on a regular basis for screening. In Stockholm county all women between ages 40 and 74 are invited to mammography screening every second year [24].

Breast cancer is typically described using three stages: The *local stage* where a tumour has grown locally in the breast, the *regional stage*, where the cancer has spread to lymph nodes and the *distant stage*, where the cancer has spread to distant organs in the body [1]. Survival decreases for later stages, where the tumour is more difficult to treat [1]. In this thesis, we only consider growth of the primary tumour and its detection.

Natural history models are used to describe the disease progression of breast cancer. Since it is not possible to observe the disease process clinically these models are important to understand onset, growth and spread of breast cancer tumours [27]. They are also commonly used for evaluating the effectiveness of screening programs [36].

There are both discrete state models and continuous models. The later can be written as random effects models and can include components such as tumour onset, growth, spread and mortality. Each of these components can in turn be dependent on several parameters that need to be estimated. Depending on the purpose and type of model the estimation procedure may differ.

Random effects models describe the disease process conditional on individual level data, such as screening history. The individual disease history cannot be observed clinically; instead it needs to be inferred statistically, using data collected at tumour detection and the screening history of the individual.

In microsimulation models (MSM), which can be either discrete or continuous, a large number of individual life histories are simulated and aggregated into population level statistics such as breast cancer incidence and mortality [32].

Calibration of the microsimulation models against observed population statistics can be done using both frequentist and Bayesian methods [18]. However, due to the calibration targets being aggregated from individual disease histories it can be difficult to compute the likelihood functions of the targets [32]. In that case methods for likelihood-free estimation are needed.

Approximate Bayesian Computation (ABC) is one of the most popular methods for likelihood-free inference. ABC is a method in Bayesian statistics that approximates the posterior distribution of the model parameters, by comparing the observed data with data simulated using parameters drawn from the prior distribution. For the last two decades the ABC algorithm has become popular as a way to perform likelihood-free parameter estimation in a number of different research fields, in particular in population genetics [5]. The method has also been used for calibration of microsimulation models, as in [18] and [33].

The disadvantage of using microsimulation models calibrated against population level statistics, is that there is no guarantee that the underlying parameters of the natural history models are identifiable.

In random effects models, estimated using individual level data, identifiability of the parameters can be ensured. However, even in these models the estimation procedure may rely on computationally complex summations over possible disease histories [2]. In some cases, the time consuming task of deriving analytical expressions can reduce computational burden [15]. However, each time the model is extended to include more components a new expression for the likelihood function needs to be derived. This makes it interesting to investigate whether ABC can be used as a computationally efficient and scalable method for fitting random effects models to individual level data.

Abrahamsson and Humphreys [2] developed a continuous breast cancer natural history model with sub-models for tumour growth, detection, and screening sensitivity in presence of screening. In the paper by Abrahamsson and Humphreys [2] the model is fitted to individual level data using maximum likelihood estimation. Due to the computational complexity of the likelihood function, convergence took several days even on high performance computers. Standard error estimation therefore needed to be carried out using approximate methods [2]. Isheden and Humphreys later [16] reduced computational complexity by deriving some quantities in the likelihood analytically.

The model assumes exponential growth. With other functional forms of tumour growth, it is unknown if there is a tractable analytic expression for the likelihood function. Also, the models have been extended to include spread to lymph nodes [16] and will most likely be extended even further in the future. It is therefore of interest to explore whether likelihood-free estimation using ABC is potentially of use for these types of models. If ABC were useful it would be interesting to develop a general procedure in which the user could explore a range of growth

functions. This could also form a basis for estimating future extensions of the model, without each time having to derive the likelihood function.

One of the main objectives of this thesis is to investigate whether ABC can be used to fit the breast cancer natural history models developed by Abrahamsson and Humphreys [2]. Here, as a first step, an ABC algorithm is constructed and evaluated using a very simple model (Plevritis et al. [27]) for data on tumour sizes among patients diagnosed in a population without mammography screening. ABC represents a research field in its own respect. A large part of the background is therefore given to providing a review of state-of-the-art ABC literature.

The outline of the thesis is as follows: In the background the natural history models are described (Section 2.1) after which the ABC literature review is presented (Section 2.2). The method is divided into two parts. In the first part (Chapter 3), ABC is used to estimate the growth models in absence of screening by Plevritis et al. [27]. In the main method part (Chapter 4) it is investigated whether ABC can be used for the natural history models in presence of screening by Abrahamsson and Humphreys [2]. Please note that no separate results section is included in this thesis. Instead the results are presented in the last sections of each method chapter. In the discussion (Chapter 5) the performance of the algorithms is discussed, as well as possible improvements and future prospects. Finally, the most important findings of the thesis are presented in the conclusions (Chapter 6).

2

Background

2.1 Random effects models for studying the natural history of breast cancer

The natural history of breast cancer can be modelled in various ways. Historically, multi-state Markov models have been widely used to model the progression of breast cancer tumours [1]. In these models it is assumed that the tumour passes through a number of discrete states, such as non-detectable cancer, preclinical cancer and clinical cancer [12]. The categories can be refined further by differentiating between tumours of different sizes [36]. While multi-state Markov models are simple, it is more realistic to model the tumour growth as continuous. In this thesis continuous tumour growth models will be considered. Specifically we will look at the models for exponential tumour growth in absence of screening by Plevritis et al. [27] (Section 2.1.1) and the models for exponential tumour growth in presence of screening by Abrahamsson and Humphreys [2] (Section 2.1.2). Additionally, models assuming logistic tumour growth will be considered (Section 2.1.4).

I would like to suggest the reader to pay special attention to the likelihood functions: For the exponential tumour growth model in absence of screening the likelihood is derived analytically (2.5). For the exponential tumour growth models in presence of screening the likelihood can be numerically approximated (2.14). For logistic tumour growth, closed form expressions have neither been derived for the models in the absence or presence of screening nor have numerical approximations been made. In Chapter 3 we will suggest that there is no analytic expression for the likelihood of the logistic tumour growth model in absence of screening, but that it can be numerically approximated.

In particular, the most important information in this section is:

- The exponential model assumptions and the likelihood function of the model in absence of screening.
- The likelihood function of the model in presence of screening.

- The logistic growth model assumptions.

Details are presented for completeness but can be omitted without loss of information.

2.1.1 Exponential tumour growth in absence of screening

Plevritis et al. [27] presented a model for studying continuous tumour growth using breast cancer epidemiological data, which is based on three assumptions: The first assumption is that a tumour grows exponentially from volume v_{cell} with inverse growth rate R. The volume V at time t is described by

$$V(t) = v_{cell} \exp(t/R). \tag{2.1}$$

The second assumption is that the inverse growth rate is a gamma distributed random variable with shape τ_1, rate τ_2 and density

$$f_R(r) = \frac{\tau_2^{\tau_1}}{\Gamma(\tau_1)} r^{\tau_1 - 1} \exp(-\tau_2 r). \tag{2.2}$$

The third assumption is that T_{sd}, the time to symptomatic detection from tumour onset, has hazard function $\eta V(t)$. That is, for a tumour that has not yet been detected at time t, the probability of detection in the interval $[t, t + dt]$ is $\eta V(t)dt$ up to an error of order $o(dt)$:

$$\mathbb{P}(T_{sd} \in [t, t + dt] \mid T_{sd} > t) = \eta V(t)dt + o(dt). \tag{2.3}$$

Using these assumptions Plevritis et al. [27] derived the distribution of tumour volume at symptomatic detection V_{sd} conditional on inverse growth rate

$$f_{V_{sd}|R=r}(v) = \eta r \exp(-\eta r(v - v_{cell})) \quad v > v_{cell} \tag{2.4}$$

and the marginal distribution of V_{sd}

$$f_{V_{sd}}(v) = \tau_1 \tau_2^{\tau_1} \eta [\tau_2 + \eta(v - v_{cell})]^{-(\tau_1+1)} \quad v > v_{cell}. \tag{2.5}$$

Abrahamsson and Humphreys [2] derived the distribution of inverse growth rate conditional on volume at symptomatic detection

$$\begin{aligned} f_{R|V_{sd}=v}(r) = &\frac{\tau_2 + \eta(v - v_{cell})}{\Gamma(\tau_1 + 1)} [r(\tau_2 + \eta(v - v_{cell}))]^{\tau_1} \\ &\cdot \exp(-r[\tau_2 + \eta(v - v_{cell})]) \qquad\qquad r \geq 0. \end{aligned} \tag{2.6}$$

In applications of these models it is typical to assume that the tumours are spherical [2][27]. Tumour diameter $d(t)$ can then be converted to tumour volume $V(t)$ using

$$V(t) = \frac{1}{6}\pi[d(t)]^3. \tag{2.7}$$

In the analysis presented by Plevritis et al. [27], tumour sizes are recorded into $B = 7$ bins:

$$\{I_i\}_{i=1}^{B} \tag{2.8}$$

where $I_i = [d_i, d_{i+1})$ for diameter or equivalently $I_i = [v_i, v_{i+1})$ for volume. Assuming independence between the tumour sizes at symptomatic detection of different individuals, the distribution of the number of detected tumours in each bin is given by the multinomial distribution. Thus the likelihood function is proportional to

$$\mathcal{L}_N(n_1, \ldots, n_B | \ \vec{\theta}) = \prod_{i=1}^{B} p_i^{n_i}. \tag{2.9}$$

Here $\vec{\theta} = (\eta, \tau_1, \tau_2)$ are the model parameters, n_i is the number of detected tumours in I_i across the population and p_i is the probability of the volume at symptomatic detection belonging to I_i:

$$p_i = \mathbb{P}(V_{sd} \in I_i) = \int_{v_i}^{v_{i+1}} f_{V_{sd}}(v) dv. \tag{2.10}$$

The analytic expression for the likelihood is presented in the appendix of [27]. Using this with breast cancer incidence data in absence of screening, maximum likelihood estimates for the model parameters (η, τ_1, τ_2) can be obtained [27].

2.1.2 Modelling exponential tumour growth in presence of screening

Abrahamsson and Humphreys [2] proposed a continuous parametric tumour growth model including information on individual mammography screening histories. The model describes the distribution of tumour size at detection given detection mode (screen detection or symptomatic detection) and screening history (times of previous negative screens). Like in Plevritis et al. [27] the model assumes exponential tumour growth (2.1), gamma distributed inverse growth rate (2.2), and a symptomatic detection hazard proportional to the tumour volume (2.3). It also includes a sub-model for screening sensitivity $S(d)$ as a function of the tumour diameter $d(t)$:

$$S(d) = \frac{\exp(\beta_1 + \beta_2 d)}{1 + \exp(\beta_1 + \beta_2 d)}. \tag{2.11}$$

Note that the tumour diameter is unobservable before detection. The screening sensitivity model can be extended by adding more covariates. In Abrahamsson and Humphreys [2] one other covariate is included. In this report only tumour diameter will be considered for simplicity.

Before describing the details of the model, we will present the general form of the likelihood function. Consider breast cancer incidence data $\mathcal{D} = \{\mathcal{D}^j\}_{j=1}^{N}$

on N individuals. Let $\mathcal{H}^j$ denote the screening history of individual j; that is, the times for previous screens and their respective outcomes (positive or negative). Let M_d^j denote the detection mode of individual j (screen detection or symptomatic detection). Finally, let V_d^j denote the volume at detection for individual j. Now the data for individual j can be written as

$$\mathcal{D}^j = (\mathcal{H}^j, M_d^j, V_d^j). \tag{2.12}$$

In Abrahamsson and Humphreys [2] the conditional probability of individual j having a tumour with volume in I_i at detection, given detection mode M_d^j and screening history $\mathcal{H}^j$, is modelled using a parametric model with parameters $\vec{\theta}$:

$$p_{ij}(\vec{\theta}) = \mathbb{P}_{\vec{\theta}}(V_d^j \in I_i | \mathcal{H}^j, M_d^j). \tag{2.13}$$

Let $x_{ij} = 1$ if the tumour volume at detection is in I_i for individual j and $x_{ij} = 0$ otherwise. Assuming conditional independence between the tumour sizes of different individuals the likelihood is given by

$$\mathcal{L}_N(\vec{\theta}) = \prod_{j=1}^{N} \prod_{i=1}^{B} p_{ij}^{x_{ij}} = \prod_{j=1}^{N} \prod_{i=1}^{B} \mathbb{P}_{\vec{\theta}}(V_d^j \in I_i | \mathcal{H}^j, M_d^j)^{x_{ij}} \tag{2.14}$$

where B is the number of bins for the tumour volume. In Abrahamsson and Humphreys [2] the model was fitted using maximum likelihood estimation. Their evaluation of the likelihood relies on computationally heavy numerical approximations since possible tumour growth trajectories needs to be summarised over [2]. Later Isheden and Humphreys [15] replaced the approximations by analytical expressions, allowing estimation of the model parameters in feasible time using the full data.

In the reminder of this section we will consider the expressions for p_{ij}. These are derived separately for screen detection and symptomatic detection in [2][15]. Notation from Isheden and Humphreys [15] will be used. For readability the j index will be dropped; but keep in mind that different individuals have different histories of screening, tumour growth, and detection.

The model has three fundamental assumptions about the population:

$$\begin{cases} \text{A1:} & \text{The rate of births in the population is constant across calendar time.} \\ \text{A2:} & \text{The distribution of age at tumour onset is constant across calendar time.} \\ \text{A3:} & \text{The distribution of time to symptomatic detection is constant across calendar time.} \end{cases} \tag{2.15}$$

A population following these assumptions is denoted a *stable disease population* [15]. An individual is seen as belonging to any of the following three disease

states at a particular point in time.

$$\begin{aligned} &\mathcal{P}_{Before} \text{ – disease free state (prior to tumour onset),} \\ &\mathcal{P}_{Tumour} \text{ – breast cancer state (as of yet undetected),} \\ &\mathcal{P}_{After} \text{ – post symptomatic detection state.} \end{aligned} \tag{2.16}$$

In Isheden and Humphreys [15] the following notation is used:

$$\begin{aligned} &C(s) \in [0, \infty) \text{ – tumour size at time point } s. \\ &A(s) = \begin{cases} 1, & \text{disease state is } \mathcal{P}_{Tumour} \text{ at time point } s, \\ 0, & \text{otherwise.} \end{cases} \\ &B(s) = \begin{cases} 1, & \text{tumour screen detected at time point } s, \\ 0, & \text{otherwise.} \end{cases} \\ &D(s) = \begin{cases} 1, & \text{tumour symptomatically detected at time point } s, \\ 0, & \text{otherwise.} \end{cases} \\ &\{s_{-i}\}_{i=1}^{K} = \text{times for the } K \text{ screens previous to time } s. \\ &\mathbf{B^c} \quad = \text{event that all screens previous to } s \text{ are negative:} \\ &\qquad (B(s_{-1}), \ldots, B(s_{-K})) = (0, \ldots, 0). \end{aligned} \tag{2.17}$$

Screen detection

The probability that a screen-detected tumour belongs to size category I_i given an individual screening history is

$$p_i = \mathbb{P}(C(s) \in I_i \mid A(s) = 1, B(s) = 1, \mathbf{B^c}). \tag{2.18}$$

Abrahamsson and Humphreys [2] derived p_i as

$$\begin{cases} p_i \propto P_A(i) \cdot P_B(i) \cdot \sum_{j \leq i} P_C(i,j) \cdot P_D(i,j) & K \geq 1 \\ p_i \propto P_A(i) \cdot P_B(i) & K = 0 \end{cases} \tag{2.19}$$

where $K \geq 1$ if there are one or more previous negative screens and $K = 0$ if there are no previous screens. P_A is the probability of screen detection at time s given that the tumour is in size interval I_i. This is approximated using the screening sensitivity evaluated at the middle point of I_i, denoted as $d_{i+\frac{1}{2}}$:

$$P_A(i) = \mathbb{P}(B(s) = 1 \mid C(s) \in I_i) \approx S(d_{i+\frac{1}{2}}). \tag{2.20}$$

P_B is the probability of a tumour belonging to size interval I_i before symptomatic detection:

$$P_B(i) = \mathbb{P}(C(s) \in I_i \mid A(s) = 1). \tag{2.21}$$

In Abrahamsson and Humphreys [2] P_B is approximated by summation over all possible future times of detection t and all tumour size intervals g greater than I_i:

$$\sum_{s \leq t} \sum_{i \leq g} \mathbb{P}(C(s) \in I_i | A(s) = 1, D(t) = 1, C(t) \in I_g) \mathbb{P}(C(t) \in I_g | D(t) = 1).$$

The summation is computationally heavy since there are many possible future times of detection. This resulted in Abrahamsson and Humphreys [2] being forced to exclude parts of the screening data when estimating model parameters [2]. Isheden and Humphreys [15] later assessed this problem by deriving the following analytical expression for P_B:

$$P_B(i) \propto \log\left(\frac{v_{i+1}}{v_i}\right) \frac{f_{V_{sd}}(v_m)}{\eta}. \tag{2.22}$$

Here $f_{V_{sd}}$ is the density given in (2.5), v_i and v_{i+1} the upper and lower bounds of I_i and $v_m \in I_i$.

P_C is the probability that all previous screens are negative, given that the tumour size at screen detection is in I_i and the tumour size at the most recent negative screen was in I_j:

$$P_C(i, j) = \mathbb{P}(\mathbf{B^c} | C(s) \in I_i, C(s_{-1}) \in I_j). \tag{2.23}$$

The probability of a screen being negative is $1 - S(d)$ and thus

$$P_C(i, j) \approx \prod_{k=1}^{K} \left[1 - S(d(s_{-k}))\right]. \tag{2.24}$$

The tumour diameter at detection $d(s)$ and the tumour diameter at the most recent screen $d(s_{-1})$ are approximated as the middle points of I_i and I_j. Using these together with the exponential growth assumption (2.1) the tumour sizes at previous screens $d(s_{-1}), \ldots, d(s_{-K})$ are obtained.

P_D is the probability that a tumour is in size interval I_j at the most recent screen, given that it is in size interval I_i at detection:

$$P_D(i, j) = \mathbb{P}(C(s_{-1}) \in I_j | C(s) \in I_i) \tag{2.25}$$

Isheden and Humphreys [15] calculated P_D using the integral

$$P_D \approx \int_{r=0}^{\infty} h(I_j, d, r) f_{R|V_{sd}=c}(r) dr. \tag{2.26}$$

Here c and d are respectively the volume and diameter corresponding to the middle point of I_i. The function $h(I_j, d, r)$ takes value one if a tumour with inverse growth rate r and volume c at time s has a diameter in I_j at time s_{-1} and zero otherwise. The density $f_{R|V_{sd}=c}(r)$ is given in (2.6).

Symptomatic detection

For the probability of symptomatic tumour detection at time s, conditional on the individual screening history, Abrahamsson and Humphreys [2] derived the following expression:

$$\begin{cases} p_i \propto P_E(i) \cdot \sum_{j \leq i} P_F(i,j) \cdot P_G(i,j) & K \geq 1 \\ p_i \propto P_E(i) & K = 0. \end{cases} \tag{2.27}$$

P_E is the probability for a symptomatically detected tumour to be in volume bin I_i (bounded by v_i and v_{i+1}):

$$P_E(i) = \mathbb{P}(V_{sd} \in I_i) = \int_{v_i}^{v_{i+1}} f_{V_{sd}}(v)dv. \tag{2.28}$$

The distribution of tumour volume at symptomatic detection, $f_{V_{sd}}(v)$, is presented in (2.5).

P_F is the probability that all screens before symptomatic detection at time s are negative, given that the tumour size at time s is in I_i and the tumour size at time s_{-1} is in I_j:

$$P_F(i,j) = \mathbb{P}(\mathbf{B^c}|C(s) \in I_i, C(s_{-1}) \in I_j, D(s) = 1). \tag{2.29}$$

In [15] this probability is calculated in the same way as P_C (2.23).

P_G is the probability that a tumour is in size interval I_j at the most recent screen, given that it is in size interval I_i at symptomatic detection:

$$P_G(i,j) = \mathbb{P}(C(s_{-1}) \in I_j|C(s) \in I_i, D(s) = 1). \tag{2.30}$$

In [15] this probability is calculated in a similar way to P_D (2.25).

2.1.3 Tumour onset

When we explore ABC for fitting growth models for data collected in the presence of screening (Chapter 4) we will need a model for tumour onset. Strandberg and Humphreys [35] introduced the Moolgavkar-Venson-Knudson (MVK) model of carcinogenesis [22] to model age at tumour onset using cohort data. The model combines four Poisson processes corresponding to cell division rate, initiation, cell death and malignant transformation. Using an identifiable reparametrisation under diagnostic data [35] the cumulative density function of the age at tumour onset A_0 can be written as:

$$F_{A_0}(t) = 1 - \left[\frac{(B-A)\exp(Bt)}{B\exp((B-A)t) - A}\right]^{\delta} \tag{2.31}$$

The model parameters are A, B and δ. For this thesis we will use the values $A = -0.075$, $B = 1.1 \cdot 10^{-4}$ and $\delta = 0.5$ taken from [35].

2.1.4 Logistic tumour growth

Instead of exponential tumour growth and gamma distributed inverse growth rates we could consider logistic tumour growth

$$V(t) = \frac{v_{max}}{\left[1 + \left(\left(v_{max}/v_{cell}\right)^{0.25} - 1\right)\exp(-0.25Rt)\right]^4} \tag{2.32}$$

with log-normal growth rate R

$$R = \exp(\mu + \sigma Z), \quad Z \sim \mathcal{N}(0,1). \tag{2.33}$$

This form of growth is taken from [41] and has previously been suggested by clinical studies [34].

In Section 3.3 we will estimate a model using logistic growth, log-normal growth rate and with the symptomatic detection hazard (2.3) from Plevritis et al. [27]. For this model the likelihood is unknown, but we will show how to numerically approximate it in Appendix B.

Weedon-Fekjær et al. [41][42] developed a growth model including individual screening histories, similar to the one developed by Abrahamsson and Humphreys [2]. The model assumes logistic growth and log-normal growth rate.

In order to evaluate the likelihood for tumour sizes at detection in presence of screening, Weedon-Fekjær et al. [41] rely on the unrealistic assumption that tumour growth rate is independent on tumour size at detection [2]. This problem was assessed by Abrahamsson and Humphreys [2] by using the model assumptions in Plevritis et al. [27]. That is, exponential tumour growth (2.1), gamma distributed inverse growth rate (2.2) and a time-to-symptomatic-detection-hazard proportional to the tumour volume (2.3).

In theory one could use the same assumptions as in Abrahamsson and Humphreys [2] but with logistic tumour growth instead of exponential growth. This could be seen as a correction of the models by Weedon-Fekjær et al [41], however it is not straight forward how to derive or approximate the resulting likelihood. If the likelihood function cannot be derived, an alternative is to resort to likelihood free inference. One such method: Approximate Bayesian Computation (ABC) will be discussed in the following section.

2.2 Approximate Bayesian Computation (ABC)

In this section the theoretical background of Approximate Bayesian Computation (ABC) is presented. ABC is an active field of research and has been used for many different types of applications in recent years. Popular fields of application are population genetics [5], ecology [3] and phylogeography [37]. In Section 2.2.1 traditional frequentist and Bayesian estimation methods are presented. In Section 2.2.2 the basic ABC algorithm is explained. Finally, in Sections 2.2.3 - 2.2.5, improvements to standard ABC which have recently been described in the literature are discussed.

2.2.1 Frequentist and Bayesian estimation based on the likelihood function

Consider data $\mathbf{y} \in \mathcal{D}$ described by a parametric model $\mathcal{P}_{\vec{\theta}} = \{f_{\vec{\theta}}(\mathbf{y}) : \vec{\theta} \in \Theta\}$. The likelihood function is a function of the model parameters $\vec{\theta} = (\theta_1, \ldots, \theta_d)$ and is defined as

$$\mathcal{L}(\vec{\theta}) = f_{\vec{\theta}}(\mathbf{y}). \tag{2.34}$$

One of the most well known techniques for estimation of model parameters in frequentist inference is *Maximum Likelihood Estimation* (MLE). In MLE the parameters that maximise the probability of the observed data are chosen as estimates, and denoted as the *Maximum Likelihood estimates* (ML estimates):

$$\hat{\vec{\theta}}_{MLE} = \underset{\vec{\theta} \in \Theta}{\operatorname{argmax}} \{\mathcal{L}(\vec{\theta})\}. \tag{2.35}$$

In Bayesian inference, the model parameters $\vec{\theta} \in \Theta$ are seen as random variables and are assigned a prior distribution:

$$\vec{\theta} \sim \pi(\vec{\theta}). \tag{2.36}$$

Inference is now made using the posterior distribution of the parameters given the observed data $\mathbf{y}$. The posterior distribution is obtained using Bayes theorem:

$$p(\vec{\theta}|\mathbf{y}) = \frac{\mathcal{L}(\mathbf{y}|\vec{\theta})\pi(\vec{\theta})}{p(\mathbf{y})} \tag{2.37}$$

where $\mathcal{L}(\mathbf{y}|\vec{\theta})$ is the likelihood function and

$$p(\mathbf{y}) = \int_{\Theta} \mathcal{L}(\mathbf{y}|\vec{\theta})\pi(\vec{\theta})d\vec{\theta}. \tag{2.38}$$

The mean of the posterior is obtained as

$$\mathbb{E}[\vec{\theta}|\mathbf{y}] = \frac{1}{p(\mathbf{y})} \int_{\Theta} \vec{\theta}\mathcal{L}(\mathbf{y}|\vec{\theta})\pi(\vec{\theta})d\vec{\theta}. \tag{2.39}$$

If the sample space Θ is high dimensional and the likelihood function complicated, the integrals (2.38) and (2.39) can be difficult to compute. To overcome this problem, Monte Carlo methods are commonly used for estimation of posterior distributions in Bayesian statistics.

Markov chain Monte Carlo (MCMC) methods, such as the Metropolis Hastings (MH) sampler [10], are commonly used for sampling from complex distributions. In Metropolis Hastings the distribution only needs to be known up to a normalising constant, which means that the marginal density of the data (2.38) does not need to be computed in order to sample from the posterior distribution of the model parameters [30]. The MH-sampler is presented in Algorithm 1.

Algorithm 1: Metropolis Hastings MCMC

Initialize $\vec{\theta}_0$
for $i \in \{1, \ldots, M\}$ **do**
  Sample $\vec{\theta}'$ from transition probability $q(\vec{\theta}'|\vec{\theta}_{i-1})$
  Calculate acceptance ratio $\alpha = \frac{\mathcal{L}(\mathbf{y}|\vec{\theta}')\pi(\vec{\theta}')q(\vec{\theta}_{i-1}|\vec{\theta}')}{\mathcal{L}(\mathbf{y}|\vec{\theta}_{i-1})\pi(\vec{\theta}_{i-1})q(\vec{\theta}'|\vec{\theta}_{i-1})}$
  Sample u from uniform distribution $U[0, 1]$
  if $u \leq \alpha$ **then**
    $\vec{\theta}_i = \vec{\theta}'$
  else
    $\vec{\theta}_i = \vec{\theta}_{i-1}$

With enough iterations the method will converge to produce samples from the posterior distribution $p(\vec{\theta}|\mathbf{y})$, for any starting value $\vec{\theta}_0$ [31]. The transition density (transition kernel) $q(\vec{\theta}'|\vec{\theta}_{i-1})$ is chosen to optimise the convergence [30].

2.2.2 Likelihood-free inference using ABC

Approximate Bayesian Computation (ABC) is a family of methods for likelihood-free estimation, used when the likelihood function in intractable or computationally heavy to evaluate. ABC belongs to the Bayesian paradigm; it generates samples from the posterior distribution of the model parameters (2.37) using a prior (2.36). In ABC a *data generating model* is needed:

$$\mathbf{y} = \mathcal{G}(\vec{\theta}). \tag{2.40}$$

The data generating model is assumed to generate samples from the unknown distribution

$$\mathbf{y} \sim p(\mathbf{y}|\vec{\theta}) \tag{2.41}$$

denoted as the *implicit likelihood*.

ABC is approximate in its nature, but is based on an exact algorithm that works as follows: A proposed parameter vector $\vec{\theta}$ is sampled from the prior $\pi(\vec{\theta})$ and used in the data generating model $\mathcal{G}(\vec{\theta})$ to obtain data $\mathbf{y}$. If the observed and generated data are equal, the proposed parameters $\vec{\theta}$ are accepted:

Algorithm 2: Exact likelihood-free method

Input: Observed data $\mathbf{y}_{obs} \in \mathcal{Y}$
Output: Sample $\vec{\theta} \sim p(\vec{\theta}|\mathbf{y}_{obs})$

```
repeat
    Sample model parameters θ⃗ from prior π(θ⃗)
    Generate data y ~ p(y|θ⃗) using model G(θ⃗)
    if y_obs = y then
        return θ⃗
```

The joint distribution of accepted data and parameters $(\mathbf{y}^*, \vec{\theta}^*)$ is

$$p_{\mathbf{y}^*,\vec{\theta}^*}(\mathbf{y}, \vec{\theta}) \propto p_{\mathbf{y}|\vec{\theta}}(\mathbf{y})\pi(\vec{\theta})\mathbb{1}_{\mathbf{y}_{obs}}(\mathbf{y}). \tag{2.42}$$

When marginalising we get

$$\begin{aligned} p_{\vec{\theta}^*}(\vec{\theta}) &\propto \int_{\mathbf{s}\in\mathcal{Y}} p_{\mathbf{y}|\vec{\theta}=\vec{\theta}}(\mathbf{s})\pi(\vec{\theta})\mathbb{1}_{\mathbf{y}_{obs}}(\mathbf{s})d\mathbf{s} = p_{\mathbf{y}|\vec{\theta}}(\mathbf{y}_{obs})\pi(\vec{\theta}) \\ &\propto p_{\vec{\theta}|\mathbf{y}=\mathbf{y}_{obs}}(\vec{\theta}). \end{aligned} \tag{2.43}$$

From this we can see that Algorithm 2 generates samples exactly from the posterior distribution of the model parameters.

In practice the data is often continuous or high dimensional, in such situations the probability of $\mathbf{y} = \mathbf{y}_{obs}$ is zero or close to zero. In ABC an approximation is introduced by choosing a *tolerance* ϵ and accepting samples for which the distance between the sample and the real data is less than the tolerance with

respect to some distance metric δ:

Algorithm 3: Basic ABC

Input: Observed data $\mathbf{y}_{obs} \in \mathcal{D}$
Output: Sample $\vec{\theta} \sim p_{ABC}(\vec{\theta}|\mathbf{y}_{obs})$

```
repeat
    Sample model parameters θ⃗ from prior π(θ⃗)
    Generate data y ~ p(y|θ⃗) using model G(θ⃗)
    if δ(y_obs, y) ≤ ε then
        return θ⃗
```

Algorithm 3 can be generalised to accept a sample with probability proportional to some kernel $K_\epsilon(\delta(\mathbf{y}, \mathbf{y}_{obs}))$:

Algorithm 4: Basic ABC using kernel

Input: Observed data $\mathbf{y}_{obs} \in \mathcal{D}$
Output: Sample $\vec{\theta} \sim p_{ABC}(\vec{\theta}|\mathbf{y}_{obs})$

```
repeat
    Sample model parameters θ⃗ from prior π(θ⃗)
    Generate data y ~ p(y|θ⃗) using model G(θ⃗)
    with probability ∝ K_ε(δ(y, y_obs))
        return θ⃗
```

Note that Algorithm 3 is a special case of Algorithm 4 using a uniform kernel $\mathbb{1}\{\delta(\mathbf{y}, \mathbf{y}_{obs}) \leq \epsilon\}$. The algorithm produces samples from the posterior distribution

$$p_{ABC}(\vec{\theta}|\mathbf{y}_{obs}) \propto \int_{\mathcal{Y}} p(\mathbf{y}|\vec{\theta})\pi(\vec{\theta})K_\epsilon(\delta(\mathbf{y}_{obs}, \mathbf{y}))d\mathbf{y} \tag{2.44}$$

which can be seen as Bayesian inference using a kernel density approximation of the likelihood function [13]. The basic assumption of ABC is

$$p_{ABC}(\vec{\theta}|\mathbf{y}_{obs}) \approx p(\vec{\theta}|\mathbf{y}_{obs}) \tag{2.45}$$

which should be valid if the tolerance ϵ is small enough. Other examples of kernels than the indicator function is the Epanechnikov kernel and the Gaussian kernel [4].

Wilkinson [44] proved that Algorithm 4 draws from the posterior for the model

$$\mathbf{y}_{obs} = \mathcal{G}'(\vec{\theta}) = \mathcal{G}(\vec{\theta}) + \xi, \quad \xi \sim K_\epsilon(\cdot) \tag{2.46}$$

where $\mathcal{G}$ and ξ are independent. We can see that this corresponds to exact sampling from the model $\mathcal{G}'(\vec{\theta})$, which can be viewed as the original model $\mathcal{G}$

with an added model error ξ. By this result it is appropriate to choose the tolerance ϵ and kernel to fit known model or measurement errors [44].

ABC suffers from the curse of dimensionality [26]. This is because the probability that the distance between the generated and observed data is smaller than the tolerance decreases with increasing dimensionality. Therefore, high dimensional data may result in very low acceptance rates for a fixed tolerance. A low acceptance rate will lead to the algorithm producing fewer samples for a fixed run time. There are two main ways to assess this problem that will be discussed in the following two sections.

2.2.3 Summary statistics for ABC

One way to improve the performance of ABC is to reduce the dimensionality of the data using summary statistics in place of the full data:

$$\mathbf{s} = S(\mathbf{y}), \quad S : \mathcal{Y} \longrightarrow \mathcal{S}. \tag{2.47}$$

Here $\dim(\mathcal{S}) < \dim(\mathcal{Y})$. If using (Bayesian) sufficient statistics we have

$$p(\vec{\theta}|\mathbf{y}) = p(\vec{\theta}|\mathbf{s}) \tag{2.48}$$

which means that no bias is introduced when using sufficient statistics in place of the full data. In practice it can be hard to find sufficient statistics for models outside of the exponential family [4]. Therefore, it is common to use non-sufficient statistics chosen to be as informative as possible. The following algorithm is identical to Algorithm 4, but with summary statistics in place of the full data.

Algorithm 5: Basic ABC using summary statistics

Input: Observed data $\mathbf{y}_{obs} \in \mathcal{D}$
Output: Sample $\vec{\theta} \sim p_{ABC}(\vec{\theta}|\mathbf{y}_{obs})$

```
Calculate s_obs = S(y_obs)
repeat
    Sample model parameters θ⃗ from prior π(θ⃗)
    Generate data y ~ p(y|θ⃗) using model G(θ⃗)
    Calculate s = S(y)
    with probability ∝ K_ε(δ(s, s_obs))
        return θ⃗
```

Good summary statistics usually take advantage of symmetries in the data [4]. Commonly an initial set of summary statistics is chosen to be informative for the problem in question [8]. After this, if the dimension of the summary statistics is much higher than the number of model parameters, some method is used to reduce the dimensionality [8]. General ways of choosing informative

summary statistics is a field of active research and many different methods has been developed. Most of these methods can be categorised into methods that project summary statistics into a lower dimensional space and methods that choose a subset of summary statistics in an optimal way [4].

Joyce and Majoram [17] described a method for optimal subset selection, introducing the new concept of approximate sufficiency. Nunes and Balding [23] developed another approach that includes a minimisation of the posterior entropy for different subsets of summary statistics.

Wegmann et al. [43] proposed using partial least squares (PLS) to reduce dimensionality. The method works by projecting the high dimensional set of summary statistics to orthogonal components that explain the variability of the model parameters in an optimal way.

Consider the quadratic loss function of the true parameter values $\vec{\theta}_0$, estimates $\hat{\vec{\theta}}$ and a positive definite matrix A:

$$L(\vec{\theta}_0, \hat{\vec{\theta}}, A) = (\vec{\theta}_0 - \hat{\vec{\theta}})' A (\vec{\theta}_0 - \hat{\vec{\theta}}). \tag{2.49}$$

Fearnhead and Prangle [13] proved for the tolerance limit $\epsilon \to 0$ that if the summary statistics are defined by $S(\mathbf{y}_{obs}) = \mathbb{E}[\vec{\theta}|\mathbf{y}_{obs}]$, the quadratic loss function is minimised by

$$\hat{\vec{\theta}} = \mathbb{E}_{ABC}[\vec{\theta}|\mathbf{s}_{obs}] := \int_{\mathbf{\Theta}} \vec{\theta}\, p_{ABC}(\vec{\theta}|\mathbf{s}_{obs}) d\vec{\theta}. \tag{2.50}$$

This means that

$$S(\mathbf{y}_{obs}) = \mathbb{E}[\vec{\theta}|\mathbf{y}_{obs}] \tag{2.51}$$

is a good choice of summary statistics for producing mean estimates. The number of statistics is reduced to the number of parameters. Using the same number of statistics as the number of parameters is also suggested by Li and Fearnhead [19]. Since the posterior mean is the target of ABC and not known before carrying out data analysis the conditional expectation in (2.51) needs to be estimated. In [13] this was done using a pilot study to simulate parameters and corresponding data, and then using regression to model the conditional expectation. The regression model was then used to produce summary statistics when running ABC. In [13] both a linear regression model using nonlinear basis functions and the Lasso [38] were tested, with similar performance.

In a comparative review, Blum [8] concluded that the methods of Nunes and Balding [23] as well as Fearnhead and Prangle [13] performed well – with the latter having the advantage of being simpler to implement [4].

2.2.4 Regression adjustment for ABC

Another way to increase the acceptance rates for high dimensional data is to increase the tolerance ϵ. While this gives higher acceptance rates, it also introduces bias. Beaumont et al. [5] proposed to use local linear regression to model the effect on the model parameters $\vec{\theta}$ of the discrepancy between $\mathbf{s}$ and $\mathbf{s}_{obs}$. Using this approach, they could increase the acceptance rate considerably, without introducing too much bias [5]. Blum and Francois [9] introduced a non-linear regression adjustment model using a two-layer neural network.

2.2.5 ABC with Markov chain Monte Carlo (ABC-MCMC)

If the prior and posterior distributions of the model parameters θ are far away from each other, the basic ABC algorithm will give low acceptance rates [21]. Majoram et al. [21] developed an ABC method that is embedded in Metropolis Hastings MCMC. As in normal MH (Algorithm 1) the method explores the parameter space to find regions of high density, using a transition kernel. The ABC-MCMC algorithm is:

Algorithm 6: ABC-MCMC

Input: Observed data $\mathbf{y}_{obs} \in \mathcal{D}$
Output: M samples from $p_{ABC}(\vec{\theta}|\mathbf{y}_{obs})$

Initialise $\vec{\theta}_0$
for $i \in \{1, \ldots, M\}$ **do**
  Sample $\vec{\theta}'$ from transition kernel $q(\vec{\theta}'|\vec{\theta}_{i-1})$
  Generate $\mathbf{y} \sim p(\mathbf{y}|\vec{\theta}')$ from $\mathcal{G}(\vec{\theta}')$
  Calculate $\mathbf{s} = S(\mathbf{y})$
  if $\rho(\mathbf{s}, \mathbf{s}_{obs}) \leq \epsilon$ **then**
    Calculate acceptance ratio $\alpha = \frac{\pi(\vec{\theta}')q(\vec{\theta}_{i-1}|\vec{\theta}')}{\pi(\vec{\theta}_{i-1})q(\vec{\theta}'|\vec{\theta}_{i-1})}$
    Sample u from uniform distribution $U[0, 1]$
    if $u \leq \alpha$ **then**
      $\theta_i = \theta'$
    else
      $\theta_i = \theta_{i-1}$

Wegmann et al. [43] improved the ABC-MCMC algorithm by setting the tolerance to δ_ϵ where

$$\mathcal{P}(\delta \leq \delta_\epsilon) = \epsilon. \tag{2.52}$$

In this context the tolerance is the ϵ-quantile of δ, where ϵ is a probability directly corresponding to the acceptance rate. In order to find the distribution of

δ a pilot study can be performed, where a large range of δ-samples are simulated using the first few steps of ABC (Algorithm 5).

Ratmann et al. [29] used a tempering scheme, where the tolerance is reduced over a burn-in phase of the ABC-MCMC. This will give a good acceptance rate even when being in the tails of the posterior and could for example be used when the prior is non-informative [4].

One weakness of ABC-MCMC as opposed to the basic ABC is that it is not as easy to parallelise. As though several chains can be run in parallel, samples in an individual chain are not produced independently.

3

ABC for models in absence of screening

In this chapter an ABC algorithm for estimating the models in absence of screening will be constructed and evaluated. We will start by presenting the ABC algorithm (Section 3.1). Then we will evaluate its performance on the exponential tumour growth model (Section 3.2) and the logistic tumour growth model (Section 3.3) though simulation studies.

3.1 The ABC algorithm

An ABC-MCMC scheme inspired by [21] and [43] was used to estimate the model parameters. It is presented in Algorithm 7.

Algorithm 7: ABC-MCMC for the tumour growth models

Input: Observed data $\mathbf{y}_{obs}$
Output: M samples from $p_{ABC}(\vec{\theta}|\mathbf{y}_{obs})$

Initialise $\vec{\theta}_0$
for $i \in \{1, \dots, M\}$ **do**
  Sample $\vec{\theta}'$ from transition kernel $q(\vec{\theta}'|\vec{\theta}_{i-1})$ as in (3.1)
  Generate $\mathbf{y} \sim p(\mathbf{y}|\vec{\theta}')$ from $\mathcal{G}(\vec{\theta}')$
  Calculate $\mathbf{s} = S(\mathbf{y})$ as in (3.2)
  if $\rho(\mathbf{s}, \mathbf{s}_{obs}) \leq \delta_\epsilon$ **then**
    if $i \equiv 0 \mod 100$ **then**
      $\delta_\epsilon = 80\%$ quantile of last 100 accepted ρ
    Calculate acceptance ratio $\alpha = \frac{\pi(\vec{\theta}')q(\vec{\theta}_{i-1}|\vec{\theta}')}{\pi(\vec{\theta}_{i-1})q(\vec{\theta}'|\vec{\theta}_{i-1})}$
    Sample u from uniform distribution $U[0, 1]$
    if $u \leq \alpha$ **then**
      $\theta_i = \theta'$
    else
      $\theta_i = \theta_{i-1}$

For the tumour growth models in absence of screening, $\mathbf{y}$ is a vector of tumour volumes at symptomatic detection.

The prior for inference $\pi(\vec{\theta})$ is chosen as the flat improper prior, in order to include situations where very little is known about the parameters beforehand. This can also be seen as the most difficult scenario for the ABC algorithm since an informative prior generally improves convergence of the algorithm [4].

The transition kernel q is the symmetric Gaussian random walk kernel:

$$q(\vec{\theta}'|\vec{\theta}_{i-1}) = f_Z(\vec{\theta}'), \quad Z \sim \mathcal{N}(\vec{\theta}_{i-1},\ \sigma_q^2 I_2). \tag{3.1}$$

The summary statistics $\mathbf{s} = (s_1, \dots, s_{24})$ are chosen as the counts in each of 24 predefined bins for the tumour volume:

$$s_j = [S(\mathbf{y})]_j = \sum_{i=1}^{N} \mathbb{1}\{y_i \in I_j\}, \quad j = 1, \dots 24 \tag{3.2}$$

where I_j denotes tumour volume bin j. This choice of summary statistics is

based on Abrahamsson and Humphreys [2], where the multinomial distribution is used for MLE of the parameter values.

The distance metric ρ is the normalised Euclidean distance weighted by estimates of the variances of the summary statistics:

$$\rho(\mathbf{s}, \mathbf{s}_{obs}) = \frac{1}{24}\sqrt{\sum_{i=1}^{24} \frac{(s_i - s_{i,obs})^2}{\mathbb{V}[s_i]}}. \tag{3.3}$$

The variances $\mathbb{V}[s_i]$ are estimated in a pilot study. Scaling the summary statistics by their respective variances will prevent the distance from being dominated by the most variable summary statistic as described in [28]. The distance metric is inspired by the one used in [11].

As in [29] a tempering scheme is used for the tolerance: Initially the tolerance is large, but after every 100 accepted iterations of ABC it is reduced to the 80% quantile of the previous 100 accepted distances until a small enough acceptance rate is obtained. The burn in of the algorithm is set so that only simulations corresponding to the smallest tolerances are kept.

After running ABC-MCMC the posterior samples are adjusted using linear regression models in a similar way as in [5]. For the exponential tumour growth model, with posterior samples $\{\tau^i, \eta^i\}_{i=1}^M$, the linear regression models are:

$$\begin{aligned} \tau^i &= \alpha_0 + \alpha_1 s_1^i + \ldots + \alpha_{24} s_{24}^i + \xi^i \\ \eta^i &= \beta_0 + \beta_1 s_1^i + \ldots + \beta_{24} s_{24}^i + \zeta^i. \end{aligned} \tag{3.4}$$

Here it is assumed that the residuals ξ^i and ζ^i are normally distributed, which can be confirmed empirically using normal quantile plots. The adjusted posterior samples are obtained as:

$$\begin{aligned} \tau_{adj}^i &= \alpha_0 + \alpha_1 s_{1,obs}^i + \ldots + \alpha_{24} s_{24,obs}^i + \xi^i \\ \eta_{adj}^i &= \beta_0 + \beta_1 s_{1,obs}^i + \ldots + \beta_{24} s_{24,obs}^i + \zeta^i \end{aligned} \tag{3.5}$$

where ξ^i and ζ^i are the residuals from before. Similar regression adjustments are used for the logistic tumour growth model. The R implementation of the ABC-MCMC algorithm and the regression adjustment is presented in Appendix C.

In the next two sections, the ABC-MCMC algorithm will be implemented for the exponential and logistic tumour growth models in absence of screening. Each section is concluded by a simulation study to investigate the performance of the ABC-MCMC algorithm on respective model.

3.2 ABC for the exponential growth model

The exponential tumour growth model in absence of screening was presented in Section 2.1.1. The model parameters are the shape τ_1 and rate τ_2 of the inverse growth rate distribution and the proportionality constant η in the symptomatic detection hazard:

$$\vec{\theta} = (\tau_1, \tau_2, \eta). \tag{3.6}$$

Since the data does not contain any temporal information, as discussed in [27], these parameters are not identifiable under the model. To ensure identifiability the parameters are restricted to

$$\tau := \tau_1 = \tau_2 \tag{3.7}$$

during the estimation procedure [27]. The same restriction is imposed during the ABC estimation and the parameter space is reduced to

$$\vec{\theta} = (\tau, \eta). \tag{3.8}$$

3.2.1 Data generating model

In ABC a data generating model $\mathcal{G}(\vec{\theta})$ needs to be specified. The data generating model $\mathcal{G}_{exp}(\vec{\theta})$, generating samples from the implicit likelihood of the exponential tumour growth model $f_{V_{sd}}(v)$ (2.5), is constructed in the following way:

1. The inverse growth rate r is sampled from (2.2), using the model parameters $\tau = \tau_1 = \tau_2$.
2. The volume at symptomatic detection V_{sd} conditional on inverse growth rate $R = r$ is sampled from (2.4), using the model parameter η.

Since the tumour volumes at symptomatic detection for different individuals are assumed to be independent, a dataset consisting of N tumour sizes at symptomatic detection can be created by repeated simulation.

3.2.2 Simulation study: evaluating the performance of ABC for the exponential growth model

In this section Algorithm 7 is applied to a synthetic dataset simulated from the data generating model using known parameter values. Given enough data points and a weak prior it is expected that the mean of the of the posterior distribution is close to the data generating parameter values. A synthetic dataset $\mathbf{y}_{obs}$ of

$N = 10000$ tumour volumes at symptomatic detection was simulated, using the data generating model with parameter values taken from [27]:

$$\vec{\theta}_{obs} = (\tau, \eta) \approx (0.848, 6.76 \cdot 10^{-5}). \tag{3.9}$$

The ABC-MCMC algorithm was initialised at $1.5 \cdot \vec{\theta}_{obs}$ and ran for 20000 iterations, which took around 3.28 minutes on a system with 4 GB RAM and an Intel Core i5-6200U 2.30 GHz CPU. The implementation in R is presented in Appendix C.1.

As described earlier, the likelihood of the volume at symptomatic detection has a closed form solution that is presented in (2.5). Therefore, ML point estimates and MCMC estimates of the posterior distribution could be produced as a reference. The ML estimates were obtained using the *optim* function in R. Metropolis Hastings MCMC was implemented according to Algorithm 1.

Kernel density estimates of the posterior distributions of τ and η obtained using ABC-MCMC, regression adjusted ABC-MCMC and Metropolis Hastings MCMC are presented in Figure 3.1. In Appendix A detailed MCMC diagnostics are presented.

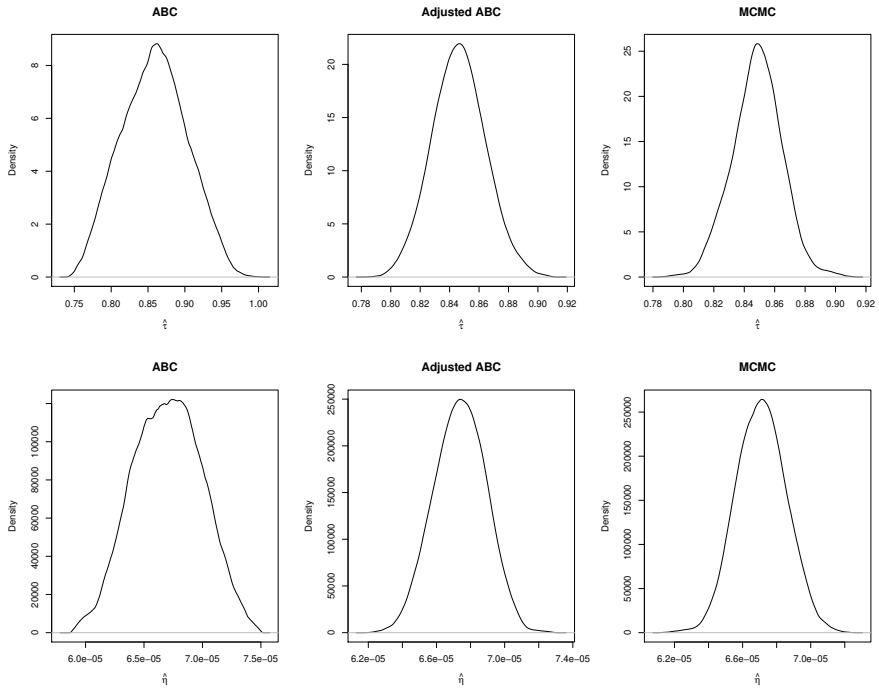

Figure 3.1: Kernel density estimates of the posterior distributions of τ (upper) and η (lower), for the exponential growth model in absence of screening. The posterior densities are produced using ABC-MCMC (left), regression adjusted ABC-MCMC (middle) and using Metropolis Hastings MCMC (right). The Epanechnikov kernel was used for the kernel density estimates.

From these results it seems like the regression adjustment reduces the spread of the posterior density estimates, which become closer to the posterior distributions obtained by MCMC. This is in accordance with theory presented in the background on ABC (Section 2.2).

Mean estimates of τ and η obtained using ABC-MCMC, regression adjusted ABC-MCMC and Metropolis Hastings MCMC are presented in Table 3.1. The data generating parameter values (i.e. the values used when creating the synthetic dataset) and ML-estimates are also provided.

	$\hat{\tau}$	$\hat{\eta}$	Run time
ABC	0.859	$6.70 \cdot 10^{-5}$	3.28 min
ABC REG	0.846	$6.74 \cdot 10^{-5}$	3.28 min
MCMC	0.848	$6.71 \cdot 10^{-5}$	10.8 s
ML	0.848	$6.76 \cdot 10^{-5}$	0.12 s
Data	0.848	$6.76 \cdot 10^{-5}$	-

Table 3.1: Mean estimates of τ and η obtained using ABC-MCMC, regression adjusted ABC-MCMC and Metropolis Hastings MCMC. ML estimates are also presented. In the last row the values used when generating the synthetic dataset are presented.

3.3 ABC for the logistic growth model

The logistic tumour growth model in absence of screening is presented in Section 2.1.4. Instead of using the parameters μ and σ in the log-normal growth rate distribution we will use its mean m and variance v. The original parameters are obtained by the transformation:

$$\begin{aligned} \mu &= \log(m^2/\sqrt{v+m^2}) \\ \sigma &= \sqrt{\log(1+(v/m^2))} \end{aligned} \tag{3.10}$$

Like in the exponential growth model, the proportionality constant η in the symptomatic detection hazard is an additional parameter. Thus, we have the parameter vector:

$$\vec{\theta} = (m, v, \eta). \tag{3.11}$$

In the same way as for the exponential model the parameters need to be restricted to ensure identifiability. This is done by specifying:

$$v = 1.31^{m/1.07} \tag{3.12}$$

This specific relationship is consistent with the estimates presented in [41] and ensures positivity of v throughout the simulation procedure. By this, we restrict the parameter space to

$$\vec{\theta} = (m, \eta). \tag{3.13}$$

3.3.1 Data generating model

For the logistic tumour growth model in absence of screening, it is not clear whether a closed form expression for the likelihood function exists. However, in Appendix B the distribution of tumour volume at symptomatic detection conditional on growth rate is derived, and the likelihood function is presented as an integral. It is also explained how the integral can be approximated numerically, to obtain ML-estimates for the model parameters.

The data generating model $\mathcal{G}_{log}(\vec{\theta})$, generating samples from the implicit likelihood $f_{V_{sd}}(v)$, is constructed in the following way:

1. The transformation (3.10) is used to obtain μ and σ from the mean and the variance.
2. The growth rate r is sampled from the log-normal distribution with parameters μ and σ, described in (2.33).
3. The volume at symptomatic detection is sampled using the distribution for tumour volume at symptomatic detection conditional on the growth rate r.

Since the tumour histories of different individuals are assumed to be independent, a dataset consisting of N tumour sizes at symptomatic detection can be created by repeated simulation.

3.3.2 Simulation study: evaluating the performance of ABC for the logistic growth model

To evaluate the performance of the Algorithm 7 on the logistic growth model, a synthetic dataset with $N = 10000$ individuals was simulated using the estimate of the mean m presented in [41] and the estimate of η presented in [27]:

$$\vec{\theta}_{obs} = (m, \eta) = (1.07, 6.76 \cdot 10^{-5}). \tag{3.14}$$

The algorithm was initialised at $1.5 \cdot \vec{\theta}_{obs}$ and ran for 10000 iterations, which took around 3.8 hours on a system with 4 GB RAM and an Intel Core i5-6200U 2.30 GHz CPU. The implementation in R is presented in Appendix C.2.

Kernel density estimates of the posterior densities of m and η obtained using ABC-MCMC, regression adjusted ABC-MCMC and Metropolis Hastings MCMC are presented in Figure 3.2. Detailed MCMC and ABC diagnostics are presented in Appendix A.

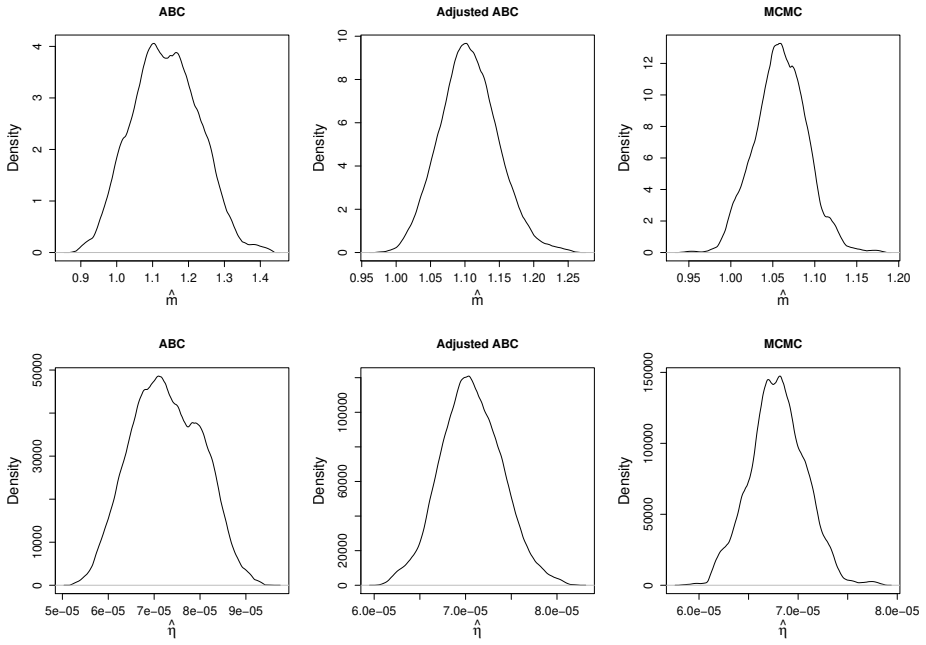

Figure 3.2: Kernel density estimates of the posterior distributions of μ (upper) and η (lower) for the logistic tumour growth model in absence of screening. Generated using: ABC-MCMC (left), ABC-MCMC with regression adjustment (center) and Metropolis Hastings MCMC (right). The Epanechnikov kernel was used for the density estimations.

In Figure 3.2 we can see that the regression adjustment reduces the spread of the posterior estimates, making them more similar to the Metropolis Hastings MCMC posteriors, in the same way as for the exponential growth model.

Mean estimates obtained using ABC-MCMC, regression adjusted ABC-MCMC and Metropolis Hastings MCMC are presented in Table 3.1 together with the data generating parameter values (i.e. the values used to generate the synthetic dataset). ML-estimates, obtained using a numerical approximation of the likelihood function described in Appendix B, are also included.

	$\hat{m}$	$\hat{\eta}$	Run time
ABC	1.14	$7.27 \cdot 10^{-5}$	4.28 h
ABC REG	1.11	$7.06 \cdot 10^{-5}$	4.28 h
MCMC	1.06	$6.79 \cdot 10^{-5}$	9.32 h
ML	1.06	$6.75 \cdot 10^{-5}$	1.85 min
Data	1.07	$6.76 \cdot 10^{-5}$	-

Table 3.2: The mean estimates of m and η obtained using ABC-MCMC without regression adjustment (ABC) and with regression adjustment (ABC REG). ML-estimates are also included. In the last row the values used when generating the synthetic dataset are presented.

From the table we can see that the regression adjustment seems to improve the mean estimates. Moreover, there is a big difference in run times. MLE is very fast in comparison to the other methods since the optimiser only needs to evaluate the likelihood numerically a few times. Metropolis Hastings MCMC is the slowest method, because the likelihood function needs to be evaluated numerically twice in every iteration. ABC turns out to be faster which indicates that one simulation from the data generating model takes shorter time than two evaluations of the likelihood function.

It is not surprising that MLE outperforms the ABC algorithms in this chapter, since the likelihood functions for the models describing data in absence of screening are fast and easy to evaluate. It is for data collected in presence of screening that ABC truly has a potential to be faster and more flexible than MLE. In some situations, it might even be impossible to use a likelihood-based approach. We now turn to consider ABC for models describing data in presence of screening.

4

ABC for models in presence of screening

So far, we have implemented ABC for estimating two different growth models based on data collected in absence of screening. However, there is little need for ABC in these settings since it is possible to compute the likelihood functions. We now develop an ABC algorithm for the random effects models in presence of screening. For these models ABC is of interest because of the computational complexity of the likelihood functions, as well as the potential for ABC to extend to future models.

In Section 4.1 we will discuss how individual screening histories can be taken into consideration in ABC, and present the implicit likelihood. In Section 4.2 a data generating model for the random effects models in presence of screening is outlined. In Section 4.3 difficulties that arise when using ABC with models for screening data are discussed. In Section 4.4 an ABC algorithm for estimating the random effects models in presence of screening is constructed. The section is concluded with a simulation study to evaluate the performance of the algorithm.

4.1 Considering individual screening histories

The models for data in absence of screening use the distribution of tumour volume at (symptomatic) detection as the likelihood for MLE:

$$\mathcal{L}(V_{sd}|\vec{\theta}) = f_{V_{sd}}(v;\vec{\theta}). \tag{4.1}$$

In the ABC implementation it was assumed that the data generating model produced samples from this likelihood. The distribution of tumour volume at detection does not differ between individuals. Population level summaries could therefore be used in ABC without much loss of information.

For the models in presence of screening (Abrahamsson and Humphreys [2]), the distribution of tumour volume at detection V_d conditional on the screening

history $\mathcal{H}$ and detection mode M_d is used as likelihood for MLE:

$$\mathcal{L}(V_d|M_d, \mathcal{H}, \vec{\theta}) = f_{V_d|M_d,\mathcal{H},\vec{\theta}}(v). \tag{4.2}$$

In presence of screening the distribution of tumour volume at detection differ between individuals, since it is dependent on the detection mode and the screening history. For example: tumours detected in screening are typically smaller than tumours detected symptomatically. Tumours detected symptomatically between screens are fast growing and typically large. Using the conditional likelihood, individual variation is accounted for in the MLE.

In ABC we need to produce samples from the implicit likelihood, using a data generating model. The implicit likelihood cannot be the conditional likelihood (4.2), since screening history and detection mode themselves are outcomes of the model. To account for individual variation, the data generating model in ABC should not only simulate the volume at detection but also the detection mode and screening history.

Using an additional model for tumour onset, we will see that the random effects models [2] can be used to generate the screening outcomes, detection mode, age at detection and the tumour volume at detection for an individual. The times for screening (screening ages) cannot be generated using the models, instead we will condition on those.

We define the individual screening history $\mathcal{H}$ as the set of ages at screening $\vec{A_{sc}}$ and the respective screening outcomes $\vec{O}$. Using this notation, the implicit likelihood of interest for ABC is:

$$\mathcal{L}(V_d, M_d, \vec{O}|\vec{A_{sc}}, \vec{\theta}) = f_{V_d,M_d,\vec{O}|\vec{A_{sc}},\vec{\theta}}(v, m, \vec{o}). \tag{4.3}$$

In the next section, we will present a data generating model that produces samples from this implicit likelihood.

4.2 Data generating model

As input the data generating model takes a vector of model parameters

$$\vec{\theta} = (\tau_1, \tau_2, \eta, \beta_1, \beta_2) \tag{4.4}$$

and the vector of the individual's age(s) at attended screen(s) $\vec{A}_{sc}$. The output is an individual history, that corresponds to a sample from the implicit likelihood. The data generating model for exponential tumour growth in presence of screening is outlined below:

1. The age at tumour onset a_0 is simulated from the tumour onset model (2.31).

2. An inverse growth rate r is simulated from the gamma distribution (2.2) with shape τ_1 and rate τ_2.

3. The tumour volume at symptomatic detection V_{sd} is sampled from the distribution (2.4) for tumour volume at symptomatic detection conditional on the inverse growth rate $R = r$, with parameter η.

4. The tumour volume as a function of the time from tumour onset $V(t)$ is obtained using the exponential tumour growth function (2.1) with the inverse growth rate r. The tumour volume as a function of age is obtained by the change of variables $t \to t - a_0$.

5. Using the tumour volume at symptomatic detection and the inverse of $V(t - a_0)$, the age at symptomatic detection is calculated.

6. The tumour volumes at the screens are calculated using $V(t - a_0)$. Using these, the screen outcomes (positive or negative) are sampled from the screening sensitivity model (2.11) with parameters β_1 and β_2. The age at screen detection is set as the age at the first positive screen.

7. The tumour volume at detection is set as the tumour volume at the age at detection.

8. The detection mode is screen detection if the age at screen detection is lower than the age at symptomatic detection, and symptomatic detection otherwise.

The screen outcomes $\vec{O}$, the detection mode M_d and the tumour volume at detection V_d are generated in the last three steps. We have thus produced a sample $(V_d, M_d, \vec{O})$ using $(\vec{A}_{sc}, \vec{\theta})$.

4.2.1 Verification of the data generating model

In this section it will be verified that the data generating model presented in the previous section produces samples from the implicit likelihood (4.3). We will use the following constants:

$$\begin{cases} N & \text{Number of individuals in the population} \\ K(j) \in \{0, 1, 2, \ldots\} & \text{Number of attended screens by individual } j \\ A_{sc}(i, j) \in (0, \infty) & \text{Age of individual } j \text{ at screen } i \end{cases} \tag{4.5}$$

For each individual j we have the following variables:

$$\begin{cases} A_0 \in [0,\infty) & \text{Age at tumour onset} \\ R \in (0,\infty) & \text{Inverse growth rate} \\ M_d \in \{\text{screen}:0,\ \text{symptomatic}:1\} & \text{Detection mode} \\ V_d \in [0,\infty) & \text{Volume at detection} \\ A_d \in [0,\infty) & \text{Age at detection} \\ A_{sd} \in [0,\infty) & \text{Age at symptomatic detection} \\ O(i) \in \{\text{negative}:0,\ \text{positive}:1\} & \text{Outcome for screen } i \end{cases} \tag{4.6}$$

Additionally, we will use the notation:

$$\vec{O} = (O(1), \ldots, O(K)) \tag{4.7}$$

$$\vec{A}_{sc} = (A_{sc}(1), \ldots, A_{sc}(K)). \tag{4.8}$$

The individual history is uniquely determined by the age at tumour onset A_0, inverse growth rate R, screening history $\mathcal{H} = \{\vec{O}, \vec{A}_{sc}\}$, detection mode M_d and the age at detection A_d. Moreover, the age of detection, detection mode and volume can be obtained by deterministic functions of the other variables:

$$A_d = f(\vec{O}, \vec{A}_{sc}, A_{sd}) = \min\{A_{scd}, A_{sd}\} \tag{4.9}$$

$$M_d = g(A_d, A_{sd}) = \begin{cases} 0 & A_{scd} < A_{sd} \\ 1 & \text{otherwise} \end{cases} \tag{4.10}$$

$$V_d = h(A_0, A_d, R) = v_0 \exp((A_d - A_0)/R) \tag{4.11}$$

where the help variable

$$A_{scd} = \vec{A}_{sc}(\min\{i : S(i) = 1\}) \tag{4.12}$$

is the age at the first positive screen. It is thus sufficient to consider the joint probability of $A_0, R, \vec{O}, A_{sd}$ when expressing the indirect likelihood function:

$$\begin{aligned} \mathcal{L}(V_d, M_d, \vec{O} | \vec{A}_{sc}, \vec{\theta}) &:= \mathbb{P}(A_0, R, \vec{O}, A_{sd} | \vec{A}_{sc}, \vec{\theta}) \\ &= \mathbb{P}(R, \vec{O}, A_{sd} | A_0, \vec{A}_{sc}, \vec{\theta}) \mathbb{P}(A_0 | \vec{A}_{sc}, \vec{\theta}) \\ &= \mathbb{P}(\vec{O}, A_{sd} | R, A_0, \vec{A}_{sc}, \vec{\theta}) \mathbb{P}(R | A_0, \vec{A}_{sc}, \vec{\theta}) \mathbb{P}(A_0 | \vec{\theta}) \\ &= \mathbb{P}(\vec{O}, A_{sd} | R, A_0, \vec{A}_{sc}, \vec{\theta}) \mathbb{P}(R | \vec{\theta}) \mathbb{P}(A_0 | \vec{\theta}) \\ &= \mathbb{P}(\vec{O} | R, A_0, \vec{A}_{sc}, A_{sd}, \vec{\theta}) \mathbb{P}(A_{sd} | R, A_0, \vec{A}_{sc}, \vec{\theta}) \mathbb{P}(R | \vec{\theta}) \mathbb{P}(A_0 | \vec{\theta}) \\ &= \mathbb{P}(\vec{O} | R, A_0, \vec{A}_{sc}, \vec{\theta}) \mathbb{P}(A_{sd} | R, A_0, \vec{\theta}) \mathbb{P}(R | \vec{\theta}) \mathbb{P}(A_0 | \vec{\theta}). \end{aligned} \tag{4.13}$$

The following assumptions are made:

1. $A_0 | \vec{\theta}$ independent of $\vec{A}_{sc}$.

2. $R|\vec{\theta}$ independent of $\vec{A}_{sc}$.
3. $A_{sd}|R, A_0, \vec{\theta}$ independent of $\vec{A}_{sc}$.
4. $\vec{O}|R, A_0, \vec{A}_{sc}, \vec{\theta}$ is independent of A_{sd}.

The first three assumptions are reasonable since tumour development is independent of screening attendance (i.e. how and when a person attends screening). The last assumption should also be valid since the screening sensitivity is modelled as only dependent on the tumour volume, which is a deterministic function of $R, A_0, \vec{A}_{sc}$.

We thus have the following expression for the indirect likelihood:

$$\mathcal{L}(V_d, M_d, \vec{O}|\vec{A}_{sc}, \vec{\theta}) = \mathbb{P}(\vec{O}|R, A_0, \vec{A}_{sc}, \vec{\theta})\mathbb{P}(A_{sd}|R, A_0, \vec{\theta})\mathbb{P}(R|\vec{\theta})\mathbb{P}(A_0|\vec{\theta}). \quad (4.14)$$

If we compare these distributions with the data generating model, we can note that $\mathbb{P}(A_0|\vec{\theta})$ corresponds to step 1, $\mathbb{P}(R|\vec{\theta})$ to step 2, $\mathbb{P}(A_{sd}|R, A_0, \vec{\theta})$ to step 3 and $\mathbb{P}(\vec{O}|R, A_0, \vec{A}_{sc}, \vec{\theta})$ to steps 4–6. Step 7 and 8 corresponds to the deterministic functions (4.9) and (4.10) respectively. This suggests that the data generating model indeed produces samples from the implicit likelihood.

4.3 ABC, screening data and the curse of dimensionality

In this section the difficulties that arise when using ABC with the models for screening data will be discussed. We will start by considering the exact inference algorithm; that is, ABC but using exact comparison of the full data (Algorithm 2). This algorithm is not computationally feasible because of the curse of dimensionality. The problem stems from comparing the full data at individual level, rather than using summary statistics. Creating informative summary statistics is a difficult task, since the individual screening histories may be very different for different individuals.

Observed screening data includes information about the tumour volume at detection V_d, age at detection A_d, detection mode M_d and screening history $\mathcal{H} = \{\vec{O}, \vec{A}_{sc}\}$. In the screening history, all screens previous to detection are negative.

Using the data generating model outlined in Section 4.2, we can simulate data from the implicit likelihood. For a population of N individuals, bold face

characters will be used to denote the vectors and indexed sets:

$$\begin{aligned} \vec{\mathbf{O}} &= \{\vec{O}_1, \ldots, \vec{O}_N\} \\ \mathbf{A_d} &= (A_d^1, \ldots, A_d^N) \\ \mathbf{V_d} &= (V_d^1, \ldots, V_d^N) \\ &\vdots \end{aligned} \tag{4.15}$$

In the exact version of ABC (Algorithm 2) we only accept simulated data that exactly match the observed data:

$$\vec{\mathbf{O}}_{obs} = \vec{\mathbf{O}}_{sim} \tag{4.16}$$
$$\mathbf{M_d}_{obs} = \mathbf{M_d}_{sim} \tag{4.17}$$
$$\mathbf{V_d}_{obs} = \mathbf{V_d}_{sim} \tag{4.18}$$
$$\mathbf{A_d}_{obs} = \mathbf{A_d}_{sim} \tag{4.19}$$

Note that the probability of acceptance for this comparison is zero, because the volume at detection V_d and age at detection A_d are both continuous random variables. Instead we introduce a distance metric for these variables:

$$\delta(\mathbf{V_d}_{obs}, \mathbf{V_d}_{sim}) \leq \epsilon \tag{4.20}$$
$$\delta(\mathbf{A_d}_{obs}, \mathbf{A_d}_{sim}) \leq \epsilon. \tag{4.21}$$

However, this algorithm is still not computationally feasible because of the high dimensionality. To understand this, consider a population of $N = 100$ individuals who all undergo one screen. For each individual there are two different outcomes for the detection mode: screen detection and symptomatic detection. Only considering the outcome of the screen, there are 2^{100} possible outcomes for the whole population. If we introduce one more screen for all individuals, there are three possible outcomes for each individual and thus 3^{100} outcomes for the whole population. The number of outcomes therefore grows exponentially with the number of individuals with a base that increases with the number of screens. Since at least a few thousand individuals are needed for estimation (we used $N = 10000$ in the previous section to get good convergence) acceptance will never occur in practice.

The curse of dimensionality stems from the comparison of data at an individual level. In absence of screening, we could define bins for the tumour volume at detection and use the number of individuals in each bin as a summary statistic. But in presence of screening, as discussed in Section 4.1, the distribution of a detected tumour is dependent on the individual screening history. Therefore, it is not sensible to only categorise an individual using the tumour volume.

One possible way to solve this issue could be to jointly define categories for the age at detection A_d, detection mode M_d and screening history $\mathcal{H}$ in such a way that the distributions of tumour size at detection are similar within each

category. In that case, we can create bins for the tumour volume and use the table of counts of each element in the Cartesian product of the set of bins and categories as a population summary statistic. In particular, we could compare the empirical distributions of the tumour volume in each category, with the corresponding ones of the data.

In the next section we will use the table of counts approach to develop an ABC algorithm for estimating the random effects models in presence of screening. This will be for the special case of a population with homogeneous screening attendance; that is, all individuals in the population attend screening the same number of times and at the same ages. In the discussion, Section 5.5, we will consider possible ways to generalise this approach to populations with heterogeneous screening attendance.

4.4 An ABC algorithm for fitting random effects models in presence of screening

In this section we will assume homogeneous screening attendance; that is, all individuals in the population attended the same number of screens K and at the same ages $\vec{A}_{sc}$. Using this assumption, we will construct an ABC algorithm for cohort data on individuals with ages in an interval $C \subset \mathbb{R}^+$.

We will adapt the ABC-MCMC algorithm (7) developed to estimate the models in absence of screening. The same distance metric (3.3), transition kernel (3.1) and the flat uninformative prior will be used. The data generating model presented in Section 4.2 will be used to produce samples from the implicit likelihood. After running ABC-MCMC, linear regression adjustments (3.5) are made to the posterior samples.

As mentioned in the previous section, we will construct summary statistics to avoid the curse of dimensionality. We start by creating B bins for the tumour volume at detection, in the same way as for the models in absence of screening. After this the cohort age interval C is divided into M sub-intervals $I_1, \ldots, I_M$ in such a way that each screen age is on the boundary between two neighbouring sub intervals. See Figure 4.1 for an example.

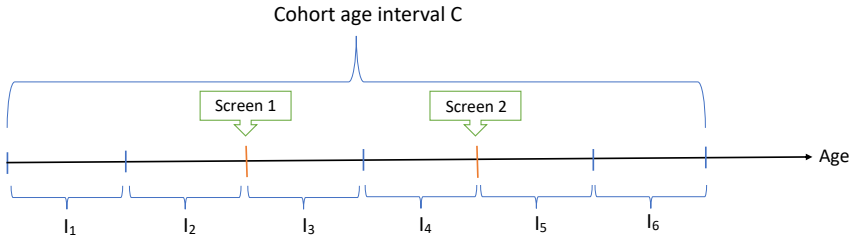

Figure 4.1: An example of how a screening history with $K = 2$ screens can be split into $M = 6$ sub-intervals. Note that both screens are on the boundary between two sub-intervals. There are 6 categories for symptomatic detection: $A_d \in I_1, \ldots, A_d \in I_6$. Additionally there are two categories for screen detection: $A_d = A_{sc}(1)$ and $A_d = A_{sc}(2)$. Thus, there are 8 categories in total.

Now we can define M categories for the age at symptomatic tumour detection $A_d \in I_1, \ldots, A_d \in I_6$, and K categories for the age at screen detection $A_d = A_{sc}(1), \ldots, A_d = A_{sc}(K)$. In total we have $K + M$ categories that together summarise the age at detection, detection mode and the screening history. Combining these categories with the B bins for tumour volume at detection, we can create a $B \times (K + M)$ table. To each element of the table we can assign the count of corresponding individual histories. This table of counts will be used as our $B \cdot (K + M)$ dimensional summary statistic for the ABC-MCMC algorithm.

When using this kind of summary statistic for ABC we introduce bias. However, by starting with a model that is known to be identifiable, we know that potential issues are due to the ABC approximation and not due to identifiability issues of the underlying model. We are no longer estimating the model parameters based on each individual screening history, but instead we have carefully designed summary statistics aggregating individuals with similar distributions of tumour volume at detection. In the natural history modelling literature (e.g. [33], [7]) it is common to calibrate models to data on incidence, detection, and mortality rates. Here, starting from the exact version of ABC for an identifiable model, we try to define as informative summary statistics as possible while still maintaining computational feasibility.

Our model parameters are:

$$\vec{\theta} = (\tau_1, \tau_2, \eta, \beta_1, \beta_2). \tag{4.22}$$

When estimating the models in absence of screening, the parameter restriction $\tau_1 = \tau_2$ had to be imposed to obtain parameter identifiability. Using screening data we can relax these parameter restrictions [2].

4.4.1 Simulation study

To evaluate the performance of the proposed algorithm a simulation study was performed assuming a cohort between age 40 and 48 that undergoes one screen at age 42. A synthetic dataset of $N = 1400000$ individuals was generated, using the data generating model with the model parameter values $\vec{\theta}_{obs}$ presented in Table 4.1. The values are MLE estimates taken from [2].

$$\begin{aligned}\tau_1 &= 2.36\\ \tau_2 &= 4.16\\ \eta &= 1.79 \cdot 10^{-5}\\ \beta_1 &= \text{-}4.75\\ \beta_2 &= 0.56\end{aligned}$$

Table 4.1: Parameter values, $\vec{\theta}_{obs}$, used when generating the synthetic dataset.

For each individual, a tumour can be detected symptomatically at any point in their lives, or screen detected at age 42. Roughly 13000 individuals had a tumour detected at an age between 40 and 48. These were aggregated using the following categories:

1. Symptomatic detection at age 40-42
2. Screen detection at age 42
3. Symptomatic detection at age 42-44
4. Symptomatic detection at age 44-48

For tumour volume 14 bins were used, with the following breakpoints in $\mathbb{R}^+$: 5, 80, 300, 700, 1500, 3000, 5000, 8000, 12500, 20000, 33000, 55000, 120000 (all in mm^3). This resulted in a 14×4 table, and thus a 56-dimensional summary statistic for ABC. In Figure 4.2 the posterior distributions of tumour volume at detection in respective category are presented.

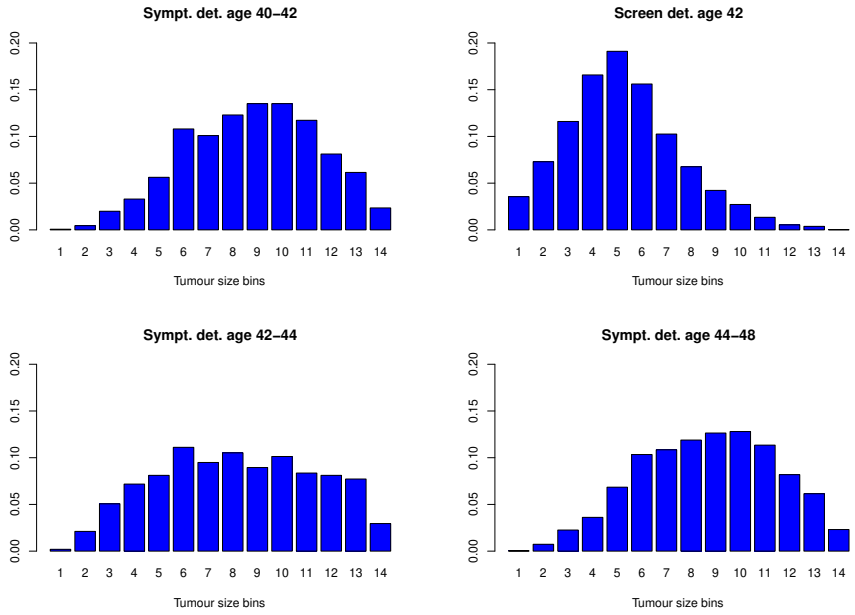

Figure 4.2: The distribution of tumour volume for the four different categories summarising detection mode and age at detection.

In the two years before the screen (top left) small tumours are not likely to be detected symptomatically. However, at screen detection (top right) smaller asymptomatic tumours are more likely to be found. In the first two years after the screen (bottom left), fewer large tumours are detected than before the screen since small and medium sized tumours are likely to be found in screening. However, some fast growing tumours with onset after the first screen may still be detected in the first two years after the screen. The distribution long after the screen (bottom right) is similar to the one before the screen. This is also reasonable since most tumours that are detected in this period are likely to have had their onset after the time for the screen, even in absence of it.

The ABC-MCMC algorithm was initialised at $1.3 \cdot \vec{\theta}_{obs}$ and ran for 20000 iterations which took 8.27 hours on a system with 4 GB RAM and an Intel Core i5-6200U 2.30 GHz CPU. The R code with detailed values of hyperparameters is presented in Appendix C.3. The resulting kernel estimates for the marginal posterior distributions of the model parameters are presented in Figure 4.3 and Figure 4.4. Point estimates of the posterior means are presented in Table 4.2 together with the data generating parameter values. Detailed ABC-MCMC diagnostics are presented in Appendix A.

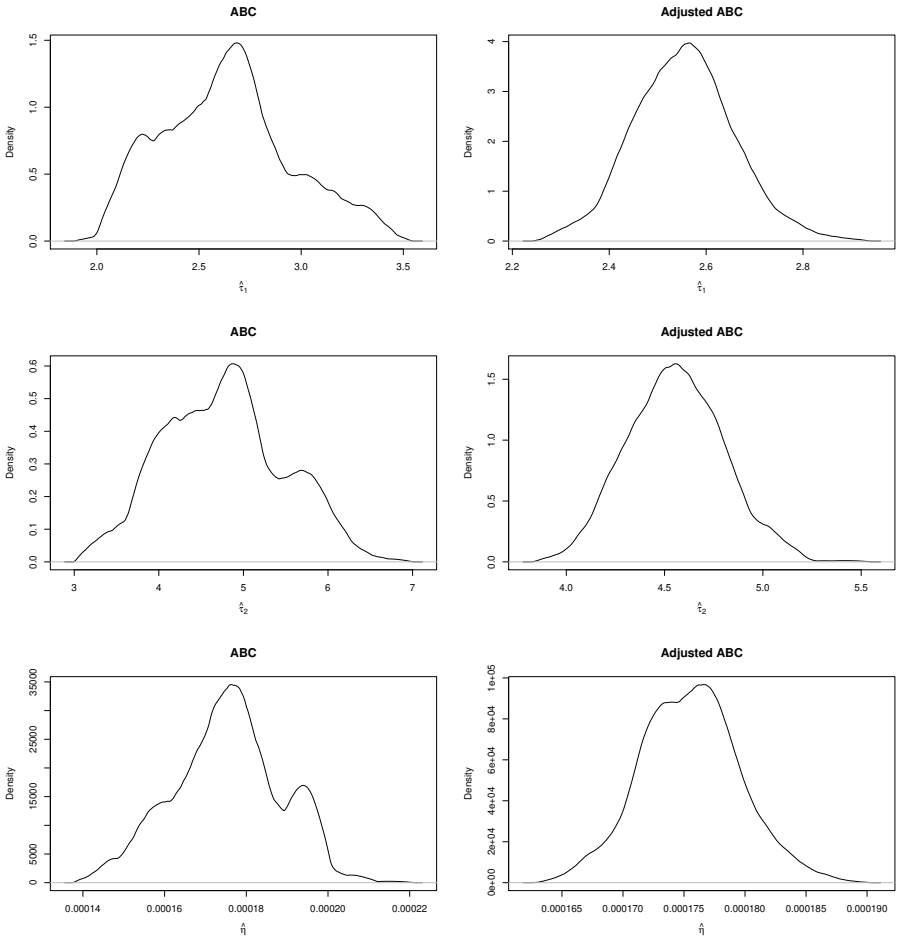

Figure 4.3: Kernel density estimates of the marginal posterior distributions of τ_1 (upper), τ_2 (centre) and η (lower). The Epanechnikov kernel was used for the kernel density estimates.

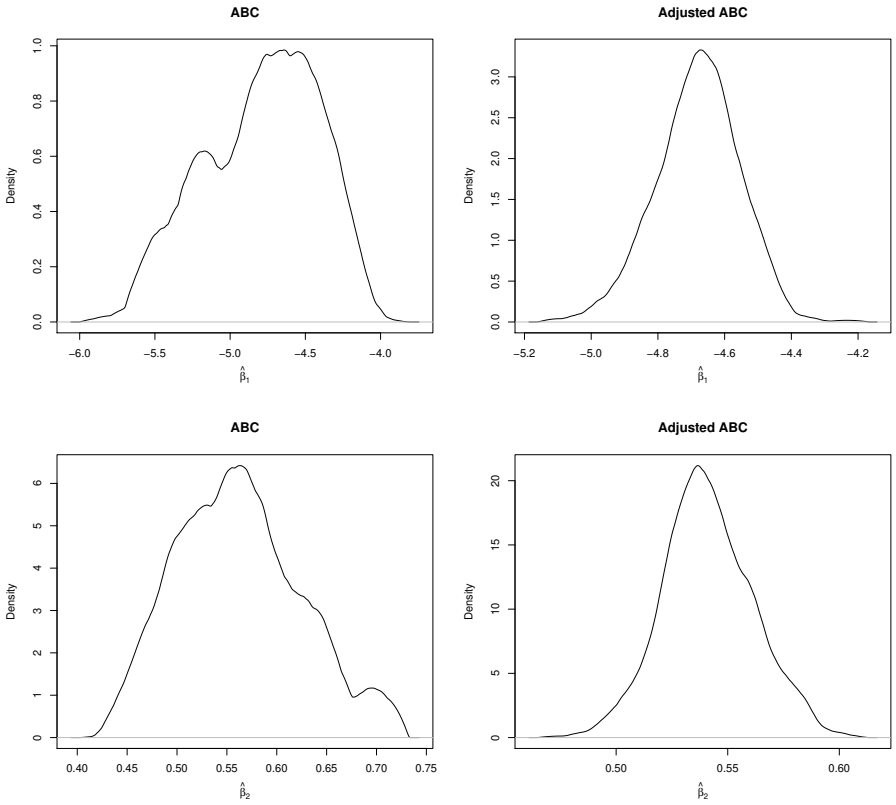

Figure 4.4: Kernel density estimates of the marginal posterior distributions of β_1 (upper) and β_2 (lower). The Epanechnikov kernel was used for the kernel density estimates.

	ABC	ABC-REG	Data
$\hat{\tau_1}$	2.64	2.55	2.36
$\hat{\tau_2}$	4.77	4.56	4.16
$\hat{\eta}$	$1.75 \cdot 10^{-5}$	$1.76 \cdot 10^{-5}$	$1.79 \cdot 10^{-5}$
$\hat{\beta_1}$	-4.79	-4.68	-4.75
$\hat{\beta_2}$	0.56	0.54	0.56

Table 4.2: Posterior mean estimates using ABC-MCMC and regression adjusted ABC-MCMC. In the last column the values used to generate the synthetic dataset are presented.

Note that the ABC-MCMC mean estimates for η, β_1 and β_2 are more accurate than the ones for τ_1 and τ_2. However, like for the models in absence of screening the regression adjustment reduces the spread of the posterior densities. It also improves the mean estimates of τ_1 and τ_2, while it introduces some bias to the estimates of β_1 and β_2.

5

Discussion

In this thesis we have investigated whether ABC can be implemented for the breast cancer random effects models in presence of screening previously described by Abrahamsson and Humphreys [2].

As a first step (Chapter 3), an ABC-MCMC algorithm was implemented for two simple breast cancer growth models in absence of screening: the first assuming exponential growth and the second assuming logistic growth. For the exponential tumour growth model, there is an analytic expression for the likelihood [27]. For the logistic growth model, the distribution of tumour volume at symptomatic detection conditional on growth rate was derived and the corresponding likelihood function evaluated numerically. Using simulation studies, the ABC-MCMC algorithm was compared with likelihood-based methods and its convergence could be confirmed.

The main part of the analysis (Chapter 4) treated ABC for the random effects models in presence of screening [2]. Since these models describe individual screening histories, the data generating model had to be specified to not only generate the tumour volume at detection but also the individual detection modes and screening outcomes.

ABC cannot be directly applied to the models in presence of screening because of the large number of different possible screening histories for each individual. This is related to the curse of dimensionality in ABC. Instead we considered a special case for the data, where all individuals in the population have the same screening attendance. Additionally, we had to include a model for tumour onset. Informative summary statistics could then be defined using categories for the age at detection and detection mode, and bins for the tumour volume at detection. Using these summary statistics with the data generating model, an ABC-MCMC algorithm for the model in presence of screening could be constructed. Convergence of this algorithm was confirmed by a simulation study using synthetic data [2].

The first section of this chapter, Section 5.1, explains why a tumour onset model is needed. In Section 5.2, the results and possible improvements of the simulation studies are discussed. In Section 5.3, the approximations introduced by ABC for

individual and population level data are discussed. Following this we will explore how the algorithm could be applied to observational data (Section 5.4) and how it could potentially be extended to include heterogeneous screening attendance (Section 5.5). Finally, a summary of the main points of the discussion will be given (Section 5.6).

5.1 The need for a tumour onset model

In Section 4.2 it was mentioned that a model for tumour onset had to be included when constructing the data generating model in presence of screening. Here we will provide a brief explanation for why the tumour onset model is needed. The data generating model needs to generate the tumour volume at detection. Simulating an inverse growth rate from (2.2), we can obtain the tumour volume as a deterministic function of the time from tumour onset (2.1). For symptomatic detection the time since tumour onset can be simulated using the symptomatic detection hazard (2.3). However, for screen detection there is no such model. By introducing a model for tumour onset (2.31) we can simulate the age at tumour onset. Now, since we condition on the individual ages at screening, we can calculate the time between tumour onset and screen detection. Using this, the tumour volume at screen detection can be obtained from (2.1). In this way, the tumour onset model enables simulation of tumour volume at screen detection.

5.2 Simulation studies

In Section 3.2.2 and Section 3.3.2 we presented simulation studies for the models in absence of screening. We assume that Metropolis Hastings MCMC (MH), more accurately estimates the true posteriors than ABC. By inspecting the mixing plots presented in Appendix A, this seems to be well motivated. Additionally, the mean estimates of the MH posteriors are close to the MLE, which is reasonable since we use the uninformative flat prior.

Comparing the ABC and MH posterior densities, we can see that ABC clearly overestimates the posterior spread. This is in accordance with what has previously been described in the ABC literature [4] and is improved using linear regression adjustment [5]. The mean estimates of ABC are close to the MH mean estimates and even closer using the regression adjustment. From these observations it seems essential to adjust the ABC estimates for accurate estimates of the posterior variance, but regression adjustment is also good for improving posterior mean estimates. In the ABC literature it is known that smaller tolerance (and thus lower acceptance rates) will reduce the bias of the posterior estimates and thus reduce the need of using regression adjustments [4]. The final acceptance rates

were 0.04 and 0.08 for the exponential and the logistic growth model, respectively.

At the end of Chapter 4, we presented a simulation study for the natural history model of breast cancer in presence of screening [2]. Here we do not have likelihood-based estimates as a reference (in particular MCMC posteriors would be of interest) but we can still validate convergence of the algorithm by comparing the ABC posterior mean estimates with the data generating parameter values.

We can see that the data generating parameter values are covered by the ABC posteriors. It thus seems likely that the ABC-MCMC algorithm converges. Like previously, the regression adjustments reduce the spread of the posterior distributions. The results for the models in absence of screening suggests that this corresponds to a correction of the posterior variance estimate.

For β_1 and β_2 the regression adjustment appears to slightly worsen the posterior mean estimates. It is unclear if this is actually the case, since it is not necessary for the true posterior mean to be equal to the data generating parameters. However, in this thesis the simplest form of regression adjustment has been used. To improve performance more advanced types, such as the ones described in [5] and [9], could be implemented. Like always in regression modelling, it is important to find a model which accurately explains the data without overfitting.

While the data generating parameter values of β_1, β_2 and η are close to the mean estimates of their respective ABC posteriors, the ones for τ_1 and τ_2 are situated in the posterior tails. The estimates for these parameters thus seem to be rather biased. Inspecting the ABC-MCMC diagnostics in Appendix A, we can see that the mixing for these parameters is in fact rather bad.

τ_1 and τ_2 are the shape and the rate of the inverse growth rate gamma distribution. Since the mean of the gamma distribution is given by τ_1/τ_2, it will be invariant when scaling these parameters equally. If looking carefully at the mixing plot we can see that there in fact seems to be a strong positive correlation between these parameters which can be responsible for the poor mixing [39]. The ABC-MCMC mixing for τ_1 and τ_2 could potentially be improved by using a correlated transition kernel or more advanced methods such as the one in [39].

In this thesis the focus has been on developing algorithms for estimating the natural history models in presence of screening. The convergence of these algorithms has been checked using simulation studies. However, with more time the algorithms could have been evaluated in greater detail. In particular it would have been interesting to include variance estimates, quantile estimates and to use several simulated datasets to evaluate robustness and coverage. Additionally, it would be interesting to investigate the robustness of the algorithms to varying sample sizes.

5.3 The approximations of ABC

In this thesis, we have used ABC for models where parameter identifiability has previously been verified. Both Plevritis et al. [27] and Abrahamsson and Humphreys [2] could ensure identifiability by fitting models to individual level data using MLE.

Parameter identifiability is not always treated carefully. In microsimulation models (MSM) calibrated to population level data, it can be difficult to know if the underlying model parameters are identifiable. Since these parameters are important to understand the disease progression, predictions of the models may be validated in some other way. In for example Berry et al. [7], several models were specified independently and later compared to validate predictions of the models.

ABC is a method that fits well into the MSM framework. In for example Rutter et al. [33] a MSM for the natural history of colorectal cancer is fitted using an algorithm based on ABC. As calibration target cancer incidence rates are used, which is a typical example of population level calibration target in the literature.

ABC is an approximate method because it introduces two different sources of bias [4]. The first is the distance metric which is used instead of exact comparison. The other is the summary statistics which are used in place of the full data. There is sometimes a trade-off between these two sources of bias: Increasing the number of summary statistics may add information about the underlying data and thus reduce bias. However, more summary statistics (higher dimensionality) will reduce the acceptance rate because of the curse of dimensionality in ABC. To prevent this, we need to increase the ABC tolerance (accept simulated data further away from the observed data) which will also introduce bias. This is related to the bias-variance trade-off in ABC: For a fixed runtime, a lower tolerance gives less bias but also fewer posterior samples and thus higher variance. Note that reducing the number of summary statistics does not always introduce bias. For example, when using sufficient statistics in place of the full data no additional bias is introduced [20].

For the models in absence of screening it is likely that almost all the bias originates from the distance metric. This is because we use many (24) fine categories for the tumour volume. In Plevritis et al. [27] the tumour volume is binned for MLE in a similar way, which introduces similar bias. The additional bias of ABC thus only originates from the distance metric for these models.

ABC has previously not been applied to natural history models using data at an individual level, as we do in this thesis. In our approach we start with a model for which parameter identifiability is verified. For this model, we create a data generating model and specify the exact inference version of ABC. Because we

know that the model parameters are identifiable to start with, we can assume that this also holds for exact ABC inference. From that point we introduce approximations in the form of summary statistics and a distance metric. By carefully designing the summary statistics to be as informative as possible, we can maintain a low bias while avoiding the curse of dimensionality.

In particular, the approach has been to aggregate individuals with similar screening and detection histories. Individuals who detect a tumour in screening are separated from individuals who detect a tumour symptomatically. For symptomatically detected individuals, information on the time relative to neighbouring screens is incorporated by aggregating on age at detection.

Figure 4.2 displays the distributions of tumour volume at detection conditional on belonging to a specific category, in terms of the age at detection and detection mode for simulated data. These distributions are important for understanding how to design the summary statistics for ABC. A category should be chosen so individuals in its possible subcategories have similar distributions of tumour volume at detection. This should be valid for populations generated using parameters in some credible interval of the prior distribution of interest (we have used the flat improper prior to simplify convergence checking, but it is in general good practice to specify a prior [40]). In this way little information is lost, but the number of statistics is reduced. Moreover, bins for the tumour volume should be defined to incorporate as much information as possible about these distributions without including too many.

As previously mentioned, we can simulate multiple datasets to investigate the properties of the ABC estimates in greater detail. By doing so, we could investigate the robustness of the algorithm to different ways of creating categories for the age at detection and detection mode.

In the present implementation of the ABC algorithm, the distance metric is simply the weighted Euclidean distance between the simulated and observed counts in each category. It would be interesting to investigate whether this distance metric can be improved by instead comparing the conditional distributions in Figure 4.2 of the observed and simulated data. Different distance metrics more appropriate for comparing distributions, e.g. the Wasserstein distance [6], could then be used in place of the Euclidean distance.

5.4 Using observational screening data

Abrahamsson and Humphreys [2] fitted the models to data from a case control study of breast cancer in Swedish women diagnosed with a primary invasive breast cancer. If we could fit the models to the same data using ABC, we could provide additional verification of the algorithm by comparing its estimates with the ML-

estimates from [2]. However, since the screening attendance is not homogeneous in the observational data the algorithm needs to be extended. Approaches to extending ABC for observational data with heterogeneous screening attendance are discussed in the next section.

5.5 Extending ABC for heterogeneous screening attendance

One of the assumptions we had to make for using ABC in presence of screening is that the screening attendance is homogeneous in the data. This allowed us to aggregate individuals with similar age and detection mode.

Screening attendance is in general heterogeneous. One possible way to assess this could be to use an additional model for simulating screening attendance. The simplest example would be to sample screening attendance from the empirical distribution obtained from the data. After doing so, this screening attendance could be used in the data generating model to generate an individual.

Under the assumptions that growth rate and screening sensitivity are independent of the age at tumour onset (which will in many cases be reasonable [2]), it would be reasonable to consider the relative time to the previous screens rather than the age at detection for forming summary statistics. For example, we could create two categories for symptomatic and screen detection, three categories for the time since last screen (e.g. 1–2 years, 2–3 years and >3 years) and similarly for other previous screens.

This approach is likely to add another layer of approximation to ABC. But if it is accurate enough, it may be a promising way to enable ABC for heterogeneous screening data. Doing so could be a first step to construct an algorithm that can generalise to eventual future models where the likelihood function is intractable. For example we could estimate the models in presence of screening but with logistic tumour growth, as described in Section 2.1.4. Another example of a potential application is models describing tumour regression.

5.6 Summary

We have constructed an ABC-MCMC algorithm for estimating a model for the natural history of breast cancer in presence of screening [2]. In doing so, we had to introduce a model for tumour onset and assume that all individuals in the population attend screening at the same ages.

ABC is approximate in two ways: the first by its use of summary statistics in place of the full data, and the second by its use of a distance metric in place of exact comparison. In our approach we have carefully chosen the summary statistics to introduce as little bias as possible.

The statistical properties of the ABC-MCMC algorithm need to be explored further. In particular, we would like to investigate robustness for different datasets, sample sizes, and ways to define the summary statistics. It would also be interesting to use the algorithm with the real data described in [2] for which ML-estimates are available for reference.

It seems to be possible to extend the algorithm to heterogeneous screening data by including a model for screening attendance. This would introduce an additional level of approximation but could potentially be very useful for future models where the likelihood function is intractable.

6

Conclusions

- An ABC-MCMC algorithm was constructed with summary statistics carefully designed to incorporate information at an individual level, for estimating a model describing the natural history of breast cancer in presence of screening [2]. This was done by using a sub-model for tumour onset.
- Although we focused on a scenario where all individuals had the same screening attendance, there appears to be ways to extend the algorithm to heterogeneous screening data. Further research is however needed.
- The statistical properties of the algorithm should be evaluated further and it should be tested on observational data where previous estimates are available for reference.

Bibliography

[1] Linda Abrahamsson. *Statistical models of breast cancer tumour progression for mammography screening data*. Phd dissertation, Karolinska Institutet, Stockholm, October 2018.

[2] Linda Abrahamsson and Keith Humphreys. A statistical model of breast cancer tumour growth with estimation of screening sensitivity as a function of mammographic density. *Statistical Methods in Medical Research*, 25(4): 1620–1637, July 2013. ISSN 0962-2802. doi: 10.1177/0962280213492843. URL https://doi.org/10.1177/0962280213492843.

[3] Mark A. Beaumont. Approximate Bayesian Computation in Evolution and Ecology. *Annual Review of Ecology, Evolution, and Systematics*, 41 (1):379–406, 2010. doi: 10.1146/annurev-ecolsys-102209-144621. URL https://doi.org/10.1146/annurev-ecolsys-102209-144621.

[4] Mark A. Beaumont. Approximate Bayesian Computation. *Annual Review of Statistics and Its Application*, 6(1):379–403, 2019. doi: 10.1146/annurev-statistics-030718-105212. URL https://doi.org/10.1146/annurev-statistics-030718-105212.

[5] Mark A. Beaumont, Wenyang Zhang, and David J. Balding. Approximate Bayesian Computation in Population Genetics. *Genetics*, 162(4):2025, December 2002. URL http://www.genetics.org/content/162/4/2025.abstract.

[6] Espen Bernton, Pierre E. Jacob, Mathieu Gerber, and Christian P. Robert. Approximate Bayesian computation with the Wasserstein distance. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 81(2): 235–269, 2019. ISSN 1467-9868. doi: 10.1111/rssb.12312. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12312. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/rssb.12312.

[7] Donald A. Berry, Kathleen A. Cronin, Sylvia K. Plevritis, Dennis G. Fryback, Lauren Clarke, Marvin Zelen, Jeanne S. Mandelblatt, Andrei Y. Yakovlev, J. Dik F. Habbema, Eric J. Feuer, and Cancer Intervention and Surveillance Modeling Network (CISNET) Collaborators. Effect of screening and adjuvant therapy on mortality from breast cancer. *The New England Journal of Medicine*, 353(17):1784–1792, October 2005. ISSN 1533-4406. doi: 10.1056/NEJMoa050518.

[8] M. G. B. Blum, M. A. Nunes, D. Prangle, and S. A. Sisson. A Comparative Review of Dimension Reduction Methods in Approximate Bayesian Computation. *Statistical Science*, 28(2):189–208, May 2013. ISSN 0883-4237, 2168-8745. doi: 10.1214/12-STS406. URL https://projecteuclid.org/euclid.ss/1369147911.

[9] Michael G. B. Blum and Olivier François. Non-linear regression models for Approximate Bayesian Computation. *Statistics and Computing*, 20(1): 63–73, January 2010. ISSN 1573-1375. doi: 10.1007/s11222-009-9116-0. URL https://doi.org/10.1007/s11222-009-9116-0.

[10] Siddhartha Chib and Edward Greenberg. Understanding the Metropolis-Hastings Algorithm. *The American Statistician*, 49(4):327–335, November 1995. ISSN 0003-1305. doi: 10.1080/00031305.1995.10476177. URL https://amstat.tandfonline.com/doi/abs/10.1080/00031305.1995.10476177. Publisher: Taylor & Francis.

[11] Katalin Csilléry, Olivier François, and Michael G. B. Blum. abc: an R package for approximate Bayesian computation (ABC). *Methods in Ecology and Evolution*, 3(3):475–479, 2012. ISSN 2041-210X. doi: 10.1111/j.2041-210X.2011.00179.x. URL https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/j.2041-210X.2011.00179.x.

[12] S. W. Duffy, H. H. Chen, L. Tabar, and N. E. Day. Estimation of mean sojourn time in breast cancer screening using a Markov chain model of both entry to and exit from the preclinical detectable phase. *Statistics in Medicine*, 14(14):1531–1543, July 1995. ISSN 0277-6715. doi: 10.1002/sim.4780141404.

[13] Paul Fearnhead and Dennis Prangle. Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(3):419–474, 2012. ISSN 1467-9868. doi: 10.1111/j.1467-9868.2011.01010.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2011.01010.x.

[14] Folkhälsomyndigheten. Dödlighet i bröstcancer, 2020. https://www.folkhalsomyndigheten.se/folkhalsorapportering-statistik/tolkad-rapportering/folkhalsans-utveckling/resultat/halsa/dodlighet-i-cancer/brostcancer-dodlighet/ [Accessed: 2020-05-02].

[15] Gabriel Isheden and Keith Humphreys. Modelling breast cancer tumour growth for a stable disease population. *Statistical Methods in Medical Research*, 28(3):681–702, November 2017. ISSN 0962-2802. doi: 10.1177/0962280217734583. URL https://doi.org/10.1177/0962280217734583.

[16] Gabriel Isheden, Linda Abrahamsson, Therese Andersson, Kamila Czene, and Keith Humphreys. Joint models of tumour size and lymph node spread for incident breast cancer cases in the presence of screening. *Statistical*

Methods in Medical Research, 28(12):3822–3842, January 2019. ISSN 0962-2802. doi: 10.1177/0962280218819568. URL https://doi.org/10.1177/0962280218819568.

[17] Paul Joyce and Paul Marjoram. Approximately sufficient statistics and bayesian computation. *Statistical Applications in Genetics and Molecular Biology*, 7(1):Article26, 2008. ISSN 1544-6115. doi: 10.2202/1544-6115.1389.

[18] Andreas Karlsson, Mark S. Clements, and Alexandra Jauhiainen. A hybrid ABC approach for calibrating microsimulation models. Manuscript, 2019.

[19] Wentao Li and Paul Fearnhead. On the asymptotic efficiency of approximate Bayesian computation estimators. *Biometrika*, 105(2):285–299, June 2018. ISSN 0006-3444. doi: 10.1093/biomet/asx078. URL https://academic.oup.com/biomet/article/105/2/285/4818354.

[20] Jean-Michel Marin, Pierre Pudlo, Christian P. Robert, and Robin J. Ryder. Approximate Bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, November 2012. ISSN 1573-1375. doi: 10.1007/s11222-011-9288-2. URL https://doi.org/10.1007/s11222-011-9288-2.

[21] Paul Marjoram, John Molitor, Vincent Plagnol, and Simon Tavaré. Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324, December 2003. doi: 10.1073/pnas.0306899100. URL http://www.pnas.org/content/100/26/15324.abstract.

[22] Suresh H. Moolgavkar and Georg Luebeck. Two-Event Model for Carcinogenesis: Biological, Mathematical, and Statistical Considerations. *Risk Analysis*, 10(2):323–341, June 1990. ISSN 0272-4332. doi: 10.1111/j.1539-6924.1990.tb01053.x. URL https://doi.org/10.1111/j.1539-6924.1990.tb01053.x. Publisher: John Wiley & Sons, Ltd.

[23] Matthew A. Nunes and David J. Balding. On optimal selection of summary statistics for approximate Bayesian computation. *Statistical Applications in Genetics and Molecular Biology*, 9:Article34, 2010. ISSN 1544-6115. doi: 10.2202/1544-6115.1576.

[24] Caroline Olsson. Mammografiscreening, 2016. https://www.1177.se/Stockholm/behandling–hjalpmedel/undersokningar-och-provtagning/bildundersokningar-och-rontgen/mammografiscreening-i-stockholms-lan/ [Accessed: 2020-05-02].

[25] World Health Organization. Cancer, 2020. https://www.who.int/news-room/fact-sheets/detail/cancer [Accessed: 2020-05-02].

[26] Umberto Picchini. An intro to ABC – approximate Bayesian computation. Lecture slides, 2016.

[27] Sylvia K. Plevritis, Peter Salzman, Bronislava M. Sigal, and Peter W. Glynn. A natural history model of stage progression applied to breast cancer. *Statistics in Medicine*, 26(3):581–595, February 2007. ISSN 0277-6715. doi: 10.1002/sim.2550.

[28] Dennis Prangle. Adapting the ABC Distance Function. *Bayesian Analysis*, 12(1):289–309, March 2017. ISSN 1936-0975, 1931-6690. doi: 10.1214/16-BA1002. URL https://projecteuclid.org/euclid.ba/1460641065.

[29] Oliver Ratmann, Ole Jørgensen, Trevor Hinkley, Michael Stumpf, Sylvia Richardson, and Carsten Wiuf. Using Likelihood-Free Inference to Compare Evolutionary Dynamics of the Protein Networks of H. pylori and P. falciparum. *PLoS Computational Biology*, 3(11), November 2007. ISSN 1553-734X. doi: 10.1371/journal.pcbi.0030230. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2098858/.

[30] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer, New York, NY, 2004. ISBN 978-1-4419-1939-7.

[31] G. O. Roberts, A. Gelman, and W. R. Gilks. Weak convergence and optimal scaling of random walk Metropolis algorithms. *Annals of Applied Probability*, 7(1):110–120, February 1997. ISSN 1050-5164, 2168-8737. doi: 10.1214/aoap/1034625254. URL https://projecteuclid.org/euclid.aoap/1034625254. Publisher: Institute of Mathematical Statistics.

[32] Carolyn M Rutter, Diana L Miglioretti, and James E Savarino. Bayesian Calibration of Microsimulation Models. *Journal of the American Statistical Association*, 104(488):1338–1350, December 2009. ISSN 0162-1459. doi: 10.1198/jasa.2009.ap07466. URL https://www.ncbi.nlm.nih.gov/pubmed/20076767.

[33] Carolyn M. Rutter, Jonathan Ozik, Maria DeYoreo, and Nicholson Collier. Microsimulation model calibration using incremental mixture approximate Bayesian computation. *Ann. Appl. Stat.*, 13(4):2189–2212, December 2019. ISSN 1932-6157. doi: 10.1214/19-AOAS1279. URL https://projecteuclid.org:443/euclid.aoas/1574910041.

[34] J. A. Spratt, D. von Fournier, J. S. Spratt, and E. E. Weber. Decelerating growth and human breast cancer. *Cancer*, 71(6):2013–2019, March 1993. ISSN 0008-543X. doi: 10.1002/1097-0142(19930315)71:6⟨2013::aid-cncr2820710615⟩3.0.co;2-v.

[35] J. Rickard Strandberg and Keith Humphreys. Statistical models of tumour onset and growth for modern breast cancer screening cohorts. *Mathematical Biosciences*, 318:108270, 2019. ISSN 0025-5564. doi: https://doi.org/10.1016/j.mbs.2019.108270. URL http://www.sciencedirect.com/science/article/pii/S0025556419305115.

[36] K. H. X. Tan, L. Simonella, H. L. Wee, A. Roellin, Y.-W. Lim, W.-Y. Lim, K. S. Chia, M. Hartman, and A. R. Cook. Quantifying the natural history of breast cancer. *British journal of cancer*, 109(8):2035–2043, October 2013. ISSN 1532-1827 0007-0920 0007-0920. doi: 10.1038/bjc.2013.471.

[37] Alan R. Templeton. Coherent and incoherent inference in phylogeography and human evolution. *Proceedings of the National Academy of Sciences*, 107 (14):6376–6381, April 2010. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.0910647107. URL https://www.pnas.org/content/107/14/6376. Publisher: National Academy of Sciences Section: Biological Sciences.

[38] Robert Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. ISSN 0035-9246. URL https://www.jstor.org/stable/2346178. Publisher: [Royal Statistical Society, Wiley].

[39] Brandon M. Turner, Per B. Sederberg, Scott D. Brown, and Mark Steyvers. A Method for Efficiently Sampling From Distributions With Correlated Dimensions. *Psychological methods*, 18(3):368–384, September 2013. ISSN 1082-989X. doi: 10.1037/a0032222. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4140408/.

[40] Stefan Van Dongen. Prior specification in Bayesian statistics: Three cautionary tales. *Journal of Theoretical Biology*, 242(1):90–100, September 2006. ISSN 0022-5193. doi: 10.1016/j.jtbi.2006.02.002. URL http://www.sciencedirect.com/science/article/pii/S0022519306000609.

[41] Harald Weedon-Fekjær, Bo H Lindqvist, Lars J Vatten, Odd O Aalen, and Steinar Tretli. Breast cancer tumor growth estimated through mammography screening data. *Breast Cancer Research : BCR*, 10(3):R41, 2008. ISSN 1465-5411. doi: 10.1186/bcr2092. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2481488/.

[42] Harald Weedon-Fekjær, Steinar Tretli, and Odd O Aalen. Estimating screening test sensitivity and tumour progression using tumour size and time since previous screening. *Statistical Methods in Medical Research*, 19(5): 507–527, October 2010. ISSN 0962-2802. doi: 10.1177/0962280209359860. URL https://doi.org/10.1177/0962280209359860.

[43] Daniel Wegmann, Christoph Leuenberger, and Laurent Excoffier. Efficient Approximate Bayesian Computation Coupled With Markov Chain Monte Carlo Without Likelihood. *Genetics*, 182(4):1207–1218, August 2009. ISSN 0016-6731. doi: 10.1534/genetics.109.102509. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2728860/.

[44] Richard David Wilkinson. Approximate Bayesian computation (ABC) gives exact results under the assumption of model error. *Statistical Applications in Genetics and Molecular Biology*, 12(2):129–141, 2013. ISSN 1544-6115. doi: 10.1515/sagmb-2013-0010. URL https://www.degruyter.com/view/j/sagmb.2013.12.issue-2/sagmb-2013-0010/sagmb-2013-0010.xml.

A

Algorithm diagnostics

The ABC-MCMC algorithm for exponential tumour growth in absence of screening had a final acceptance rate of 0.0354. The mixing is presented in Figure A.1. Note how the mixing is initially very bad, to later converge. This behaviour stems from the adaptive tolerance, where initially simulated data far from the observed is accepted. However, using an adaptive tolerance the initial guess can be further away from the mode of the posterior than otherwise.

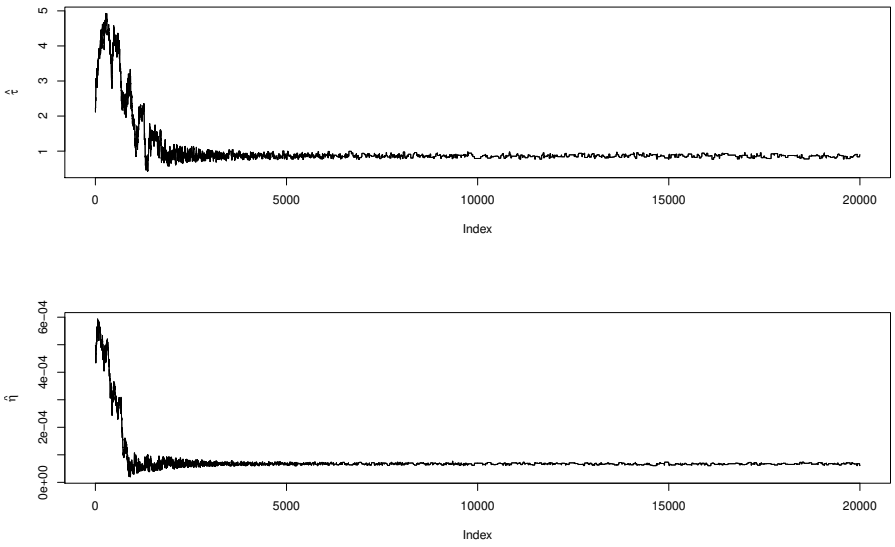

Figure A.1: ABC-MCMC mixing for the exponential tumour growth model in absence of screening.

For the exponential tumour growth model, the Metropolis Hastings MCMC algorithm had a final acceptance rate of 0.255. In Figure A.2 the mixing and ACF plots are presented, indicating good performance. We can thus rely on the Metropolis Hastings MCMC estimates when evaluating the ABC-MCMC performance.

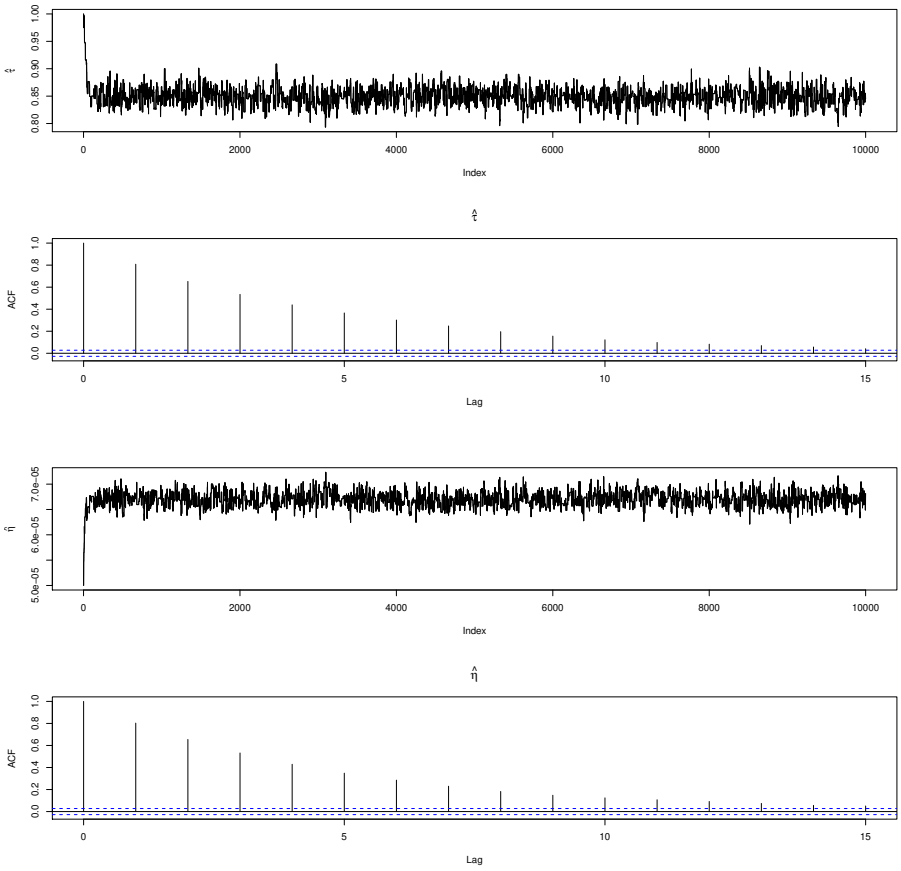

Figure A.2: Metropolis Hastings MCMC mixing and ACF plots for the exponential tumour growth model in absence of screening. These plots indicate that the performance of the algorithm is good, and that samples are indeed generated from the posterior distribution.

For the logistic tumour growth model in absence of screening, ABC-MCMC had a final acceptance rate of 0.0762. The mixing is displayed in Figure A.3. The plots indicate a good performance, which is also suggested by the simulation study.

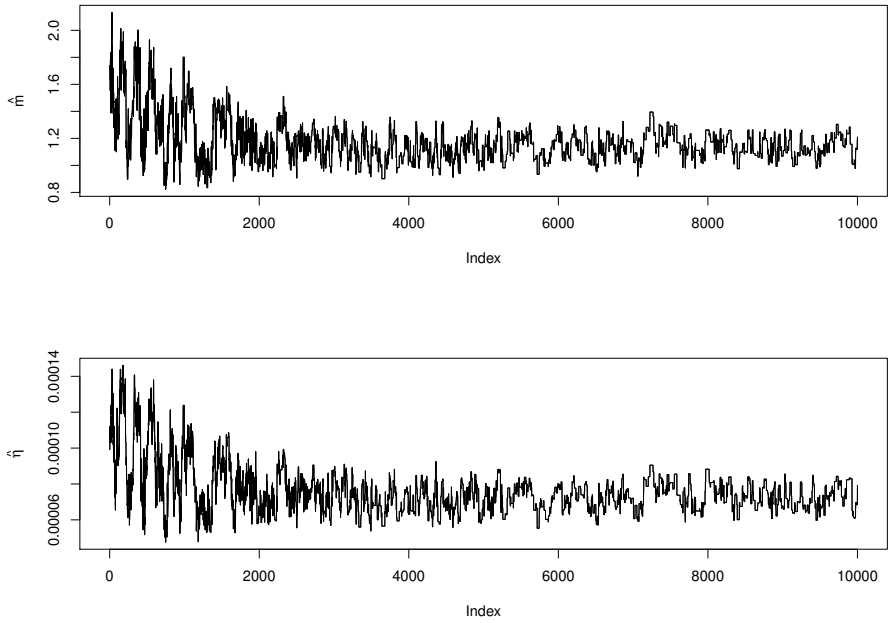

Figure A.3: ABC mixing for the logistic tumour growth model in absence of screening.

MCMC for the logistic tumour growth model in absence of screening had a final acceptance rate of 0.166. In Figure A.4 mixing and ACF plots are presented.

For the exponential random effects natural history model in presence of screening, mixing plots for the parameters τ_1, τ_2 and η are displayed in Figure A.5 and for the parameters β_1 and β_2 in Figure A.6. The plots indicate decent performance for τ_1 and τ_2, and better performance for the other parameters. This is supported by the findings of the simulation study.

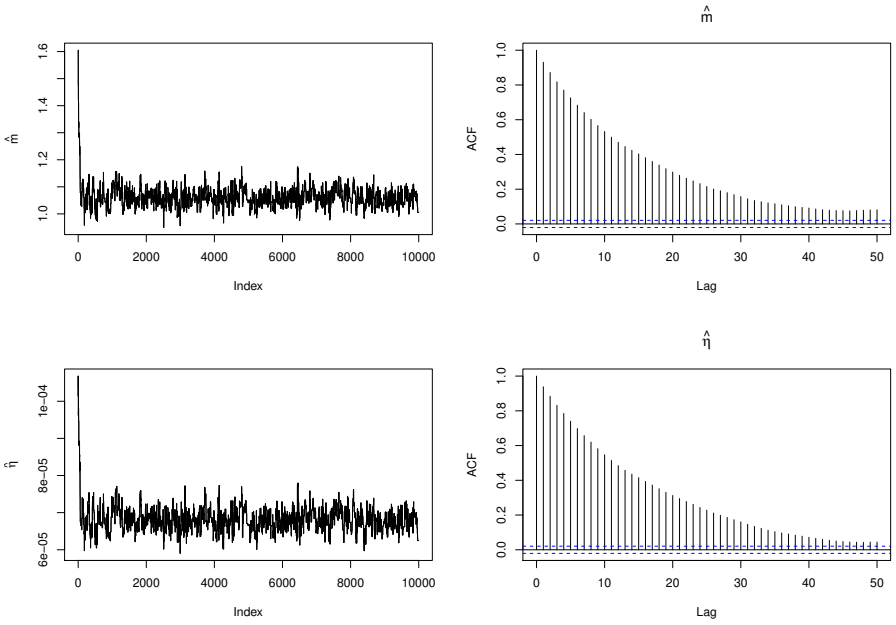

Figure A.4: MCMC mixing and ACF plots for the logistic tumour growth model in absence of screening. The final acceptance rate was 0.166. The mixing is not perfect but still indicates a higher accuracy than ABC-MCMC.

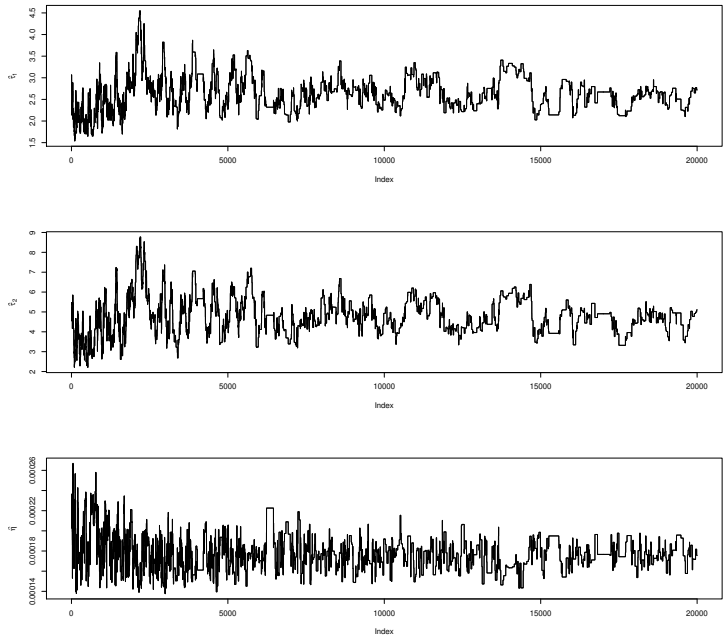

Figure A.5: ABC-MCMC mixing for the exponential natural history model in presence of screening. The mixing is for the parameters τ_1, τ_2 and η.

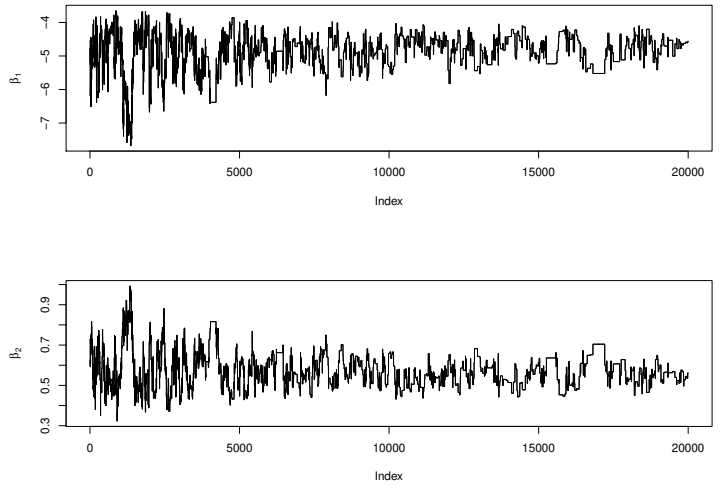

Figure A.6: ABC-MCMC mixing for the exponential natural history model in presence of screening. The mixing is for the parameters β_1 and β_2

B

The likelihood of the logistic growth model in absence of screening

In this appendix we will derive an integral expression for the likelihood function of the logistic growth model for data in absence of screening. We will also describe how it can be approximated numerically. In the process we will derive closed form expressions of the probability density and cumulative density functions of tumour volume at symptomatic detection V_{sd} conditional on inverse growth rate $R = r$.

In Plevritis et al. [27] the cdf of volume at symptomatic detection conditional on the inverse growth rate $R = r$ is derived as:

$$P(V_{sd} < x | R = r) = 1 - \exp\left(- \int_0^{V^{-1}(x)} \eta V(t) dt \right). \tag{B.1}$$

Using the function for logistic tumour growth with inverse growth rate $R = r$

$$V(t) = \frac{V_{max}}{\left[1 + \left((V_{max}/V_{cell})^{0.25} - 1 \right) \exp(-0.25rt) \right]^4} \tag{B.2}$$

and defining

$$a := (V_{max}/V_{cell})^{0.25} - 1 \tag{B.3}$$

$$b := 0.25r \tag{B.4}$$

$$c := V_{max} \tag{B.5}$$

$$I = \int_0^{V^{-1}(x)} \frac{1}{(1 + ae^{-bt})^4} dt \tag{B.6}$$

we get the inverse of the growth function

$$V^{-1}(x) = -\frac{1}{b} \log \left(\frac{1}{a} \left[\left(\frac{x}{c} \right)^{-0.25} - 1 \right] \right) \tag{B.7}$$

and the conditional cdf

$$P(V_{sd} < x|R = r) = 1 - \exp\left(-\eta c I\right). \tag{B.8}$$

The integral I is computed using WolframAlpha

$$\begin{aligned} I =& \frac{11 + \frac{27}{\left(\frac{x}{c}\right)^{-0.25}-1} + \frac{18}{\left[\left(\frac{x}{c}\right)^{-0.25}-1\right]^2}}{6b\left[1 + \frac{1}{\left(\frac{x}{c}\right)^{-0.25}-1}\right]^3} + \frac{1}{b}\log\left(a + \frac{a}{\left(\frac{x}{c}\right)^{-0.25} - 1}\right) \\ &- \frac{11a^3 + 27a^2 + 18a}{6b(a+1)^3} - \frac{1}{b}\log(a+1). \end{aligned} \tag{B.9}$$

Simplifying the expression we get:

$$\begin{aligned} I = \frac{1}{b}\Bigg[&\log\left(\frac{a}{1 - \left(\frac{x}{c}\right)^{0.25}}\right) - \log(a+1) - \frac{1}{3}\left(\frac{x}{c}\right)^{0.75} - \frac{1}{2}\left(\frac{x}{c}\right)^{0.5} - \\ &- \left(\frac{x}{c}\right)^{0.25} - \frac{11a^3 + 27a^2 + 18a}{6b(a+1)^3} + \frac{11}{6}\Bigg]. \end{aligned} \tag{B.10}$$

Defining the x dependent (but not r dependent) function $M(x)$ as

$$\begin{aligned} M(x) := &\log\left(\frac{a}{1 - \left(\frac{x}{c}\right)^{0.25}}\right) - \log(a+1) - \frac{1}{3}\left(\frac{x}{c}\right)^{0.75} - \frac{1}{2}\left(\frac{x}{c}\right)^{0.5} - \\ &- \left(\frac{x}{c}\right)^{0.25} - \frac{11a^3 + 27a^2 + 18a}{6b(a+1)^3} + \frac{11}{6} \end{aligned} \tag{B.11}$$

we can write the cumulative density function as:

$$P(V_{sd} < x|R = r) = 1 - \exp\left(-\frac{\eta c}{b} M(x)\right). \tag{B.12}$$

Taking the derivative of the cdf with respect to x we obtain the pdf:

$$f_{V_{sd}|R=r}(x) = \frac{\eta c}{b}\exp\left(-\frac{\eta c}{b}M(x)\right)\frac{dM(x)}{dx}. \tag{B.13}$$

The derivative is calculated using WolframAlpha:

$$\frac{dM(x)}{dx} = \frac{1}{4c\left[1 - \left(\frac{x}{c}\right)^{0.25}\right]}. \tag{B.14}$$

This gives

$$\begin{aligned} f_{V_{sd}|R=r}(x) &= \frac{\eta}{4b\left[1 - \left(\frac{x}{c}\right)^{0.25}\right]}\exp\left(-\frac{\eta c}{b}M(x)\right) \\ &= \frac{\eta}{r\left[1 - \left(\frac{x}{c}\right)^{0.25}\right]}\exp\left(-\frac{4\eta c}{r}M(x)\right) \\ &= \frac{\eta}{r}A(x)B(x)^{\eta/r} \end{aligned} \tag{B.15}$$

where we have defined

$$\begin{aligned} A(x) &= \frac{1}{1-\left(\frac{x}{c}\right)^{0.25}} \\ B(x) &= e^{-4cM(x)}. \end{aligned} \tag{B.16}$$

In order to get the likelihood function of volume at symptomatic detection, we need to marginalise out the growth rate:

$$f_{V_{sd}}(x) = \int_0^\infty f_{V_{sd}|R=r}(x,r) f_R(r) dr. \tag{B.17}$$

Log-normal growth rate with parameters μ and σ gives

$$f_R(r) = \frac{1}{r\sigma\sqrt{2\pi}} \exp\left(-\frac{(\log(r)-\mu)^2}{2\sigma^2}\right). \tag{B.18}$$

Thus we have

$$f_{V_{sd}}(x) = \frac{\eta A(x)}{\sigma\sqrt{2\pi}} \int_0^\infty \frac{1}{r^2} B(x)^{\eta/r} \exp\left(-\frac{(\log(r)-\mu)^2}{2\sigma^2}\right) dr. \tag{B.19}$$

An attempt was made to solve this integral using WolframAlpha, but it could not find any analytical solutions. Instead, the integral can be numerically approximated using the built in *integrate* function in R. Since the tumour volumes at detection for different individuals are assumed to be independent, we get the joint likelihood function for N individuals:

$$L_n(\vec{\theta}) = \prod_{i=1}^{N} f_{V_{sd}^i}(x_i). \tag{B.20}$$

To find the ML-estimates, the built in R function *optim* was used. In order to enable unconstrained optimisation via the standard *Nelder-Mead* algorithm, an exponential transformation was used for η. Details can be found in the R code in Appendix C.2.

C

R-code

C.1 Exponential tumour growth model in absence of screening

The R implementations of ABC-MCMC, Metropolis Hastings MCMC and ML estimation of the exponential tumour growth model in absence of screening are presented here. Please note that all functions are in the end of the code. The code was built for R version 3.6.1.

```
### OBS FUNCTIONS IN THE END ###

### Libraries ###
library("pracma") # mod, size functions
library("tseries") # ACF plots for MCMC diagnostics


############## DATA GENERATION (synthetic data) #######
set.seed(1)
n ← 10000   # Number of observations
tau ← exp(-0.165)
gammavar ← exp(-9.602)
y ← M_exp(n, tau, gammavar)
# Summary stats: counts in each category
sobs ← summary_stats(y)


################## ML-estimates ####################
set.seed(1)
start_time ← Sys.time()
n ← 10000
tau ← exp(-0.165)
gammavar ← exp(-9.602)
w ← c(tau, gammavar)
y ← M_exp(n, w[1], w[2])
opt ← optim(w, Ln_exp, gr=NULL,
              vcd=y, negative=TRUE, method="L-BFGS-B",
```

```
28              lower=c(0.01,0.000001), upper=c(2,1))
29 ML_time ← Sys.time() - start_time
30 print(ML_time)
31 print(opt$par)
32
33
34 ###################### ABC-MCMC  ######################
35 # Euclidean distance metric weighted by sd estimates
36 # Tolerance tempering
37 # Normal transition kernel with sd = rho
38
39 # V[s] estimates #
40 set.seed(1)
41 n_mad ← 1000
42 smad ← numeric(24)
43 tmp ← matrix(0,n_mad,24)
44 for(i in 1:n_mad){
45   y ← M_exp(n, tau, gammavar)
46   tmp[i,] ← summary_stats(y)
47 }
48 smad ← apply(tmp, 2, sd)
49
50 # Hyperparameters (adjust these)
51 M ← 3000                      # Number of iterations
52 burn_in ← 2500                # Burn in
53 trans_cv ← c(0.5, 0.5)        # transition kernel var
54 w ← 1.5*c(tau, gammavar)
55 epsilon ← 2.0                  # Tolerance
56 quantile ← 90                  # Quantile for tempering
57 final_acc_rate ← 0.01          # Final accetance rate
58
59 # Initialization (do not change)
60 set.seed(0)
61 start_time ← Sys.time()
62 d ← length(w)                      # Number of parameters
63 n_stat ← length(sobs)              # Number of statistics
64 w_post ← matrix(NaN,M,d)           # stores postrior values
65 ssim ← numeric(n_stat)
66 ssim_mat ← matrix(NaN,M,n_stat) # Stores simulated stats
67 sdist_vec ← rep(NaN, M)
68 n_accABC ← 0
69 acc_vec ← numeric(M)
70 acc_rate ← 1
71 rho ← trans_cv*w        # sd of transition kernel
72
73 for(iter in 1:M){
74
75   if(mod(iter,1000)==0){
76     print(iter)
77     acc_rate ← mean(acc_vec[(iter-999):(iter-1)])
```

```
78    }
79
80    # Sample from the transition kernel
81    wp ← rnorm(d, mean=w, sd=rho)
82
83    # Simulate from the MODEL using the proposed parameter
          values
84    y_sim ← M_exp(n, wp[1], wp[2])
85
86    # Calculate sufficient statistics from the samples
87    ssim ← summary_stats(y_sim)
88
89    # Caclulate normalized euclidian distence between observed
          and simulated statistics
90    sdist ← sqrt(sum(((ssim-sobs)/smad)^2))/n_stat
91
92    # If the distance between the data statistics and the
          simulated statistics
93    # is bigger then epsilon, jump to next iteration
94    if(sdist ≥ epsilon){
95      w_post[iter,] ← w
96      acc_vec[iter] ← 0
97      next
98    }
99
100   # Epsilon is set to the q:th quantile of accepted errors
101   # every 100th iteration
102   n_accABC ← n_accABC + 1
103   ssim_mat[iter,] ← ssim
104   sdist_vec[n_accABC] ← sdist
105   if(mod(n_accABC, 100) == 0 && acc_rate > final_acc_rate){
106     sdist_sorted ← sort(sdist_vec[(n_accABC-99):n_accABC])
107     epsilon ← sdist_sorted[quantile]
108     cat(iter, "", epsilon, " \n") #Print the epsilon values
109   }
110
111   prior_ratio ← 1
112
113   if(is.nan(prior_ratio)){
114     warning("Prior ratio: NAN")
115   }
116   if(is.infinite(prior_ratio)){
117     warning("Prior ratio: INF")
118   }
119
120   alpha ← min(1, prior_ratio) # MH acceptence probability
121
122   U ← runif(1)
123   if(alpha > U){
124     w_post[iter,] ← wp
```

```
125     w ← wp
126     acc_vec[iter] ← 1
127   } else {
128     w_post[iter,] ← w
129     acc_vec[iter] ← 0
130   }
131 }
132 w_post_ABC ← w_post
133 abc_time ← Sys.time() - start_time
134 print("ABC-MCMC time")
135 print(abc_time)
136
137 # MIXING PLOTS
138 par(mfrow=c(d,1))
139 plot(w_post[,1],type='l', ylab=expression(hat(tau)))
140 lines(w_post[,1])
141 plot(w_post[,2],type='l', ylab=expression(hat(eta)))
142 lines(w_post[,2])
143
144 # POSTERIOR MEANS #
145 w_mean ← colMeans(w_post[burn_in:M,])
146 print("ABC-MCMC acceptance rate:")
147 print(mean(acc_vec[burn_in:M]))
148 print("ABC-MCMC means:")
149 print(w_mean)
150 print("ABC-MCMC mean relative error in %")
151 print((w_mean-c(tau,gammavar))/c(tau,gammavar)*100)
152
153 # LINEAR REGRESSION ADJUSTMENT  #
154 w_post_adj ← vector(mode="list", length=d)
155 w_mean_adj ← numeric(d)
156 for(i in 1:d){
157   lm_data = data.frame("w"=w_post[burn_in:M,i], ssim_mat[
        burn_in:M,])
158   lm_tmp ← lm(w ~ ., data = lm_data, na.action = na.omit)
159   w_post_adj[[i]] ← sum(lm_tmp$coefficients * c(1,sobs), na.
        rm=TRUE) + lm_tmp$residuals
160   w_mean_adj[i] ← mean(w_post_adj[[i]])
161 }
162 print("ABC-MCMC regression adjusted means:")
163 print(w_mean_adj)
164 print("ABC-MCMC adjusted mean relative error in %")
165 print((w_mean_adj-c(tau,gammavar))/c(tau,gammavar)*100)
166
167 ################ Metropolis Hastings MCMC ##################
168 # Normal transition kernel with sd = rho
169 set.seed(0)
170 start_time ← Sys.time()
171
172 # Hyperparameters (adjust these)
```

```
M ← 10000                    # Number of iterations
burn_in ← 5000               # Burn in
trans_cv ← c(0.03, 0.03)     # transition kernel sd
w ← c(1, 0.00005)            # initial parameters

# Initialization (do not change)
d ← length(w)              # Number of parameters
w_post ← matrix(NaN,M,d) # stores posterior samples
acc_vec ← numeric(M)
rho ← trans_cv*w           # sd of transition kernel

for(iter in 1:M){

  if(mod(iter,1000)==0){
    print(iter)
  }

  # Sample from the transition kernel
  wp ← rnorm(d, mean=w, sd=rho)

  # Uninformative (improper) flat prior
  prior_ratio ← 1

  # Exp growth model Likelihood ratio
  Ln_wp ← Ln_exp(wp, y)
  Ln_w  ← Ln_exp(w, y)
  likelihood_ratio ← exp(Ln_wp - Ln_w)
  if(is.nan(likelihood_ratio)){
    message(iter, ": Likelihood ratio: NaN")
  }
  if(is.infinite(likelihood_ratio)){
    message(iter, ": Likelihood ratio: INF")
  }

  alpha ← prior_ratio*likelihood_ratio
  alpha ← min(1, alpha, na.rm=TRUE)  # MH acceptence
      probability

  U ← runif(1)
  if(alpha > U){
    w_post[iter,] ← wp
    w ← wp
    acc_vec[iter] ← 1
  } else {
    w_post[iter,] ← w
    acc_vec[iter] ← 0
  }
}
w_MCMC ← w_post
MCMC_time ← Sys.time() - start_time
```

```
222 print("MCMC time")
223 print(MCMC_time)
224
225 # MCMC mixing plots #
226 par(mfrow=c(4,1))
227 plot(w_post[,1],type='l', ylab=expression(hat(tau)))
228 lines(w_post[,1])
229 acf(w_post[burn_in:M,1], lag=15, main=expression(hat(tau)))
230 plot(w_post[,2],type='l', ylab=expression(hat(eta)))
231 lines(w_post[,2])
232 acf(w_post[burn_in:M,2], lag=15, main=expression(hat(eta)))
233
234 # MCMC posterior means #
235 w_mean <- colMeans(w_post[burn_in:M,])
236 print("ABC-MCMC acceptance rate:")
237 print(mean(acc_vec[burn_in:M]))
238 print("ABC-MCMC means:")
239 print(w_mean)
240
241
242 ################# POSTERIOR PLOTS (ALL) ##################
243 par(mfrow=c(2,3))
244 plot(density(w_post_ABC[10000:M,1], kernel="epanechnikov",
245     adjust=1.4), xlab=expression(paste(hat(m))), main="ABC")
246 plot(density(w_post_adj[[1]], kernel="epanechnikov",
247     adjust=1.0), xlab=expression(paste(hat(m))), main="
        Adjusted ABC")
248 plot(density(w_MCMC[2000:M,1], kernel="epanechnikov",
249     adjust=1.2), xlab=expression(paste(hat(m))), main="MCMC"
        )
250 plot(density(w_post_ABC[10000:M,3], kernel="epanechnikov",
    adjust=1.4), xlab=expression(paste(hat(eta))), main="ABC"
    )
251 plot(density(w_post_adj[[3]], kernel="epanechnikov",
252     adjust=1.0), xlab=expression(paste(hat(eta))), main="
        Adjusted ABC")
253 plot(density(w_MCMC[2000:M,2], kernel="epanechnikov",
254     adjust=1.2), xlab=expression(paste(hat(eta))), main="
        MCMC")
255
256
257
258 ################## FUNCTIONS #################
259
260 M_exp <- function(nobs, tau, gammavar){
261   tau1 <- tau
262   tau2 <- tau
263   gam  <- gammavar
264
```

```
265   grr<-rgamma(nobs,tau1,rate=tau2)  # Gamma inverse growth
          rates
266   u<-runif(nobs,0,1)
267   v0<-(0.5^3)*pi/6
268   vcd<-v0-(log(1-u))/(gam*grr)
269
270   return(vcd)
271 }
272
273 Ln_exp <- function(w, vcd, negative=FALSE){
274   if(any(w<=0))
275     return(-Inf)
276   tau <- w[1]
277   tau1 <- tau
278   tau2 <- tau
279   gammavar <- w[2]
280   v_cell<-(0.5^3)*pi/6
281   if(negative)
282     L <- -sum(log(tau1)+tau1*log(tau2)+log(gammavar)+(-(tau1
          +1))*log(tau2+gammavar*(vcd-v_cell)))
283   else
284     L <- sum(log(tau1)+tau1*log(tau2)+log(gammavar)+(-(tau1
          +1))*log(tau2+gammavar*(vcd-v_cell)))
285   return(L)
286 }
```

C.2 Logistic tumour growth model in absence of screening

The R code for ABC-MCMC, Metropolis Hastings MCMC and ML estimation of the logistic tumour growth model in absence of screening is presented here. Please note that all functions are in the end of the code. The code was built for R version 3.6.1.

```
### OBS ALL FUNCTIONS IN THE END ###
### Libraries ###
library("pracma")  # mod, size functions
library("tseries") # ACF plots for MCMC


############ DATA GENERATION (synthetic data) #########
set.seed(1)
start_time <- Sys.time()
n <- 10000  # Number of individuals
m <- 1.07
v <- exp(log(1.31)/1.07*m)
gammavar <- exp(-9.602) # eta (old name)
y <- M_log(n, m, v, gammavar)
print(Sys.time()-start_time)
# Summary stats: counts in each category
sobs <- summary_stats(y)
print(sobs)


############### ML estimates #################
start_time <- Sys.time()
opt <- optim(c(0.5, -4), fn=Ln_log, x=y, opt=TRUE)
w_ML <- c(opt$par[1], 10^(opt$par[2]))
ML_time <- Sys.time() - start_time
print(ML_time)
print(w_ML)


#################### ABC-MCMC ######################
# Euclidian distance weighted by V[s]
# Adaptive adjustment of tolerance
# Normal transition kernel with sd = rho

### Estimation of V[s] ###
set.seed(0)
start_time <- Sys.time()
n_mad <- 200
tmp <- matrix(0,n_mad,24)
for(i in 1:n_mad){
```

```
  y_mad ← M_log(n, m, v, gammavar)
  tmp[i,] ← summary_stats(y_mad)
}
smad ← apply(tmp, 2, sd)
print(Sys.time()-start_time)
set.seed(0)
start_time ← Sys.time()

# Hyperparameters (adjust these)
M ← 10000                       # Number of iterations
burn_in ← 5000                  # Burn in
trans_cv ← c(0.1, 0, 0.1)       # transition kernel var
w ← 1.5*c(m, v, gammavar)       # initial parameters
epsilon ← 0.63                  # tolerance
quantile ← 0.9          # Quantile for tol. tempering
final_acc_rate ← 0.01   # Stop crit. for tempering
tempering_freq ← 100    # Decay tol every 100 acc. iter.

# Initialization (do not change)
d ← length(w)
n_stat ← length(sobs)
w_post ← matrix(NaN,M,d)         # stores postrior values
ssim ← numeric(n_stat)
ssim_mat ← matrix(NaN,M,n_stat) # Stores simul statistics
sdist_vec ← rep(NaN, M)          # Stores statistic distances
n_accABC ← 0
acc_vec ← numeric(M)
acc_rate ← 1
rho ← abs(trans_cv*w)            # sd of transition kernel

for(iter in 1:M){

  if(mod(iter,100)==0){
    print(iter)
  }
  # Plot of w over iterations (mixing)
  if(mod(iter,5)==0){
    par(mfrow=c(d,1))
    for(i in 1:d){
      plot(w_post[,i],type='l')
      lines(w_post[,i])
    }
  }
  if(mod(iter,500)==0){
    print(iter)
    acc_rate ← mean(acc_vec[(iter-499):(iter-1)])
  }
  # Sample from the transition kernel
  wp ← rnorm(d, mean=w, sd=rho)
  if(any(is.nan(wp)) | wp[2]≤0
```

```
91      | wp[3]≤0 | any(is.infinite(wp))){
92      message(cat("WARNING: ", wp))
93      w_post[iter,] ← w
94      acc_vec[iter] ← 0
95      next
96    }
97    # Simulate from the MODEL using the proposed params
98    y_sim ← M_log(n, wp[1], wp[2], wp[3])
99
100   # Calculate sufficient statistics from the samples
101   ssim ← summary_stats(y_sim)
102
103   # Normalized euclidean distance between observed and
104   # simulated statistics
105   sdist ← sqrt(sum(((ssim-sobs)/smad)^2))/n_stat
106
107   # If the dist between the data stats and the simul stats
108   # is bigger then epsilon, jump to next iter
109   if(sdist ≥ epsilon){
110     w_post[iter,] ← w
111     acc_vec[iter] ← 0
112     next
113   }
114
115   # Epsilon is set to the q:th quantile of accepted errors
116   # every 100th iteration
117   n_accABC ← n_accABC + 1
118   ssim_mat[iter,] ← ssim
119   sdist_vec[n_accABC] ← sdist
120   if(mod(n_accABC, tempering_freq)==0
121                 && acc_rate>final_acc_rate){
122     epsilon ← quantile(sdist_vec[(n_accABC-(
123                     tempering_freq-1)):n_accABC],
124                     quantile, names=FALSE)
125     cat(iter, "", epsilon, " \n") #Print the epsilon values
126   }
127
128   # Uninformative (improper) flat prior
129   prior_ratio ← 1
130
131   alpha ← min(1, prior_ratio) # MH acceptance probability
132   U ← runif(1)
133   if(alpha > U){
134     w_post[iter,] ← wp
135     w ← wp
136     acc_vec[iter] ← 1
137   } else {
138     w_post[iter,] ← w
139     acc_vec[iter] ← 0
140   }
```

```
}
abc_time <- Sys.time() - start_time
print("ABC-MCMC time")
print(abc_time)

# ABC MIXING PLOTS #
par(mfrow=c(2,1))
plot(w_post[0:M,1],type='l', ylab=expression(hat(m)))
lines(w_post[0:M,1], ylab=expression(hat(m)))
plot(w_post[0:M,3],type='l', ylab=expression(hat(eta)))
lines(w_post[0:M,3], ylab=expression(hat(eta)))

# ABC POSTERIOR MEANS #
burn_in <- 4000
w_mean <- colMeans(w_post[burn_in:M,])
print("ABC-MCMC acceptance rate:")
print(acc_rate)
print("ABC-MCMC means:")
print(w_mean)
print("ABC-MCMC mean relative error in %")
print((w_mean-c(m,v*1.5,gammavar))/c(m,v,gammavar)*100)

### LINEAR REGRESSION ADJUSTMENT ###
w_post_adj <- vector(mode="list", length=d)
w_mean_adj <- numeric(d)
for(i in c(1,3)){
  lm_data = data.frame("w"=w_post[burn_in:M,i], ssim_mat[
      burn_in:M,])
  lm_tmp <- lm(w ~ ., data = lm_data, na.action = na.omit)
  w_post_adj[[i]] <- sum(lm_tmp$coefficients * c(1,sobs), na.
      rm=TRUE) + lm_tmp$residuals
  w_mean_adj[i] <- mean(w_post_adj[[i]])
}
print("ABC-MCMC regression adjusted means:")
print(w_mean_adj)
print("ABC-MCMC adjusted mean relative error in %")
print((w_mean_adj-c(m,v*1.5,gammavar))/c(m,v,gammavar)*100)


############### Metropolis Hastings MCMC #################
# Normal transition kernel with sd = rho
set.seed(0)
start_time <- Sys.time()

# Hyperparameters (adjust these)
M <- 10000                    # Number of iterations
burn_in <- 5000               # Burn in
trans_cv <- c(0.05, 0.05)     # transition kernel var
w <- 1.5*c(m, gammavar)       # initial parameters

```

```
# Initialization (do not change)
d ← length(w)                 # Number of parameters
w_post ← matrix(NaN,M,d)      # stores postrior values
acc_vec ← numeric(M)
rho ← trans_cv*c(m,gammavar) # sd of transition kernel

for(iter in 1:M){

  if(mod(iter,10)==0){
    print(iter)
  }
  # Mixing
  if(mod(iter,10)==0){
    par(mfrow=c(d,1))
    for(i in 1:d){
      plot(w_post[,i],type='l')
      lines(w_post[,i])
    }
  }
  # Sample from the transition kernel
  wp ← rnorm(d, mean=w, sd=rho)

  prior_ratio ← 1 # Flat prior

  # Exp growth model Likelihood ratio
  Ln_wp ← Ln_log(wp, y)
  Ln_w  ← Ln_log(w, y)
  likelihood_ratio ← exp(Ln_w - Ln_wp)
  if(is.nan(likelihood_ratio)){
    message(iter, ": Likelihood ratio: NaN")
  }
  if(is.infinite(likelihood_ratio)){
    message(iter, ": Likelihood ratio: INF")
  }

  alpha ← prior_ratio*likelihood_ratio
  alpha ← min(1, alpha, na.rm=TRUE)  # MH acceptance prob

  U ← runif(1)
  if(alpha > U){
    w_post[iter,] ← wp
    w ← wp
    acc_vec[iter] ← 1
  } else {
    w_post[iter,] ← w
    acc_vec[iter] ← 0
  }
}
w_MCMC ← w_post
MCMC_time ← Sys.time() - start_time
```

```
239 print("MCMC time")
240 print(MCMC_time)
241
242 # MCMC MIXING PLOTS #
243 par(mfrow=c(2,2))
244 plot(w_post[,1],type='l', ylab=expression(hat(m)))
245 lines(w_post[,1])
246 acf(w_post[burn_in:M,1], lag=50, main=expression(hat(m)))
247 plot(w_post[,2],type='l', ylab=expression(hat(eta)))
248 lines(w_post[,2])
249 acf(w_post[burn_in:M,2], lag=50, main=expression(hat(eta)))
250 w_MCMC <- w_post
251
252 # MCMC POSTERIOR MEAN #
253 w_mean_MCMC <- colMeans(w_post[burn_in:M,])
254 print("ABC-MCMC acceptance rate:")
255 print(mean(acc_vec[burn_in:M]))
256 print("ABC-MCMC means:")
257 print(w_mean_MCMC)
258
259
260 ########## ALL POSTERIOR PLOTS ############
261 par(mfrow=c(2,3))
262 par(cex.axis=1.3, cex.lab=1.5)
263 plot(density(w_post_ABC[4000:M,1], kernel="epanechnikov",
264     adjust=1.4), xlab=expression(paste(hat(m))), main="ABC")
265 plot(density(w_post_adj[[1]], kernel="epanechnikov",
266     adjust=1.0), xlab=expression(paste(hat(m))), main="
        Adjusted ABC")
267 plot(density(w_MCMC[2000:M,1], kernel="epanechnikov",
268     adjust=1.2), xlab=expression(paste(hat(m))), main="MCMC"
        )
269 plot(density(w_post_ABC[4000:M,3], kernel="epanechnikov",
270     adjust=1.4), xlab=expression(paste(hat(eta))), main="ABC
        ")
271 plot(density(w_post_adj[[3]], kernel="epanechnikov",
272     adjust=1.0), xlab=expression(paste(hat(eta))), main="
        Adjusted ABC")
273 plot(density(w_MCMC[2000:M,2], kernel="epanechnikov",
274     adjust=1.2), xlab=expression(paste(hat(eta))), main="
        MCMC")
275
276
277 ################  FUNCTIONS  ######################
278 M_log <- function(nobs, m, v, gammavar){
279   v <- exp(log(1.31)/1.07*m)
280   location <- log(m^2 / sqrt(v + m^2))
281   shape <- sqrt(log(1 + (v / m^2)))
282   gr <- rlnorm(n=nobs, location, shape)
283   # Numerical inverse sampling
```

```
284   volcd ← apply(as.array(gr),
285                 MARGIN=1, FUN=inv_sampl,
286                 gammavar=gammavar)
287   return(volcd)   # Volumes at symptomatic detection
288 }
289
290 inv_sampl ← function(r, gammavar){
291   u ← runif(1)
292   root ← uniroot(f=vsd_cdf, interval=c(0, vmax), r, gammavar
          , u)$root
293   return(root)
294 }
295
296 summary_stats ← function(vcd, N=24){
297   gupperbounds←c(1.5,linspace(2.5,46,17), 48, 50, 55, 70,
          90, 128)
298   glowerbounds←c(0.5,linspace(1.5,45,17), 46, 48, 50, 55,
          70, 90)
299   gsizeintervals←length(gupperbounds)
300   gmid←(gupperbounds+glowerbounds)/2
301   gmid←(gupperbounds*glowerbounds)^(1/2)
302   nobs ← length(vcd)
303   dcd←(6*vcd/pi)^(1/3)
304   gsize←matrix(-1 ,nobs,1)
305   for(s in 1:gsizeintervals){
306     gsize[dcd ≥ glowerbounds[s] & dcd < gupperbounds[s]] ← s
307   }
308   stat ← numeric(24)
309   for(i in 1:24){
310     stat[i] ← sum(gsize == i)
311   }
312   return(stat)
313 }
314
315 # pdf of V_sd|R=r
316 vsd_pdf ← function(r, x, gammavar){
317   vcell←(0.5^3)*pi/6
318   vmax←(128^3)*pi/6
319   a ← (vmax/vcell)^0.25 - 1
320   b ← 0.25*r
321   c ← vmax
322   M←numeric(length(x))
323   dM_dx←numeric(length(x))
324   if(all(x>0) && all(x<vmax)){
325     M ← (log(a/(1-(x/c)^(0.25))) - log(a+1)
326           - 1/3*(x/c)^(0.75) - 1/2*(x/c)^(0.5) - (x/c)^
                (0.25)
327           - (11*a^3+27*a^2+18*a)/(6*(a+1)^3) + 11/6)
328
329     dM_dx ← 0.25/(c*(1-(x/c)^(0.25)))
```

```
330   }
331   return(gammavar*c/b*exp(-gammavar*c/b*M)*dM_dx*as.numeric(
          r>0)*as.numeric(gammavar>0))
332 }
333
334 # cdf of V_sd|R=r
335 vsd_cdf <- function(x, r, gammavar, u=0){
336   n <- length(r)
337   vcell<-(0.5^3)*pi/6
338   vmax<-(128^3)*pi/6
339   a <- (vmax/vcell)^0.25 - 1
340   b <- 0.25*r
341   c <- vmax
342   cdf<-numeric(n)-u
343   if(x>0 && x<vmax){
344     M <- (log(a/(1-(x/c)^(0.25))) - log(a+1)
345           - 1/3*(x/c)^(0.75) - 1/2*(x/c)^(0.5) - (x/c)^
                (0.25)
346           - (11*a^3+27*a^2+18*a)/(6*(a+1)^3) + 11/6)
347     cdf <- (1-exp(-gammavar*c/b*M))*as.numeric(r>0)*as.
          numeric(gammavar>0)-u
348   } else if(x>=vmax) {
349     cdf <- 1-u
350   }
351   return(cdf)
352 }
353
354 L_integr <- function(r, x, mv, gammavar){
355   v <- exp(log(1.31)/1.07*mv)
356   location <- log(mv^2 / sqrt(v + mv^2))
357   shape <- sqrt(log(1 + (v / mv^2)))
358   return(vsd_pdf(r,x,gammavar) * dlnorm(r,location, shape))
359 }
360
361 L_loghelp <- function(x, m, gammavar){
362   integrate(L_integr, lower=0, upper=Inf, x=x, mv=m,
          gammavar=gammavar, subdivisions=5000, rel.tol=1e-8,
          stop.on.error=FALSE)$value
363 }
364
365 L_log <- function(x, m, gammavar){
366   apply(as.array(x), MARGIN=1, FUN=L_loghelp, m=m, gammavar=
          gammavar)
367 }
368
369 # negative log likelihood
370 Ln_log <- function(w, x, optim=FALSE){
371   m <- w[1]
372   if(optim){
373     gammavar <- 10^(w[2])
```

```
374   } else{
375     gammavar ← w[2]
376   }
377   Ln_vec ← apply(as.array(x), MARGIN=1, FUN=L_log, m=m,
          gammavar=gammavar)
378   return(-sum(log(Ln_vec)))
379 }
```

C.3 Model in presence of screening

The R implementation of ABC-MCMC for the natural history model in presence of screening is presented here. Please note that all functions are in the end of the code. The code was built for R version 3.6.1.

```
### OBS! All functions are included in the END! ###
### Libraries
library("pracma") # mod, size functions

### DATA GENERATION (synthetic data)
set.seed(1)
n ← 1400000 # Number of individuals
tau1 = 2.36
tau2 = 4.16
eta=exp(-8.63)
beta1=-4.75
beta2=0.56
w_obs ← c(tau1, tau2, eta, beta1, beta2)
d ← length(w_obs)
sobs ← as.vector(simulate_model(n, tau1=tau1, tau2=tau2,
                 eta=eta, beta1=beta1, beta2=beta2))
n_stat ← length(sobs)

### Histograms of V|(A_d, M_d)
par(mfrow=c(2,2))
set.seed(0)
a ← simulate_model(n, tau1=tau1, tau2=tau2, eta=eta,
                     beta1=beta1, beta2=beta2)
rownames(a) ← c("1", "2", "3", "4", "5", "6", "7", "8", "9",
     "10", "11", "12", "13", "14")
barplot((t(a)/colSums(a))[1,], ylim=c(0,0.20), col="blue",
    xlab="Tumour size bins", main="Sympt. det. age 40-42")
barplot((t(a)/colSums(a))[4,], ylim=c(0,0.20), col="blue",
    xlab="Tumour size bins", main="Screen det. age 42" )
barplot((t(a)/colSums(a))[2,], ylim=c(0,0.20), col="blue",
    xlab="Tumour size bins", main="Sympt. det. age 42-44" )
barplot((t(a)/colSums(a))[3,], ylim=c(0,0.20), col="blue",
    xlab="Tumour size bins", main="Sympt. det. age 44-48" )

### Estimation of standard deviation of summary statistics
set.seed(1)
start_time ← Sys.time()
n_mad ← 100
tmp ← matrix(1, n_mad, n_stat)
for(i in 1:n_mad){
  tmp[i,] ← as.vector(simulate_model(n,tau1=tau1,tau2=tau2,
                     eta=eta, beta1=beta1, beta2=beta2))
}
```

```
ssd ← apply(tmp, 2, sd)
print(Sys.time()-start_time)

### Finding an initial epsilon
set.seed(1)
n_pilot ← 100
tmp ← numeric(n_pilot)
for(i in 1:n_pilot){
  ssim ← as.vector(simulate_model(n, tau1=tau1*1.3,
                    tau2=tau2*1.3, eta=eta*1.3,
                    beta1=beta1*1.3, beta2=beta2*1.3))
  tmp[i] ← sqrt(sum(((ssim-sobs)/ssd)^2))/n_stat
}
quantile(tmp, 0.01)

### ABC-MCMC
# Euclidean distance metric weighted by sd estimates
# Adaptive adjustment of tolerance
# Flat improper prior
# Normal (symmetric) transition kernel
set.seed(0)
start_time ← Sys.time()

# Hyperparameters (adjust these)
M ← 20000                          # Number of iterations
burn_in ← 8000                     # Burn in
trans_cv ← c(0.1,0.1,0.1,0.1,0.1)  # trans kernel variance
w ← 1.3*w_obs                      # initial parameters
epsilon ← 0.40                     # Tolerance
quantile ← 80                      # For tolerance decay
final_acc_rate ← 0.01              # Final tolerance

# Initialization (do not change)
w_post ← matrix(NaN,M,d)         # stores postrior values
ssim ← numeric(n_stat)           # Stores temporary simulstat
ssim_mat ← matrix(NaN,M,n_stat)  # Simul stats for all iter.
sdist_vec ← rep(NaN, M)          # Stores statistic distances
n_accABC ← 0                     # Counter for ABC accptance
acc_vec ← numeric(M)             # 1 if iter was acc 0 else
acc_rate ← 1
rho ← abs(trans_cv*w_obs)        # sd of transition kernel

for(iter in 1:M){

  # print acceptance rates every 1000 iter
  if(mod(iter,1000)==0){
    print(iter)
    acc_rate ← mean(acc_vec[(iter-999):(iter-1)])
  }
  # Plots of w over iterations (mixing)
```

```
89    if(mod(iter,10)==0){
90      par(mfrow=c(3,2))
91      for(i in 1:d){
92        plot(w_post[,i],type='l')
93        lines(w_post[,i])
94      }
95    }
96    # Sample from the transition kernel
97    wp ← rnorm(d, mean=w, sd=rho)
98
99    # If tau2<0 or eta<0 reject
100   if(wp[2]≤0 || wp[3]≤0){
101     w_post[iter,] ← w
102     acc_vec[iter] ← 0
103     next
104   }
105   # Simulate from the MODEL using the proposed parameter
          values
106   ssim ← as.vector(simulate_model(n, tau1=wp[1], tau2=wp[2],
          eta=wp[3],
107                                   beta1=wp[4], beta2=wp
                                          [5]))
108
109   # Caclulate normalized euclidian distence between observed
          and simulated statistics
110   sdist ← sqrt(sum(((ssim-sobs)/ssd)^2))/n_stat
111
112   # If the distance between the data statistics and the
          simulated statistics
113   # is bigger then epsilon, jump to next iteration
114   if(sdist ≥ epsilon){
115     w_post[iter,] ← w
116     acc_vec[iter] ← 0
117     next
118   }
119
120   # Epsilon is set to the q:th quantile of accepted
121   # errors every 100th iteration
122   n_accABC ← n_accABC + 1
123   ssim_mat[iter,] ← ssim
124   sdist_vec[n_accABC] ← sdist
125   if(mod(n_accABC, 100) == 0 && acc_rate > final_acc_rate){
126     sdist_sorted ← sort(sdist_vec[(n_accABC-99):n_accABC])
127     epsilon ← sdist_sorted[quantile]
128     cat(iter, "", epsilon, " \n") #Print the epsilon values
129   }
130
131   # # Uninformative (improper) flat prior
132   prior_ratio ← 1
133
```

```
134   if(is.nan(prior_ratio)){
135     warning("Prior ratio: NAN")
136   }
137   if(is.infinite(prior_ratio)){
138     warning("Prior ratio: INF")
139   }
140
141   alpha ← min(1, prior_ratio) # MH acceptence prob
142
143   U ← runif(1)
144   if(alpha > U){
145     w_post[iter,] ← wp
146     w ← wp
147     acc_vec[iter] ← 1
148   } else {
149     w_post[iter,] ← w
150     acc_vec[iter] ← 0
151   }
152 }
153 abc_time ← Sys.time() - start_time
154 print("ABC-MCMC time")
155 print(abc_time)
156
157 ### MIXING PLOTS
158 # tau1, tau2, eta
159 par(mfrow=c(3,1))
160 plot(w_post[,1],type='l', ylab=expression(hat(tau)[1]))
161 lines(w_post[,1])
162 plot(w_post[,2],type='l', ylab=expression(hat(tau)[2]))
163 lines(w_post[,2])
164 plot(w_post[,3],type='l', ylab=expression(hat(eta)))
165 lines(w_post[,3])
166 # beta1, beta2
167 par(mfrow=c(2,1))
168 plot(w_post[,4],type='l', ylab=expression(hat(beta)[1]))
169 lines(w_post[,4])
170 plot(w_post[,5],type='l', ylab=expression(hat(beta)[2]))
171 lines(w_post[,5])
172
173 ### ABC POSTERIOR MEANS
174 w_mean ← colMeans(w_post[burn_in:M,])
175 print("ABC-MCMC acceptance rate:")
176 print(mean(acc_vec[burn_in:M]))
177 print("ABC-MCMC means:")
178 print(w_mean)
179 print("ABC-MCMC mean relative error in %")
180 print((w_mean-w_obs)/w_obs*100)
181
182 ### LINEAR REGRESSION ADJUSTMENT
183 w_post_adj ← vector(mode="list", length=d)
```

```
w_mean_adj ← numeric(d)
for(i in 1:d){
  lm_data = data.frame("w"=w_post[burn_in:M,i],
                       ssim_mat[burn_in:M,])
  lm_tmp ← lm(w ∼ ., data = lm_data, na.action = na.omit)
  w_post_adj[[i]] ← sum(lm_tmp$coefficients * c(1,sobs),
                        na.rm=TRUE) + lm_tmp$residuals
  w_mean_adj[i] ← mean(w_post_adj[[i]])
}
print("ABC-MCMC regression adjusted means:")
print(w_mean_adj)
print("ABC-MCMC regression adjusted means rel error in %")
print((w_mean_adj-w_obs)/w_obs*100)
w_ABC_adj ← w_post_adj
w_ABC ← w_post[burn_in:M,]

### POSTERIOR DESNITY PLOTS
# tau1, tau2, eta
par(mfrow=c(3,2))
plot(density(w_ABC[,1], kernel="epanechnikov", adjust=1.5),
  xlab=expression(paste(hat(tau)[1])), main="ABC")
plot(density(w_ABC_adj[[1]],kernel="epanechnikov",adjust=1),
  xlab=expression(paste(hat(tau)[1])), main="Adjusted ABC")
plot(density(w_ABC[,2], kernel="epanechnikov", adjust=1.5),
  xlab=expression(paste(hat(tau)[2])), main="ABC")
plot(density(w_ABC_adj[[2]],kernel="epanechnikov",adjust=1),
  xlab=expression(paste(hat(tau)[2])), main="Adjusted ABC")
plot(density(w_ABC[,3], kernel="epanechnikov", adjust=1.5),
  xlab=expression(paste(hat(eta))), main="ABC")
plot(density(w_ABC_adj[[3]],kernel="epanechnikov",adjust=1),
  xlab=expression(paste(hat(eta))), main="Adjusted ABC")
# beta1, beta2
par(mfrow=c(2,2))
plot(density(w_ABC[,4],kernel="epanechnikov",adjust=1.5),
  xlab=expression(paste(hat(beta)[1])), main="ABC")
plot(density(w_ABC_adj[[4]],kernel="epanechnikov",adjust=1),
  xlab=expression(paste(hat(beta)[1])), main="Adjusted ABC")
plot(density(w_ABC[,5], kernel="epanechnikov", adjust=1.5),
  xlab=expression(paste(hat(beta)[2])),main="ABC")
plot(density(w_ABC_adj[[5]],kernel="epanechnikov",adjust=1),
  xlab=expression(paste(hat(beta)[2])),main="Adjusted ABC")


############### FUNCTIONS ###############
age_onset_cdf ←  function(t, u=0, A=-0.075, B=1.1*10^(-4),
                          delta=0.5){
  return(1 - ((B-A)*exp(B*t)/(B*exp((B-A)*t)-A))^delta - u)
}

inv_sampl ← function(u, A, B, delta){
```

```
234   root ← uniroot(f=age_onset_cdf, interval=c(1e-5, 10000), u
          , A, B, delta)$root
235   return(root)
236 }
237
238 ages_screen_model ← function(n){
239   return(42)
240 }
241
242 age_onset_model ← function(n, A=-0.075, B=1.1*10^(-4),
243                            delta=0.5){
244   dt ← 0.005
245   t ← seq(0, 250, dt)
246   dens ← A*B*delta*(B-A)^delta*exp(delta*B*t)*(1-exp((B-A)*t
          ))/(
247               B*exp((B-A)*t)-A)^(delta+1)
248   onset_ages ← sample(t, n, replace=TRUE, p=dens)
249   return(onset_ages)
250 }
251
252 growth_rate_model ← function(n, tau1=2.36, tau2=4.16){
253   rates ← rgamma(n, tau1, rate=tau2)
254   return(rates)
255 }
256
257 age_sd_model ← function(growth_rates, ages_onset,
258                           eta=exp(-8.63)){
259   n ← length(growth_rates)
260   u ← runif(n,0,1)
261   v0 ← (0.5^3)*pi/6
262   vdet ← v0 - (log(1-u))/(eta*growth_rates)
263   tdet ← growth_rates*log(vdet/v0)
264   age_det ← ages_onset + tdet
265   return(age_det)
266 }
267
268 screen_out_model ← function(growth_rates, ages_onset,
269         ages_screen, beta1=-4.75, beta2=0.56){
270   n ← length(growth_rates)
271   S ← numeric(n)
272   v0 ← (0.5^3)*pi/6
273   a0 ← ages_onset
274   vol ← v0*exp((ages_screen-a0)/growth_rates)
275   diam ← (6*vol/pi)^(1/3)
276   screen_sens ← 1/(1+exp(-(beta1+beta2*diam)))*(
277                  (sign(ages_screen-a0)+1)/2)
278   u ← runif(n)
279   S[a0<ages_screen & screen_sens>u] ← 1
280   return(S)
281 }
```

```
282
283 age_det_model ← function(screen_outs, ages_scr, ages_sd){
284   age_scrdet ← ifelse(screen_outs, 42, Inf)
285   ages_det ← pmin(age_scrdet, ages_sd)
286   return(ages_det)
287 }
288
289 det_mode_model ← function(ages_det, ages_sd){
290   det ← as.numeric(ages_sd == ages_det)
291   return(det)
292 }
293
294 vol_det_model ← function(ages_onset,ages_det,growth_rates){
295   v0 ← (0.5^3)*pi/6
296   vol ← v0*exp((ages_det-ages_onset)/growth_rates)
297   return(vol)
298 }
299
300 cat_table ← function(vols_det, ages_det, det_modes,
301                      ages_screen){
302   ind_coh ← ages_det ≥ 40 & ages_det ≤ 48
303   vols ← vols_det[ind_coh]
304   ages ← ages_det[ind_coh]
305   detmodes ← det_modes[ind_coh]
306   agesscr  ← 42
307   vol_breaks ← c(0, 5, 80, 300, 700, 1500, 3000, 5000, 8000,
308                  12500,20000, 33000, 55000, 120000, Inf)
309   n_volbreaks ← length(vol_breaks)
310   age_breaks ← c(40, 42, 44, 48)
311   n_agebreaks ← length(age_breaks)
312   n_screens ← length(agesscr)
313   n_agecat ← n_agebreaks-1+n_screens
314   n_volcat ← n_volbreaks-1
315   table ← matrix(0, n_volcat, n_agecat)
316   for(j in 1:n_agecat){
317     for(i in 1:n_volcat){
318       if(j < n_agebreaks){
319         tmp ← sum(detmodes == 1 &
320                     vols ≥ vol_breaks[i] &
321                     vols <  vol_breaks[i+1] &
322                     ages ≥ age_breaks[j] &
323                     ages <  age_breaks[j+1])
324       } else{
325         tmp ← sum(vols ≥ vol_breaks[i] &
326                     vols <  vol_breaks[i+1] &
327                     ages == agesscr[j-n_agebreaks+1])
328       }
329       table[i,j] ← tmp
330     }
331   }
```

```
332   return(table)
333 }
334
335 simulate_model ← function(nobs, tau1=2.36, tau2=4.16,
336             eta=exp(-8.63), beta1=-4.75, beta2=0.56,
337             A=-0.075, B=(1.1*10^(-4)), delta = 0.5){
338   ages_screen ← ages_screen_model(nobs)
339   growth_rates ← growth_rate_model(nobs, tau1, tau2)
340   ages_onset ← age_onset_model(nobs, A, B, delta)
341   ages_sd ← age_sd_model(growth_rates, ages_onset, eta)
342   screen_outs ← screen_out_model(growth_rates, ages_onset,
343                                  ages_screen, beta1, beta2)
344   ages_det ← age_det_model(screen_outs,ages_screen,ages_sd)
345   det_modes ← det_mode_model(ages_det, ages_sd)
346   vols_det ← vol_det_model(ages_onset, ages_det,
347                            growth_rates)
348   tab ← cat_table(vols_det, ages_det, det_modes,
349                   ages_screen)
350   return(tab)
351 }
```

TRITA -SCI-GRU 2020:089