Mikhail Shubin
Department of Mathematics and Statistics
University of Helsinki
Finland

Bayesian Inference for Spatio-Temporal Models

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Academic Dissertation. To be presented, with the permission of the Faculty of Science of the University of Helsinki, for public examination in Exactum (Gustaf Hällströmin katu 2b), Helsinki, auditorium CK112, on June 29, 2016 at 12 noon.

Supervisor	**Jukka Corander**	University of Helsinki, Finland
Pre-examiners	**Tanel Tenson**	University of Tartu, Estonia
	Gianpaolo Scalia-Tomba	University of Rome Tor Vergata, Italy
Opponent	**Birgitte Freiesleben de Blasio**	University of Oslo, Norway
Custos	**Jukka Corander**	University of Helsinki, Finland

LICENCES AND COPY RIGHTS

PRINTED BY HANSAPRINT

ISBN 978-951-51-2289-6 (paperback)
ISBN 978-951-51-2290-2 (PDF, available at ethesis.helsinki.fi)

First printing, June 2016

It is simply this: do not tire, never lose interest, never grow indifferent –
lose your invaluable curiosity and you let yourself die.
It’s as simple as that.

– Tove Jansson

To Marina

Abstract

The dissertation presents five problem-driven research articles, representing three research domains related to micro-organisms causing infectious disease. Articles I and II are devoted to the A(H1N1)pdm09 influenza ('swine flu') epidemic in Finland 2009-2011. Articles III and IV present software tools for analysing experimental data produced by Biolog phenotype microarrays. Article V studies a mismatch distribution as a summary statistic for the inference about evolutionary dynamics and demographic processes in bacterial populations.

All addressed problems share the following two features: (1) they concern a dynamical process developing in time and space; (2) the observations of the process are partial and imprecise.

The problems are generally approached using Bayesian Statistics as a formal methodology for learning by confronting hypothesis to evidence. Bayesian Statistics relies on modelling: constructing a generative algorithm mimicking the object, process or phenomenon of interest.

Acknowledgement

I wish to thank my PhD supervisor Jukka Corander for believing in me and for supplying me with amazingly interesting projects;

Kari Auranen for his patience and perfectionism;

Sirkka-Liisa Varvio for introducing me to the influenza project;

Elina Numminen for being the one who disagrees;

Jukka Kohonen for sharing his enthusiasm to knowledge;

Minna Vehkala for collaboration;

my parents for love, support and education;

my wife, Marina, for being with me through cold winter nights.

I would like to thank all participants of Helsinki Statistics discussion club. Our long discussions revealed to me that Statistics is a subject one could devote a life to. Without you I would never been able to finish this dissertation[1].

In addition, I wish to thank the following people for assisting me with writing this introduction: Jukka Corander, Elina Numminen, Jukka Kohonen, Jani Anttila, Tanel Tenson, Gianpaolo Scalia-Tomba and Marina Shubina.

[1] This is a cliché phrase, but I really mean it. In all the parallel universes where the Statistics club had not existed this dissertation is not yet completed.

List of Articles

I *Mikhail Shubin, Mikko Virtanen, Salla Toikkanen, Outi Lyytikäinen and Kari Auranen*
Estimating the burden of A(H1N1)pdm09 influenza in Finland during two seasons.
2013, *Epidemiology and Infection*.
doi:10.1017/S0950268813002537

MS designed and performed the research. MS, KA, OL wrote the paper. MV, ST, OL provided the data. OL, KA provided the biological expertise.

II *Mikhail Shubin, Artem Lebedev, Outi Lyytikäinen and Kari Auranen*
Revealing the true incidence of pandemic A(H1N1)pdm09 influenza in Finland during the first two seasons - an analysis based on a dynamic transmission model.
2016, *PLOS Computational Biology*.
doi: 10.1371/journal.pcbi.1004803

MS designed and performed the research. OL provided the data. OL, KA provided the biological expertise. MS implemented `R` and `Python` code, AL implemented `C` code. All authors contributed in writing the paper.

III *Minna Vehkala, Mikhail Shubin, Thomas R. Connor, Nicholas R. Thomson, Jukka Corander*
Novel R Pipeline for Analyzing Biolog Phenotypic Microarray Data.
2014, *PLOS ONE*.
doi:10.1371/journal. pone.0118392

MV, MS wrote the paper. TC, NRT provided the data. MV, JC designed the pipeline. MV, MS implemented the scripts. MS designed the figures. TC, NRT, JC read and approved the manuscript.

IV *Mikhail Shubin, Katharina Schaufler, Karsten Tedin, Minna Vehkala, Jukka Corander*
Identifying multiple potential metabolic cycles from Biolog experiments.
2016, *Submitted Manuscript*.

MS designed and coded the algorithm. KS, KT acquired the data and provided the biological expertise. All authors contributed in writing the paper.

V *Mikhail Shubin, Elina Numminen, Michael U. Gutmann, William P. Hanage, Jukka Corander*
Statistical properties of the allelic mismatch distribution in neutrally evolving haploid populations.
2016, *Submitted Manuscript*.

MS designed and performed the simulations. WH provided the data. All authors contributed in writing the paper.

Contents

Preface

Let $p(x|y)$ be a probability that a statement x is true given that a statement y is true. If y is always true we can write $p(x)$; if x consists of two statements a and b we can write $p(a, b|y)$ or $p(b, a|y)$. Probabilities satisfy the following conditions:

$$p(x|y) \geq 0,$$

$$\int_x p(x|y)dx = 1,$$

$$\int_x p(x, y)dx = p(y),$$

$$p(x, y) = p(x|y)p(y).$$

In the continuous case $p(x|y)$ is called a probability density function. In a discrete case integration is substituted with summation and $p(x|y)$ is called a probability mass function. Using the last expression we can easily show that $p(x|y)p(y) = p(y|x)p(x)$, therefore:

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}.$$

This is known as the Bayes' theorem or Bayes' rule. It was first proven by Thomas Bayes, published by Richard Price in 1763 and reinvented in its modern form by Pierre-Simon Laplace in 1774 [1].

The simple expression known as the Bayes' theorem is the essence of the Bayesian Statistics. Everything done in 242 years after its discovery addresses particularities: how to apply, compute and understand it. Why Bayes' theorem is so important? It formalizes the learning process: obtaining a new knowledge by confronting hypothesis to evidence.

The present dissertation consists of five problem-driven research articles, each demonstrating how the Bayesian inference can be applied in microbiology. Articles I and II are devoted to the A(H1N1)pdm09 influenza ('swine flu') epidemic in Finland 2009-2011. Articles III and IV present software tools for analysing experimental data produced by Biolog phenotype microarrays. Article V studies a mismatch distribution as a summary statistics for the inference of structure in bacterial populations. All addressed problems share the following two features:

- They concern a dynamical process developing in time and space: spread of a virus; growth of bacteria on the experimental plate; neutral evolution of a population.
- The observations of the process are partial and imprecise: there is an under-reporting of infected influenza cases; measurement errors caused by a laboratory equipment; sampling bias.

These problems are handled by constructing a spatio-temporal model of the latent process and estimating the parameters of such model in a Bayesian framework. The next chapter describes briefly the methodological platform. The subsequent chapters are devoted to models and computational methods. The last chapter of this introduction summarizes articles, presenting their context, research goals and results.

Through the text, boxes like this are used to link the introduction to the research articles.

Bayesian Thinking

There is a long tradition [2, 3, 4] of establishing the Bayesian Statistics as a theory for a decision-making in the presence of uncertainty[2]. To make the best possible decision we need three ingredients:

- Knowledge about the problem's domain;
- Observations which can support or refute this knowledge;
- Utility function, quantifying the gain different decisions will bring us in the different states of nature.

First, the existing knowledge is formalized as model and prior. The model presents certain knowledge[3] while the prior codifies uncertainty. The model, prior and observations are combined in Bayes' theorem to produce the updated uncertainty represented by the posterior. The posterior is combined with the utility function in the process of Bayesian decision making to choose the decision maximizing the expected utility. The scheme is shown in the following picture:

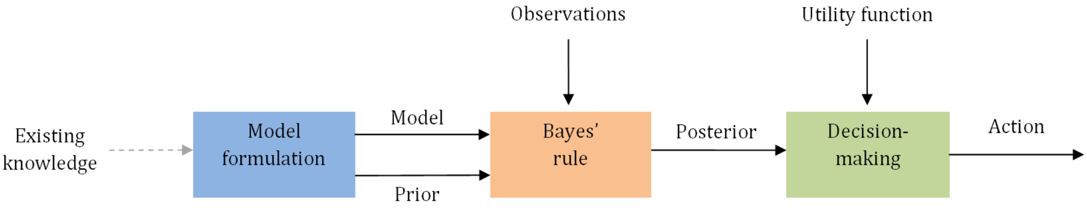


Figure 1: The idealized version of the Bayesian thinking. ⟶ and --→ show the flow of the formal and informal knowledge, respectively.

The absence of feedback is an essential feature of this scheme, guaranteeing its objectivity and fairness. The model formulation

[2] i.e. everywhere, as the certain knowledge is doubtfully possible outside of the formal domains of Mathematics and Logic.

[3] certain knowledge: knowledge which is reliable and precise enough to safely ignore its uncertainty.

should not be affected by the observations. In the ideal world, the model is constructed before the observations are obtained. The posterior should not be affected by the decision-making process. Statistics serves as a blind witness, not knowing what it is testifying for. Model formulation, Bayesian inference and decision making are separate processes. They may even be done by separate people. For example, a model of an epidemic spreading across a country is formulated by a field expert (e.g. virologist) while the utility function evaluating different control policies is set by a decision maker (e.g. health official). Statistician plays a role of the medium, conducting the Bayesian inference.

This scheme is inapplicable directly to scientific research distanced from decision making. The end product of the scientific inquiry is not an action but new knowledge. The different scheme is thus:

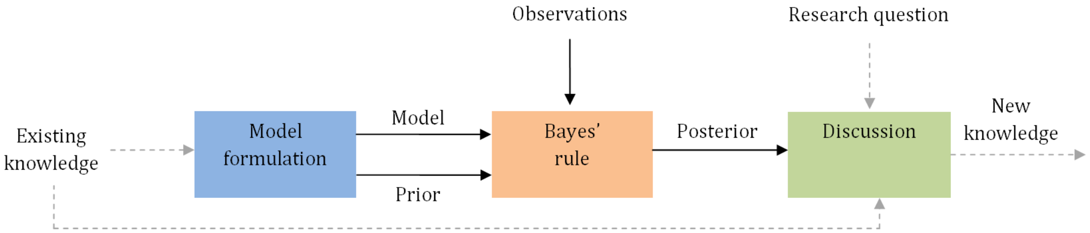


Figure 2: The idealized version of the Bayesian thinking in science. ⟶ and --→ show the flow of the formal and informal knowledge, respectively.

The formally defined mathematical objects: 'utility function' and 'decision' are replaced by informal vague terms: 'research question' and 'new knowledge'. Posterior distribution is now the last formal result of the research. Discussion plays a role of reverse-modelling, it deformalizes the posterior translating it from the language of mathematical abstraction into a natural language. Discussion is able to link back the knowledge which was not used in constructing the model. It is supposedly done by the same researcher who constructed the model. Discussion is not limited to the posterior but may address the model as well.

Not every statistical problem can be solved analytically. If the Bayesian rule can not be applied directly, computational[4] methods are used instead. The scheme would look differently:

[4] Here by analytical solution I mean an exact answer, as achieved by pure math, pen and paper. By computational solution I mean an approximate answer achieved with a computer.

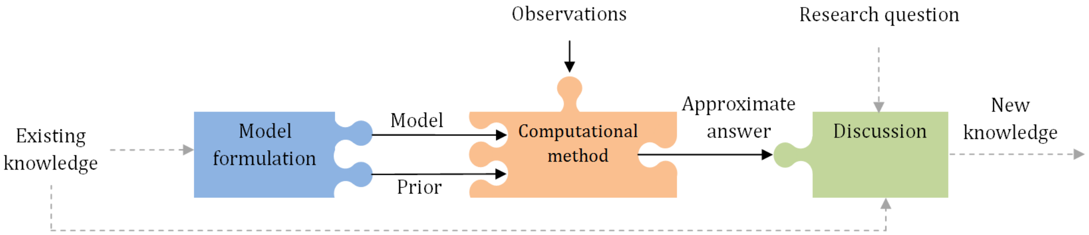


Figure 3: The practical version of the Bayesian thinking in science. ⟶ and --→ show the flow of the formal and informal knowledge, respectively. Components of thinking are constructed to fit each other.

Instead of the exact posterior distribution a computational method provides an approximate answer such as an approximation of the posterior, point estimates, interval estimate or a model selection criterion. This scheme still has no feedback loop. However, components should be selected minding each other. The appropriate choice of inference method depends on the research question, model and available data. The priors could be chosen to facilitate computational methods. The discussion should take into account the assumptions, simplifications and approximations made during the model formulation and computational inference.

Pre-selecting these components to perfectly fit each other could be impossible in a real project. Bayesian thinking needs to be a holistic process:

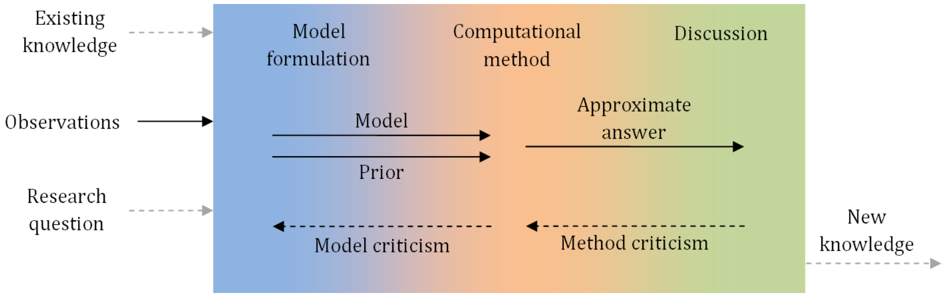


Figure 4: The practical version of the Bayesian inference in science. ⟶ and --→ show the flow of the formal and informal knowledge, respectively.

The model, prior and computational method are iteratively tuned to fit each other in processes called model criticism and method criticism. Thus, feedback loops are created. Knowledge is generated during the whole process of Bayesian thinking including the criticisms, e.g. learning that certain models are unable to describe certain data could bring valuable insight.

The use of feedback loops expels us from the safe haven of idealized Bayesian thinking. It may lead to scientific malpractices, both voluntary and unconscious. Firstly, the model is now constructed taking the observables into account. This may give data too much weight or lead to overfitting. In the worst case the whole procedure of Bayesian inference turns into a useless tautology, as the model would be able to explain nothing but observations.

Secondly, iterative nature of Bayesian thinking may tempt us to continue tuning the model and the inference method until the result would satisfy our personal reasons, e.g. until they will agree with our personal beliefs. I don't know any perfect solution to prevent these malpractices. Personally, I tried to document all the rationales for choosing the model and inference methods.

How these principles are related to the presented articles? Did I see the data before formulating the models and inference methods?

Articles I and II are not just devoted to the same problem (influenza epidemic in Finland in 2009-2011) but use the same dataset. Paper II expands on the ideas of paper I, using a more complex dynamical transmission model simulated on a cluster computer. The model used in Article II was formulated after working with the dataset for several years.

Articles III and IV are suggesting new methods to handle Biolog phenotype microarray data. These methods were built on models formulated while working with the dataset.

On the contrary, Article V uses ideas developed on a purely theoretical basis, without observing any data.

Modelling

Modelling is one of the basic tools of the scientific method. It substitutes the studied object, process or phenomenon with its simplified version – model, keeping the relevant features and excluding or simplifying irrelevant ones [5]. In this introduction I would use a narrow definition:

A model is a computer program capturing a real-life object, process or phenomenon, taking the input values x (parameter) and returning output values y (observations, data) using reasonable computational resources. Sampling (generating) the values of y with the parameters x is denoted with $y \sim p(y|x)$.

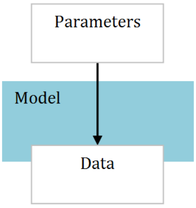


Figure 5: Model is represented as a direct acyclic graph.

This definition is very practical: every model satisfying it can be analysed by the Bayesian thinking using the framework described in this introduction. This definition is synonymous to the concept of simulator, generating algorithm and a program in a probabilistic programming [6]. In the model-program computations are used as a universal tool for capturing reality. In particular, randomly generated values are used to represent unknown quantities or known mechanisms in a simplified fashion.

The observations y represent all quantities that can be directly compared to reality. The parameters x capture the quantities of interest what we are trying to estimate. Models can be split into several chained programs so one model's output is another model's input. These intermediate values will be referred as the hidden states and denoted[5] with h.

A simple example of a model is a program emulating a fair coin toss: it takes no input parameters and returns either 'head' or 'tail' with equal chance. On the other side of complexity scale we can find astrophysical simulators, taking cosmological

[5] There are no single notation convention. For example, Cressie and Wikle [7] use θ for parameters, Y for hidden states and Z for observables. Parameters used to define other parameters are sometimes called metaparameters or hyperparameters and denoted with γ or ϕ.

constants as input and emulating galaxy collisions or black holes.

What models are used in the research articles?

Articles I and II use epidemiological models, capturing the spread of influenza virus across Finland. Articles III and IV use model of metabolic signals produced by bacteria during a Biolog phenotype microarray experiment. Article V uses a model of neutral evolution in the structured population.

Models as Probability Distributions

Every model generating data y using parameters x defines a probability distribution $p(y|x)$ known as the likelihood. Philosophically speaking, the program and the corresponding probability distribution $p(y|x)$ are the same entity. In practice, however, having the program does not guarantee knowing the distribution. Sometimes one could sample $x \sim p(y|x)$ but can't compute $p(y|x)$ and vice versa.

Types of Models

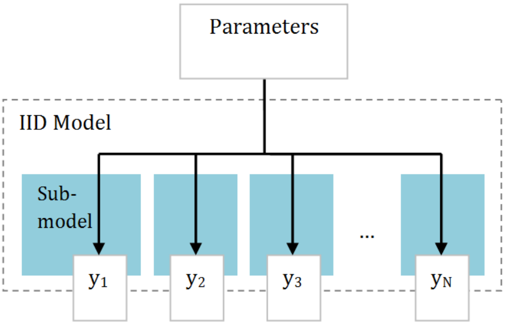
The following list presents few basic classes of models relevant to research articles. The classification is based on the ways how we can split the model and its output into submodels.

Spatio-Temporal Models is a class of models describing dynamical processes happening in space. The hidden states and observation are split into subsets $h_{s,t}$ and $y_{s,t}$ indexed by space s and time t.

Bayesian inference can target different parts of the model. The common examples are:

x	Parameter estimation
h	Inference about states
$h_{s,t>T}$ or $y_{s,t>T}$	Prediction
$h_{s\notin S,t}$ or $y_{\notin S,t}$	Extrapolation

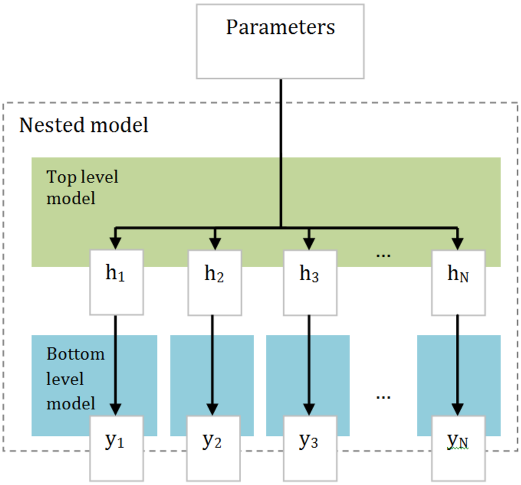
Models with independent and identically distributed (IID) variables. If a model can be split into n identical sub-programs sharing the same input to generate n subsets of data $y = (y_i)_{u\in 1\ldots N}$, the model is referred as having IID assumptions. The model could be represented by the following graph and expression:



$$\begin{aligned} y_1 &\sim p(y|x) \\ y_2 &\sim p(y|x) \\ &\cdots \\ y_N &\sim p(y|x) \end{aligned}$$

Inference on such models could utilize multiple independent evidence. However, the IID assumption is often unrealistic outside the experimental setting in controllable environment.

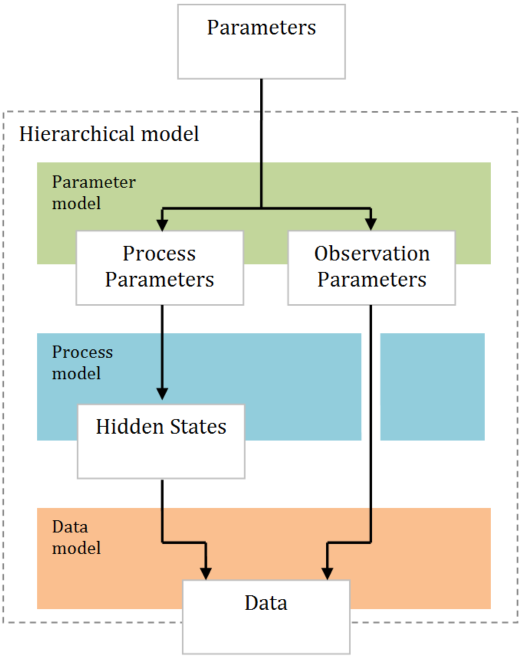
Nested Models is a class of models which can be split into two levels of hierarchy. The top level model generates a vector of hidden states $h = (h_i)_{i \in 1 \ldots N}$. The bottom level models generate subsets of data $y = (y_i)_{u \in 1 \ldots N}$ independently, each using its own subsets of hidden states [8]. Nested model could be represented by the following graph and expression:



$$\begin{aligned} h_1, h_2 \ldots h_N &\sim p(h_1, h_2 \ldots h_N | x) \\ y_1 &\sim p(y_1 | h_1) \\ y_2 &\sim p(y_2 | h_2) \\ &\ldots \\ y_N &\sim p(y_N | h_N) \end{aligned}$$

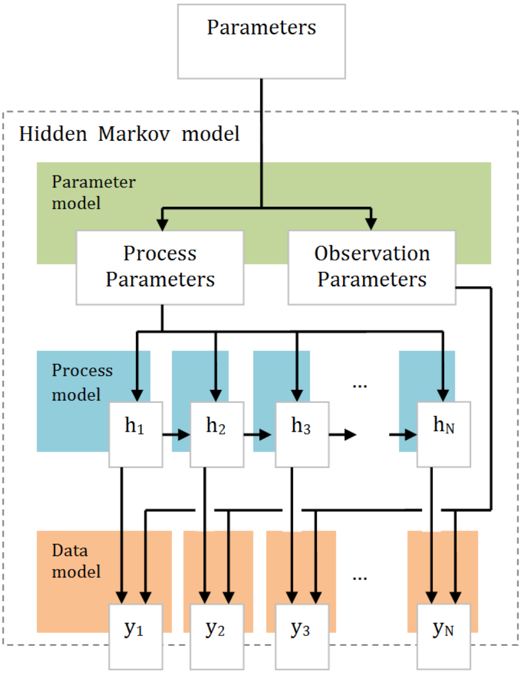
In a spatio-temporal nested model, hidden states are usually generated assuming *smoothness*: states what are close in time and space are more likely to be similar than the distant ones. Smoothness assumption plays an important role in the Bayesian Statistics. It is realistic in many cases and it allows sharing evidence among non-IID data subsets.

Hierarchical Models is a class of models which can be split into three separate programs [7]: the parameter model generates two sets of parameters denoted with x_y and x_h. The process model generates hidden states h using process parameters x_h. The data model uses hidden states and observation parameters x_y to generate the observations y.



parameter model	$x_y, x_h \sim p(x_y, x_h \mid x)$
process model	$h \sim p(h \mid x_h)$
data model	$y \sim p(y \mid h, x_y)$

Hidden Markov models is a class of spatio-temporal nested hierarchical models with two additional restrictions: hidden state h_t should depend only on the parameters and the previous hidden state h_{t-1}; observation y_t should depend only on the parameters and the hidden state h_t.



parameter model	$x_y, x_h \sim p(x_y, x_h \| x)$
process model	$h_1 \sim p(h_1 \| x_h)$
	$h_t \sim p(h_t \| h_{t-1}, x_h)$
data model	$y_t \sim p(y_t \| h_t, x_y)$

What model types are used in the research articles?

Articles I, II and V use Hidden Markov models.

Articles III and IV use general spatio-temporal models.

Identical experimental replicates are treated as IID samples.

Prior Selection

Model requires input parameters x. Prior distribution $p(x)$ captures the belief about the input parameters before observing the data. While the model codifies certain knowledge (it simulates one's ideas of how the reality works) priors may express uncertainty. Here I suggest a classification of priors according to the amount of information they bring:

Non-informative priors reflect the absence of any information regarding the parameters. Constructing an uninformative prior is a complex mathematical and philosophical task. It is sometimes solved by an improper prior – a function which does not satisfy the probability axioms but may be used as an approximation. Improper priors require extra attention to avoid problems in inference.

Regularizing priors slightly restrict the parameters space to prevent the model from exhibiting unwanted singular behaviour.

Informative priors represent actual knowledge. They typically strongly restrict the parameter space. Informative prior may represent subjective knowledge.

Fixed values are not technically prior any more, but a part of a model. They are used to represent certain knowledge or to simplify the model by locking less important parameters.

In practice, priors are mostly constructed with standard probability distributions, so we can both sample $x \sim p(x)$ and compute $p(x)$. Formulating a prior probability distribution for the parameters $p(x)$ defines the existence of a prior distribution for the hidden states h and observables $p(y)$.

What priors are used in the research articles?

Articles I and II use mostly informative priors, obtained through expert opinions and literature review.

Articles III and IV present computational tools, where users have to define the priors themselves.

Model Complexity

> Everything should be made as simple as possible, but not simpler.
> – attributed to Albert Einstein

> A model should be as big as an elephant.
> – Leonard Jimmie Savage

> All models are wrong but some are useful.
> – George E. P. Box

> All models are useful but some are wrong.
> – Jani Anttila

How exactly should models be constructed? As the epigraphs imply, there exist multiple schools of thought about this issue. This section brings some general discussion on model complexity.

I separate two approaches: *analytical modelling* and *synthetic modelling*. The first approach (also known as the Aristotelian idealization [9]) suggests that a model should be as simple as possible. Analytical model consists of a minimal set of rules needed to replicate the studied object, process or phenomenon. For example, in infectious disease epidemiology this could be an SIR model, suggested in 1927 [10, 11]. It consists of only three differential equations and is able to describe a process resembling an epidemic outbreak.

The synthetic modelling suggests that a model should follow our conception of reality as closely as possible. The limit to model's complexity is set by our knowledge and our ability to analyse it (e.g. access to computational resources). In infectious disease epidemiology a FluTE model [12] can serve as an exam-

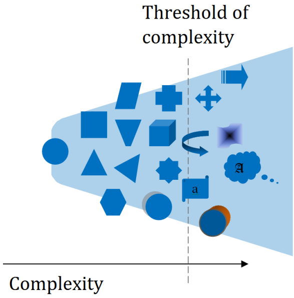


Figure 6: The set of all possible models. Should we use the simplest model possible or the most complex model which we can handle using the available computational resources?

ple. This model simulates the spread of influenza by recreating the low-level behaviour of individual people. FluTE requires not only extensive computational resources to run a virtual epidemic, but detailed census data (distribution of the household sizes, contact frequencies, etc) and information on influenza virology (latent periods, transmission probabilities, etc).

Bayesian methodology generally prefers synthetic modelling [3]. Naturally, to get the best answer one should use all the available information. However, more complex models generally require more complex computational methods, resulting in more layers of approximation. This makes the Bayesian inference less transparent and may nullify or even revert the gain from inclusion of additional knowledge.

The complexity of analytical modelling depends on a set of features we consider to be *essential* for object, phenomenon or process of interest. Usually we can define such a large set of essential features that the difference between analytical and synthetic modelling disappears. Therefore the problem of model complexity can be interpreted as a problem of essential features.

How the models used in the research articles are constructed?

Articles I and II use synthetic models, the most complex models possible for available computational resources. This allows integrating multiple sources of information in the inference and addressing more complex questions.

Articles III – V use analytical models, the simplest models which are able to replicate the phenomenon of interest: metabolic signals from active and not-active bacteria (III), several metabolic cycles reflected in signals from active bacteria (IV), neutral evolution affected by population structure (V). Articles III – V are not linked to any particular problem. Constructing more complex models would be rather impossible due to lack of specific knowledge.

Computational methods

Construction of a model implies a likelihood $p(y|x)$. Construction of the prior $p(x)$ and the model imply the prior distribution for the observables $p(y)$. This three distributions can be plugged together into the Bayes' rule

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}.$$

to estimate the posterior $p(y|x)$.

We would like to *study* the posterior: visualize it, estimate its mean, median and mode, its shape (e.g. how many modes it has), its correlation structure etc. If $p(x|y)$ is too complex to be studied analytically, computational methods are used.

This chapter attempts to classify the computation methods used in Bayesian statistics. It does not aim at providing neither comprehensive nor exclusive overview, but rather giving a context for the methods used in the research articles.

Computational methods can be separated into three groups: point estimation, proxy distribution and sampling. The following picture shows the methods relation within the groups:

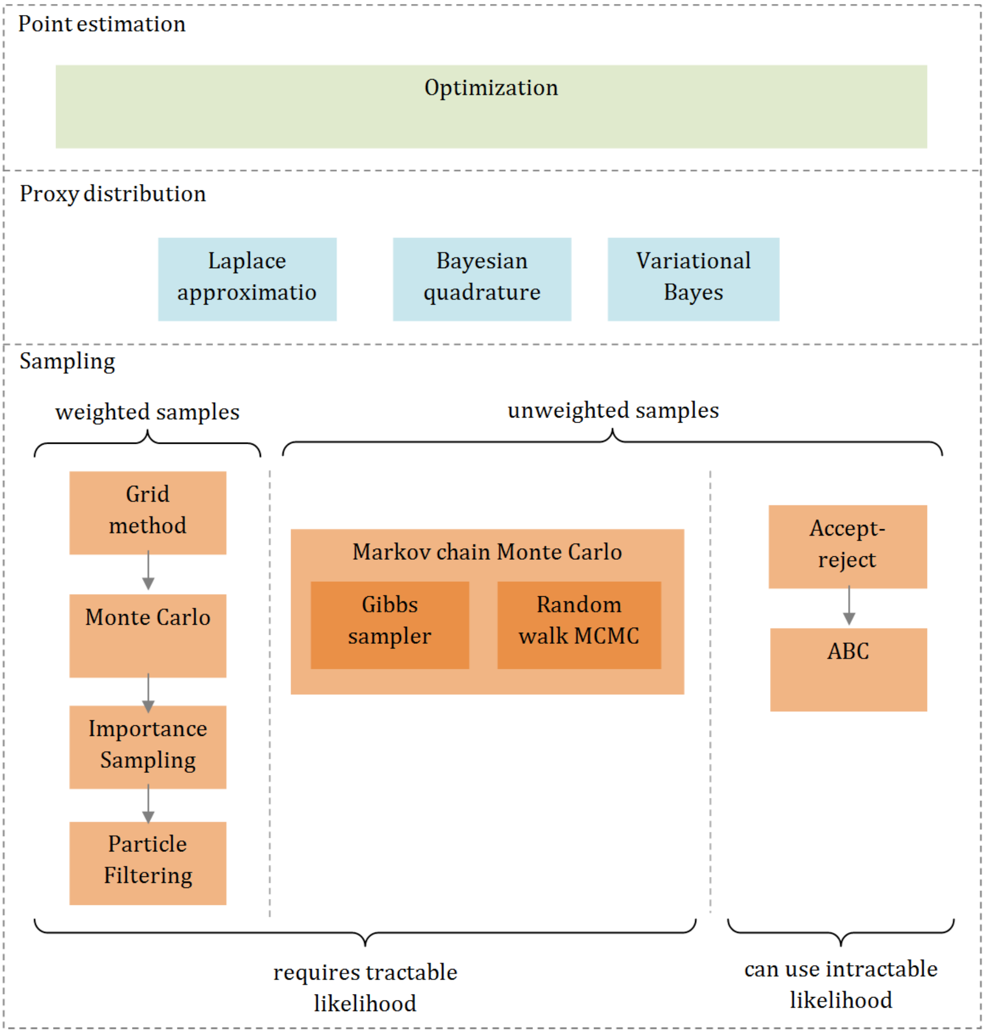


Figure 7: Classification of computational methods. Arrows represent method extensions.

Point Estimation

A single summary of the target posterior (such as mean or mode) is estimated. This can be done by any suitable optimization method.

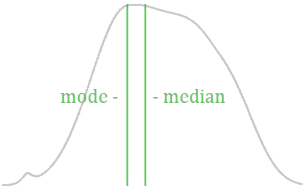


Figure 8: The median and the mean: two point estimates of a target distribution (gray line).

Proxy Distribution

The target posterior is approximated with another probability distributions, which is close to the target but is easier to analyse.

Laplace approximation (also known as the Laplace's Method or saddle-point approximation) is a simple example of proxy methods. It uses a normal distribution with mean at a mode of the target posterior and covariance defined using a second order derivatives at the posterior mode. The mode could be found with optimization and derivatives could be estimated numerically [13].

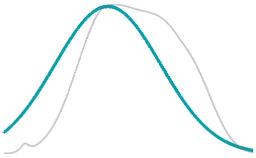


Figure 9: The target distribution (gray) is approximated a Normal distribution (cyan).

Variational Bayes methods are based on minimizing the Kullback-Leibler divergence between the target and a proxy distribution. It is not limited to a particular form of a proxy distribution [13].

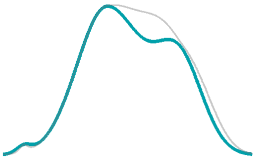


Figure 10: The target distribution (gray line) is approximated with a mixture of normal distributions (cyan)

Bayesian quadrature approximates the target with a non-parametric (e.g. Gaussian) process fitted to the target using the principles of Bayesian inference [14].

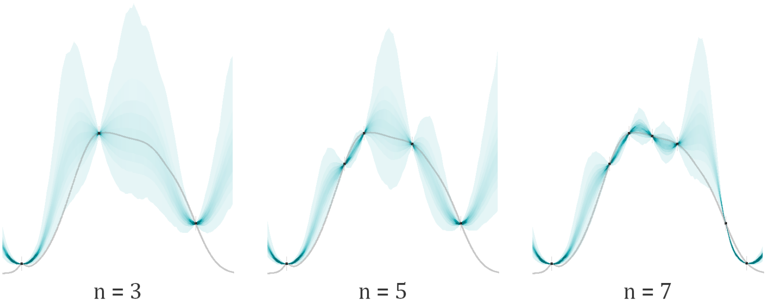


Figure 11: The target distribution (gray line) is approximated with a Gaussian process (cyan), build upon n=3, 5 and 7 points.

Sampling

Sampling methods approximate the target posterior with a set of particles $\{x_i\}_{i \in 1 \ldots n}$ or weighted particles $\{x_i, w_i\}_{i \in 1 \ldots n}$, here x is a vector of parameters and w is a positive number.

Sampling approximation is good when taking a random particle from a set imitates sampling from the target distribution. In the unweighted case the particle is selected uniformly, in the weighted case - with the corresponding probability weight $w_i / \sum w$. For most of the sampling methods the precision grows with the number of used particles n and becomes perfect with $n \to \infty$.

The set of generated particles provides a very practical way to do visualization and estimate various statistics. For example, mean and quantiles of the target distribution are approximated by mean and quantiles of the set. Any change of variables in the target distribution can be mirrored by a corresponding transformation on the set.

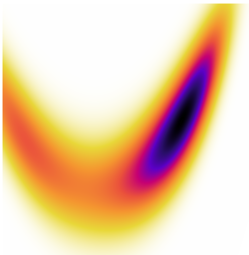


Figure 12: The heatmap of the true target distribution, used for the few next examples.

Grid method is a deterministic method. The target distribution $p(x|y)$ is measured on a grid of n points from the domain of x. These points are pre-allocated to uniform cover the domain. For example, in one dimensional interval $[a, b]$ the optimal allocation would be $x \in \{a, a + \frac{b-a}{n-1}, \ldots, b\}$. The target distribution is then represented by a set of weighted samples $\{x_i, p(x_i|y)\}$. Grid methods work well only for low dimension as the complexity of allocation grows exponentially with dimensionality of a parameters' domain.

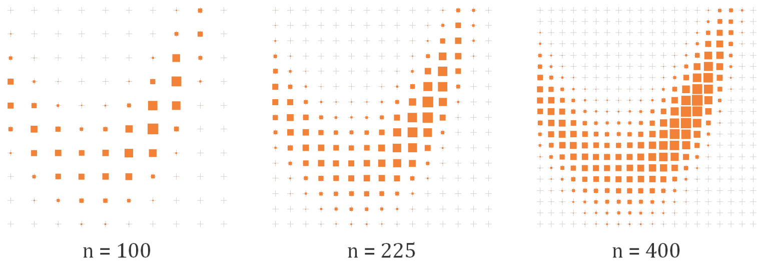


Figure 13: Three sets of particles obtained with a Grid Method with n=100, 225 and 500 particles. All particles are marked by + signs and by orange dots with areas corresponding to the particles' weight.

Monte Carlo method also represents the target distribution with $\{x_i, p(x_i|y)\}$. The difference is that points x_i are randomly and uniformly sampled from the domain of x. Monte Carlo scales much better than the grid method [15] as its convergence is guaranteed by the law of the large numbers independently of the number of dimensions.

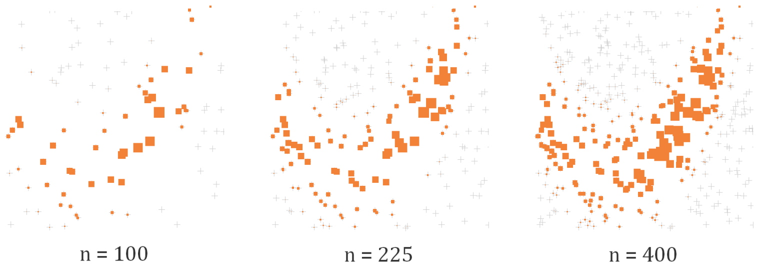


Figure 14: Three sets of particles obtained with a Monte Carlo method with $n = 100, 225$ and 500 particles. All particles are marked by + signs and by orange dots with areas corresponding to the particles' weight.

Importance Sampler (IS) is an extension of the Monte Carlo method [15], enabling to utilize knowledge about $p(x|y)$ to study it. The values of x are sampled not uniformly, but from an *importance distribution* $g(x|y)$. To compensate for the uneven spread of samples, the weights are divided by the sampling probability: the target distribution is represented by $\{x_i, p(x|y)/g(x|y)\}$.

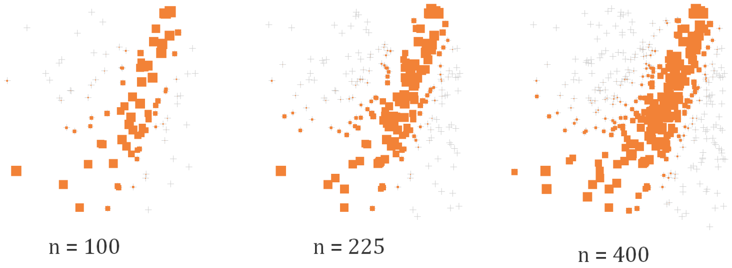


Figure 15: Three sets of particles obtained with an Importance sampler.

Importance sampler is more efficient when $g(x|y)$ is similar to $p(x|y)$. The importance distribution is constructed either to reflect the prior knowledge about $p(x|y)$, or in such a way that $p(x|y)/g(x|y)$ could be simplified.

If the importance function is well selected, the weights $p(x|y)/g(x)$ would be near 1. It is natural to measure the goodness of importance function an expected log weight of a sample. This is, in fact, a Kullback-Leibler divergence, therefore optimizing the importance function is an example of a variational Bayes method.

Particle Filtering (PF) (also known as the Particle Monte Carlo and Population Monte Carlo) is an extension of the Importance sampling [16]. Assume the hidden states and observation are indexed by time, $h = (h_1, h_2, \dots, h_T)$, $y = (y_1, y_2, \dots, y_T)$ and a partial posterior $p(x, h_{1\dots t}|y_{1\dots t})$ can be defined for each t. To generate x and h from an importance distribution $h \sim g(h, x|y)$ hidden states are sampled sequentially: first we sample
$x \sim g(x)$, then
$h_1 \sim g(h_1|x, y_1)$, then
$h_2 \sim g(h_2|x, h_1, y_{1,2})$, then
$h_2 \sim g(h_3|x, h_{1,2}, y_{1,2,3})$ and so on until
$h_T \sim g(h_T|x, h_{1\dots T-1}, y)$.

Sometimes one can early predict if a particle's weight $w = p(h, x|y)/g(h, x|y)$ is going to be big or small seeing its partial weights $w_t = p(h_{1\dots t}, x|y_{1\dots t})/g(h_{1\dots t}, x|y_{1\dots t})$. In this case one could immediately stop generating the useless low-weight particle and start a new one.

This is exactly what PF is trying to achieve. To be able to compare samples, they are not generated sequentially but all at once. First, the set of particles $\{x, h_1, w_1\}_{i \in 1\dots n}$ is generated. Then each individual particle is propagated to $\{x, h_{1,2}, w_2\}_i$, then to $\{x, h_{1,2,3}, w_3\}_i$ and so on. Note that each such cloud of weighted particles describes the partial posterior $p(x, h_{1\dots t}|y_{1\dots t})$.

The filtering occurs on so called resampling steps, when a set of particles $\{h_{1\dots t}, w_t\}_{i \in 1\dots n}$ is reorganized so that all particles would have the same weight: particles with a small weight are removed while particles with a large weight are split. The simplest way to do it is a multinomial distribution, but there are other approaches [17].

Markov chain Monte Carlo (MCMC) is a family of methods where unweighed particles are generated sequentially [15].

Assume a Markov chain with an initial state x^0 and the transition rule:

$$x^* \sim g(x^*|x^t)$$
$$x^{t+1} = \begin{cases} x^* & \text{with probability} \quad min\ 1, \dfrac{p(x^*|y)g(x^t|x^*)}{p(x^t|y)g(x^*|x^t)} \quad ; \\ x^t & \text{otherwise} \end{cases}$$

here x^* is a proposed new value for x, $g()$ is a proposal distributions and the superscript denotes iteration index. It can be shown that if $p()$ and $g()$ are fulfilling several obvious criteria the Markov chain would have a unique stationary distribution which is exactly $p(x|y)$. Knowing that every Markov chain eventually converges to its stationary distribution we can generate samples from $p(x|y)$ given sufficient time.

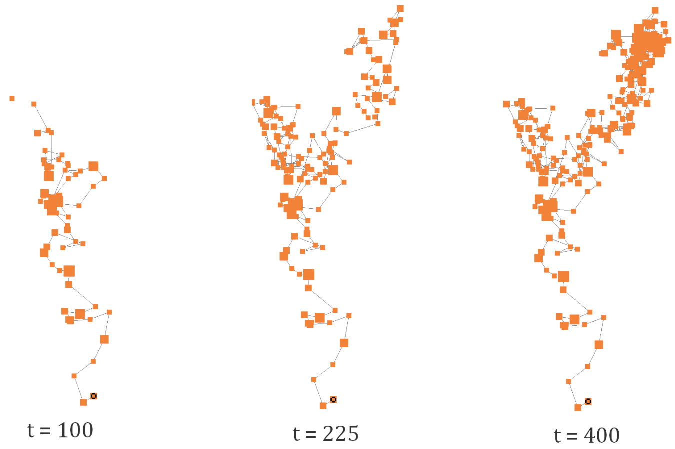


Figure 16: The set of particles obtained with MCMC at t=100, 255 and 400th iterations. The size of each point corresponds to the number of particles with this coordinates (each particle has the same weight). The gray line traces the order of sampling. The chain was initiated form far from the distribution mode, but was able to converge to the true posterior.

In the generalized scheme the vector of parameter x is divided into subsets $x_1, x_2 \ldots x_m$. During each iteration only a single component x_i, $i = i(t)$ is updated.

$$x_i^* \sim g_i(x_i^*|x^t)$$
$$x^* = (x_1^t \dots x_{i-1}^t, \quad x_i^*, \quad x_{i+1}^t \dots x_m^t)$$
$$x^{t+1} = \begin{cases} x^* & \text{with probability} \quad min\ 1, \dfrac{p(x^*|y)g_i(x_i^t|x^*)}{p(x^t|y)g_i(x_i^*|x^t)} \\ x^t & \text{otherwise} \end{cases} ;$$

here $g_i()$ is the proposal function for i^{th} component, x^* is a parameter vectors containing the proposed value of x_i.

In a random walk MCMC the acceptance probability is simplified to posterior ratio:

$$min\ 1, \frac{p(x^*|y)}{p(x'|y)}$$

In a Gibbs Sampler the acceptance probability is always equal to 1, i.e. all proposals are automatically accepted.

MCMC has different names depending on what proposal distribution is used. The general formula is referred to as Metropolis-Hastings method. The MCMC with symmetric proposal distribution $g_i(x^*|x) = g_i(x|x^*)$ is called random walk MCMC. The MCMC where the proposal distribution is equal to the marginal posterior $g_i(x^*|x) = p(x^*|y)$ is called Gibbs sampler.

Accept-Reject. In this method, we first sample a parameter x^* from its prior $x^* \sim p(x)$, then sample a pseudodata y^* using the model $y^* \sim p(y|x^*)$. If the generated pseudodata is identical to the real data $y = y^*$, the parameter x^* is accepted. Otherwise, it is rejected. The target distribution $p(x|y) \sim p(x)p(y|x)$ is represented by a set of unweighed accepted samples x^*.

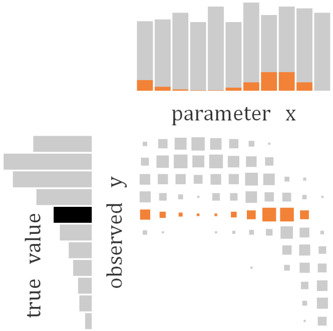


Figure 17: Accept-Reject method. All generated pairs of <parameters, pseudodata> are shown in gray. Accepted pairs are shown in orange.

On a bright side, Accept-Reject method does not ask to compute any probability densities. We do not need to know any mathematics behind the model, we just need to be able to sample from it. This makes Accept-Reject applicable for so-called intractable models (i.e. models where likelihood cannot be computed easily) such as very complex models and black boxes. On a negative side, Accept-Reject does not work for any realistic problem as a chance of hitting x with x^* is practically zero.

Approximate Bayesian Inference (ABC). To bring Accept-Reject to practical use, two layers of approximation are suggested [18, 19]. First, instead of comparing the data and pseudo-data we are comparing their summary statistics $ss(x)$ and $ss(x^*)$. Second, instead of a perfect match $ss(x) = ss(x^*)$ we require similarity. The parameter x^* is accepted if $d(ss(x), ss(x^*)) < \varepsilon$, here d is a distance and ε is a pre-set threshold.

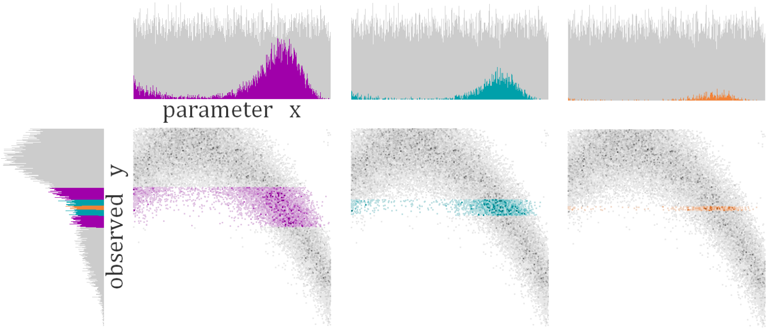


Figure 18: ABC method. Three sets of particles obtained for different ε. All generated pairs of <parameters, pseudodata> are shown in gray. Accepted pairs are shown in magenta (big ε), cyan (mid ε) and orange (small ε).

ABC opens a whole world of intractable models to be explored. It also raises a set of questions: how to choose a distance, summary statistics, threshold ε.

Nested Computational Methods

Sometimes we need to estimate a probability distribution $p(x|y)$ such that $p(x|y) = \int p(x,h|y)dh$ where $p(x,h|y)$ is still analytically intractable, but is easier to handle. In this situations one could use a nested computational method. The inner algorithm approximates $p(x,h|y)$ while the outer algorithm studies the distribution $p(x|y)$ using the estimates from the inner algorithm. This adds another layer of approximation errors, but enables addressing more complex problems. There are multiple examples of nested methods:

Particle Gibbs uses MCMC as an outer method and Particle filter as an inner method [20]. The proposal function takes one of the particles left after the work of a particle filter, thus reducing a computational burden.

Exact Approximate MCMC uses MCMC as an outer method [21, 22]. Any numerical approximation can be used inside. If the inner approximation satisfies a certain condition, the whole method still converges to the true distribution (thus the name).

Bayesian Optimization for Likelihood-Free Inference (BOLFI) uses Bayesian quadrature as an outer method, ABC as an inner method [23].

Integrated Nested Laplace Approximation (INLA). If the target distribution comes from the class of Latent Gaussian models, INLA uses a set of certain complex methods to approximate it [24].

Sequential ABC uses ABC, sequentially adopting the prior distribution in a manner of a particle filter [25].

What computational methods are used in the present articles?

Article I uses Gibbs Sampler. The method was implemented in `Python`.

Article II uses several algorithms. MCMC was used as a main tool. Exact-approximate MCMC was used to check the convergence of the main method. Optimization was used to set an initial position for MCMC and to perform a sensitivity analysis. The method were implemented in `Python` with bottleneck parts written in `C`.

Article III uses a Gibbs Sampler implemented in `BUGS`.

Article IV presents an optimization function written in `R`.

Article V discusses a summary statistics for ABC.

Method Criticism

It may not be immediately clear if a chosen computational method fits the problem or if amount of generated samples is high enough. Moreover, computational methods, like other computer programs, may contain bugs. Tests should be performed. There are multiple testing methods ranging from a visual inspection to formal criteria. In the following list I summarize commonly used methods. Note that sometimes dysfunctional computational methods could be a sign of inadequate model.

Posterior predictive check tests if the estimated posterior distribution actually corresponds to the observed data. It generates pseudo-observations using the values of the parameters sampled from the posterior and compares them to the real observations.

Leave-data-out class of approaches relies on separating the dataset into subsets $y = (y_1, y_2)$. The inference is made using only the training part y_1 by learning the distribution $p(x|y_1)$. This posterior is then used to predict the known test dataset y_2.

Mock-up data analysis does not require the actual data y. The mock data y^* are generated from the model with pre-set parameter values x'. The computational method is then used to infer back the x'. If the method fails this task, there is a fare chance it would not succeed with a real problem too.

Several independent runs are important for sequential methods such as MCMC what are in danger of locking themselves in a local maxima of the posterior. To control for this, sequential methods are usually repeated several times starting from different initial points.

Let $\overset{\approx}{p}\ [a|b]$ denotes a computational estimate of a distribution $p(a|b)$ and $p(a|b \sim g(b))$ denotes a probability distribution of a given that b is sampled from $g(b)$. Then we can formalize a posterior predictive check as estimating the probability

$$p(y|x \sim \overset{\approx}{p}\ [x|y]);$$

leave-data-out as estimating the probability

$$p(y_2|x \sim \overset{\approx}{p}\ [x|y_1]);$$

and mock-up data analysis as estimating the probability

$$\overset{\approx}{p}\ [x'|y^* \sim p(y|x')].$$

What tests are used in the research articles? In the articles I, II and IV mock-up data test and independent runs were applied. In addition, in Article II the same posterior distribution was estimated by several different methods. Article V used independent runs.

Research articles

This chapter summarizes the research articles presenting their context, research goals and results.

Influenza Epidemiology

In Finland 2009-2011, A(H1N1)pdm09 influenza (also known as 'swine flu') caused two epidemic seasons. The first season was part of the global A(H1N1)pdm09 pandemic and received lots of attention from media and medical institutions. The second season was classified as a normal influenza winter outbreak. A national vaccination campaign was undertaken in 2009-2010.

Multiple control measures can be utilized against an epidemic: vaccination, quarantine, school closures etc. To choose the best countermeasure decision makers should understand how dangerous the infection is, how it spreads, which social groups are most vulnerable and so on. To address these questions it is important to know the true number of infections. However, for such large scale disease as influenza estimating the true incidence may be problematic.

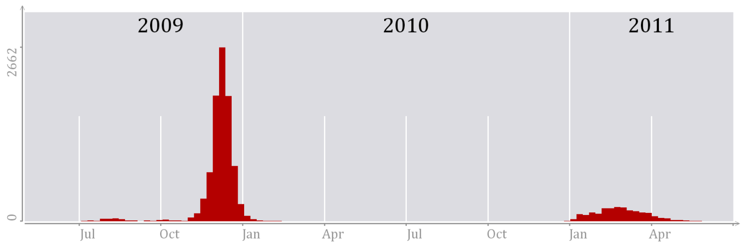


Figure 19: The numbers of confirmed A influenza cases during two seasons in Finland in 2009-2011

In Finland 2009, several registers were collecting the data on the influenza spread. But the registers alone could not reveal the whole picture due to the high underreporting rate: only a fraction of infected individuals shows any symptoms and only a fraction of these seek medical care; only a fraction of doctors tests their patients for A(H1N1)pdm09 and submits results to the registry. The probabilities of these events may depend on the age of a sick person, severity of a sickness, geographical region and time of infection, since the public concern about the pandemic changed with time. Underreporting causes the *fundamental unidentifiability* of the true attack rate. It is impossible to tell a widespread hard-to-detect infection from a rare but easy-to-detect one using only the number of registered cases. Additional sources of information should be used.

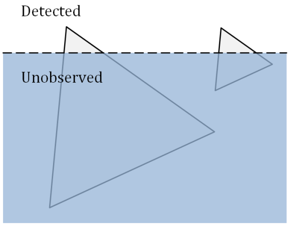


Figure 20: The fundamental unidentifiability of the registries: it is impossible to estimate the true number of infections using only the registered data.

Bayesian approach solves the problem of fundamental unidentifiability by amalgamating several sources of information. In addition to registers one could use experts opinion, general understanding of how the influenza spreads, estimates of the influenza burden in other countries and during other pandemics. This approach is sometimes referred as Bayesian evidence synthesis [26]. The epidemiological models used to study influenza pandemics can be classified as *static* or *dynamic*. In static models, the infection pressure is captured as a probability of becoming infected during the whole season [27, 28, 29]. In dynamic models, the spread of the infection is modelled explicitly [30]. The static approach requires less computational resources and less specific knowledge, while the dynamic one enables to address more complex questions.

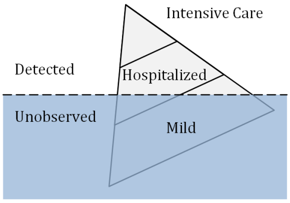


Figure 21: The concept of tilted iceberg: more severe infections have a bigger chance to be detected.

Article I – Estimating the burden of A(H1N1)pdm09 influenza in Finland during two seasons.

This is a research paper aiming at estimating the true burden of 'swine flu' epidemic, i.e. the true number of infected during the influenza outbreaks.

The Article uses a static model. Two seasons are modelled, vaccination placed between seasons. The true numbers of infections are estimated for each age group and geographical region. Computations took about two days on a server computer.

The main methodological feature of this paper is a Bayesian evidence synthesis. We have combined several data sources, including registers of different reliability, experts opinion and a literature review. This allowed us to bypass the fundamental unidentifiability of register data and estimate the attack rate in the population. We conclude that totally 5% of the population are estimated to be infected during two seasons. This means that only 1 infection per 25 was detected.

While the main text of the paper concentrates on epidemiological details, supplementary material (included) covers the details of modelling and computational methods.

Article II – Revealing the true incidence of pandemic A(H1N1) pdm09 influenza in Finland during the first two seasons - an analysis based on a dynamic transmission model.

This is a research paper aiming at:

- improving the results of Article I using more accurate model;
- estimating the effect of the vaccination campaign: how many potential infections were prevented by conducting the vaccination;
- developing a general methodology what could be applied for other epidemics.

The Article uses a dynamic model. To simulate the spread of infection we used contact rates estimated in Finland by Polymod survey [31]. Stratification by geographical regions was abandoned due to computation burden. Computations took about two months on a 16-core cluster computer.

The dynamical model in combination with several data sources and informative priors allowed us to bypass the fundamental unidentifiability of register data and estimate the true attack rate. This estimate was larger than in Article I: 500 000 individuals (9% of the population). Dynamical model was able to simulate a scenario where vaccination campaign had never started. In this case, the study predicts, the incidence could reach as high as 50% of the population.

The main methodological contribution of this Article is introduction of time-dependent model parameters. The transmissibility of the virus changes with time, representing the weather variation, public holidays and population response to the epidemic. The detection probability for a mild infection changes with time, representing shifts in public and governmental concerns about the 'swine flu' epidemic.

Six appendixes for this Article cover estimating the contact rates (Supplement 1), modelling (2), methods (3), present additional results (4), model and method criticism (5) and discuss the difference between the continuous and discrete time SIR models (6).

Phenotype Microarrays Data Analysis

Biolog Phenotype micriarrays (PM) is a laboratory equipment capable of multiple parallel screening of bacterial responses to different conditions such as nutrition and antibiotics [32]. The PM acts as multiple parallel Petri dishes: bacteria are placed on a plate containing small wells filled with different substrates. The bacterial metabolic activity is then measured by a by-product of metabolism and recorded in arbitrary units. Measurements are taken automatically for the duration of the experiment (usually several days) producing a time series referred to as a *metabolic signals*. In a benevolent condition bacteria are active, high signal is registered; in a poison or antibiotics bacteria die; only small signal is visible.

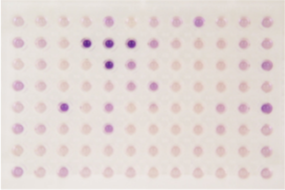


Figure 22: Biolog PM plate. The photo is taken from the Biolog homepage [32]

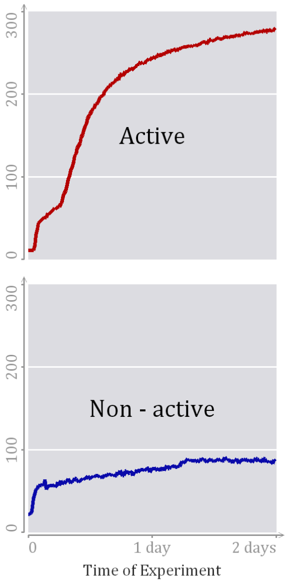


Figure 23: Example of signals produced by metabolically active and non-active bacteria

Profiling bacteria – learning their behaviour in different substrates – can be used to identify them, study their reaction on drugs and gene knockout. However, PM metabolic signals are subjects to measurement noise, both well-specific non-normal noise and plate-wide variation. The bacterial response to a substrate is stochastic. Complexity of an experimental setup and a variation in bacterial metabolism sometimes lead to different metabolic signals in seemingly identical conditions.

A number of different software packages have been developed for analysing and comparing metabolic signals. The simplest methods assign a single statistic (e.g. maximum intensity reached) to a signal [33, 34]. Model-based methods are fitting growth models such as logistic [35], Gompertz [35, 36], and Richards [35] into a signal.

Article III – Novel R pipeline for analyzing Biolog phenotypic microarray data.

This is a software paper presenting an `R` package for analysing PM metabolic signals: clustering them into active and non-active signals, removing the plate-wide variation, identifying effects of experimental conditions. It possesses the following benefits absent in analogous software, available at that time:

- The metabolic signals produced by active and non-active bacteria are fundamentally different. Our package recognizes this difference and uses two models: logistic for active and linear for non-active signals.
- We provided a function for normalizing arrays, i.e. removing plate-wide experimental noise. Active and non-active signals are normalized separately.
- We provided a function for identifying effects of experimental conditions on metabolic signals. The probabilistic model of such effects is used. This part of the package is implemented in `WinBUGS`.

Article IV – Identifying multiple potential metabolic cycles from Biolog experiments.

This is a software paper presenting an `R` package for identifying multiple periods of activity in PM metabolic signals. This is done by decomposing a target signal into a set of growth model, each potentially representing a biologically meaningful event such as a change in bacterial metabolic pathway.

This is the first package able to do so – all analogous software are fitting only a single parametric model into a metabolic signal.

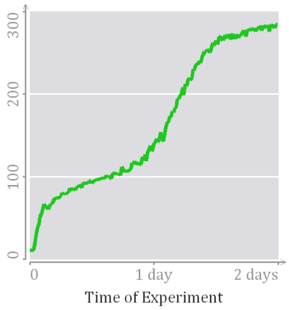


Figure 24: Example of signals with two periods of growth, hinting that bacteria may have undertaken two metabolic cycles during the experiment.

Mismatch Distributions

A mismatch distribution (MMD) is a common summary statistic used to describe genetic diversity of a population. MMD is a probability distribution of the expected number of different genes (i.e. Hamming distance) that two individuals randomly picked from the population would have.

MMD has been proposed for analysing genotype data and conducting inference on population structure, as it could reflect the signs of past events such as bottlenecks or exponential growth [37, 38]. However, learning the MMD of a real population is nearly impossible, as in practice only a few non-independent samples are taken. Simulation-based inference methods (such as ABC) allow to account for a sampling biases.

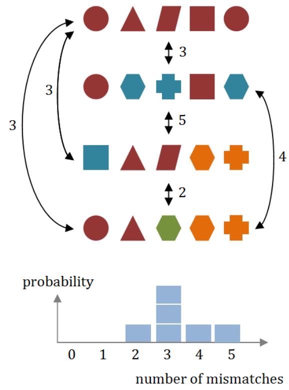


Figure 25: Toy example of the population of four haplotypes, five sites each. The top panel shows the pairwise mismatch distance between haplotypes. The bottom figure presents the mismatch distribution.

Article V – Statistical properties of the allelic mismatch distribution in neutrally evolving haploid populations. This is a research paper. It is based on a neutral evolution model similar to the one used by Numminen et al. [38]. The neutral evolution is affected by the population structure, which is modelled implicitly by introducing stochastic migration and local epidemic outbreaks (microepidemics).

The paper rigorously and comprehensively analyses a dependence of mismatch distribution on parameters of the model. It discusses the possible use of MMD as a summary statistic in ABC. It concludes that MMD is not a comprehensive statistic and should be used only alongside other summaries or informative priors.

Bibliography

[1] S. McGrayne, *The theory that would not die: how Bayes' rule cracked the enigma code, hunted down Russian submarines, and emerged triumphant from two centuries of controversy*. New Haven Conn: Yale University Press, Sep 2011.

[2] E. T. Jaynes, *Probability theory the logic of science*. Cambridge, UK New York, NY: Cambridge University Press, Jun 2003.

[3] D. V. Lindley, "The philosophy of statistics," *Journal of the Royal Statistical Society. Series D (The Statistician)*, vol. 49, no. 3, pp. 293–337, 2000.

[4] J. M. Bernardo and A. F. M. Smith, *Bayesian theory*. Chichester, New York: Wiley, May 2000.

[5] V. Kuznetsov, "Model in the phylosopy of science," in *Phylosophical dictionary (Russian, online)*, 2010. Russian, online: www.lomonosov-fund.ru/enc/ru/encyclopedia.

[6] N. D. Goodman and J. B. Tenenbaum, *Probabilistic Models of Cognition (online)*. Online, http://probmods.org.

[7] N. Cressie and C. K. Wikle, *Statistics for spatio-temporal data*. Hoboken, N.J: Wiley, Apr 2011.

[8] A. Gelman and J. Hill, *Data analysis using regression and multilevel/hierarchical models*, vol. Analytical methods for social research. Cambridge University Press, 2006.

[9] R. Frigg and S. Hartmann, "Models in science," in *The Stanford Encyclopedia of Philosophy (online)*, fall 2012 ed., 2012. Online: http://plato.stanford.edu/archives/fall2012/entries/models-science/.

[10] W. O. Kermack and A. G. McKendrick, "A contribution to the mathematical theory of epidemics," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 115, pp. 700–721, Aug 1927.

[11] O. Diekmann and J. A. P. Heesterbeek, *Mathematical epidemiology of infectious diseases: model building, analysis, and interpretation*. Wiley series in mathematical and computational biology, Chichester, New York: John Wiley, Mar 2000.

[12] D. L. Chao, M. E. Halloran, V. J. Obenchain, and I. M. Longini, “FluTE, a publicly available stochastic influenza epidemic simulation model,” *PLoS Computational Biology*, vol. 6, Jan 2010.

[13] D. J. C. MacKay, *Information theory, inference, and learning algorithms*. Cambridge, UK New York: Cambridge University Press, Oct 2003.

[14] M. Kennedy, “Bayesian quadrature with non-normal approximating functions,” *Statistics and Computing*, vol. 8, pp. 365–375, Mar 1998.

[15] P. Koistinen, “Lecture notes on comutational statistics (online),” 2012. Online: `wiki.helsinki.fi/display/mathstatKurssit/Computational+statistics,+spring+2012`.

[16] P. Djurić, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. Bugallo, and J. Miguez, “Particle filtering,” *IEEE Signal Processing Magazine*, vol. 20, pp. 19–38, Sep 2003.

[17] R. Douc, O. Cappé, and E. Moulines, “Comparison of resampling schemes for particle filtering,” *In 4th International Symposium on Image and Signal Processing and Analysis (ISPA)*, vol. abs/cs/0507025, 2005.

[18] J.-M. Marin, P. Pudlo, C. P. Robert, and R. J. Ryder, “Approximate bayesian computational methods,” *Statistics and Computing*, vol. 22, pp. 1167–1180, Oct 2011.

[19] M. Sunnåker, A. G. Busetto, E. Numminen, J. Corander, M. Foll, and C. Dessimoz, “Approximate bayesian computation,” *PLoS Compututational Biology*, vol. 9, p. e1002803, Jan 2013.

[20] R. H. Christophe Andrieu, Arnaud Doucet, “Particle markov chain monte carlo method,” *Journal of the Royal Statistical Society*, 2010.

[21] D. Wilkinson, “The pseudo-marginal approach to “exact approximate” MCMC algorithms (online),” Sep 2010. Online: `darrenjw.wordpress.com/2010/09/20/`.

[22] A.-M. Lyne, M. Girolami, Y. AtchadĺĘ, H. Strathmann, and D. Simpson, “On russian roulette estimates for bayesian inference with doubly-intractable likelihoods,” *Statistical Science 2015, Vol. 30, No. 4, 443-467*, Nov 2015.

[23] M. U. Gutmann and J. Corander, “Bayesian optimization for likelihood-free inference of simulator-based statistical models,” *Journal of Machine Learning Research*, Aug 2015.

[24] H. Rue, S. Martino, and N. Chopin, “Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 71, no. 2.

[25] E. Numminen, L. Cheng, M. Gyllenberg, and J. Corander, “Estimating the transmission dynamics of streptococcus pneumoniae from strain prevalence data,” *Biometrics*, vol. 69, pp. 748–757, Jul 2013.

[26] A. M. Presanis, R. G. Pebody, B. J. Paterson, B. D. Tom, P. J. Birrell, A. Charlett, M. Lipsitch, and D. De Angelis, "Changes in severity of 2009 pandemic A/H1N1 influenza in England: a Bayesian evidence synthesis," *BMJ*, vol. 343, p. d5408, Sep 2011.

[27] M. D. Van Kerkhove, S. Hirve, A. Koukounari, A. W. Mounts, R. Allwinn, D. Bandaranayake, A. Bella, A. Bone, F. Carrat, M. S. Chadha, M. Chen, C. Y. Chi, C. M. Cox, M. Cretikos, N. Crowcroft, J. Cutter, X. de Lamballerie, K. Dellagi, G. Doukas, S. Dudareva-Vizule, A. M. Fry, G. L. Gilbert, W. Haas, P. Hardelid, P. Horby, Q. Sue Huang, O. Hungnes, N. Ikonen, K. Iwatsuki-Horimoto, N. Z. Janjua, I. Julkunen, J. M. Katz, Y. Kawaoka, A. Lalvani, D. Levy-Bruhl, H. C. Maltezou, J. McVernon, E. Miller, A. Mishra, M. Moghadami, S. D. Pawar, C. Reed, S. Riley, C. Rizzo, L. Rosella, T. M. Ross, Y. Shu, D. M. Skowronski, S. Sridhar, A. Steens, B. V. Tandale, M. Theodoridou, M. van Boven, K. Waalen, J. R. Wang, J. T. Wu, C. Xu, S. Zimmer, C. A. Donnelly, and N. M. Ferguson, "Estimating age-specific cumulative incidence for the 2009 influenza pandemic: a meta-analysis of A(H1N1)pdm09 serological studies from 19 countries," *Influenza and Other Respiratory Viruses*, Jan 2013.

[28] A. M. Presanis, R. G. Pebody, P. J. Birrell, B. D. M. Tom, H. K. Green, H. Durnall, D. Fleming, and D. De Angelis

[29] A. Steens, S. Waaijenborg, P. F. Teunis, J. H. Reimerink, A. Meijer, M. van der Lubben, M. Koopmans, M. A. van der Sande, J. Wallinga, and M. van Boven, "Age-dependent patterns of infection and severity explaining the low impact of 2009 influenza A (H1N1): evidence from serial serologic surveys in the Netherlands," *American Journal of Epidemiology*, vol. 174, pp. 1307–1315, Dec 2011.

[30] P. J. Birrell, G. Ketsetzis, N. J. Gay, B. S. Cooper, A. M. Presanis, R. J. Harris, A. Charlett, X. S. Zhang, P. J. White, R. G. Pebody, and D. De Angelis, "Bayesian modeling to unmask and predict influenza A/H1N1pdm dynamics in London," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 108, pp. 18238–18243, Nov 2011.

[31] J. Mossong, N. Hens, M. Jit, P. Beutels, K. Auranen, R. Mikolajczyk, M. Massari, S. Salmaso, G. S. Tomba, J. Wallinga, J. Heijne, M. Sadkowska-Todys, M. Rosinska, and W. J. Edmunds, "Social contacts and mixing patterns relevant to the spread of infectious diseases," *PLoS Medicine*, vol. 5, p. e74, Mar 2008.

[32] "Biolog homepage." `http://www.biolog.com/`.

[33] B. R. Bochner, "New technologies to assess genotype-phenotype relationships," *Nat. Rev. Genet.*, vol. 4, pp. 309–314, Apr 2003.

[34] B. R. Bochner, "Global phenotypic characterization of bacteria," *FEMS Microbiol. Rev.*, vol. 33, pp. 191–205, Jan 2009.

[35] L. A. Vaas, J. Sikorski, B. Hofner, A. Fiebig, N. Buddruhs, H. P. Klenk, and M. Goker, "opm: an R package for analysing OmniLog(R) phenotype microarray data," *Bioinformatics*, vol. 29, pp. 1823–1824, Jul 2013.

[36] M. DeNittis, A. Querol, B. Zanoni, J. L. Minati, and R. Ambrosoli, "Possible use of Biolog methodology for monitoring yeast presence in alcoholic fermentation for winemaking," *J. Appl. Microbiol.*, vol. 108, pp. 1199–1206, Apr 2010.

[37] W. J. Ewens, *Mathematical population genetics 1. Theoretical introduction*. New York: Springer, 2004.

[38] E. Numminen, M. Gutmann, M. Shubin, P. Marttinen, G. Méric, W. van Schaik, T. M. Coque, F. Baquero, R. J. Willems, S. K. Sheppard, E. J. Feil, W. P. Hanage, and J. Corander, "The impact of host metapopulation structure on the population genetics of colonizing bacteria," *Journal of Theoretical Biology*, vol. 396, pp. 53–62, May 2016.