Bayesian Probabilistic Population Projections: Do It Yourself

Hana Ševčíková, University of Washington
Adrian E. Raftery, University of Washington and University College Dublin
Patrick Gerland, United Nations

September 26, 2013

Abstract

A Bayesian approach for probabilistic population projections has recently been used by the United Nation Population Division in the preparation of the 2012 revision of the World Population Prospects. The methods have been implemented in publicly available open-source software as a collection of R packages. In this paper, we demonstrate how to easily reproduce such population projections, including probabilistic projections of total fertility rate and life expectancy. The packages allow any analysts to generate variations of the UN projections, to use their own data, to impute missing data or to apply the methods to sub-national datasets. Using a flexible expression language, probabilistic results can be summarized and visualized in graphs, maps, or population pyramids. The software can be conveniently controlled from a graphical user interface.

1 Introduction

A systematic framework for producing probabilistic population projections for all countries, both developed and developing, has recently been proposed by Raftery et al. (2012). It consists of probabilistically projecting total fertility rate and life expectancy using Bayesian hierarchical models (Alkema et al., 2011; Raftery, Chunn, Gerland & Ševčíková, 2013), converting the results to age-specific rates, and projecting the population forward using the cohort-component method applied to each trajectory simulated from their predictive distributions. The median projection from the method has been used as the UN's official medium projection for all countries in the 2012 revision of the *World Population Prospects*, or WPP (United Nations, 2013).

Here we describe a suite of software packages developed to allow users beyond the UN to implement the methodology. These are four R packages: **bayesTFR** to project total fertility rate, **bayesLife** to project life expectancy, **bayesPop** to project population, and a graphical user interface, **bayesDem**. All are freely available. We also introduce a package for interactive visual exploration of various indicators derived from WPP data, called **wppExplorer**.

In this paper, we show how to reproduce such probabilistic population projections, including probabilistic projections of total fertility rate and life expectancy. The packages allow any analyst to generate variations of the UN projections, to use their own data, to impute missing data or to apply the methods to sub-national datasets. We also introduce a flexible expression language which allows probabilistic results to be summarized and visualized in graphs, maps or population pyramids. The software can be conveniently controlled from a graphical user interface.

The paper is organized as follows. In Section 2 we review the basic methodology underlying the probabilistic projections. In Section 3, we describe the **bayesTFR** package for projecting the total fertility rate. In Section 4, we outline the **bayesLife** package for projecting life expectancy. In Section 5, we describe the **bayesPop** package for population projection. This covers methods for obtaining future population trajectories, for viewing the trajectories in various ways and for producing probabilistic population pyramids. It also describes the expression language for projecting a wide range of derived population quantities. Finally, Section 6 discusses the graphical user interface **bayesDem** and **wppExplorer**.

2 Methodology

Almost all methods used for predicting population $P_{c,t}$ in country c at time period t are based on the demographic balancing equation

$$P_{c,t} = P_{c,t-1} + B_{c,t} - D_{c,t} + M_{c,t}$$

where B denotes the number of births, D the number of deaths and M net migration. In most applications this equation is solved deterministically using the cohort component method (Whelpton, 1928, 1936) which decomposes it into age- and sex-specific components.

The traditional UN methodology of population projections for future time periods $t > 0$ follows the cohort component method and for that purpose, it uses the following inputs for each country:

- Sex- and age-specific population estimates at the initial time $t = 0$
- Projections of total fertility rate (TFR)
- Projections of fertility distribution over ages
- Projections of sex ratio at birth
- Projections of male and female life expectancy at birth (e_0)
- Historical data on sex- and age-specific death rates (for $t \leq 0$)
- Projections of sex- and age-specific net migration

In each future time period t, the projected TFR is converted to age-specific fertility rates using the fertility distribution over ages at t. Using the historical data on death rates, the projected life expectancy is converted to age-specific mortality rates using a variant of the Lee-Carter method (Lee & Carter, 1992). Then the cohort component model is applied.

To communicate uncertainty in the context of this deterministic approach, until recently, the UN used three scenarios, the high, medium and low variants. Here, the medium variant is the main projection. The high and low variants have been generated by adding plus or minus half a child to the TFR, respectively, and applying the method above. Such approach suffers from not having a probabilistic basis and can lead to inconsistencies (Lee & Tuljapurkar, 1994; National Research Council, 2000).

Recently approaches to probabilistic projections of two main input components were introduced, namely TFR (Alkema et al., 2011) and life expectancy (Raftery, Chunn, Gerland & Ševčíková, 2013). Raftery et al. (2012) describes a methodology for combining these components into probabilistic population projections. The idea is to

1. simulate a large set of trajectories of future values of TFR,
2. simulate an equal number of trajectories of life expectancy,

3. convert each of the trajectories into a future trajectory of sex- and age-specific population quantity, using the current UN methodology as described above.

The resulting set of values is viewed as a sample from the predictive distribution of population projections.

This process is fully supported by open-source packages implemented in R (Ihaka & Gentleman, 1996). The simulation of TFR trajectories (1.) is implemented in a package called **bayesTFR** (Ševčíková et al., 2011). Life expectancy trajectories (2.) can be generated using the package **bayesLife** (Ševčíková & Raftery, 2013b). Finally, population projections by age and sex (3.) can be obtained using the package **bayesPop** (Ševčíková & Raftery, 2013a). A graphical user interface (GUI) for the three packages is provided by the R package **bayesDem** (Ševčíková, 2013a). One can generate probabilistic projections of TFR and life expectancy, and combine those results into probabilistic population projections from a single interface, see Figure 1.

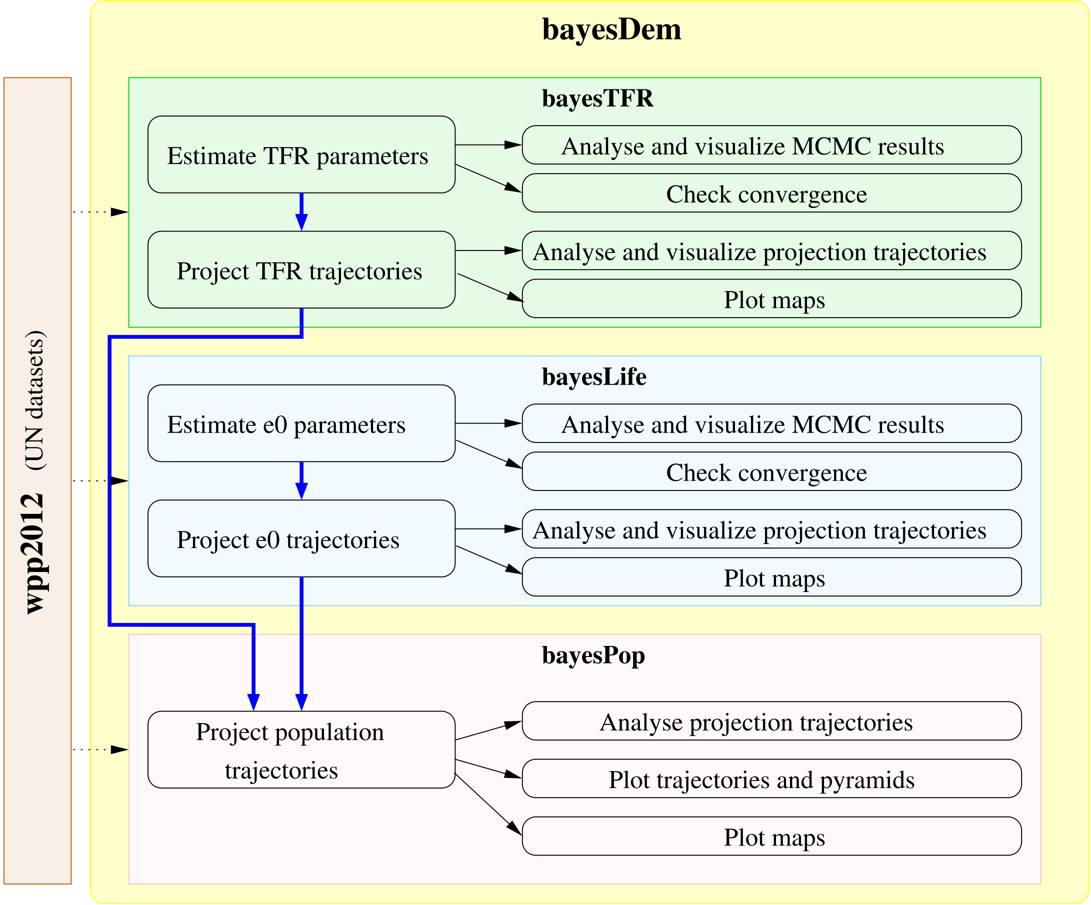

bayesDem
wpp2012 (UN datasets)
bayesTFR
Estimate TFR parameters
Analyse and visualize MCMC results
Check convergence
Project TFR trajectories
Analyse and visualize projection trajectories
Plot maps
bayesLife
Estimate e0 parameters
Analyse and visualize MCMC results
Check convergence
Project e0 trajectories
Analyse and visualize projection trajectories
Plot maps
bayesPop
Project population trajectories
Analyse projection trajectories
Plot trajectories and pyramids
Plot maps

Figure 1: Structure of packages supported by **bayesDem**. Boxes shown on the left-hand side (connected by blue arrows) depict the main steps needed for generating probabilistic population projections. Boxes on the right-hand side show supporting functionality of the packages. The packages operate on UN datasets included in the **wpp2012** package.

In what follows, we assume that the reader is somewhat familiar with basic R syntax. Furthermore, in order to reproduce the code examples in the text, R and the three packages, **bayesTFR**, **bayesLife** and **bayesPop** should be installed. One can accomplish the package installation simply by calling

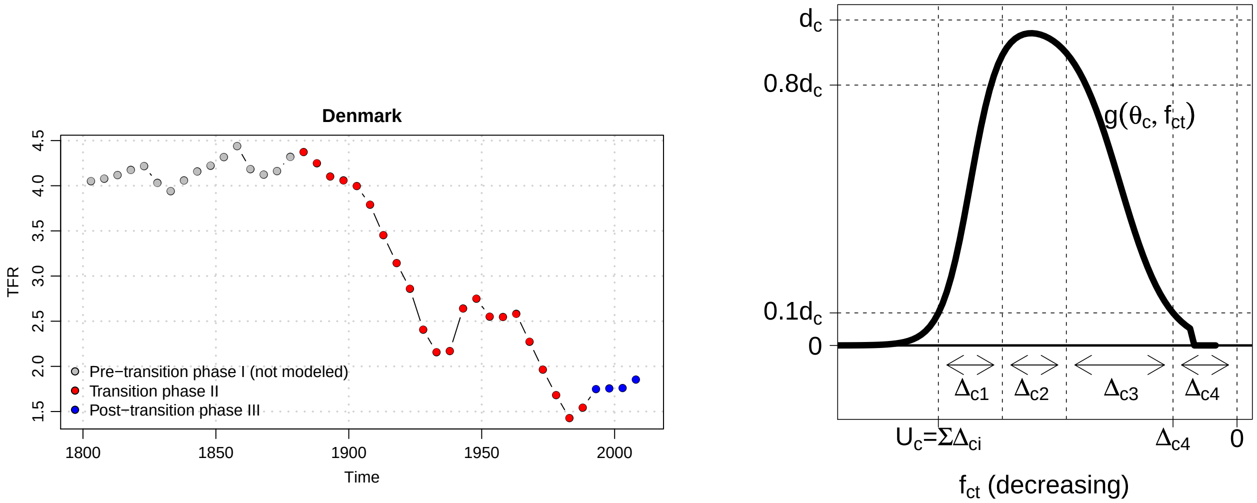


Figure 2: Left panel: TFR time series for Denmark divided into three phases. Right panel: Hypothetical double logistic decline function plotted agains TFR.

```
> install.packages("bayesPop", dependencies=TRUE)
```

from the R environment. This installs all three packages at once. To load them into the environment, type the command

```
> library(bayesPop)
```

When describing a function, only its main functionality will be mentioned, without going into great detail about its arguments. More information can be obtained from the functions' help pages, using ?*function_name*.

3 Total Fertility Rate

The **bayesTFR** package (Ševčíková et al., 2011) implements a methodology for probabilistic projection of TFR as proposed by Alkema et al. (2011) and Raftery, Alkema & Gerland (2013).

The model is based on the observation that the evolution of the TFR includes three broad phases, referred to as Phase I, II and III: (I) a high-fertility pre-transition phase, (II) the fertility transition in which the TFR decreases from high fertility levels towards or below replacement level fertility, and (III) a low-fertility post-transition phase, which includes recovery from below-replacement fertility toward replacement fertility and oscillations around replacement-level fertility. The left panel of Figure 2 shows an example of a TFR time series for Denmark divided into the three phases.

Phase II, the fertility transition phase, is modeled by a random walk with a nonconstant drift. The drift, or TFR decline, is a double logistic function with a country-specific set of parameters that defines the shape of the decline curve (see right panel of Figure 2). These parameters are estimated using a Bayesian hierarchical model in which the country-specific decline curves arise from a "world" distribution. Alkema et al. (2011) and Ševčíková et al. (2011) describe this part of the model in more detail.

Phase III, the post-transition phase is modeled as a first-order autoregressive model where the model parameters are allowed to vary between countries, following a Bayesian hierarchical model (Raftery, Alkema & Gerland, 2013).

3.1 Simulating TFR Future Trajectories

This section explains how to generate TFR future trajectories using the **bayesTFR** package. After loading the package into the environment as shown in Section 2, choose a directory on which the package will operate:

```
> tfr.dir <- "/my/TFR/directory"
```

There are three main steps to obtain TFR projections, each of which translates into one function call:

1. Estimate the parameters of the Phase II model using Markov Chain Monte Carlo (MCMC):

   ```
   > mc1 <- run.tfr.mcmc(output.dir=tfr.dir, iter="auto", wpp.year=2012,
                          start.year=1950, present.year=2010)
   ```

   Note that such a call is set up to produce long MCMC chains, ideally passing convergence diagnostics. Thus, its processing can take a very long time, possibly multiple days. For a toy simulation, one can set the argument `iter` to a small number, for example on the order of tens. The argument `wpp.year` determines which historical data are being used; in this case it is a dataset from the R package **wpp2012** (Ševčíková et al., 2013). For some countries, that dataset goes back to year 1750 and the argument `start.year` determines the first time period to be used, i.e. data prior to `start.year` are ignored. If an analyst wishes to (possibly partly) replace the default UN dataset with his/her own data, an argument `my.tfr.file` can be used that points to a user-defined historical TFR dataset.

2. Estimate the parameters of the Phase III model by MCMC:

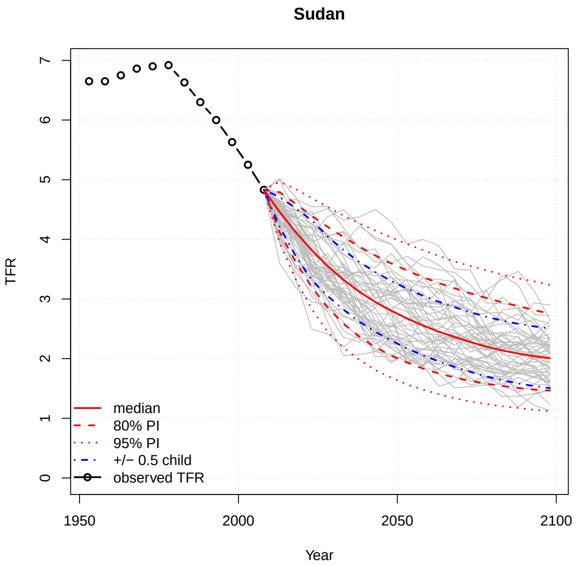
   ```
   > mc2 <- run.tfr3.mcmc(sim.dir=tfr.dir, iter="auto")
   ```

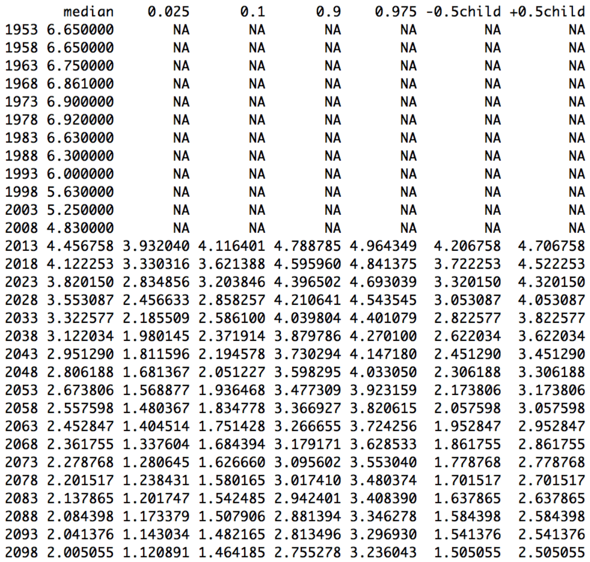
   The Phase III MCMCs will run substantially faster than step 1. Nevertheless again, `iter` should be decreased for a toy simulation. Any time-specific arguments are inherited from the Phase II simulation. In the estimation, the historical TFR values from countries that reached Phase III between `start.year` and `present.year` are used. Again, the default historical dataset can be overwritten with a user-specific one using the optional argument `my.tfr.file`.

3. Using the estimated parameters, generate future TFR trajectories:

   ```
   > tfr.pred <- tfr.predict(sim.dir=tfr.dir, end.year=2100, use.diagnostics=TRUE)
   ```

   The `use.diagnostics` argument should be set to `TRUE` only if `iter="auto"` was used in the estimation commands. In such a case, the MCMC burn-in and thin (or equivalently number of trajectories) is automatically determined from existing diagnostics. If a toy simulation was generated, these values should be provided explicitly, e.g. `burnin=10`, `burnin3=10`, `nr.traj=100`. This command produces a set of future TFR trajectories for each country included in the historical dataset and stores it in the simulation directory. Any missing values prior to `present.year` (set in Step 1) are treated as projections and are imputed by the median of the posterior distribution. Note that only consecutive missing values at the end of the time series are allowed.





```
        median    0.025      0.1      0.9    0.975 -0.5child +0.5child
1953 6.650000       NA       NA       NA       NA        NA        NA
1958 6.650000       NA       NA       NA       NA        NA        NA
1963 6.750000       NA       NA       NA       NA        NA        NA
1968 6.861000       NA       NA       NA       NA        NA        NA
1973 6.900000       NA       NA       NA       NA        NA        NA
1978 6.920000       NA       NA       NA       NA        NA        NA
1983 6.630000       NA       NA       NA       NA        NA        NA
1988 6.300000       NA       NA       NA       NA        NA        NA
1993 6.000000       NA       NA       NA       NA        NA        NA
1998 5.630000       NA       NA       NA       NA        NA        NA
2003 5.250000       NA       NA       NA       NA        NA        NA
2008 4.830000       NA       NA       NA       NA        NA        NA
2013 4.456758 3.932040 4.116401 4.788785 4.964349  4.206758  4.706758
2018 4.122253 3.330316 3.621388 4.595960 4.841375  3.722253  4.522253
2023 3.820150 2.834856 3.203846 4.396502 4.693039  3.320150  4.320150
2028 3.553087 2.456633 2.858257 4.210641 4.543545  3.053087  4.053087
2033 3.322577 2.185509 2.586100 4.039804 4.401079  2.822577  3.822577
2038 3.122034 1.980145 2.371914 3.879786 4.270100  2.622034  3.622034
2043 2.951290 1.811596 2.194578 3.730294 4.147180  2.451290  3.451290
2048 2.806188 1.681367 2.051227 3.598295 4.033050  2.306188  3.306188
2053 2.673806 1.568877 1.936468 3.477309 3.923159  2.173806  3.173806
2058 2.557598 1.480367 1.834778 3.366927 3.820615  2.057598  3.057598
2063 2.452847 1.404514 1.751428 3.266655 3.724256  1.952847  2.952847
2068 2.361755 1.337604 1.684394 3.179171 3.628533  1.861755  2.861755
2073 2.278768 1.280645 1.626660 3.095602 3.553040  1.778768  2.778768
2078 2.201517 1.238431 1.580165 3.017410 3.480374  1.701517  2.701517
2083 2.137865 1.201747 1.542485 2.942401 3.408390  1.637865  2.637865
2088 2.084398 1.173379 1.507906 2.881394 3.346278  1.584398  2.584398
2093 2.041376 1.143034 1.482165 2.813496 3.296930  1.541376  2.541376
2098 2.005055 1.120891 1.464185 2.755278 3.236043  1.505055  2.505055
```

Figure 3: Future TFR trajectories as graph (left) and table (right) for Sudan.

A simulation generated with these three steps is sufficient for use in a population projection. In addition, the package contains various functions to explore the future TFR trajectories, such as trajectories graphs and TFR world maps. To retrieve projections from the disk, i.e. to obtain the object `tfr.pred` in later R session, then create a graph for one country and view tabular values of such graph, one can do:

```
> tfr.pred <- get.tfr.prediction(tfr.dir)
> tfr.trajectories.plot(tfr.pred, "Sudan", nr.traj=50)
> tfr.trajectories.table(tfr.pred, "Sudan")
```

Results of the last two commands can be seen in Figure 3. More details on **bayesTFR** functions and methodology can be found in Ševčíková et al. (2011) and in the package help pages.

4 Life Expectancy

In WPP 2012 for the first time, the UN Population Division used a probabilistic model to project life expectancy at birth (e_0). It follows a Bayesian hierarchical model introduced by Raftery, Chunn, Gerland & Ševčíková (2013) which is implemented in the R package **bayesLife** (Ševčíková & Raftery, 2013b).

Similarly to TFR, life expectancy is modeled using a random walk with drift where the drift, or life expectancy increase, is modeled by a double-logistic function with country-specific parameters determining the shape of the function. This model allows one to pool information about the rates of gains across countries by assuming that each set of country-specific double-logistic parameters is randomly sampled from a common (world) distribution.

The population projection methodology requires projections of female and male life expectancy simultaneously. Generating them independently from the model of Raftery, Chunn, Gerland & Ševčíková (2013) is unsatisfactory because there is generally a strong relationship between them, and ignoring this can lead to future trajectories of female and male life expectancy that diverge unrealistically.

Lalic & Raftery (2012) proposed a method for joint projections of female and male life expectancy that models the gap between them, the gap being defined as female e_0 minus male e_0. In such a model, projections of female life expectancy are generated by the Bayesian hierarchical model, and are then combined with projections of the gap to produce projections of male life expectancy.

4.1 Simulating Future Trajectories of Life Expectancy

The implementation of the Bayesian hierarchical model in **bayesLife** closely follows the structure of the **bayesTFR** package. Many functions in one package have their analogues in the other, and their names differ only in the part called "tfr" versus "e0". As for TFR, we first set a directory for the e_0 simulation:

```
> e0.dir <- "/my/e0/directory"
```

Obtaining e_0 projections involves two steps:

1. Estimate parameters for female e_0 via MCMC:

   ```
   > mc <- run.e0.mcmc(output.dir=e0.dir, sex="Female", iter="auto", wpp.year=2012)
   ```

   The note from above regarding long processing time for an "auto" setting applies here as well. The function has many of the same arguments as `run.tfr.mcmc` with the same meaning, for example `my.e0.file` for specifying user-defined (female) historical data, or arguments specifying time (`*.year`), some of which are left out here as their defaults are to be used.

2. Using estimated parameters, generate future female and male e_0 trajectories.

   ```
   > e0.pred <- e0.predict(sim.dir=e0.dir, end.year=2100, use.diagnostics=TRUE)
   ```

   Again, the `use.diagnostics` argument is to be used in combination with an "auto" simulation only. This call first generates trajectories of female e_0, then estimates the gap model, predicts the gap, and finally produces trajectories of male e_0. Any user-defined data on male e_0 would be passed here via an optional `my.e0.file` argument.

The package supports various ways of viewing the results. In addition to viewing trajectories for each sex separately, one can create graphs of the marginal distribution as well as the joint distribution of life expectancy for the two sexes:

```
> e0.pred <- get.e0.prediction(e0.dir)
> e0.trajectories.plot(e0.pred, "Brazil", nr.traj=10, both.sexes=TRUE)
> e0.joint.plot(e0.pred, "Brazil", nr.points=100, pi=80, years=c(2010, 2050, 2100))
```

The results are shown in Figure 4. Other functions for exploring results are available in **bayesLife**, such as creating maps, or generating plots for all countries at once.

5 Population Projection

5.1 Producing Future Population Trajectories

As mentioned in Section 2, probabilistic population projections implemented in **bayesPop** incorporate two probabilistic components, namely TFR and sex-specific e_0. For each country, the prediction function applies the cohort component method to each trajectory, keeping the remaining input components constant.

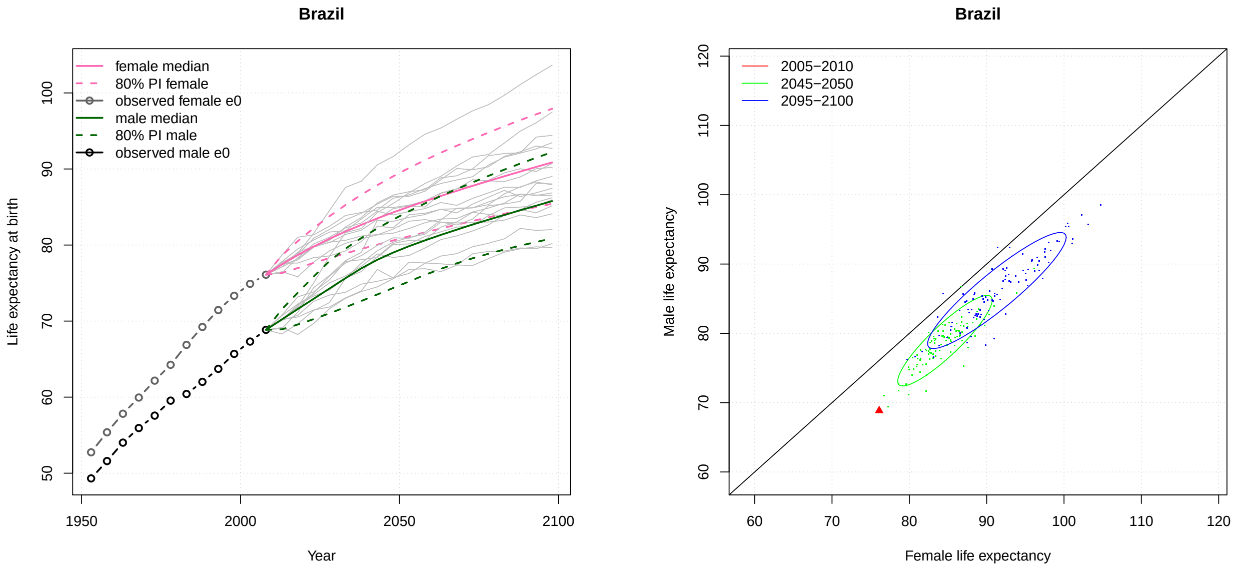


Figure 4: Future e_0 marginal trajectories (left) and joint female-male distribution for three different time points (right) with 80% probability interval for Brazil.

Thus in standard cases, to generate future population trajectories, all one needs to do is to point the code to the place on disk where TFR and e_0 are stored, in our case the `tfr.dir` and `e0.dir` directories. The results are stored in a separate directory:

```
> pop.dir <- "/my/pop/directory"
> pop.pred <- pop.predict(end.year=2100, start.year=1950, present.year=2010,
                          wpp.year=2012, output.dir=pop.dir, nr.traj=1000,
                          inputs=list(tfr.sim.dir=tfr.dir,
                                      e0F.sim.dir=e0.dir, e0M.sim.dir="joint_"))
```

The keyword "joint_" directs the function to extract the male e_0 projections from the female simulation directory, i.e. it indicates that e_0 was simulated jointly for female and male. The `start.year` determines the first time period for observed death rates to be used in the Lee-Carter method, and `present.year` is the time period of the initial population data. Obviously, `end.year` cannot go beyond the end year used in TFR and e_0 projections, nor beyond the end year of other projection inputs, such as migration. If the number of trajectories to be produced (`nr.traj`) is smaller than the number of available TFR and e_0 trajectories, these are equidistantly thinned. The deterministic input components are taken from the given wpp package, here **wpp2012**. However, the `inputs` argument allows one to overwrite each of them using tab-delimited text files. Instead of one's own projection of TFR and e_0, it is possible to use the UN-predicted quantiles (included in the **wpp2012** package) by leaving all the `*.sim.dir` elements equal to `NULL` (the default). An optional logical argument `keep.vital.events` can be used for storing additional data generated during projection, such as births and deaths. By default these are not kept, as it more than doubles the amount of data stored.

To access such population prediction objects in a later session one can use:

```
> pop.pred <- get.pop.prediction(pop.dir)
```

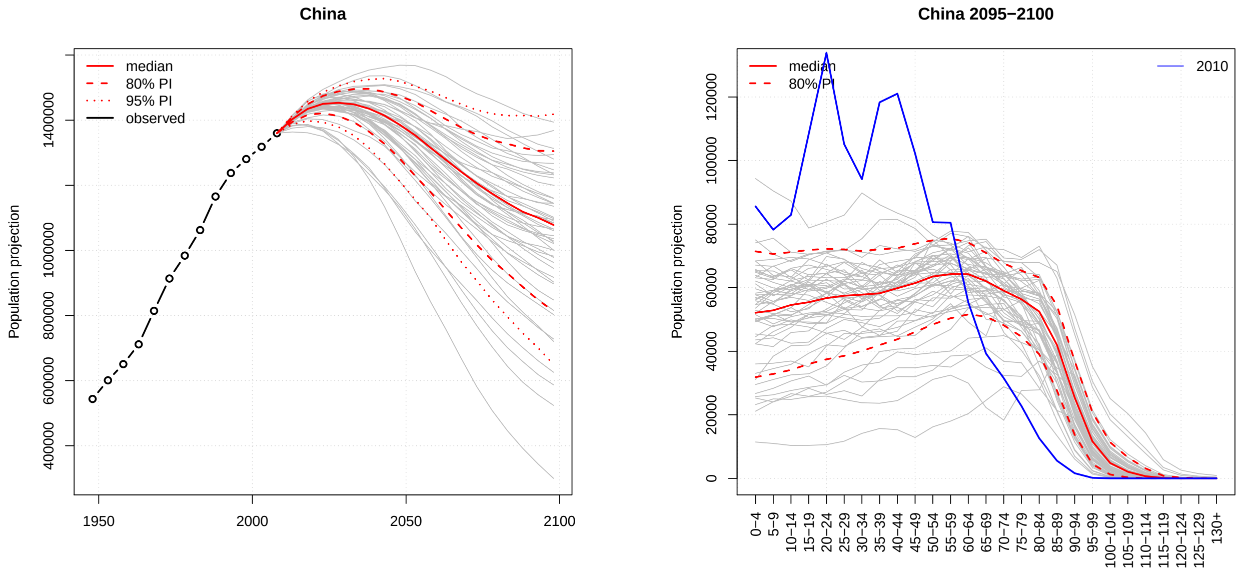


Figure 5: Projected trajectories for China. Left panel shows the population projection by time; the right panel shows the population projection by age for 2100 (red lines) and for the present year, 2010, (blue line).

5.2 Viewing Population Trajectories

Population trajectories can be viewed on a country-specific basis. A simple `summary` function gives a quick look at various quantiles of the country's projections, e.g.:

```
> summary(pop.pred, country="Italy")
```

The following code shows how to create plots of trajectories by time as well as by age, including adding curves to existing plots. Results are shown in Figure 5.

```
> country <- "China"
> pop.trajectories.plot(pop.pred, country=country, sum.over.ages=TRUE, nr.traj=50)
> pop.byage.plot(pop.pred, country=country, year=2100, nr.traj=50, pi=80, ylim=c(0,130000))
> pop.byage.plot(pop.pred, country=country, year=2010, add=TRUE, show.legend=FALSE, col="blue")
> legend("topright", legend=2010, col="blue", lty=1, bty="n")
```

If `sum.over.ages` in `pop.trajectories.plot` is `FALSE`, separate plots for each age group are generated. The function also accepts arguments for specifying sex and age where age is defined as an index to the vector (0-4, 5-9, 10-14, . . ., 125-129, 130+), thus of length 27. A graph for male population aged 0-14 would have arguments `sex="male", age=1:3`, or women in child-bearing age would be defined as `sex="female", age=4:10`.

Numerical analogous to the trajectory plots is implemented in the functions `pop.trajectories.table` and `pop.byage.table`, respectively. Trajectory plots can be created for all countries at once using `pop.trajectories.plotAll` and `pop.byage.plotAll`.

5.3 Probabilistic Population Pyramids

The package supports plotting probabilistic population pyramids for given country and one or more given years. There are two different kinds of pyramids – a *classic pyramid* consisting of boxes, and

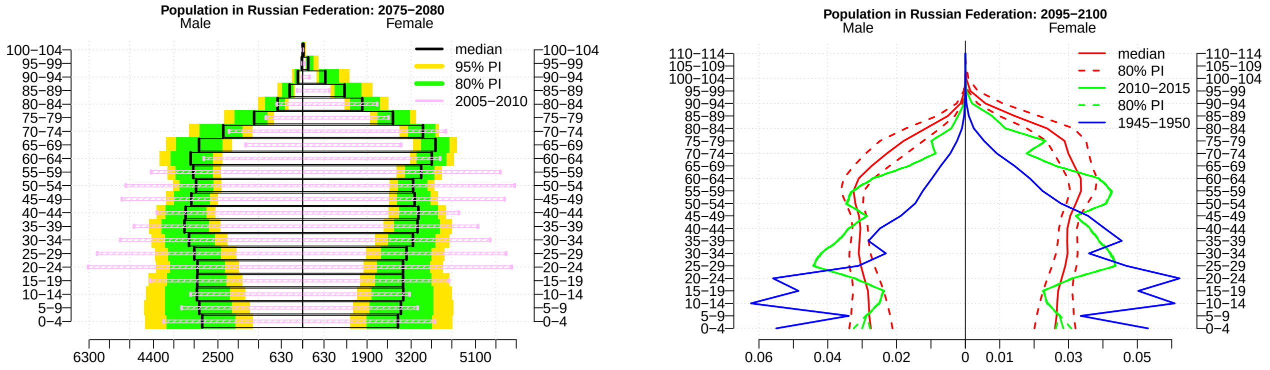


Figure 6: Probabilistic population pyramid for Russia: a classic type comparing two time periods on the left, and trajectory type comparing three time periods on a proportion scale on the right.

a so called *trajectory pyramid* which is created using age trajectories such as the ones on the right in Figure 5. The classic pyramid can display projections for up to two years in one pyramid with one set of probability intervals; the trajectory pyramid can include any number of years and any number of probability intervals. Here is an example of generating the two pyramid types, see Figure 6:

```
> country <- "Russian Federation"
> pop.pyramid(pop.pred, country, year=c(2080, 2010))
> pop.trajectories.pyramid(pop.pred, country, year=c(2100, 2015, 1950),
                           nr.traj=0, proportion=TRUE, age=1:23, pi=80)
```

Here the trajectory pyramid uses the argument `proportion` to switch the x-axis to a proportional scale, which is useful when comparing pyramids from different time periods. Both functions also accept various arguments for changing the appearance of the pyramids, such as colors, height and thickness of boxes etc.

In addition to creating pyramids for the results of the `pop.predict` function, both pyramid functions can also be applied to any user-defined data that can be fitted into a pyramid data structure. See the package documentation for more detail.

5.4 bayesPop Expressions

So far we have explored data resulting directly from the `pop.predict` function, which provides information about sex- and age-specific population projections. It is often of interest, however, to analyse various derivations and transformations of these quantities, such as potential support ratio, mean age of child-bearing, median age etc. For this purpose, the package implements a simple expression language that allows one to compute such quantities on the fly.

A **bayesPop** expression is a collection of *basic components* connected via usual arithmetic operators and combined using parentheses. Standard R functions and pre-defined functions can be also used within expressions.

A *basic component* of an expression is a character string consisting of four sub-strings, the first two of which are mandatory. They must be in the following order:

1. Measure identification. The following upper-case characters are currently allowed: **P** for Population, **D** for Deaths, **B** for Births, **S** for Survival ratio, **F** for Fertility rate, **M** for Mortality rate, **Q** for probability of dying, and **G** for net miGration. All but the P and G indicators can be used only if `keep.vital.events` was set to `TRUE` when generating predictions. P and G are always available.

2. Country part which can be either a numerical country code or two- or three-character ISO 3166 code, or characters "XXX" which serves as a wildcard for a country code. For example, "P528", "PNL", and "PNLD" are all expressions for the total population of the Netherlands.

3. Sex sub-string (optional) which is either "_F" or "_M", specifying female or male indicator, respectively. An expression consisting of two basic components "P528_F / P528" gives the ratio of the female population to the total population in the Netherlands.

4. Age sub-string (optional). If used, the basic component is concluded by an array of age indices. Such an array is enclosed by either brackets ("[ ]") or curly braces ("{ }"). The former invokes a summation of counts over given ages, and the latter is used when no summation is desired. If the age sub-string is missing, counts are automatically summed over all ages. To use all ages without summing, empty curly braces can be used. For example, the female population of India of child-bearing age can be expressed as "PIND_F[4:10]". Indicators S, M and Q allow an index −1 which corresponds to the age group 0–1, and an index 0 which corresponds to the age group 1–4.

Not all combinations of the four parts above make sense. For example, fertility rate can be combined only with female sex and a subset of the age groups, namely child-bearing ages (indices 4 to 10). Births are also restricted to those age groups. As another example, all the rate-like indicators (S, F, M, Q) should include all four components, as summing over sexes or age groups is meaningless for this type of measure.

Basic components can be combined using arithmetic operations, R functions, or pre-defined functions in the package. Among the most useful pre-defined functions are `gmedian`, `gmean` for group median and group mean, respectively, and `pop.apply` for applying a function along the age axis of the data. Here are a few examples of bayesPop expressions and their use in the package functions (please refer to the help page `?pop.expressions` for more details):

Summarising by time: These country-specific expressions can be used in `pop.trajectories.plot` and `pop.trajectories.table`.

- Median age of France:
  "pop.apply(PFR{}, gmedian, cats=seq(0, by=5, length=28))"
- Average age of women in the USA in child-bearing age:
  "pop.apply(PUSA_F{4:10}, gmean, cats=seq(15, by=5, length=8))"
- Ratio of German to French total population: "PDE / PFR"
- Potential support ratio of India: "PIND[5:13] / PIND[14:27]"

Summarising by age: Country-specific expressions that can be used in `pop.byage.plot` and `pop.byage.table` in which time must be given.

- Log of male mortality rate for Spain: "log(MES_M{})"
- Migration per capita in the Netherlands: "GNL{} / PNL{}"

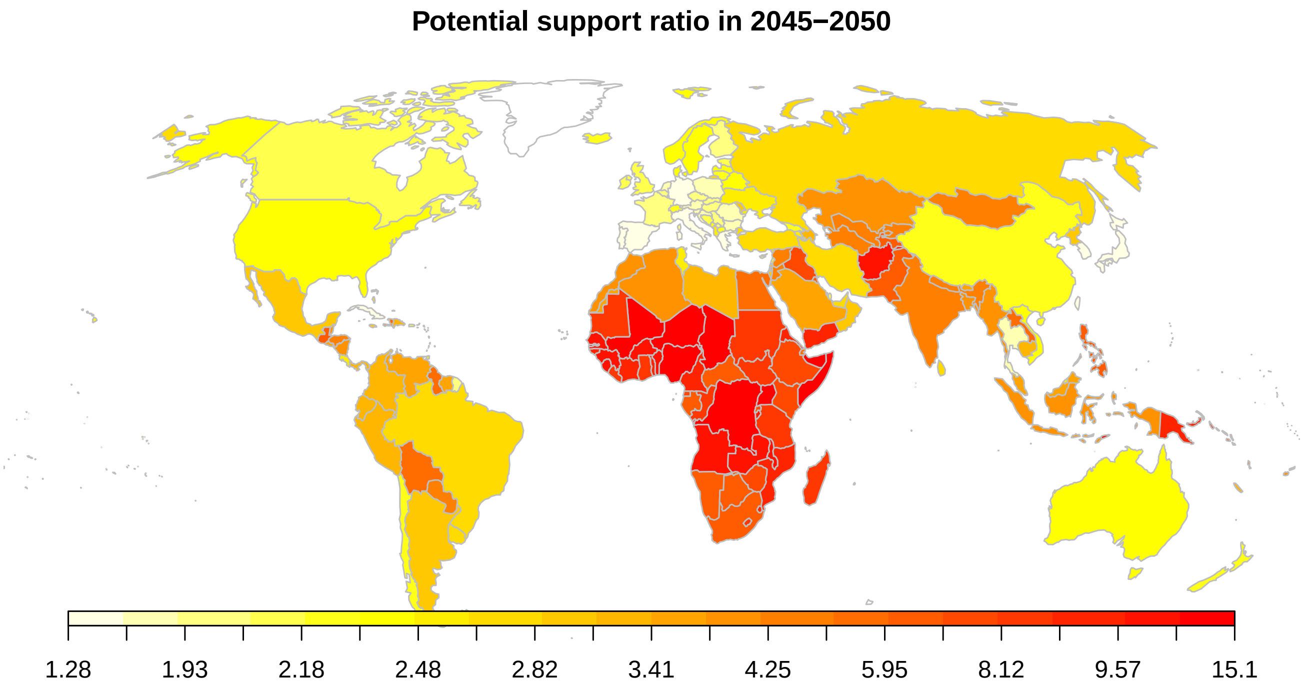


Figure 7: World map of the median projection of potential support ratio in 2045-2050.

- Number of births by mother's age per woman of the same age in Hungary: "BHU{} / PHU_F{4:10}"

All countries: The character string "XXX" can be used in functions for plotting maps, such as `pop.map` and `pop.map.gvis`, or a function `write.pop.projection.summary` for exporting projection results into an ASCII file. These expressions should be constructed the same way as the "by time" expressions above, but the country code should be replaced by "XXX", e.g. the following code generates the map in Figure 7 for potential support ratio:

```
> pop.map(pop.pred, expression = "PXXX[5:13] / PXXX[14:27]", year=2050,
          main="Potential support ratio in 2045-2050", numCats=20)
```

The same applies when using expressions in `pop.trajectories.plotAll`. Function `pop.byage.plotAll` accepts expressions containing "XXX" constructed as expressions "by age" above.

5.5 Aggregations

In addition to producing population estimates and projections at the country level, the UN also provides projections for numerous country aggregates of interest, such as geographic regions and trading blocs. **bayesPop** offers two methods for producing aggregations, namely an Independence method and a Regional method. The *Independence method* treats population trajectories as independent between countries, and thus, the aggregation is accomplished by simply summing population counts on each trajectory across countries of the regions in question. In the *Regional method*, aggregations are generated using the cohort component method as described in Section 2 using aggregated input components. Here is an example of aggregating over continents and over the whole world:

```
> pop.aggr <- pop.aggregate(pop.pred, method="independence",
```

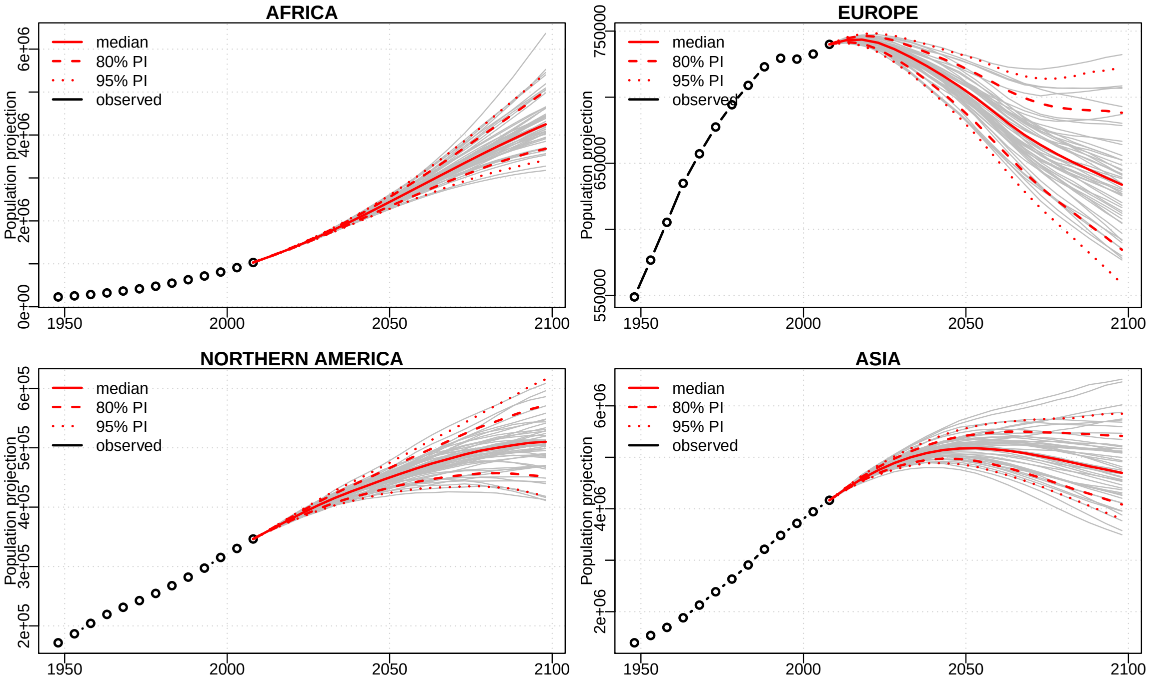


Figure 8: Population trajectories for aggregated regions.

```
        # World, Africa, Europe, Northern America, Asia, Latin Am.
        regions=c(900, 903, 908, 905, 935, 904), verbose=TRUE)
```

The region codes must correspond to the column “area_code” of the `UNlocations` dataset in the **wpp2012** package. Alternatively, user-defined aggregations are also supported, see `?pop.aggregate` for more information.

Results of the aggregation are stored in the same directory as `pop.pred` and can be retrieved in later sessions by

```
> get.pop.aggregation(pop.dir)
```

Both functions above accept an argument `name`. Thus, one can label an aggregation and keep multiple aggregation objects in the same directory.

The stored data have the same structure as the non-aggregated prediction object. Thus, any of the summarising and plotting function described in the previous sections can be used, including in combination with expressions:

```
> par(mfrow=c(2,2))
> for (region in c(903, 908, 905, 935))
        pop.trajectories.plot(pop.aggr, region, sum.over.ages=TRUE, nr.traj=50)

> pop.pyramid(pop.aggr, 900, year=c(2100, 2010), proportion=TRUE)

> pop.byage.plot(pop.aggr, expression="P903{} / P900{}", year=2100, pi=80,
        main = "African population to world population 2095-2100",
        nr.traj=50)
```

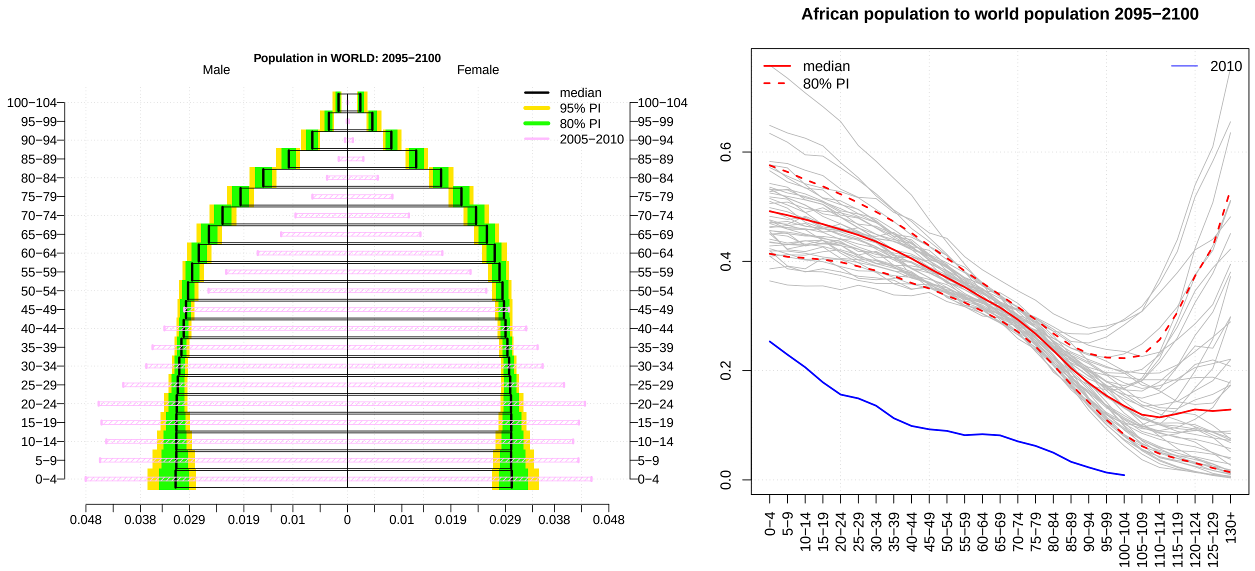


Figure 9: Left panel: Population pyramid for the world in 2010 and 2100. Right panel: Proportion of Africans to the world population by age in 2100 and the same indicator in 2010 (blue line).

```
> pop.byage.plot(pop.aggr, expression="P903{} / P900{}", year=2010,
           nr.traj=0, add=TRUE, show.legend=FALSE, col="blue")
> legend("topright", legend=2010, col="blue", lty=1, bty="n")
```

Resulting graphs are shown in Figures 8–9. Note that several of the countries in Africa are experiencing generalised HIV/AIDS epidemics, and for these countries the projected life expectancies in WPP 2012 were generated by **bayesLife** using different settings than used for other countries. The results shown here are based on the life expectancies published in WPP 2012.

6 Graphical User Interface

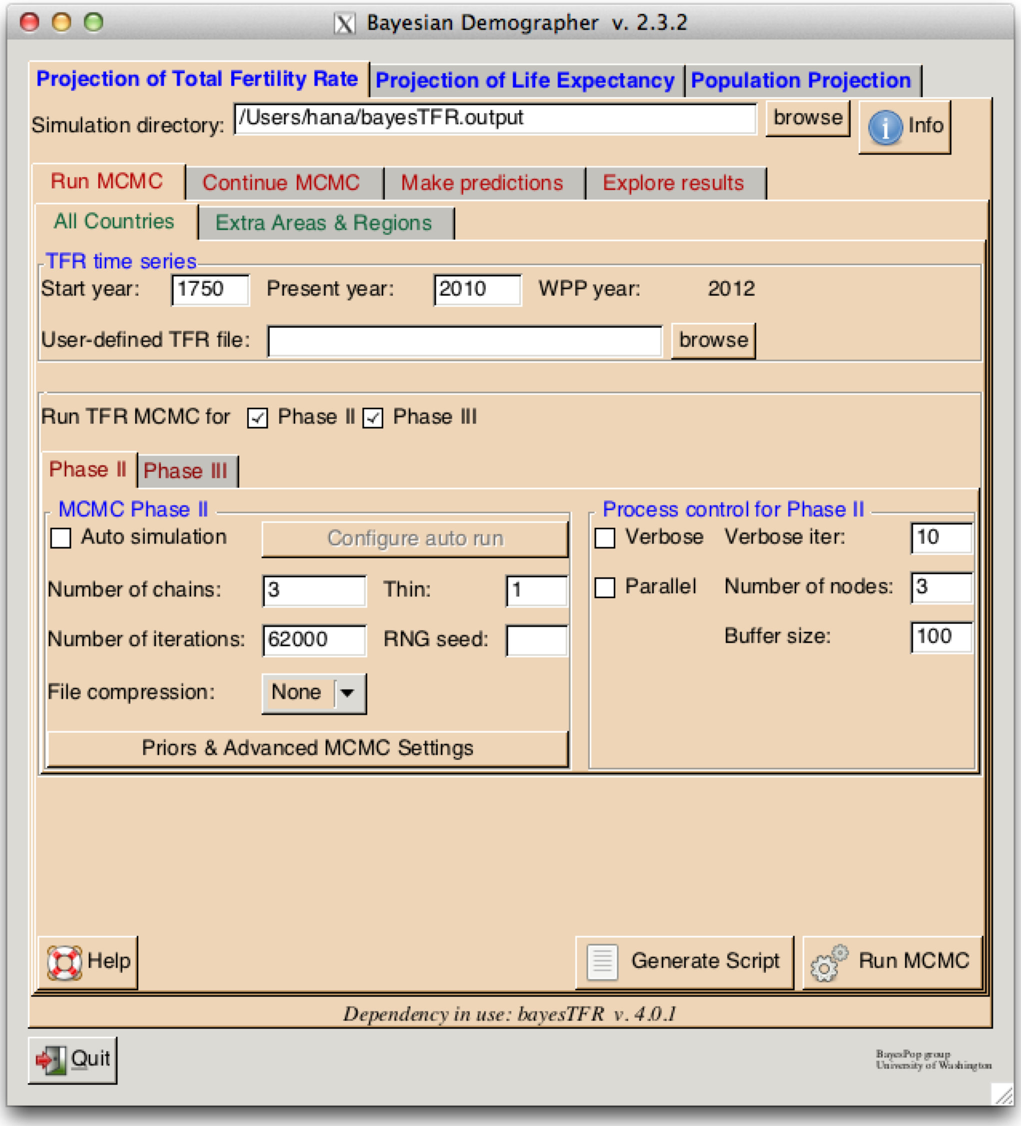
6.1 bayesDem: Bayesian Demographer

A graphical user interface for **bayesTFR**, **bayesLife**, and **bayesPop** is implemented in the R package **bayesDem** (*Bayesian Demographer*). One can generate probabilistic projections of TFR, life expectancy, and combine those results into probabilistic population projections from a single interface. In addition, it offers functionality for exploring results, such as trajectories, maps, population pyramids and others.

Bayesian Demographer can be loaded into and started from the R interface by

```
> library(bayesDem)
> bayesDem.go()
```

which opens the main window of the GUI (see Figure 10). There are three upper level tabs each of which corresponds to one of the underlying packages. Each package operates on its own directory which are entered in the “Simulation directory” field on the top of the GUI. The “Info” button beside it gives information about estimation and projection objects already stored in the given directory.



Simulation directory: /Users/hana/bayesTFR.output

Start year: 1750 Present year: 2010 WPP year: 2012

User-defined TFR file:

Run TFR MCMC for ☑ Phase II ☑ Phase III

□ Auto simulation

Number of chains: 3 Thin: 1

Number of iterations: 62000 RNG seed:

File compression: None

□ Verbose Verbose iter: 10

□ Parallel Number of nodes: 3

Buffer size: 100

Dependency in use: bayesTFR v. 4.0.1

Figure 10: Bayesian Demographer: Graphical user interface for **bayesTFR**, **bayesLife**, and **bayesPop**.

Each main tab is organised into sub-tabs, each of which corresponds to a function (or group of related functions) of the corresponding package. They are ordered from left to right in the same way an analyst would progress on his/her road to population projection along the blue arrows in Figure 1.

In the bottom right corner there is usually a button that invokes the actual function after collecting its arguments from the workspace. A button "Generate Script" shows how the function is invoked. The user can use this functionality to copy and paste function calls into a batch file which can be then processed outside of the GUI. This is especially useful and recommended for time-consuming processes, such as MCMC runs or generating predictions. In order to help users with the meaning of the various inputs, there is a "Help" button in the bottom left corner which shows help pages for the underlying R function(s).

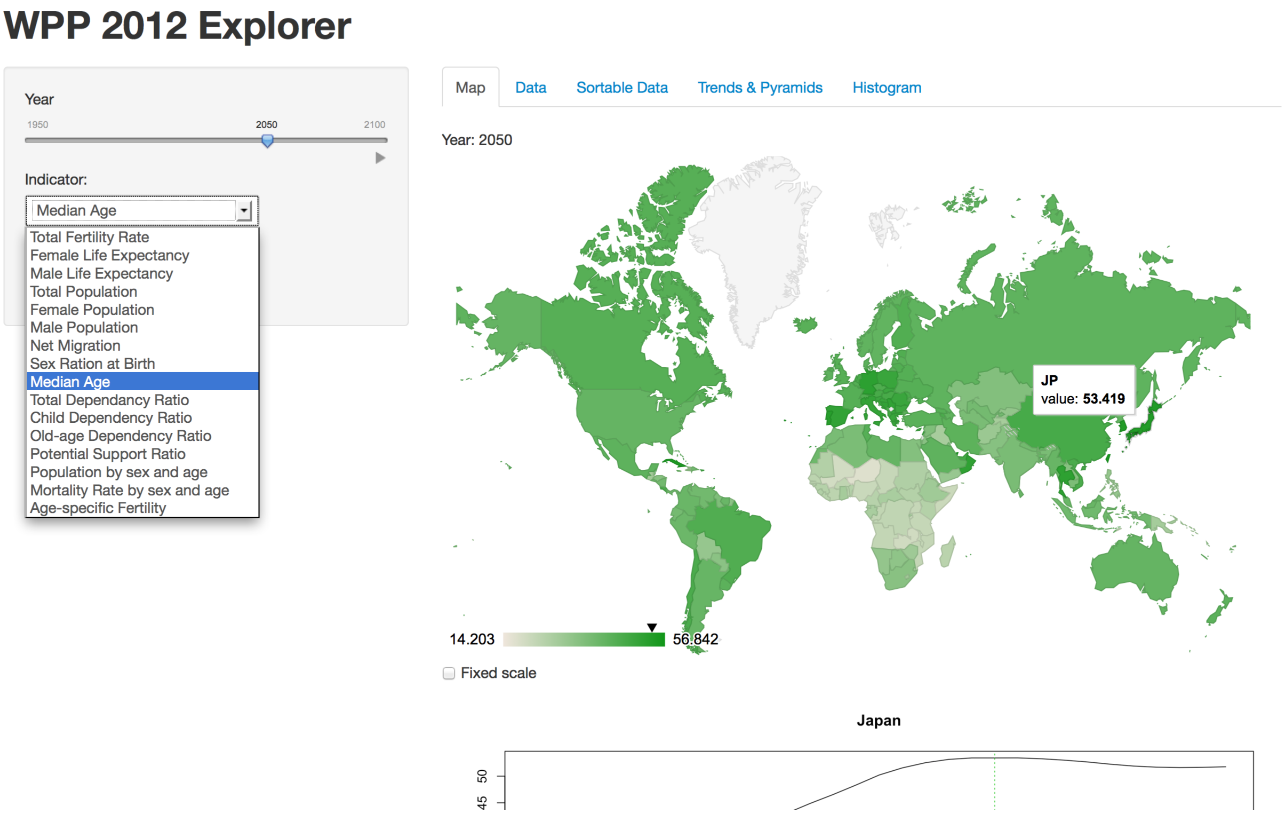
All functions described in this paper and many more can be accessed through the GUI.

6.2 wppExplorer: Interface for UN Estimates and Projections

Many datasets from the latest WPP revision are included in the R package **wpp2012** (Ševčíková et al., 2013). We have developed a package for easy visualisation of these datasets and their derivatives, called **wppExplorer** (Ševčíková, 2013b), which is based on package **shiny** (RStudio & Inc., 2013). One can load and start the exploration by

```
> library(wppExplorer)
> wpp.explore()
```

It opens an interface in user's default browser and offers interactive maps, tables, time series plots, histograms and pyramids. Where available, uncertainty is also included. Figures 11 and 12 show examples of the interface. An optional argument to `wpp.explore` can be used to switch to explore data from previous revisions of WPP, namely 2010 or 2008, in which case R packages **wpp2010** or **wpp2008** would be used.



WPP 2012 Explorer

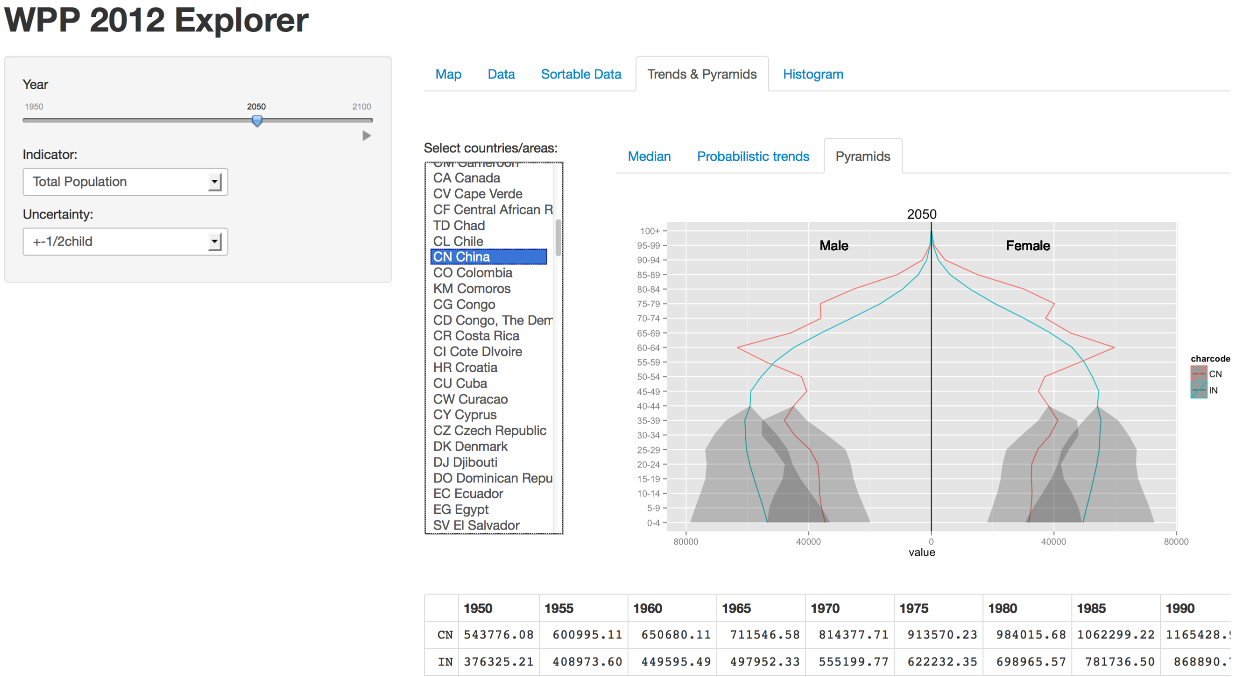
Year: 2050

Indicator: Median Age

Figure 11: wppExplorer: Interactive exploration of the **wpp** packages (map of median age).

7 Discussion

We have shown the basic functionality of our demographic packages, namely **bayesTFR**, **bayesLife**, **bayesPop**, **bayesDem**, and **wppExplorer**. They can be used to reproduce some of the UN WPP 2012 demographic projections and visualize results. The packages offer additional features not mentioned in this paper, such as special handling of small areas, imputing missing values, analysing MCMCs,



WPP 2012 Explorer

Year: 2050

Indicator: Total Population

Uncertainty: +-1/2child

Select countries/areas:

CM Cameroon
CA Canada
CV Cape Verde
CF Central African R
TD Chad
CL Chile
CN China
CO Colombia
KM Comoros
CG Congo
CD Congo, The Dem
CR Costa Rica
CI Cote DIvoire
HR Croatia
CU Cuba
CW Curacao
CY Cyprus
CZ Czech Republic
DK Denmark
DJ Djibouti
DO Dominican Repu
EC Ecuador
EG Egypt
SV El Salvador

	1950	1955	1960	1965	1970	1975	1980	1985	1990
CN	543776.08	600995.11	650680.11	711546.58	814377.71	913570.23	984015.68	1062299.22	1165428.
IN	376325.21	408973.60	449595.49	497952.33	555199.77	622232.35	698965.57	781736.50	868890.

Figure 12: wppExplorer: Interactive exploration of the **wpp** packages (Population pyramid of China and India).

exporting projections, etc. We refer the reader to the corresponding package documentation for more details.

Acknowledgements: This work was supported by the Eunice Kennedy Shriver National Institute of Child Health and Human Development under grants R01 HD054511 and R01 HD070936. Raftery's research was also supported by Science Foundation Ireland E.T.S. Walton visitor award 11/W.1/I2079.

References

Alkema, L., Raftery, A. E., Gerland, P., Clark, S. J., Pelletier, F., Buettner, T., & Heilig, G. K. (2011). Probabilistic projections of the total fertility rate for all countries. *Demography*, *48*, 815–839.

Ihaka, R. & Gentleman, R. (1996). R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, *5*, 299–314.

Lalic, N. & Raftery, A. E. (2012). Joint probabilistic projection of female and male life expectancy. Presented at the annual meeting of Population Association of America. `http://paa2012.princeton.edu/abstracts/120140`.

Lee, R. D. & Carter, L. (1992). Modeling and forecasting the time series of US mortality. *Journal of the American Statistical Association*, *87*, 659–671.

Lee, R. D. & Tuljapurkar, S. (1994). Stochastic population forecasts for the United States: Beyond high, medium, and low. *Journal of the American Statistical Association*, *89*, 1175–1189.

National Research Council (2000). *Beyond Six Billion: Forecasting the World's Population*. Washington, D.C.: National Academy Press.

Raftery, A. E., Alkema, L., & Gerland, P. (2013). Bayesian population projections for the United Nations. *Statistical Science, in press.*

Raftery, A. E., Chunn, J. L., Gerland, P., & Ševčíková, H. (2013). Bayesian probabilistic projections of life expectancy for all countries. *Demography*, *50*, 777–801.

Raftery, A. E., Li, N., Ševčíková, H., Gerland, P., & Heilig, G. K. (2012). Bayesian probabilistic population projections for all countries. *Proceedings of the National Academy of Sciences*, *109*, 13915–13921.

RStudio & Inc. (2013). *shiny: Web Application Framework for R.* R package version 0.7.0.

United Nations (2013). *World Population Prospects: The 2012 Revision.* New York, NY: United Nations.

Ševčíková, H. (2013a). *bayesDem: Graphical User Interface for bayesTFR, bayesLife and bayesPop.* R package version 2.3-2.

Ševčíková, H. (2013b). *wppExplorer: Explorer of World Population Prospects.* R package version 1.0-3.

Ševčíková, H., Alkema, L., & Raftery, A. E. (2011). bayesTFR: An R package for probabilistic projections of the total fertility rate. *Journal of Statistical Software*, *43*, 1–29.

Ševčíková, H., Gerland, P., Andreev, K., Li, N., Gu, D., & Spoorenberg, T. (2013). *wpp2012: World Population Prospects 2012.* R package version 2.0-0.

Ševčíková, H. & Raftery, A. (2013a). *bayesPop: Probabilistic Population Projection.* R package version 4.1-1.

Ševčíková, H. & Raftery, A. E. (2013b). *bayesLife: Bayesian Projection of Life Expectancy.* R package version 2.0-1. Original WinBugs code written by Jennifer Chunn.

Whelpton, P. K. (1928). Population of the United States, 1925–1975. *American Journal of Sociology*, *31*, 253–270.

Whelpton, P. K. (1936). An empirical method for calculating future population. *Journal of the American Statistical Association*, *31*, 457–473.