Bayesian Demography with **iNZight**

Tom Elliott[1,2] and John Bryant[2]

[1]School of Health, Victoria University of Wellington
[2]Department of Statistics, University of Auckland
[3]Bayesian Demography Ltd

June 2021

1 Introduction

Demographic data are often most useful when they are disaggregated. Data on mortality rates, for instance, are more likely to lead to better insights and better policies if they are disaggregated by ethnicity and locality, instead of being reported only at the national level. Businesses, community organizations, and city councils need data about their own communities, not the country as a whole.

Demographers and statisticians have, in recent years, made substantial progress on methods for analysing disaggregated data. Traditional techniques for demographic estimation and forecasting struggle to deal with small numbers and random variation. New techniques provide excellent options for smoothing through the random variation, and for quantifying the associated uncertainties. These new techniques are, however, often inaccessible to potential users without advanced skills in statistics and computing.

Implementing these techniques in **iNZight** (Elliott et al., in press), a graphical user interface (GUI) written with R, makes them accessible to a much wider group of users. Doing so, however, requires some careful design choices. The challenge is to steer users through the process of building complex models without overwhelming them, and finding the appropriate balance between flexibility, robustness, and ease of use.

In this paper, we describe a prototype of an **iNZight** module for Bayesian demography. The paper provides a brief introduction to Bayesian demography

and to **iNZight**. It then describes the implementation details, and illustrates the use of the module through two case studies.

2 What is Bayesian demography?

Many problems in applied demography, from measuring changes in life expectancy, to forecasting births, to estimating populations in small areas, have a common structure. The data come in the form of tables of cell counts, rather than the classic rows-for-cases, columns-for-variables format.[1] Demographic data, in other words, look like

	Female	**Male**
Age 0–4	34	82
Age 5–9	17	23
Age 10+	45	41

rather than

Person	**Age**	**Sex**
1	3	Female
2	14	Male
3	5	Male
4	7	Female
5	10	Female
6	12	Female
7	0	Male

The number of dimensions for demographic data is typically small, especially when compared with the datasets used in much of modern statistical modelling and machine learning. However, the level of detail required for each dimension can be large: for instance, analyses of mortality often use over 101 separate age groups, and there is strong interest in variation across all these age groups. Moreover, there is often interest in interactions: how age profiles differ between females and males, for instance, and how these differentials evolve over time.

[1] R users may be familiar with this being referred to as 'tidy' format.

The distinctive datasets and questions of demography have given rise to distinctive methods, so that there has traditionally been only limited overlap between applied demography and mainstream applied statistics. However, as the toolkit of mainstream applied statistics has expanded, and as demographers have begun to grapple with problems such as sampling variation or the analysis of time series, there has been extensive cross-fertilization between demography and statistics (Alho and Spencer, 2005).

Although statistical demography in general has been developing quickly in recent years, *Bayesian* approaches to statistical demography have grown particularly fast (Bijak and Bryant, 2016). Bayesian statistics is an alternative to the "classical" or "frequentist" approaches that dominated statistical practice for much of the twentieth century. From a theoretical point of view, the distinctive feature of Bayesian statistics is its willingness to represent all forms of uncertainty in quantitative terms through probability distributions. But for applications, the distinctive feature of Bayesian statistics is its extreme flexibility, which allows it to handle complex problems that might otherwise be intractable.

The most prominent example of Bayesian demography is the national and global population forecasts by the United Nations, which are done using Bayesian methods (Gerland et al., 2014). However, Bayesian methods have also been widely used at the subnational level, such as for small area estimation and forecasting of fertility, mortality, and migration rates (e.g. Alexander et al., 2017; Schmertmann and Gonzaga, 2018; Burstein et al., 2019; Zhang and Bryant, 2020). Bayesian approaches have proven particularly useful in demographic applications involving small sample sizes, measurement error, and forecasting.

3 Bringing Bayesian demographic methods to a wider group of users

Tools for general-purpose Bayesian computing have been improving fast (e.g. Carpenter et al., 2017; Salvatier et al., 2016). However, using these tools to design models for the distinctive problems of applied demography requires programming and statistical skills that are not common among applied practitioners.

R packages providing facilities for Bayesian demographic modelling have begun to appear. The team responsible for the United Nations population projections has published open source R packages implementing all their methods

(Ševčíková and Raftery, 2016). Another important example is the **SUMMER** package, which implements a specific class of models for estimating childhood mortality in small areas (Li et al., 2021).

There are also some excellent non-Bayesian packages for statistical demography. The **StMoMo** package, for instance, implements a large number of different models for mortality at all ages (Villegas et al., 2018). The **demography** package implements models for fertility, mortality, and migration, and does population projections, though only for data that have dimensions "age", "sex", and "time" (Hyndman, 2019).

Packages **dembase**, **demest**, and **demlife**, which were developed at Statistics New Zealand with contributions from external collaborators, provide general facilities for Bayesian demographic modelling. The **dembase** package provides tools for manipulating demographic data, the **demest** package does Bayesian model-fitting and forecasting, and the **demlife** package uses estimates of mortality rates to construct life tables and estimates of life expectancy. There are no restrictions over the dimensions that are included in the models, and the estimation models cover a wide variety of use cases in applied demography (Bryant and Zhang, 2019). The statistical models come with defaults that give acceptable performance for many problems. All three packages are open source, and are on the Statistics New Zealand GitHub repository, `://github.com/statisticsnz/R`. The packages are currently used to produce official statistics on mortality, and further applications are being investigated.

Although the **dembase**, **demest**, and **demlife** packages do not require advanced skills in demography or statistics, they do require some proficiency with R. Even this requirement is an important barrier for many potential users of Bayesian demographic methods. Small national statistical offices, community or local-government organizations, or small businesses, for example, often do not have the resources to support specialist R programmers. These types of users can, however, quickly learn to use **iNZight**. Adding functionality for Bayesian demography to **iNZight** is therefore a practical way of bringing these methods to a much wider audience.

3.1 An introduction to iNZight

iNZight is a free, open-source GUI built with R for visualising and analysing data without the need to code. The point-and-click interface provides easy access to common data visualisation techniques, statistical analysis methods, and

data wrangling, and is well suited to both education and applied settings. For the latter, the simplicity of design makes **iNZight** ideal for organisations with limited resources, enabling analysts without coding skills to perform common statistical procedures with ease.

Recently, **iNZight** has been equipped with an *add-on* system, making it extensible to new data types and analytic methods. It works by allowing developers to write a simple, self-contained GUI interface using **gWidgets2** (Verzani, 2019) with input controls for unique graphical and analytic outputs. Users just need to install the add-on to **iNZight** to gain the new functionality on top of what **iNZight** already offers (for example importing and wrangling data).

3.2 A Bayesian demography module for iNZight

An important advantage of building on top of an existing, extensible system such as **iNZight** instead of developing a standalone app is the ability to take advantage of all the existing functionality. This includes common processes such as data import and wrangling tools (including aggregation for standard and survey data). However, it also comes with an existing build and deploy system (**iNZight** features a standalone Windows installer that does not require users to install or interface with R).

A GUI interface to complex modelling packages like **demest** does not need to offer the full set of features—the vast majority of users only need a small set of features and good defaults for everything else to do what they need. For this reason, we put together an **iNZight** module interfacing with the **dembase**, **demest**, and **demlife** packages introduced above. Currently, the prototype implements Normal, Poisson, and Binomial models for demographic data, with additional methods for mortality data (namely life expectancy calculations using the **demlife** package).

3.2.1 Interface layout

The module is structured in a way that asks for the least amount of information from the user as possible before providing "guesses" that can be overridden when necessary. The interface is divided into several sections, each collecting information needed for the next. In this way, users are not overwhelmed with too many decisions at once—instead, they simply fill out each input as it appears.

The first panel is where the overall model type is specified, which starts by asking the user to specify the response variable in the data set. In many cases,

this will be `counts` or similar, but may be something more informative such as `deaths` or `income`. For the latter, the module will attempt to detect the model type from the variable. For example, a variable called `deaths` will automatically apply a *deaths* model, which (in the future) will provide special presets for the model parameters. If **iNZight** cannot decide on a model type, the user can easily pick one from the drop-down. Based on the model chosen (manually or automatically), the response framework is chosen (Normal, Poisson, or Binomial) describing how we will model the response. In our example, the response `deaths` will be modelled using a Poisson distribution, which is standard for (mortality) rates.

After choosing the response and modelling framework, the next section allows users to modify the variables used in the model. Check boxes control the inclusion of individual variables, alongside information about their types to be used in the model. Confirming the variables automatically displays a plot of the raw response (counts or measurements) by the chosen explanatory variables, giving the user an opportunity to check everything looks sensible, and make an initial visual exploration of the data.

The next section allows users to specify the remaining parts of the model. This includes the formula for the (transformed) response as a function of the explanatory variables, using the standard R formula syntax, including interactions with the colon (`:`) or asterix (`*`). Examples are given in Section 4.

Specifying the formula automatically generates a list of parameters in the model, each of which can be given its own hyper-prior. By default, the module uses the defaults supplied by **demest** (e.g., exchangeable priors are used for `sex` and `region` variables) but optionally a dynamic linear model (DLM) can be fitted to any of the terms. In the prototype, DLMs (with optional trend and damping) are the only additional prior, but future versions will have a greater range of options.

Within the **iNZight** prototype module, users must manually specify simulation parameters, as shown in Fig. 4. However, future versions will use updated versions of the **demest** package(s) and automatically run the Markov chain Monte Carlo (MCMC) simulations until convergence is obtained. This is a necessary condition for an interface targeting non-expert users. Once the simulations are complete, the results are shown automatically in the graph as the posterior median and credible regions for each of the observed counts, along with the naïve estimate based on raw counts or proportions. Within the prototype, there are limited features for working with the resulting object, namely

viewing the results (default), as well as looking at the posterior distributions of various model parameters by choosing them from the menu.

Once the MCMC simulation has finished, the graph is updated with the posterior fit for the underlying rate/measurement, overlaid with the naïve estimates. This gives a quick visual guide of how well the model fits the data. Posterior distributions of individual parameters can be explored using the parameter tree. Additionally, if a time variable was included, forecasts can be obtained.

All of this is performed using simple, intuitive GUI interface widgets, such as drop-down boxes, sliders, and text inputs. At no point do users need to interact with code (except arguably when specifying the model formula). The hope is that this will fulfill the basic needs of users who need only do simple demographic inference occasionally.

4 Bayesian demography with iNZight

To demonstrate the new **iNZight** module, we demonstrate two demographic analyses on pre-tabulated data using **iNZight**. Model details can be found on the Github repository `https://github.com/terourou/small-area-estimation` (in the `vignettes` directory).

4.1 Life expectancy

Our first example is annual estimates of life expectancy in Iceland, by sex, for the period 2015–2019. The input data on deaths and population come from the Statistics Iceland online database.[2] Given the small size of the Icelandic population (360,100 people in 2019), cell counts in the deaths and population data are small. The average number of deaths in each combination of age, sex, and year is 16, and 8% of combinations have counts of 0. Traditional methods for estimating mortality and life expectancy would struggle with data this disaggregated.

The data is first formatted into an R data frame with a column for age, sex, and year, plus columns for the count and total population size in each combination of the variables.[3] The result after importing into **iNZight** is shown

[2] Tables *Deaths by municipalities, sex and age 1981-2019* and *Population by municipalities, sex and age 1 January 1998-2020 - Current municipalities*, accessed on 9 February 2021.

[3] **iNZight** works with 'tidy' data.

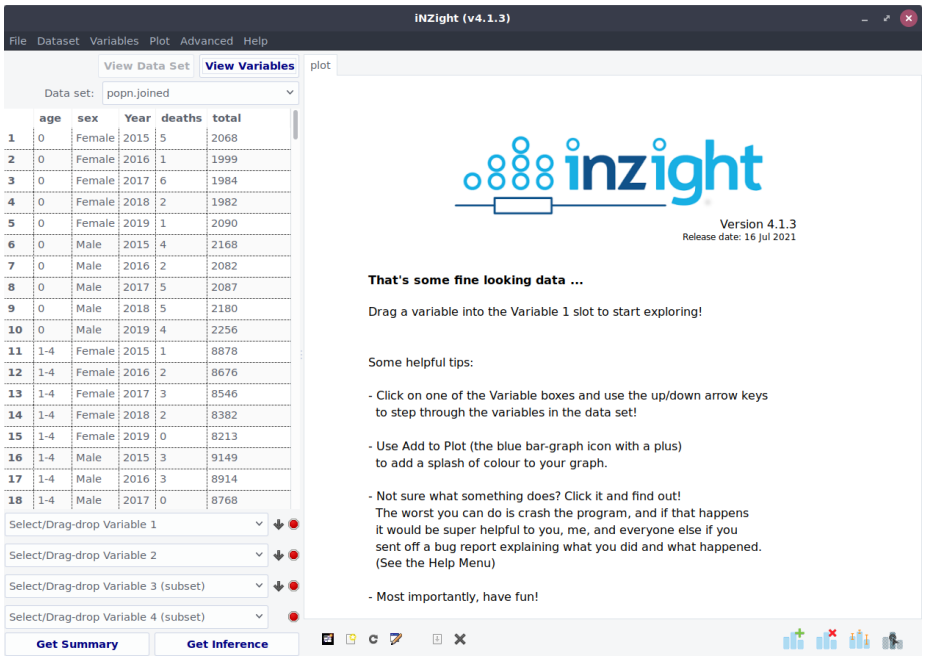

Figure 1: iNZight main window with life expectancy data loaded.

in Fig. 1. We can construct a hierarchical model to estimate life expectancy from the death counts in each cell by starting the small area estimation module from **iNZight**'s **Advanced** menu.

First, the response (`deaths`) is chosen from the variable drop-down, which automatically populates the model type and framework based on the variable name. After choosing `total` as the *exposure* variable, the first part of the model specification process is complete, which is registered by clicking the **Save** button. Next we choose which additional variables to include in the model (by default all are included), and, after saving, **iNZight** produces a graph of the raw data (which can be tweaked in the **Plot modifications** panel). The first three panels are displayed, along with the graph generated, in Fig. 2. Behind the scenes, **iNZight** converts the 'tidy' data to a pair of demographic arrays (see Section 2) for death counts and totals.

We now move to model specification which, by default, uses a linear combination of the explanatory variables (age, sex, and year) to model the underlying mortality rate in each cell. This is easily modified using standard R formula syntax, as shown in Fig. 3, where we are fitting a complex hierarchical model

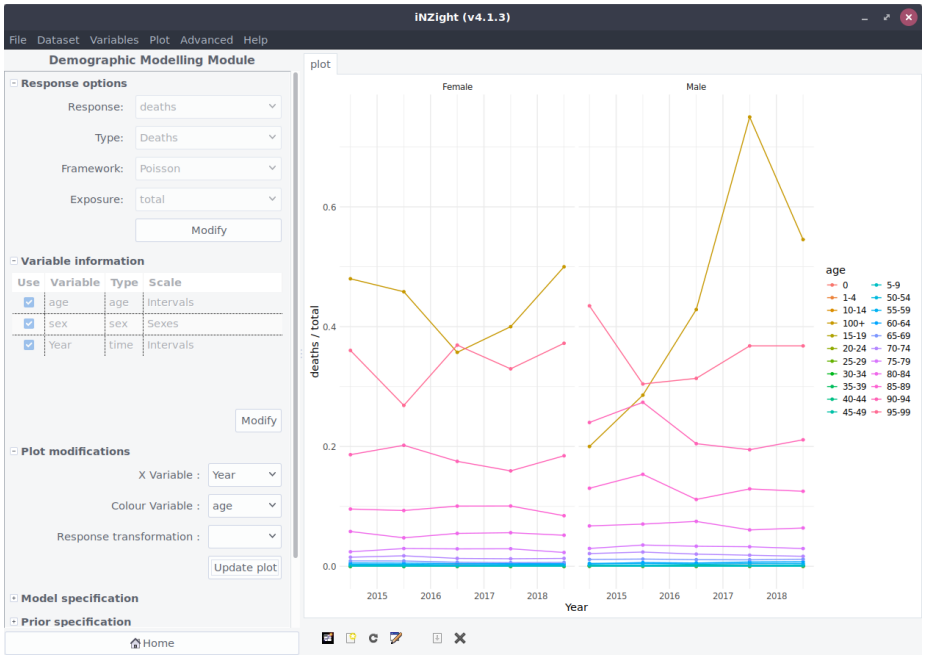

Figure 2: The demographic modelling module displays a simple graph of naïve death rate estimates automatically after specifying the response, framework, and explanatory variables. The Plot Modifications panel (left) allows users to change the x variable and colour variable.

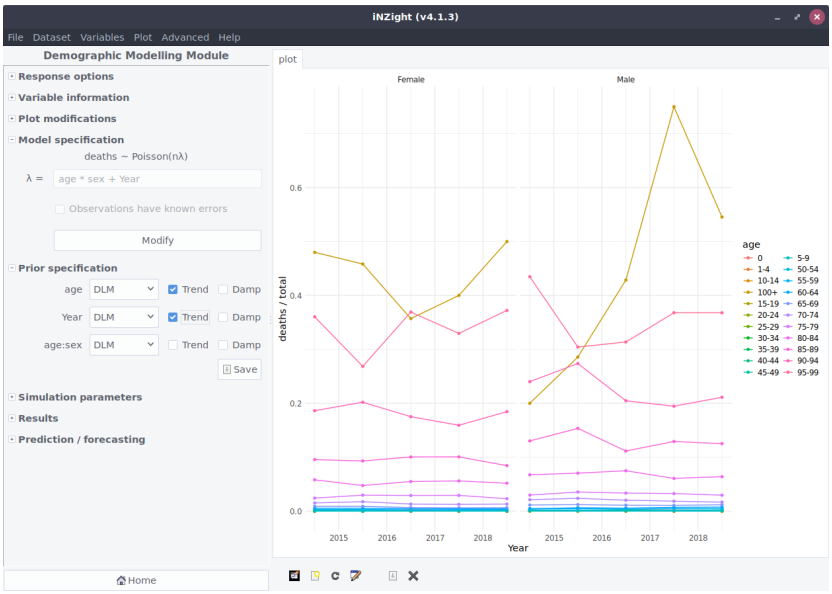

Figure 3: Model specification with the module simply requires specifying the formula for the underlying rate parameter, λ, and specifying the priors for the variables.

of counts y_{ijk} in populations of size N_{ijk}, for age i, sex j, and year k, where the number of deaths is Poisson with rate $N_{ikj}\lambda_{ijk}$. The underlying rate is a function of age, sex, and year, with an interaction between age and sex (using the colon, `:`). After specifying the model, the software presents a list of parameters that can be given priors: currently, the software uses either the defaults provided by **demest**, but may use a DLM instead. DLMs are useful for time-variables (e.g., age and year) where adjacent values are correlated. The model details are handled by the **demest** package behind-the-scenes so the user does not have to.

The standard summary indicator for mortality rates is life expectancy, a complicated non-linear transformation of mortality rates. Using traditional statistical methods, calculating point estimates and error bands for life expectancy is complicated. Using Bayesian methods, however, it is easy. The **demlife** package contains a set of methods for producing life expectancy estimates from death rate estimates, all accessed from a single option in the **iNZight** module. By choosing **Life expectancy** from the drop-down, life expectancies for each sex over time are calculated and graphed, as shown in Fig. 5.

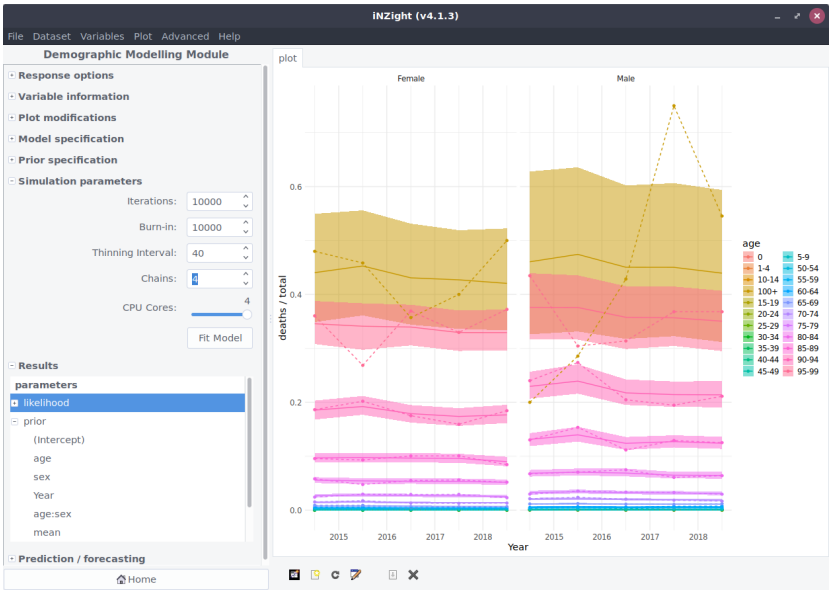

Figure 4: Specification of MCMC simulation parameters, along with the final graph resulting from fitting the Bayesian hierarchical model to the data, showing the posterior mean (solid line), 95% credible interval (shaded region), and naïve estimates (points, dotted lines).

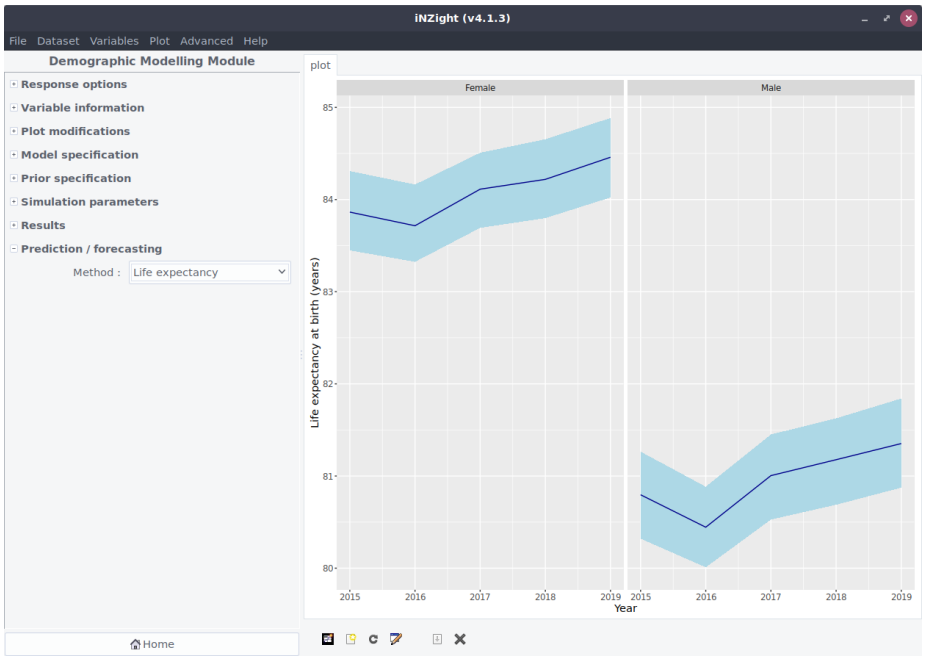

Figure 5: Life expectancy estimate from the model. Traditionally, a complex non-linear transformation is required, but here a simple drop-down selection is all that is required.

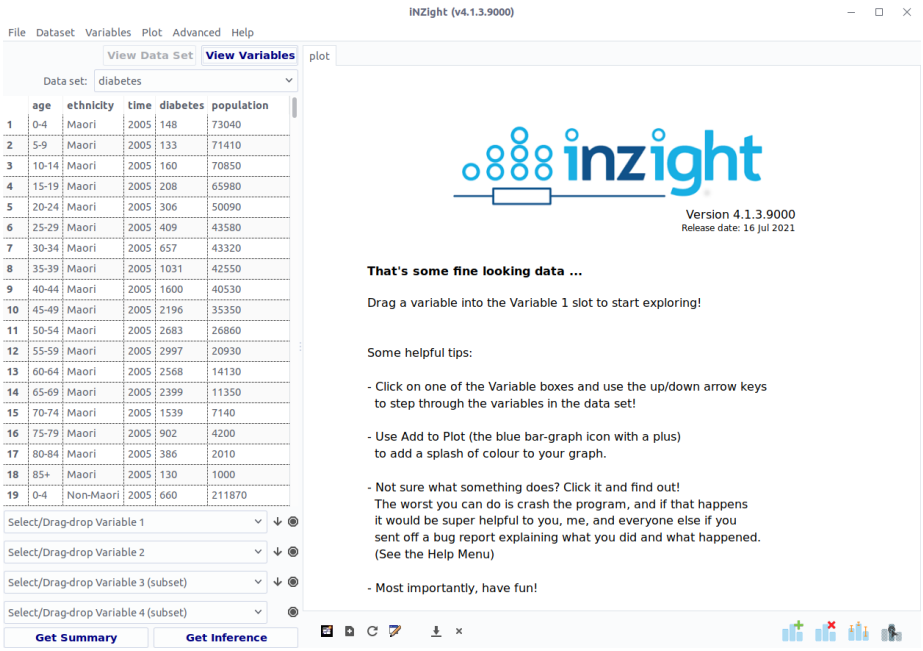

Figure 6: iNZight main window with diabetes data loaded.

4.2 Diabetes

Our second example is the estimation and forecasting of diabetes prevalence in Aotearoa New Zealand, by 5-year age group and by ethnicity (Māori vs non-Māori). The data on numbers of people with diabetes come from the Virtual Diabetes Register, and were supplied as a custom tabulation by the Ministry of Health.[4] The data on population come from the Infoshare database on the Statistics New Zealand website.[5]

Once again, the data is prepared for **iNZight**, with columns for each of the predictive variables *age* (in 5-year intervals), *time* (required for forecasting), and *ethnicity* (Māori or non-Māori). Attached to these categories are the population counts (`population`) and the numbers of people with diabetes (`diabetes`). This is imported directly into **iNZight** as shown in Fig. 6. We will develop a model to estimate the probability of someone in each age group having diabetes, and use the model to forecast rates for the future.

[4] https://www.health.govt.nz/

[5] Tables *Estimated Resident Population by Age and Sex (1991+) (Annual-Jun)* and *Māori Ethnic Group Estimated Resident Population by Age and Sex (1991+)*, accessed from `https://www.health.govt.nz/` on 14 April 2021.

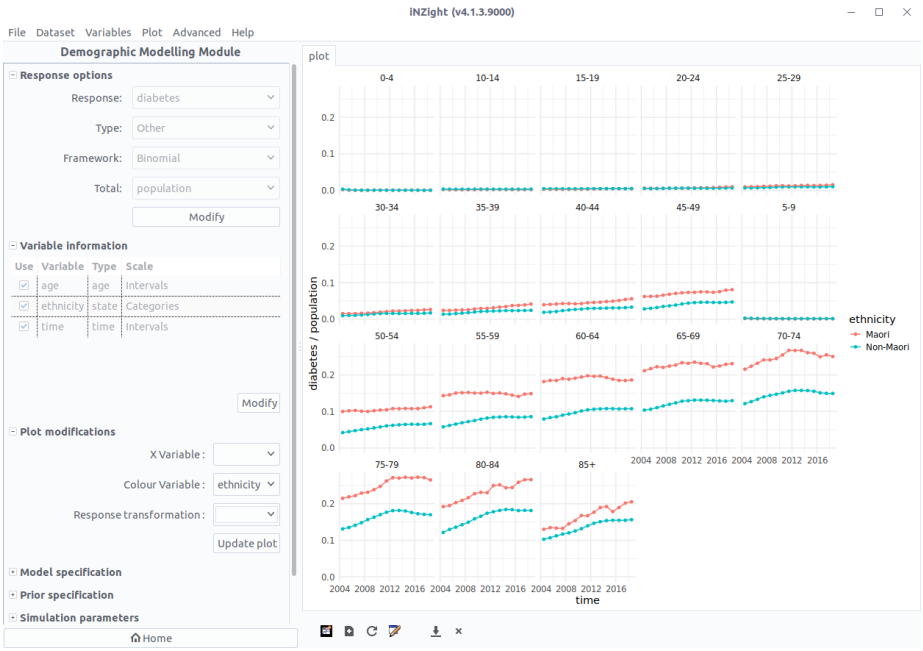

Figure 7: The demographic modelling module displays the basic graph obtained after specifying the response, framework, and explanatory variables. This shows the naïve estimates of diabetes prevalence by ethnicity and age groups over time.

As before, we start by specifying the response information in the top-left of Fig. 7, in this case the number of people in each group with diabetes (`diabetes`). This time, there is no built-in model for diabetes, so we must specify `other` for the response type, and choose `binomial` for the framework. The population total for the binomial probability calculation is the total population in each group (`population`). After confirming the model and the variable information (we are using all three explanatory variables), we get a graph of diabetes prevalence (displayed in Fig. 7). It is clear that Māori from about age 40 experience higher rates of diabetes, and that the overall prevalence of diabetes has been increasing over time. However, to predict what proportion of the population will have diabetes in the future, we need to fit a hierarchical Bayesian model to these data and include multiple DLMs for the various predictors (and their interactions).

The specification of the model this time includes all two-way interactions, specified using R syntax.[6] The response `diabetes` is a binomial outcome, where

[6]This could more succinctly be written using `(age + ethnicity + year)^2`.

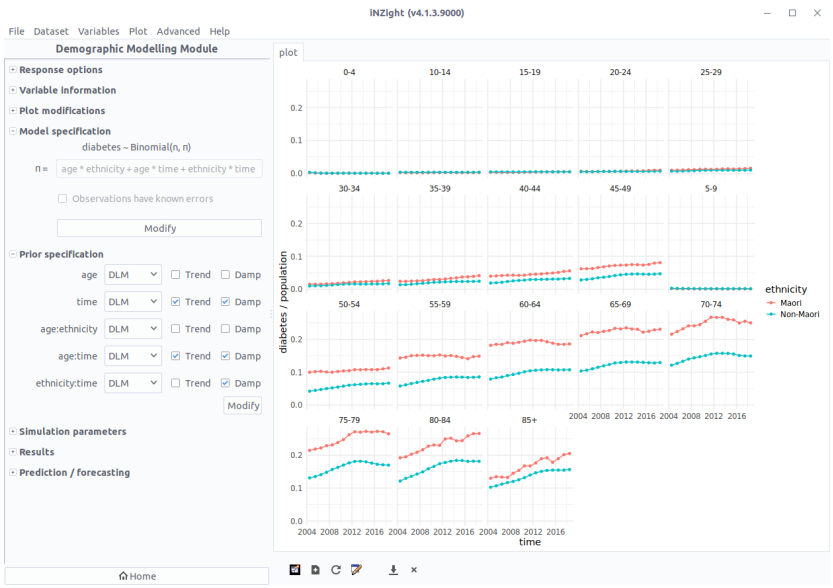

Figure 8: Model specification is a simple process of specifying the formula for the underlying rate parameter, and adding priors for the variables, if desired.

the log odds of having diabetes is a function of age, ethnicity, year, and the two-way interactions `age:ethnicity`, `age:year`, and `ethnicity:year`. A three-way interaction could be included, if required, but this increases computational requirements. For the parameter priors, all but `ethnicity` (categorical) were fitted with DLMs, which allow for forecasting for future values of `year`. The features of the priors are displayed in Fig. 8.

Finally, we can specify the simulation parameters.[7] Clicking the **Fit Model** button compiles the hierarchical model and estimates the posterior distribution using MCMC techniques. The results are displayed automatically once complete, as shown in Fig. 9. Unlike the life expectancy example, the posterior distributions are much more closely aligned to the data, due in part to the additional complexity with interactions and DLMs. This is necessary for making useful forecasts.

Like before, doing the final forecast within the **iNZight** module is as simple as selecting the **Forecast** option from the drop-down. The results, shown in Fig. 10, show that there remains considerable uncertainty—particularly in older age groups—in diabetes prevalence, but the general trend for most Māori groups

[7] Only required for the prototype.

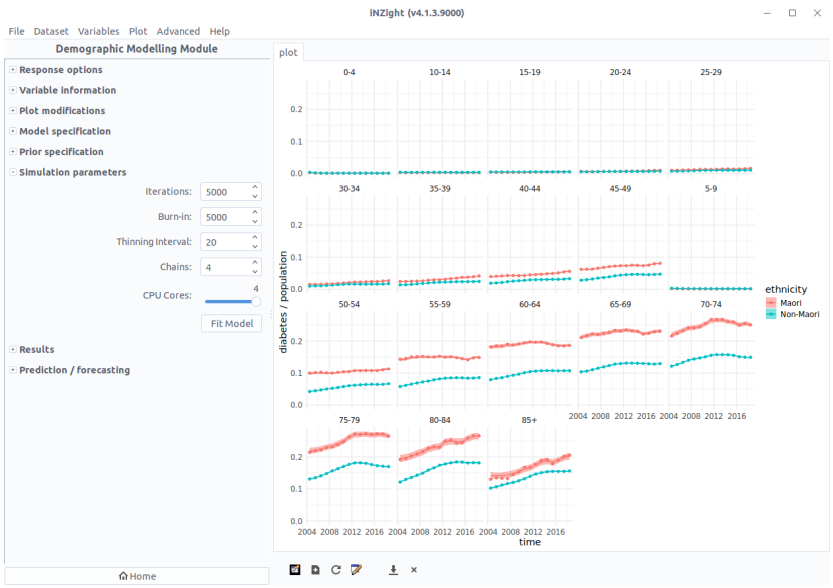

Figure 9: Specification of MCMC simulation parameters, along with the final graph resulting from fitting the Bayesian hierarchical model to the data. The posterior mean (solid line) very closely matches the naïve estimate (points).

suggests . . . a continual increase?, while non-Māori exhibit a plateau or reduction in diabetes prevalence?.

5 Future work

Work is currently underway on a suite of new R packages to replace **dembase**, **demest**, and **demlife**. Development versions of the code are at `https://github.com/bayesiandemography`. The new packages will have an improved interface and new algorithms that should give shorter computation times and require less input from users. Future versions of the **iNZight** Bayesian demography module will incorporate these new features. One particular focus is developing ways of capturing empirical regularities in demographic data, such as standard age profiles for mortality, fertility, and migration rates, and building these automatically into models, with the aim of increasing computation speeds, robustness, and reliability.

Figure 10: Diabetes forecasts by age group and ethnicity.

Acknowledgements

This project was funded by an MBIE Endeavour Grant, ref 62506 ENDRP. The work presented here is a collaboration between Te Rourou Tātaritanga (`https://terourou.org`) and Bayesian Demography Ltd.

Acronyms

DLM dynamic linear model. 6, 10, 14, 15

GUI graphical user interface. 1, 4, 5, 7

MCMC Markov chain Monte Carlo. 6, 7, 11, 15, 16

References

M. Alexander, E. Zagheni, and M. Barbieri. A flexible Bayesian model for estimating subnational mortality. *Demography*, 54(6):2025–2041, 2017.

J. Alho and B. Spencer. *Statistical Demography and Forecasting*. Springer-Verlag, 2005.

J. Bijak and J. Bryant. Bayesian demography 250 years after Bayes. *Population Studies*, 70(1):1–19, 2016.

J. Bryant and J. L. Zhang. *Bayesian demographic estimation and forecasting*. CRC Press, 2019.

R. Burstein, N. J. Henry, M. L. Collison, L. B. Marczak, A. Sligar, S. Watson, N. Marquez, M. Abbasalizad-Farhangi, M. Abbasi, F. Abd-Allah, et al. Mapping 123 million neonatal, infant and child deaths between 2000 and 2017. *Nature*, 574(7778):353–358, 2019.

B. Carpenter, A. Gelman, M. D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. Brubaker, J. Guo, P. Li, and A. Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(i01), 2017. doi: http://hdl.handle.net/10. URL `https://ideas.repec.org/a/jss/jstsof/v076i01.html`.

T. Elliott, C. Wild, A. Sporle, and D. Barnett. **iNZight**: A graphical user interface for data visualisation, exploration, and analysis. *Journal of Statistical Software*, in press. URL `https://inzight.nz/papers/?paper=2021_jss`.

P. Gerland, A. E. Raftery, H. Ševčíková, N. Li, D. Gu, T. Spoorenberg, L. Alkema, B. K. Fosdick, J. Chunn, N. Lalic, et al. World population stabilization unlikely this century. *Science*, 346(6206):234–237, 2014.

R. J. Hyndman. ***demography**: Forecasting Mortality, Fertility, Migration and Population Data*, 2019. URL `https://CRAN.R-project.org/package=demography`. R package version 1.22.

Z. R. Li, B. D. Martin, Y. Hsiao, J. Godwin, J. Wakefield, S. J. Clark, G.-A. Fuglstad, and A. Riebler. ***SUMMER**: Spatio-Temporal Under-Five Mortality Methods for Estimation*, 2021. URL `https://CRAN.R-project.org/package=SUMMER`. R package version 1.1.0.

J. Salvatier, T. V. Wiecki, and C. Fonnesbeck. Probabilistic programming in Python using **PyMC3**, 2016.

C. P. Schmertmann and M. R. Gonzaga. Bayesian estimation of age-specific mortality and life expectancy for small areas with defective vital records. *Demography*, 55(4):1363–1388, 2018.

J. Verzani. ***gWidgets2**: Rewrite of **gWidgets** API for Simplified GUI Construction*, 2019. URL `https://CRAN.R-project.org/package=gWidgets2`. R package version 1.0-8.

A. M. Villegas, V. K. Kaishev, and P. Millossovich. **StMoMo**: An R package for stochastic mortality modeling. *Journal of Statistical Software*, 84(3):1–38, 2018. doi: 10.18637/jss.v084.i03.

H. Ševčíková and A. E. Raftery. **bayesPop**: Probabilistic population projections. *Journal of Statistical Software*, 75(5):1–29, 2016. doi: 10.18637/jss.v075.i05.

J. L. Zhang and J. Bryant. Bayesian disaggregated forecasts: Internal migration in iceland. In *Developments in Demographic Forecasting*, pages 193–215. Springer, Cham, 2020.