

Data Impact Program



An R Package and Shiny Application for Estimating Excess Mortality

July 2024





Acknowledgements

This work was led by the Data Impact Program at Vital Strategies and supported by the Bloomberg Philanthropies Data for Health Initiative. This document was authored by Zehang (Richard) Li and Samuel J. Clark.



1 Introduction

Excess mortality is the difference between the expected and observed mortality in a given time period. Excess mortality has emerged as one of the most important measures of the COVID-19 pandemic's impact. Many excess mortality models have been developed in the literature. Despite the different model specifications, almost all excess mortality methods operate under the same framework: statistical models are used to estimate counterfactual expected deaths without COVID-19 during the pandemic using historical mortality data, and excess mortality is computed as the contrast between the observed and the predicted mortality.

Different statistical models have been developed to estimate the counterfactual expected deaths with varying data availability. For example, when only a single series of mortality data is available, Chen et al. (2021) developed a time series model to predict expected mortality and the model has been applied to model sub-population excess mortality for each population group separately. Paglino et al. (2023) develops a spatial Poisson model for subnational excess mortality estimation in the United States under a Bayesian framework, which allows information sharing across different spatial units. When covariate data are available, Mengjuan et al. (2022) develops a negative binomial regression model, estimated with penalized least squares. Finally on the global level, the World Health Organization (WHO) produces global and country-specific excess mortality using a joint overdispersed Poisson model with a range of different all-cause mortality and covariate data (Knutson et al. 2023; Msemburi et al. 2023).

In this document, we introduce an open-source R package and the associated Shiny application to estimate monthly and weekly excess mortality by sex and age using aggregated data. In particular, we focus on the data sparse setting, where we only utilize all-cause mortality in the form of aggregated counts, potentially by sex and age, and optionally, information on population size. Two estimators are implemented in the package: a simple estimator based on historical average, and a Bayesian Poisson regression approach. The new Poisson regression model is similar in nature to some of the existing work mentioned above (Paglino et al. 2023; Knutson et al. 2023; Msemburi et al. 2023). The Bayesian framework allows flexible modeling and straightforward interpretation of the



uncertainty. In the rest of the document, we first describe the two models and then compare them in an analysis of excess mortality in two country examples. Finally, we provide a description of the associated Shiny application and discuss additional recommendations.

2 Method

When sex- and age-specific mortality data are available, we estimate excess mortality for each sex, age, sex-age combination, and the overall population separately. This is a practical choice to avoid a potentially much more complicated joint model. Therefore, we describe the methods below for a single series of death counts for simplicity.

Denote observed deaths by Y_t for time period $t = 1, 2, \dots, T_0$ during the pre-pandemic period and $t = T_0 + 1, \dots, T$ for pandemic period. Excess mortality model proceeds by first estimating the expected number of deaths in the pandemic period, using pre-pandemic data; and then computing the difference between the observed and predicted number of deaths to obtain the expected excess death counts. That is, we denote \tilde{Y}_t to be the predicted death count in time period t had there being no pandemic, then the excess mortality at time period t is $E_t = Y_t - \tilde{Y}_t$.

Some additional notations are necessary to describe the models. We let $J[t], t = 1, 2, \dots, T$ be the month or week index for time period t . For example, if the analysis is performed at the monthly level, and the 14th time period in the data is March, then $J[14] = 3$. We let $P_t, t = T_0 + 1, T_0 + 2, \dots, T$ to denote the population size at time t .

2.1 Previous simple baseline method

The previous method does not make use of the population data P_t . It estimates the expected death count as the average death counts in the same month/week during pre-pandemic years. That is,

$$\tilde{Y}_t = \frac{\sum_{i: J[i]=J[t], i \leq T_0} Y_i}{\sum_{i: J[i]=J[t], i \leq T_0} 1}$$



Note that under this model, the expected death counts for the same week/month over different years remains the same, thus it does not account for any across-year variation or time trend.

The standard error of the expected death count \tilde{Y}_t is estimated by the sample standard deviation of the death counts in the same month/week during pre-pandemic years, divided by the square root of the number of observations used to compute the sample average.

Finally, the 95% lower and upper confidence interval of the expected deaths are computed by the Wald type interval

$$\left(\tilde{Y}_t - 1.96 \times SE(\tilde{Y}_t), \tilde{Y}_t + 1.96 \times SE(\tilde{Y}_t) \right)$$

The excess death counts are computed by

$$E_t = Y_t - \tilde{Y}_t$$

and the 95% confidence interval is given by

$$\left(Y_t - \tilde{Y}_t - 1.96 \times SE(\tilde{Y}_t), Y_t - \tilde{Y}_t + 1.96 \times SE(\tilde{Y}_t) \right)$$

2.2 Issues with the simple baseline method

There are two main issues with the simple baseline method. First, the model does not take population change into account. The simple average of historical death counts can be an underestimate of the expected deaths if the population size increased, and overestimate if population size declined. Second, the confidence interval is constructed based on the asymptotic sampling distribution of the sample mean. For historical data with only 5 to 10 years, the standard error estimation can be highly unstable and thus the uncertainty interval needs to be viewed skeptically. Both issues can be addressed by the Poisson regression method described in the next section.



2.3 Poisson regression method

The Poisson regression model assumes the sampling distribution of the observed data to be

$$\tilde{Y}_t \mid \mu_t \sim \text{Pois}(\mu_t P_t)$$

where μ_t is a latent parameter for the time-varying mortality rate. When population size is unknown, P_t is treated as a constant and omitted from the model. The mortality rate μ_t is further assumed to decompose into three components

$$\log(\mu_t) = \alpha + \beta t + \eta_t + \gamma_{J[t]}$$

where α is the global intercept; βt captures the long-term linear trend in time; η_t captures the long-term non-linear trend; and $\gamma_{J[t]}$ is the within-year seasonal effect. These latent parameters are given the following priors:

- $\alpha \sim N(0,1000)$ and $\beta \sim N(0,1000)$ are fixed effects following a normal prior with a large variance.
- The long-term non-linear effect η_t follows an autoregressive process of order 1 (AR(1)) over a low-resolution knots. More specifically, we put K evenly spaced knots over the entire time period, and we can write $\eta_t = \sum_{k=1}^K w_{tk} x_k + e_t$, where x_1, \dots, x_K is an AR(1) process on the K knots, i.e.,

$$x_k = \rho x_{k-1} + \epsilon_k$$

where $\epsilon_k \sim N(0, \sigma_\eta^2)$. The weights w_{tk} for each time period t is assigned such that when the time point is between two knots, the weight is inversely proportional to the distance from each knot. For formulation allows us to represent the observed non-linear change over time to be a slowly-varying function over a reduced set of time points. We put 4 evenly spaced knots per year by default. PC priors are used for the AR(1) process hyperparameters so that $P(\rho > 0.7) = 0.9$ and $P(\sigma_\eta > 1) = 0.01$.

- The seasonal effect $\gamma_{J[t]}$ is a random effect with first order random walk prior, i.e.

$$\gamma_j = \gamma_{j-1} + \epsilon'_j$$

where $\epsilon'_j \sim N(0, \sigma_\gamma^2)$. PC prior is used for σ_γ^2 such that $P(\sigma_\gamma > 1) = 0.01$.

The model is estimated by the INLA software (Rue, Martino, and Chopin 2009; Rue et al. 2017). Posterior median and 95% posterior credible intervals of \tilde{Y}_t and $Y_t - \tilde{Y}_t$ are obtained from the fitted model directly, and used as the final estimates and uncertainty intervals.



3 Case Study I

We use the mortality data from Country Example I to illustrate the use of the package to compute and visualize excess mortality. The analysis can be done either in R with the **ExcessMortalityApp** package, or by using the Shiny App described in the next section. All codes used for the analysis is included in the Appendix.

3.1 Input data

The input data for the Shiny App or the functions in **ExcessMortalityApp** package is in the “long format”, where each row correspond to the death count of a given (sub-)population at a given time period. The dataset should include at least the following required columns, with the exact column names:

- *year*: numerical value, e.g., 2015, 2016, ...
- *month* or *week*: numerical value. For monthly data input, the *month* field is required to be from 1 to 12. For weekly data input, the *week* field is required to be from 1 to 53.
- *deaths*: numerical value. This is the death count for the corresponding year and month/week.

In general, we recommend having at least 3 to 5 years of historical data to accurately estimate the baseline expected mortality. Longer time series helps improve model estimation and reduce uncertainty.

There are additional columns that can be included in the data. The column names of these fields can be user-defined.

- **sex**: if death count of a given row is for a specific sex.
- **age**: if death count of a given row is for a specific age group. Please note the age needs to be grouped into bins.
- **population**: the total population corresponding to the row. For example, for a row corresponding to male and age group of 65 or above in a certain month. The *death*



field specifies the number of male deaths above age 65 in that month, and the *population* field specifies the total population of male above age 65 in that month.

In this case study, we first aggregated death counts by month, sex, and three age groups (0 to 65, 65 to 85, and 85+). Different age group splits are also allowed. Population data by sex and age are obtained from WPP 2022 estimates (medium variant projection for year 2022) and merged with the mortality data.

Table 3.1 shows the first 6 rows of the aggregated dataset. For dataset with both sex- and age- stratification, the sex-specific, age-specific, and overall mortality results are also computed, so there is no need to fit the model multiple times with different breakdowns. Additional data input formats without the sex and age breakdown can be found in the example dataset from the Shiny App.

3.2 Visualize results for a single (sub)-population

We fit both models on this dataset for each sex, age, sex-age combination, and the overall population. Figure 3.1 compares the expected death counts estimated by the two models. Expected death counts after 2020 are predictions based on historical data, so they have larger uncertainties. For both models, the uncertainty bands are for the expected death counts (i.e., the mean function) so we do not expect 95% of the observed points to fall within the intervals. But we can see that for the simple baseline model, the expected deaths do not track the observed deaths in terms of the overall trend, as it does not take population growth into account. This is illustrated more clearly when evaluating sub-population results, e.g., for females over 85 years old in Figure 3.2.

The excess mortality estimates and the 95% uncertainty intervals are shown in Figure 3.3.



Table 3.1: Example input data format.

year	month	sex	age	deaths	population
2015	1	Female	0 to 65	2,596	21,977,568
2015	2	Female	0 to 65	2,332	21,977,568
2015	3	Female	0 to 65	2,509	21,977,568
2015	4	Female	0 to 65	2,397	21,977,568
2015	5	Female	0 to 65	2,560	21,977,568
2015	6	Female	0 to 65	2,499	21,977,568

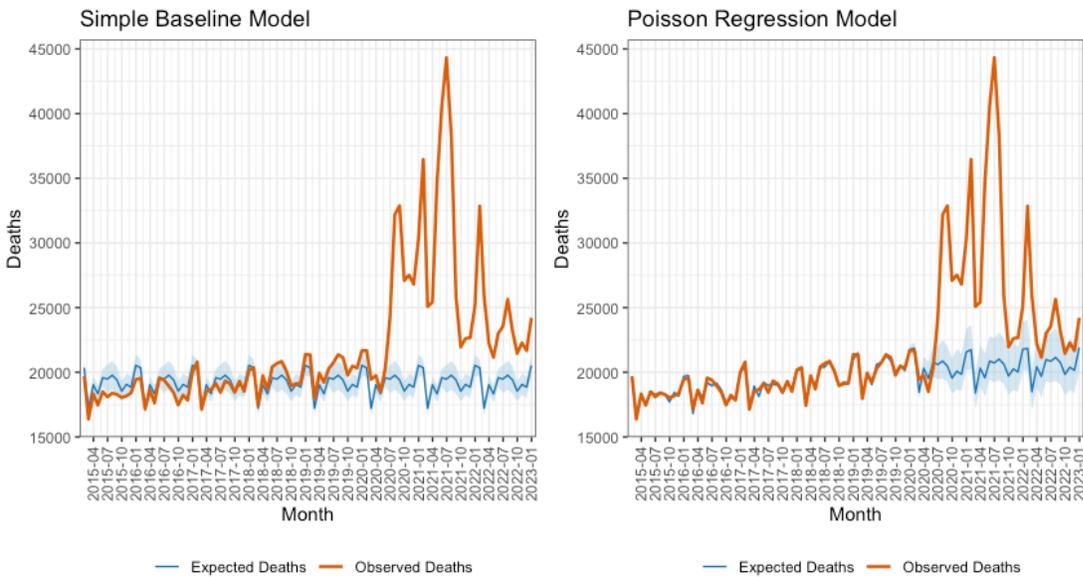


Figure 3.1: Country Example I: The expected and observed death counts for the overall population. 95% uncertainty intervals are shown for the expected death counts.

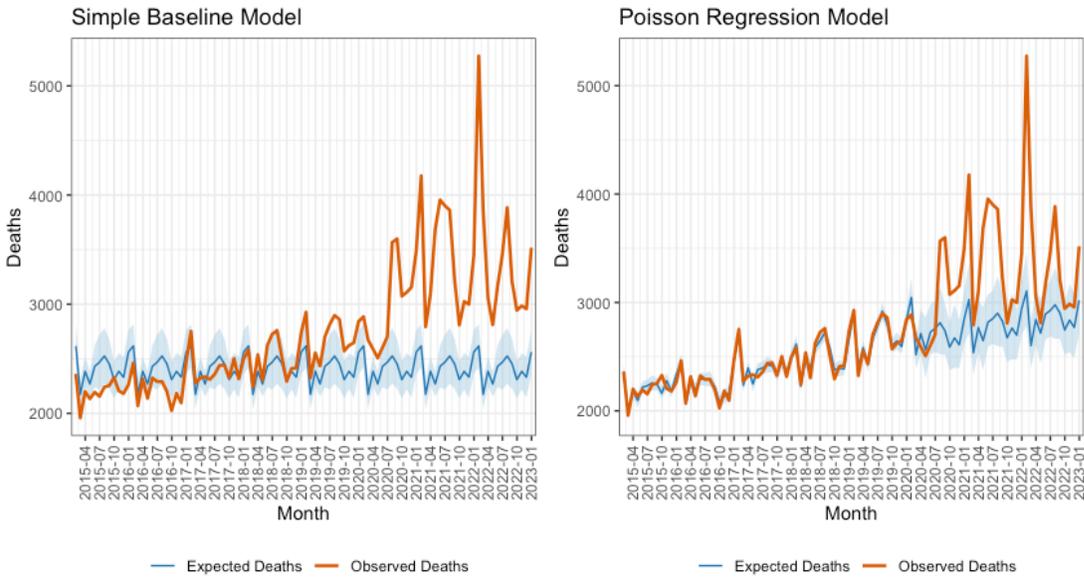


Figure 3.2: Country Example I: The expected and observed death counts for females over 85 years old. 95% uncertainty intervals are shown for the expected death counts.

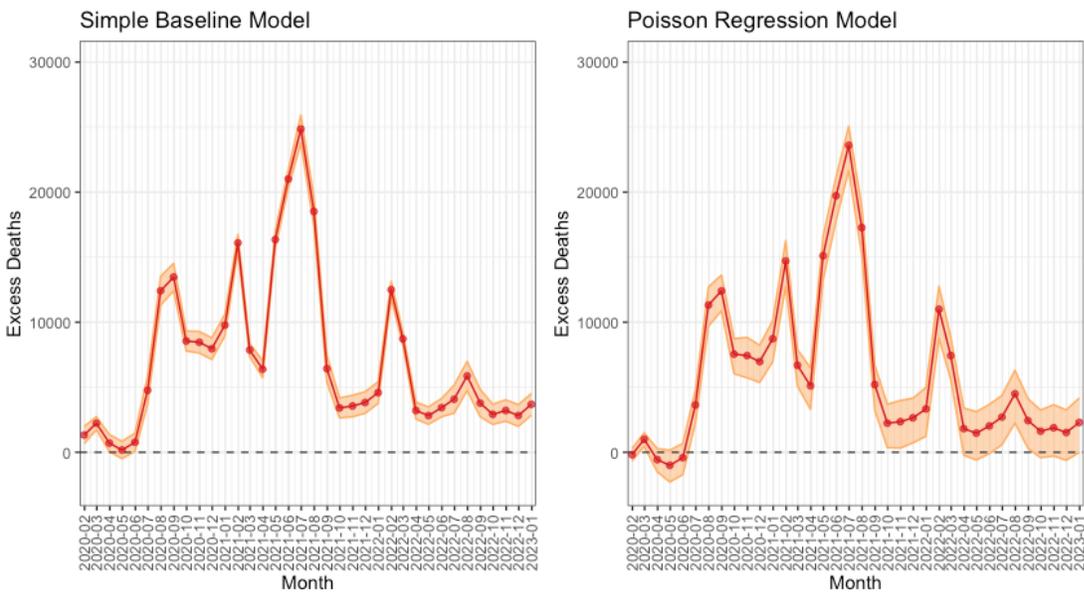


Figure 3.3: Country Example I: The excess death counts for the overall population and the associated 95% uncertainty interval.



3.3 Visualize results for multiple sub-populations

We can also compare results by sex, age, or both sex and age. Figure 3.4 shows the comparison of expected and observed death counts for all age-sex combinations. Figure 3.5 shows the comparison of sub-population excess mortality estimates and their uncertainties. Figure 3.6 shows the excess mortality estimates overlaid on the same plot for all sub-populations.

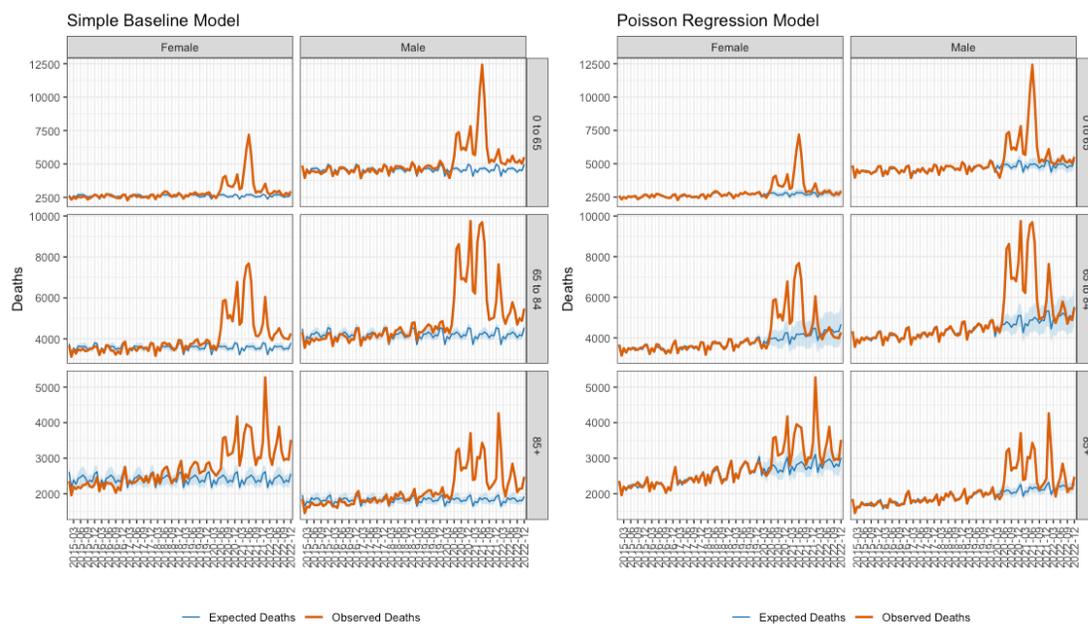


Figure 3.4: Country Example I: The expected and observed death counts for all sub-populations. 95% uncertainty intervals are shown for the expected death counts.

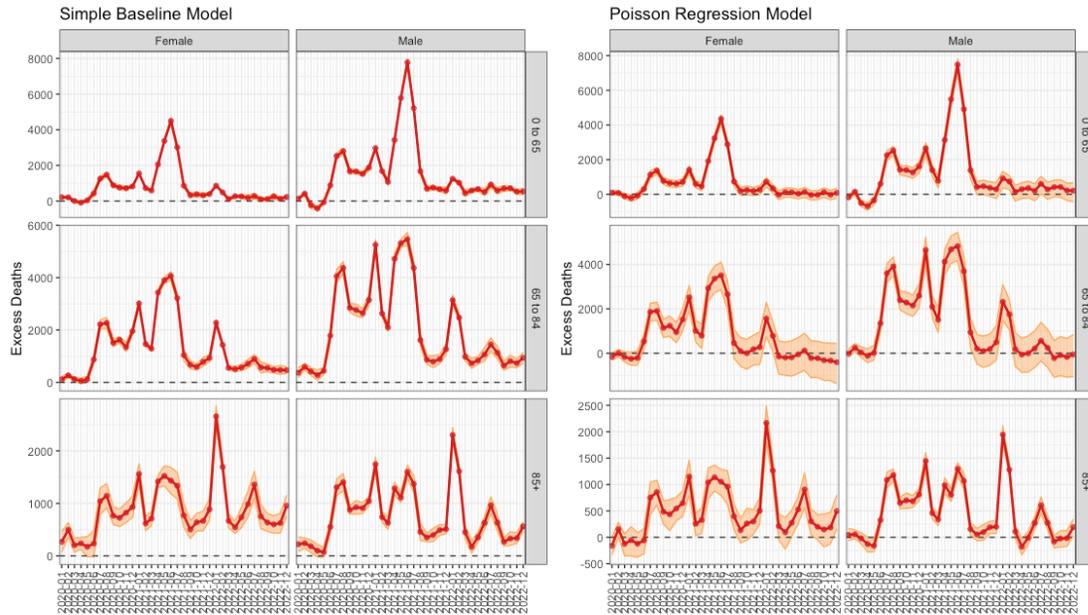


Figure 3.5: Country Example I: The excess death counts for all sub-populations and the associated 95% uncertainty interval.

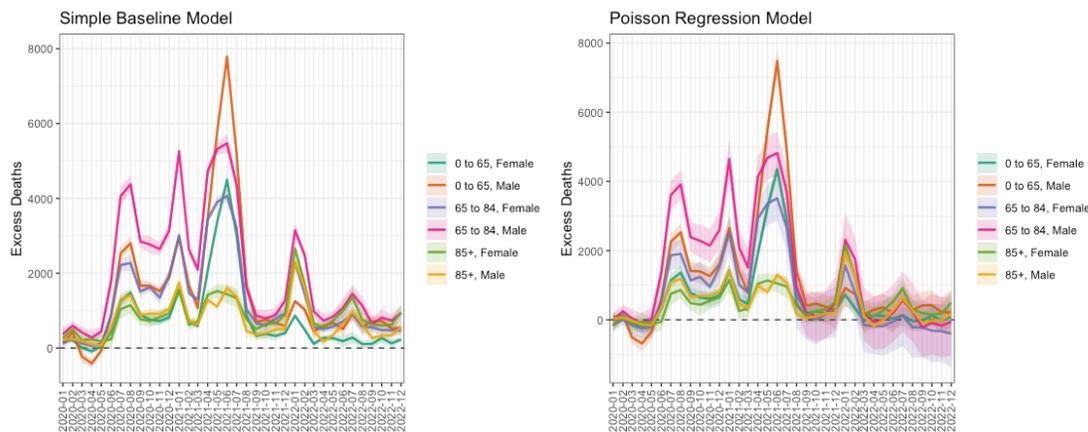


Figure 3.6: Country Example I: The excess death counts for all sub-populations and the associated 95% uncertainty interval, overlaid on the same plot.



3.4 Additional comparison to the Poisson regression model without population information

Lastly, we compare the Poisson regression model with the same model without using population size information. Such scenarios where users do not have population information may arise in practice. Figure 3.7 shows the comparison of the estimated expected death counts. When population information is not known, the linear trend term in the Poisson regression model can partially account for the overall linear increase or decrease of death counts, but it cannot account for any non-linear change in population size and will increase the uncertainty of the estimates. Therefore, we highly recommend fitting the model with population size information whenever possible.

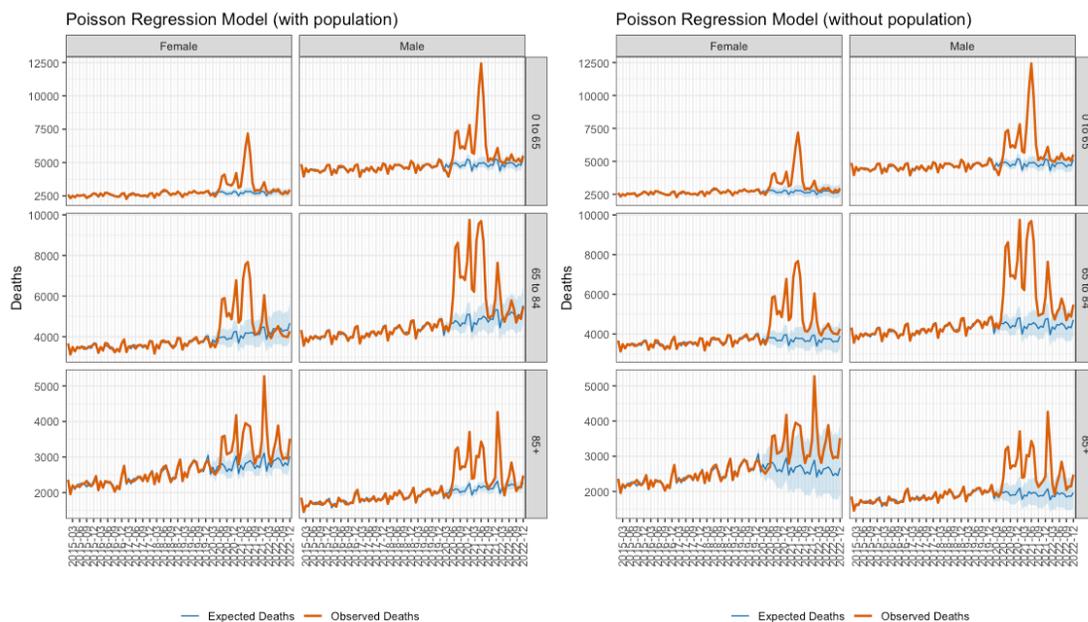


Figure 3.7: Country Example I: The expected and observed death counts for the overall population. 95% uncertainty intervals are shown for the expected death counts.



4 Case Study II

We now turn to a second example using the mortality data from Country Example II. The data are organized into deaths by sex and deaths by age groups separately, with incomplete data in the last quarter of 2022. The population size information was obtained from WPP 2022 estimates (medium variant projection for year 2022) and merged with the mortality data.

This dataset consist of smaller death counts and exhibit weaker seasonality pattern in mortality rates than the previous country example. As a result, the estimated uncertainty intervals of the expected mortality are larger. Similar plots are generated as in the previous example. Figures 4.1 and 4.2 compare the expected death counts by sex and age, estimated by the two models. Figures 4.2 and 4.4 show the comparison of sub-population excess mortality estimates and their uncertainties.

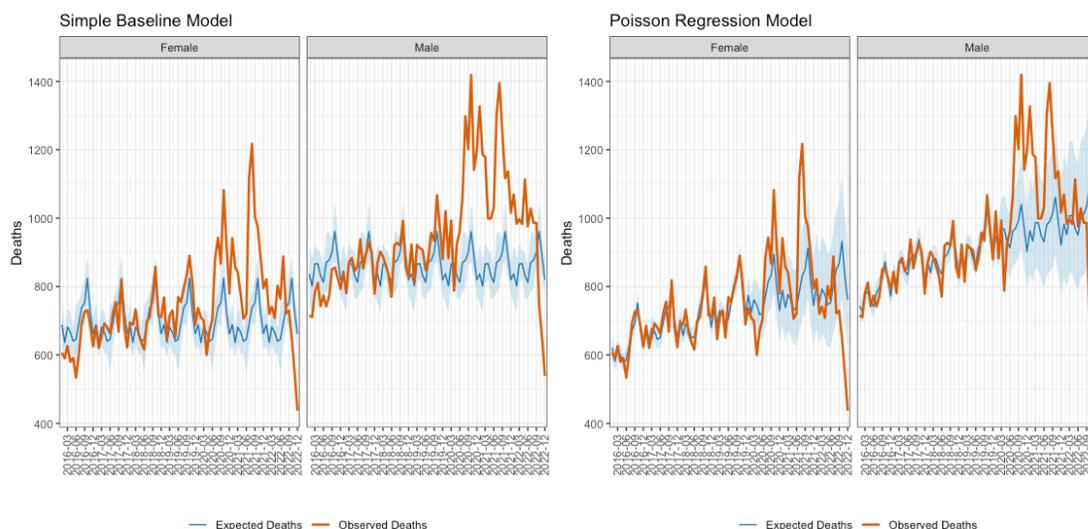


Figure 4.1: Country Example II: The expected and observed death counts for male and female. 95% uncertainty intervals are shown for the expected death counts.

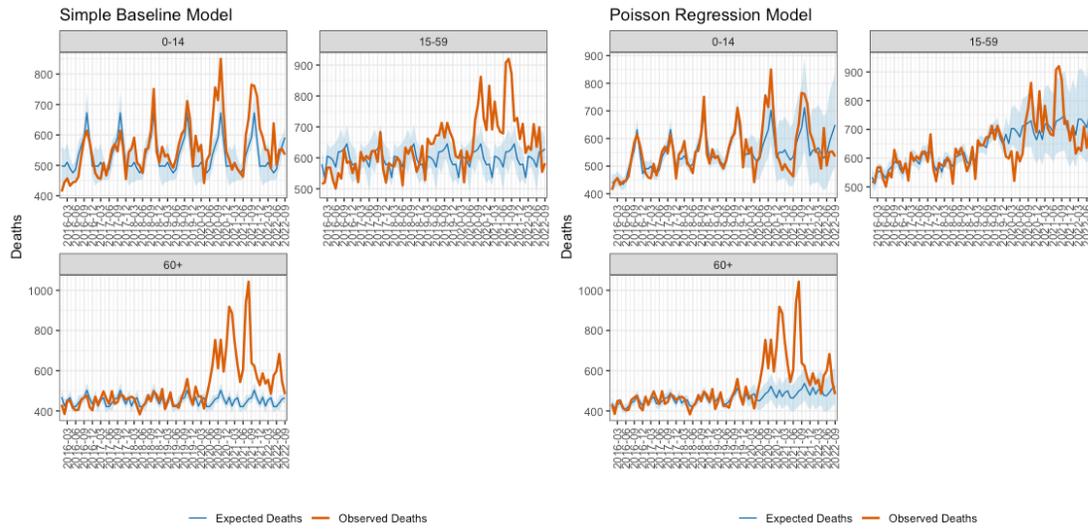


Figure 4.2: Country Example II: The expected and observed death counts for the three age groups. 95% uncertainty intervals are shown for the expected death counts.

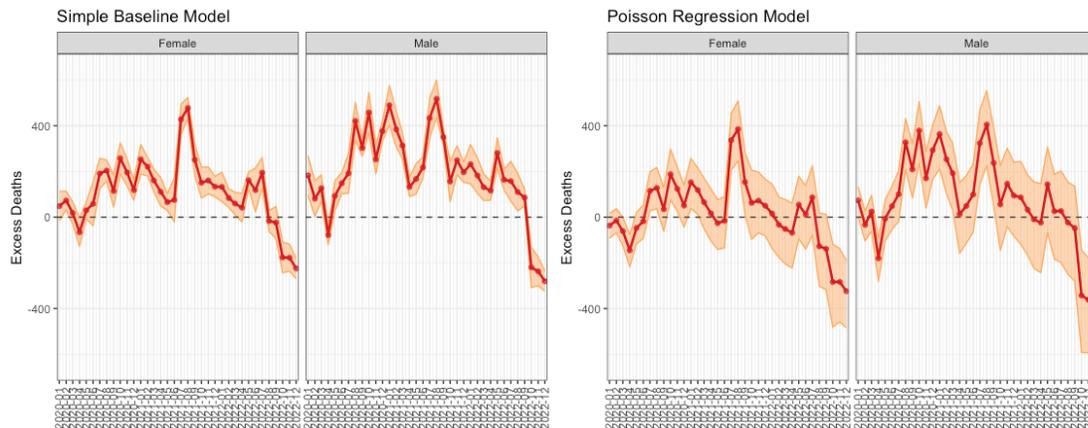


Figure 4.3: Country Example II: The excess death counts for male and female, and the associated 95% uncertainty interval.

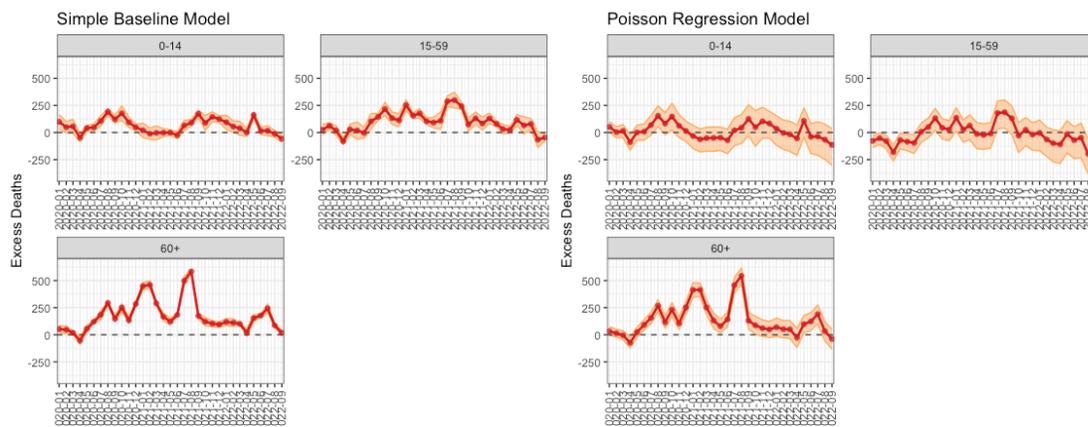


Figure 4.4: Country Example II: The excess death counts for different age groups, and the associated 95% uncertainty interval.

5 The Shiny App

All of the results in the previous session can be obtained in R by running command line codes. However, for users who do not have R installed, or for users who prefer a more interactive interface, we have developed a Shiny App that provides the same functionality as the R package. The Shiny App can be accessed at <https://zehanqli.shinyapps.io/ExcessMortalityApp/>. The App allows users to upload their own data, choose relevant time resolution, fit the model, and produce visualizations for the overall population, specific sub-populations, or comparisons as shown in the previous Section. The App also provides a historical data explorer tab, which allows users to visualize the raw death counts. All the plots on the App are interactive with additional zooming and pop-up upon hovering features for easier exploration. Results and figures can be downloaded in multiple formats. A screenshot of the web application is included in Figure 5.1.



Excess Mortality Calculator

Developed by the openVA team (<https://openVA.net>). This work is supported by Vital Strategies 

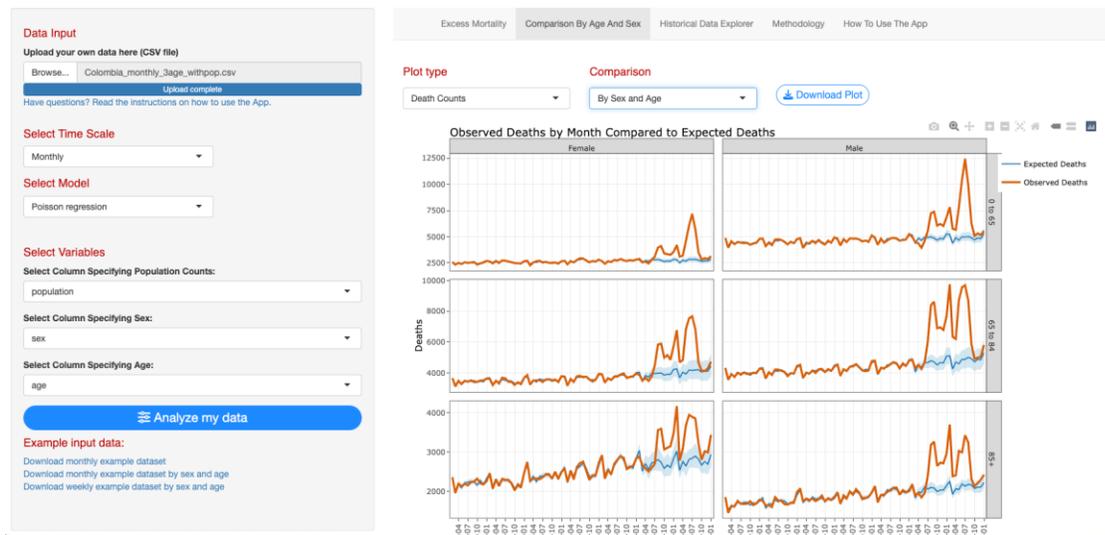


Figure 5.1: A screenshot of the ExcessMortalityApp Shiny application.

Data Security. Data that are uploaded to the application are stored temporarily and cleared after the application is closed. Users may access the source R code in the GitHub repository to verify that there is no connection with permanent storage.

6 Discussion

In this document, we describe the method and usage of the **ExcessMortalityApp** package and the Shiny App. The models are developed for data sparse settings, so we assume little extra information other than aggregated counts. We identify two main issues with the simple baseline method, namely ignoring population change and unstable uncertainty interval, and developed a new Bayesian Poisson regression alternative. Results from the two models are compared based on mortality data from two country examples.

There are several directions of future research, software development, and model exploration. First, the current model treats each sub-population separately. This modeling choice is appealing for a general tool as it does not require user to have certain types of data structures such as requiring the minimal number of sub-populations or sample sizes, etc. However, for many specific use cases, the models can be improved by borrowing information across different sub-populations. The Bayesian Poisson regression model can



be readily extended to such cases but require more model tuning and validation. Second, the model uses only mortality data before 2020 to compute expected mortality. The accuracy of long-term projection into the future may deteriorate. It is an open area of research how to combine information after 2020 to produce a more relevant baseline of expected mortality. Lastly, the proposed method is similar in nature to many of the existing excess mortality models, thus we expect it to produce similar baseline mortality projections compared to many alternative model formulations. However, little work has been done in the literature to compare the different model assumptions and specifications under different data availability scenarios so there is a need for a better guideline to choose between different model classes.

References

Chen, Yea-Hung, M Maria Glymour, Ralph Catalano, Alicia Fernandez, Tung Nguyen, Margot Kushel, and Kirsten Bibbins-Domingo. 2021. “Excess Mortality in California During the Coronavirus Disease 2019 Pandemic, March to August 2020.” *JAMA Internal Medicine* 181 (5): 705–7.

Knutson, Victoria, Serge Aleshin-Guendel, Ariel Karlinsky, William Msemburi, and Jon Wakefield. 2023. “Estimating Global and Country-Specific Excess Mortality During the COVID-19 Pandemic.” *The Annals of Applied Statistics* 17 (2): 1353–74.

Mengjuan, Duan, Mark S Handcock, Bart Blackburn, Fiona Kee, Viema Biaukula, Tamano Matsui, and Babatunde Olowokure. 2022. “Tool for Tracking All-Cause Mortality and Estimating Excess Mortality to Support the COVID-19 Pandemic Response.” *Western Pacific Surveillance and Response Journal: WPSAR* 13 (2): 1.

Msemburi, William, Ariel Karlinsky, Victoria Knutson, Serge Aleshin-Guendel, Somnath Chatterji, and Jon Wakefield. 2023. “The WHO Estimates of Excess Mortality Associated with the COVID-19 Pandemic.” *Nature* 613 (7942): 130–37.



Paglino, Eugenio, Dielle J Lundberg, Zhenwei Zhou, Joe A Wasserman, Rafeya Raquib, Anneliese N Luck, Katherine Hempstead, et al. 2023. "Monthly Excess Mortality Across Counties in the United States During the COVID-19 Pandemic, March 2020 to February 2022." *Science Advances* 9 (25): eadf9742.

Rue, Håvard, Sara Martino, and Nicolas Chopin. 2009. "Approximate Bayesian Inference for Latent Gaussian Models by Using Integrated Nested Laplace Approximations." *Journal of the Royal Statistical Society Series B: Statistical Methodology* 71 (2): 319–92.

Rue, Håvard, Andrea Riebler, Sigrunn H Sørbye, Janine B Illian, Daniel P Simpson, and Finn K Lindgren. 2017. "Bayesian Computing with INLA: A Review." *Annual Review of Statistics and Its Application* 4: 395–421.

Appendix:

R codes to reproduce the case study

```
library(ExcessMortalityApp)
library(INLA)
library(ggplot2)
library(patchwork)
library(knitr)
data <- read.csv("../data/data1_monthly_3age_withpop.csv")
knitr::kable(head(data), caption = "Example input data format.")
out0 <- base_model(time_case = "Monthly", T = 12, years = c(2015:2022),
                  morData = data, sexCol = "sex", ageCol = "age"
,
                  popCol = "population", timeCol = "month")
out <- smooth_model(time_case = "Monthly", T = 12, years = c(2015:2022),
                   morData = data, sexCol = "sex", ageCol = "age"
,
                   popCol = "population", timeCol = "month")
g1 <- mortality_plot(model = out0, sex = "All", age = "All",
                    month_or_week = "Monthly", plot_show = "Death Coun
```



```
ts")
g2 <- mortality_plot(model = out, sex = "All", age = "All", timeCol = "month",
                    month_or_week = "Monthly", plot_show = "Death Counts")
ts")
(g1 + ggtitle("Simple Baseline Model") ) +
(g2 + ggtitle("Poisson Regression Model"))
g1 <- mortality_plot(model = out0, sex = "Female", age = "85+",
                    month_or_week = "Monthly", plot_show = "Death Counts")
ts")
g2 <- mortality_plot(model = out, sex = "Female", age = "85+", timeCol = "month",
                    month_or_week = "Monthly", plot_show = "Death Counts")
ts")
(g1 + ggtitle("Simple Baseline Model") ) + g2 + ggtitle("Poisson Regression Model")
g1 <- mortality_plot(model = out0, sex = "All", age = "All",
                    month_or_week = "Monthly", plot_show = "Excess Death Counts")
g2 <- mortality_plot(model = out, sex = "All", age = "All", timeCol = "month",
                    month_or_week = "Monthly", plot_show = "Excess Death Counts")
(g1 + ggtitle("Simple Baseline Model") + ylim(c(-2500, 30000))) +
(g2 + ggtitle("Poisson Regression Model") + ylim(c(-2500, 30000)))
g1 <- compare_plot(model = out0, by = "By Sex and Age",
                  month_or_week = "Monthly",
                  plot_show = "Death Counts")
g2 <- compare_plot(model = out, by = "By Sex and Age",
                  month_or_week = "Monthly", timeCol = "month",
                  plot_show = "Death Counts")
(g1 + ggtitle("Simple Baseline Model") ) +
(g2 + ggtitle("Poisson Regression Model"))
g1 <- compare_plot(model = out0, by = "By Sex and Age",
                  month_or_week = "Monthly",
                  plot_show = "Excess Death Counts")
g2 <- compare_plot(model = out, by = "By Sex and Age",
                  month_or_week = "Monthly", timeCol = "month",
                  plot_show = "Excess Death Counts")
(g1 + ggtitle("Simple Baseline Model") ) +
(g2 + ggtitle("Poisson Regression Model"))
g1 <- compare_plot(model = out0, by = "By Sex and Age",
                  month_or_week = "Monthly",
                  plot_show = "Excess Death Counts (Overlay)")
```



```
g2 <- compare_plot(model = out, by = "By Sex and Age",
                  month_or_week = "Monthly", timeCol = "month",
                  plot_show = "Excess Death Counts (Overlay)")
(g1 + ggtitle("Simple Baseline Model") ) +
(g2 + ggtitle("Poisson Regression Model"))
data$population <- NA
out1 <- smooth_model(time_case = "Monthly", T = 12, years = c(2015
:2022),
                    morData = data, sexCol = "sex", ageCol = "age
",
                    popCol = "population", timeCol = "month")
g1 <- compare_plot(model = out, by = "By Sex and Age",
                  month_or_week = "Monthly", timeCol = "month",
                  plot_show = "Death Counts")
g2 <- compare_plot(model = out1, by = "By Sex and Age",
                  month_or_week = "Monthly", timeCol = "month",
                  plot_show = "Death Counts")
(g1 + ggtitle("Poisson Regression Model (with population)") ) +
(g2 + ggtitle("Poisson Regression Model (without population)"))
data_by_sex <- read.csv("../data/data2_by_sex.csv")
data_by_sex$age <- "All"
data_by_age <- read.csv("../data/data2_by_age.csv")
data_by_age$sex <- "All"
out0_by_sex <- base_model(time_case = "Monthly", T = 12, years = c
(2016:2022),
                        morData = data_by_sex, sexCol = "sex", ageCol
= "age",
                        popCol = "population", timeCol = "month")
out_by_sex <- smooth_model(time_case = "Monthly", T = 12, years =
c(2016:2022),
                          morData = data_by_sex, sexCol = "sex", ageCol
= "age",
                          popCol = "population", timeCol = "month")
out0_by_age <- base_model(time_case = "Monthly", T = 12, years = c
(2016:2022),
                        morData = data_by_age, sexCol = "sex", ageCol
= "age",
                        popCol = "population", timeCol = "month")
out_by_age <- smooth_model(time_case = "Monthly", T = 12, years =
c(2016:2022),
                          morData = data_by_age, sexCol = "sex", ageCol
= "age",
                          popCol = "population", timeCol = "month")
g1 <- mortality_plot(model = out0_by_sex, sex = "All", age = "All"
```



```
,
      month_or_week = "Monthly", plot_show = "Death Counts")
g2 <- mortality_plot(model = out_by_sex, sex = "All", age = "All",
  timeCol = "month",
  month_or_week = "Monthly", plot_show = "Death Counts")
(g1 + ggtitle("Simple Baseline Model") ) +
(g2 + ggtitle("Poisson Regression Model"))
g1 <- compare_plot(model = out0_by_sex, by = "By Sex",
  month_or_week = "Monthly",
  plot_show = "Death Counts")
g2 <- compare_plot(model = out_by_sex, by = "By Sex",
  month_or_week = "Monthly", timeCol = "month",
  plot_show = "Death Counts")
(g1 + ggtitle("Simple Baseline Model") ) +
(g2 + ggtitle("Poisson Regression Model"))
g1 <- compare_plot(model = out0_by_age, by = "By Age",
  month_or_week = "Monthly",
  plot_show = "Death Counts")
g2 <- compare_plot(model = out_by_age, by = "By Age",
  month_or_week = "Monthly", timeCol = "month",
  plot_show = "Death Counts")
(g1 + ggtitle("Simple Baseline Model") ) +
(g2 + ggtitle("Poisson Regression Model"))
g1 <- compare_plot(model = out0_by_sex, by = "By Sex",
  month_or_week = "Monthly",
  plot_show = "Excess Death Counts")
g2 <- compare_plot(model = out_by_sex, by = "By Sex",
  month_or_week = "Monthly", timeCol = "month",
  plot_show = "Excess Death Counts")
(g1 + ggtitle("Simple Baseline Model") + ylim(c(-650, 650))) +
(g2 + ggtitle("Poisson Regression Model") + ylim(c(-650, 650)))
g1 <- compare_plot(model = out0_by_age, by = "By Age",
  month_or_week = "Monthly",
  plot_show = "Excess Death Counts")
g2 <- compare_plot(model = out_by_age, by = "By Age",
  month_or_week = "Monthly", timeCol = "month",
  plot_show = "Excess Death Counts")
(g1 + ggtitle("Simple Baseline Model") + ylim(c(-400, 650))) +
(g2 + ggtitle("Poisson Regression Model") + ylim(c(-400, 650)))
```