How to recommend and teach Bayesian modelling

Stories from a QUT internship with the Department of Health Western Australia

James Hogg

17 November, 2022

Structure/talking points

- Introduction to internship and project
- Modelling work
- `R` code

Introduction

- QUT-approved internship for 3-months
- Paused PhD
- Got paid a **very** nice salary

Supervision

- Industry Partner: Alex Xiao (Department of Health Western Australia (DOH WA))
- QUT: Susanna Cramb

Project details

Improving Health Insights in **Western Australia** with novel **modelling** method development and implementation

- **Improve** internal "Public Health Atlas" using **modelling**
- Help staff gain **modelling skills** to apply to **future** DOH projects

Project aims

- Map **estimates** for **multiple**

Project aims

- Map **estimates** for **multiple**
  - **Types of health data** (Administrative, survey and burden of disease)

Project aims

- Map **estimates** for **multiple**
  - **Types of health data** (Administrative, survey and burden of disease)
  - **Conditions** (cancer, smoking, backpain, etc)

Project aims

- Map **estimates** for **multiple**
  - **Types of health data** (Administrative, survey and burden of disease)
  - **Conditions** (cancer, smoking, backpain, etc)
  - **Epidemiology indicators** (age-standardised rates, standardised incidence ratios, years of life lost, etc)

Project aims

- Map **estimates** for **multiple**
  - **Types of health data** (Administrative, survey and burden of disease)
  - **Conditions** (cancer, smoking, backpain, etc)
  - **Epidemiology indicators** (age-standardised rates, standardised incidence ratios, years of life lost, etc)
  - **Years** (up to 10)

Project aims

- Map **estimates** for **multiple**
  - **Types of health data** (Administrative, survey and burden of disease)
  - **Conditions** (cancer, smoking, backpain, etc)
  - **Epidemiology indicators** (age-standardised rates, standardised incidence ratios, years of life lost, etc)
  - **Years** (up to 10)
  - **Geographical levels** (3)
    - Health districts (HD): 34
    - Local government areas (LGA): 138
    - Statistical area level 2 (SA2): 254

Project aims

- Map **estimates** for **multiple**
  - **Types of health data** (Administrative, survey and burden of disease)
  - **Conditions** (cancer, smoking, backpain, etc)
  - **Epidemiology indicators** (age-standardised rates, standardised incidence ratios, years of life lost, etc)
  - **Years** (up to 10)
  - **Geographical levels** (3)
    - Health districts (HD): 34
    - Local government areas (LGA): 138
    - Statistical area level 2 (SA2): 254
- Use **spatio-temporal Bayesian models**

Project aims

- Map **estimates** for **multiple**
  - **Types of health data** (Administrative, survey and burden of disease)
  - **Conditions** (cancer, smoking, backpain, etc)
  - **Epidemiology indicators** (age-standardised rates, standardised incidence ratios, years of life lost, etc)
  - **Years** (up to 10)
  - **Geographical levels** (3)
    - Health districts (HD): 34
    - Local government areas (LGA): 138
    - Statistical area level 2 (SA2): 254
- Use **spatio-temporal Bayesian models**
- QUT role/goals
  - **Explore** the wealth of data
  - **Suggest/recommend** the most suitable Bayesian models for *all* the DOH WA data
  - Complete three core **deliverables**

QUT Deliverables

Deliverable 1

- **Short report** describing model recommendations

QUT Deliverables

Deliverable 1

- **Short report** describing model recommendations

Deliverable 2

- **Longer report** with details of model recommendations

QUT Deliverables

Deliverable 1

- **Short report** describing model recommendations

Deliverable 2

- **Huge report** with details of model recommendations *and* . . .
    - Bayesian inference and computation
    - Hierachical and spatio-temporal models
    - Basic linear algebra
    - R functions
    - Plots and tables for illustrative purposes

QUT Deliverables

Deliverable 1

- **Short report** describing model recommendations

Deliverable 2

- **Huge report** with details of model recommendations *and* . . .
  - Bayesian inference and computation
  - Hierachical and spatio-temporal models
  - Basic linear algebra
  - R functions
  - Plots and tables for illustrative purposes

Deliverable 3

- In-person **three-day training**

Considerations

Considerations

Modelling will be complete by **DOH WA team**

- **Some** have formal and extensive training in statistics
- **Some** have experience with modelling
- **None** have any knowledge of Bayesian statistics
- **No/limited** skills with `R`

Considerations

Modelling will be complete by **DOH WA team**

- **Some** have formal and extensive training in statistics
- **Some** have experience with modelling
- **None** have any knowledge of Bayesian statistics
- **No/limited** skills with `R`

Scope and size

- **Huge** variation in structure and type of data

Considerations

Modelling will be complete by **DOH WA team**

- **Some** have formal and extensive training in statistics
- **Some** have experience with modelling
- **None** have any knowledge of Bayesian statistics
- **No/limited** skills with `R`

Scope and size

- **Huge** variation in structure and type of data
- **HUGE** number of conditions

Considerations

How **HUGE**?

For illustration consider:

- Conditions: 660
- Geographical levels: 3
- Sex: 3
- Types (adjusted, unadjusted): 2

Considerations

How <u>**HUGE**</u>?

For illustration consider:

- Conditions: 660
- Geographical levels: 3
- Sex: 3
- Types (adjusted, unadjusted): 2

11,880 Bayesian ST models

Considerations

How HUGE?

For illustration consider:

- Conditions: 660
- Geographical levels: 3
- Sex: 3
- Types (adjusted, unadjusted): 2

11,880 Bayesian ST models

Assume 1 hour of computation per model: 1.3 years

Considerations

What does this mean for the **model recommendations**?

Considerations

What does this mean for the **model recommendations**?

- **Reduce** complexity
- **Avoid** condition-by-condition model selection
- Make models widely applicable (i.e. **generic**)
- **Avoid** recent developments - rely on *tried and tested* approaches
- Keep the number of models to a **minimum**
- *Double dip* where possible

The data and models

Three broad categories

<u>Administrative data</u>

- Mortality
- Emergency department (ED) attendances
- Hospitalisations
- Notifiable communicable diseases
- Cancer incidence

<u>Survey data</u>

- Risk factor data

<u>Burden of disease data</u>

- Mortality data
- Prevalence data

The data and models

Three broad categories

Administrative data **Registries**

- Mortality
- emergency department (ED) attendances
- Hospitalisations
- Notifiable communicable diseases
- Cancer incidence

Survey data **Yearly Health and Wellbeing Surveillance System (HWSS) survey**

- Risk factor data

Burden of disease data **Registries and surveys**

- Mortality
- Prevalence

Administrative data

We recommend **three** Bayesian Spatio-temporal disease mapping models. . .

Administrative data

We recommend **three** Bayesian Spatio-temporal disease mapping models. . .

. . . which only need **raw counts** (and **populations**).

Administrative data

We recommend **three** Bayesian Spatio-temporal disease mapping models...

... which only need **raw counts** (and **populations**).

	Input data by						
Model	Area	Year	Age	**Input data**	**Offset term**	**Key model output** calculation ‡	**Software**
SIR_ST	✓	✓		Counts	Expected counts	Fitted counts ÷ offset	`CARBayesST`
ASR_ST	✓	✓		Counts	Counts ÷ ASRs	Fitted counts ÷ offset	`CARBayesST`
ASRA_ST	✓	✓	✓	Counts	Population	Fitted counts (then calculate ASR)	`nimble`

Survey data

We recommend **three** Bayesian Spatio-temporal small area estimation models. . .

Survey data

We recommend **three** Bayesian Spatio-temporal small area estimation models...

... need **individual-level survey data**, **area-level summaries** and **area-level census data**.

Survey data

We recommend **three** Bayesian Spatio-temporal small area estimation models...
... need **individual-level survey data**, **area-level summaries** and **area-level census data**.

Model	Sample weights	Population counts	Input data	Covariates	Model output	Software
MrP_ST		✓	Individual-level survey data	Individual- and area-level covariates	Fitted probabilities	`nimble` `mcmcsae`
WMrP_ST	✓	✓	Individual-level survey data	Individual- and area-level covariates	Fitted probabilities	`nimble`
FHELN_ST	✓	✓	Area-by-Year proportion estimates and sampling variances	Area-level census covariates	Fitted probabilities	`mcmcsae`

Burden of Disease data

We recommend **three** models

Burden of Disease data

We recommend **three** models

Model	**BoD Metric**	Area	Year	Age	**Input data**	**Offset term**	**Key model output** **calculation**	**Software**
		Input data by						
ASRA_ST	YLL, YLD	✓	✓	✓	Counts/Point prevalence	Population/adjusted population	Fitted counts (then calculate YLL or YLD)	`nimble`
ASRAME_ST	YLD	✓	✓	✓	Prevalence estimates and sampling variances	Adjusted population	Fitted counts (then calculate YLL or YLD)	`nimble`
WMrP_ST	YLD	Individual-level survey data			Binary outcome	NA	Fitted probabilities§	`nimble`

Burden of Disease data

We recommend **three** models

Model	BoD Metric	Input data by: Area	Year	Age	Input data	Offset term	Key model output calculation	Software
ASRA_ST	YLL, YLD	✓	✓	✓	Counts/Point prevalence	Population/adjusted population	Fitted counts (then calculate YLL or YLD)	`nimble`
ASRAME_ST	YLD	✓	✓	✓	Prevalence estimates and sampling variances	Adjusted population	Fitted counts (then calculate YLL or YLD)	`nimble`
WMrP_ST	YLD	Individual-level survey data			Binary outcome	NA	Fitted probabilities§	`nimble`

Software

- `R` as the statistical software
- Bayesian `R` packages
  - `CARBayesST`:
  - `nimble`:
  - `mcmcsae`:

Software

- `R` as the statistical software
- Bayesian `R` packages
  - `CARBayesST`: **Fast and easy to use** but not **flexible**
  - `nimble`: **Flexible** but **complex and slow**
  - `mcmcsae`: **Fast, easy to use** and **relatively flexible**

Software

- `R` as the statistical software
- Bayesian `R` packages
  - `CARBayesST`: **Fast and easy to use** but not **flexible**
  - `nimble`: **Flexible** but **complex and slow**
  - `mcmcsae`: **Fast, easy to use** and **relatively flexible**
- Developed a series of `R` wrapper functions *(46 and counting)*

Code wrapper for CARBayesST

```r
SIR_model <- SampleCBST(y ~ offset(log(E)),
                        # Number of MCMC samples to draw for
                        ↪  each chain
                        n.sample = 2500,
                        # burn-in
                        nburnin = 1250,
                        # amount to thin by
                        thin = 1,
                        # define the dataset
                        data = df,
                        # binary contiguity weight matrix
                        W = W,
                        # area and year variables in df
                        area = "M_id",
                        year = "T_id",
                        # offset term as a numeric vector
                        ofs = df$E,
                        # observed count as a numeric vector
                        y = df$y)
```

Example of MCMC diagnostic warnings

```
> message(ASRAST_fit$messages)
Median Rhat: 1
0.01% of Rhats larger than 1.01
Max Rhat = 1.02 (rho)
0.01% of ess_bulk are too small
Min ess_bulk = 319.88 (rho)
0% of ess_tail are too small
Min ess_tail = 443.56 (rho)
Average posterior draws per minute: 2051.13
```

Code wrapper for nimble

```
ASRA_model <- SampleNimble( # BUGS code
                            code = code,
                            # data list
                            nD = nD,
                            # initial value function
                            nI = nI(),
                            # constant list
                            nC = nC,
                            # parameters to monitor
                            monitors = monitors,
                            # total iterations per chain
                            niter = 4000,
                            # burn-in per chain
                            nburnin = 2000,
                            thin = 20,
                            nchains = 4,
                            # check samplers are correct
                            print_samplers = T,
                            # optimize sampling
                            # of the fixed effects
                            optimBeta = T,
                            beta_name = "B_qr",
                            # use an RW_block sampler
                            sampler_name = "RW_block",
                            # decrease adaption during burn-in
                            adaptInterval = 10 # defaults to 200)
```

Thank you for your attention!