Spell analysis in multistate Markov models

Tim Riffe*[1]

[1]Max Planck Institute for Demographic Research

October 28, 2017

Abstract

Multistate matrix population models are typically used to compute statistics such as the state occupancies or the moments thereof. Recently formulas have been derived to compute the average number of spells for a given state, which combined with the average state occupancy yields the average spell duration. I've since wondered what the age pattern to average spell durations is. Here I demonstrate how to use simulations to estimate a variety of age and other patterns of different state spells.

1 Introduction

The amount of population-level measures that one might devise for a particular population-level process is dizzying, so it may beg the question from the outset why we might desire to have more. Matrix-based manipulations of incidence-based models are rather undeveloped with respect to tenure-statistics, and these might be of interest for a variety of substantive reasons. By tenure statistics I refer to the statistics of particular spells or episodes of a state. Namely, in incidence-based models with bidirectional flows (i.e. allowing for recovery and then re-onset, and so on), a hypothetical individual may pass through a state (say sickness) many times before death. Typically a transition matrix manipulation would only give us the average time spent sick (or moment statistics thereof). Recently matrix calculations have been described for how to calculate the average number of episodes of a given state (Dudel and Myrskylä 2017a). Combined with the average state occupancy, this information yields the average duration of episodes.

One may wonder how the average spell duration changes with age, and for this there is no ready matrix expression (although it would of course be possible). I will procede using simulations rather than matrix calculations because it will save the work of deriving and checking dozens of formulas. In this way, we have the liberty to change definitions without incuring methodological setbacks. Since we simulate, we get stochastic stationary distributions of each measurement, which I'll represent using fan-chart visualizations. This approach is not all that different from that proposed by Laditka and Wolf (1998), but I take things a bit farther by proposing a suite of age realignments and resulting age-like patterns of episode statistics.

2 Data

The point of departure for all calculationsis a transition matrix. To demonstrate, I use a published transition matrix from a recent study of working life expectancy in the United States (Dudel and Myrskylä 2017b). This matrix refers to black females aged 50-100 in 1994. The same sorts of things can be done with any age-stage matrix. Actually technically you could do the same with an ageless matrix, if the simulated sequence steps are interpretable as age.

3 Simulation

I take advantage of the recently published `R` package `markovchain` (Spedicato 2017), which includes a random state sequence generator function `rmarkovchain()` that mereley requires a transition matrix to

*riffe@demogr.mpg.de

do its work.[1] I generate a large number, say 10k or more trajectories to operate on as the stationary population. Each trajectory consists in a `character` vector of states [`Employed`,`Inactive`,`Retired`,`Dead`], where dead is of course an absorbing state, but the other states can be switched on and off annually from ages 50 and higher.[2]

A glimpse of the first 10 randomly generated individuals is shown in Figure 1. These ten individuals will be recycled in all of the following data manipulations used to demonstrate concepts. All aggregate calculations of age patterns (and so on) are done on the full simulated population. In this case, I simulated assuming that one starts in a state of employment at age 50, but the starting state could also easily be a mixture of states.

Figure 1: Ten randomly generated state sequences from the 1994 transition matrix of black females (Dudel and Myrskylä 2017b)

4 Running clocks and alignment

Beyond counting episodes, one may wish to aggregate statistics on each episode in novel ways. For instance, conditional on being in state s in age x, what is the average duration of spell that one finds oneself in? From the previous example, I take episodes of inactivity as the reference state. I chose inactivity in part because each sequence in this simulation starts with employment. Each sequence therefore has no left truncation problem, but aggregate statistics over individuals certainly still do.

4.1 clocks

To calculate the average spell duration by age, first convert our state sequences to a data object something like Figure 2a. Using chronological age as the reference, one may also wish to calculate time spent or left in the state episode, per Figure 2[3]. In either of these cases, value alignment is with respect to episode entry or exit, but aggregation alignment remains pegged to age. Statistics across individuals in an array with therefore produce age patterns.

It is clear that one may fill episodes with other markers, such as episode order or episode fractions, and that one could further condition age patterns of total duration, time spent, or time left on episode order. If spells are filled with 1s, then aggregation results in prevalence. Note, time *left* in the episode has no left-truncation problem, also not in the aggregate. These series of values, that I call clocks, are then aggregated in some way, and aggregation is always within some external structuring classes that remain to be defined.

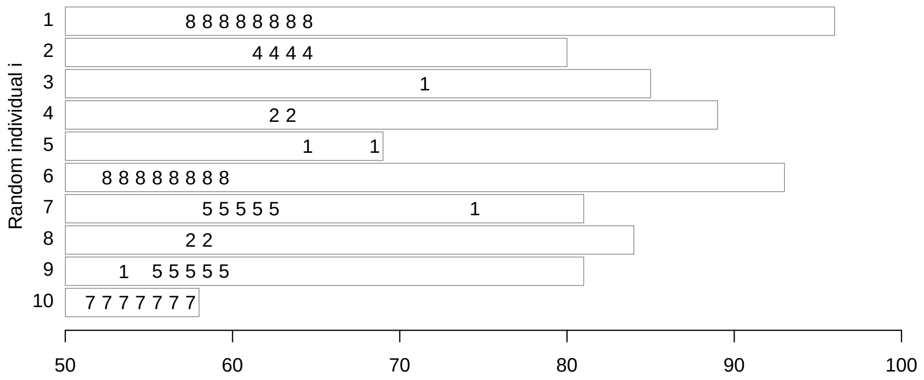
[1]There are some trivial object definition steps to convert the employment matrix into a conformable markov object before feeding it to the random generator.

[2]It would also be possible to further graduate transition rates to standardized month or week units to produce higher resolution sequences, but this is unnecessary for the present treatment.

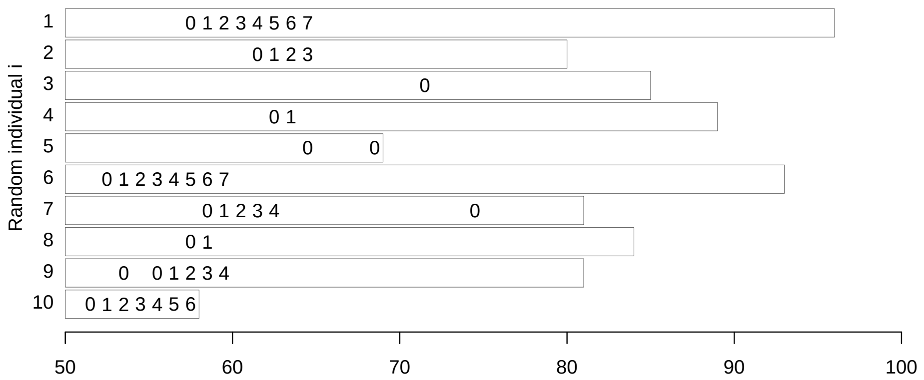
[3]Actually, I'd increment values by $\frac{1}{2}$ for mid-state clocking, but decimals would squeeze the figure too much.

Figure 2: Each age observation of *inactivity* from Figure 1 is imputed with the total or the running time spent/left in the episode. It's probably better to add $\frac{1}{2}$ to the displayed *running* values.

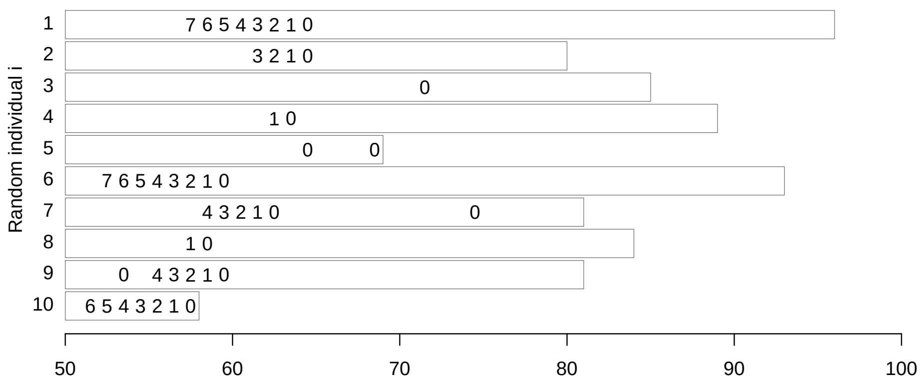
(a) Static; Total episode duration.



(b) Running; Time spent in episode.



(c) Running; Time left in episode

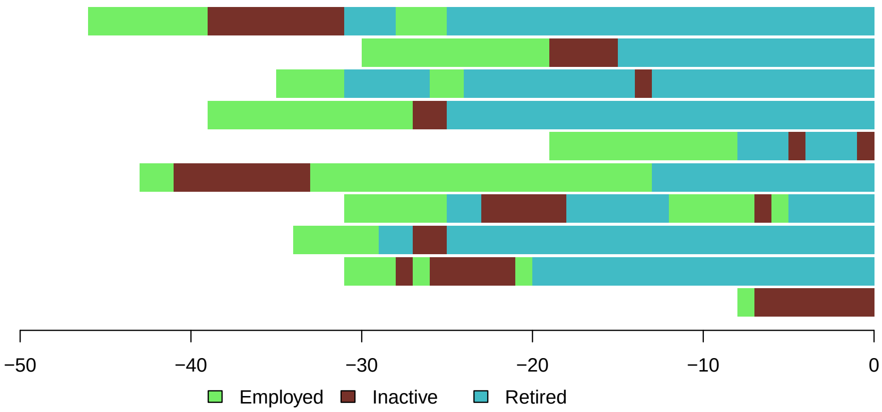


4.2 Alignment

Episodic clock values are aggregated according to some structuring criteria, such as chronological age, but other choices are also available and may be more compelling. Classifying life lines by age is the same as left-aligning life lines on the event of birth. For late-life processes, birth is usually a long ways off, and empirical regularity may be be found with respect to other alignment criteria. Aligning lifelines requires two choices: 1) a reference moment or event must be selected, and 2) the alignment direction must be chosen. A reference event could be any instance of entry, exit, or other compelling anchor point, such as a spell midpoint.

Aggregated patterns would certainly turn out different if we were to right-align on the moment of death, per Figure 3. This particular realignment doesn't seem so compelling for the demonstrated process, but I suppose it would be illuminating for health states and the sequence of events leading up to death.

Figure 3: Each sequence from Figure 1 is realigned on death.



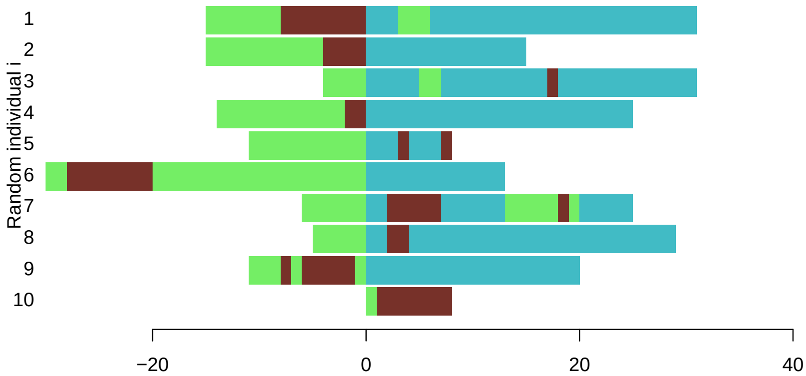
We need not restrict ourselves to alignment on birth or death, for indeed spells themselves, perhaps chosen by spell order (first or last) or the largest spell duration, could also be alignment criteria. In this case, alignment may be on entry, exit, midpoint, or some other spell reference, and it could be justified in any of the ways that we are accustomed to in word processors: left, right, centered, or justified (there's an odd one!).

Figure 4 shows a set of three alignment selections out of the many possible choices. Figure 4a left-aligns on entry to *first* retirement (if any). One could also choose last, longest, or some other episode of retirement, or of course right-align on exit. Figure 4b left-aligns on entry into each individual's longest spell of inactivity, whereas Figure 4c right-aligns on exit from the same spell.

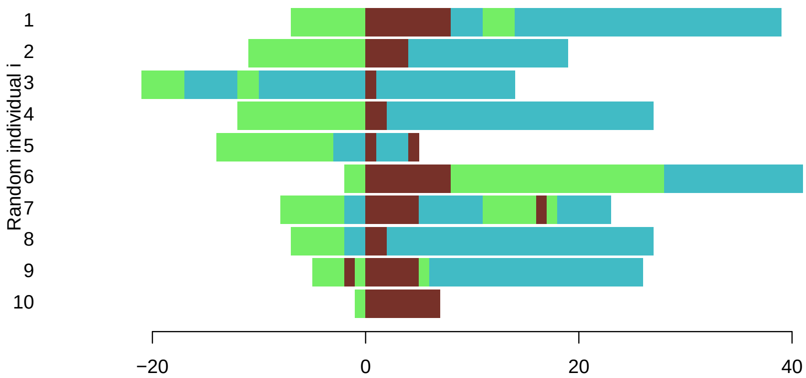
The purpose of realigning is to bring transitions into focus. We do not showcase *sorting* in this treatment, as this is another can of worms. Rather, here we would like to operate on aggregations of individual sequences, in which case between-individual sorting is unimportant. We'd still like to make statements about population-level characteristics.

Figure 4: Alignment on transient states.

(a) Left-aligned on *first* retirement.



(b) Left-aligned on entrance to *longest* spell of inactivity



(c) Right-aligned on exit from *longest* spell of inactivity

References

C. Dudel and M. Myrskylä. Estimating the expected number and length of episodes using markov chains with rewards. (under review), 2017a.

Christian Dudel and Mikko Myrskylä. Working life expectancy at age 50 in the united states and the impact of the great recession. *Demography*, Oct 2017b. ISSN 1533-7790. doi: 10.1007/s13524-017-0619-6. URL `https://doi.org/10.1007/s13524-017-0619-6`.

Sarah B Laditka and Douglas A Wolf. New methods for analyzing active life expectancy. *Journal of Aging and Health*, 10(2):214–241, 1998.

Giorgio Alfredo Spedicato. Discrete time markov chains with r. *The R Journal*, 07 2017. URL `https://journal.r-project.org/archive/2017/RJ-2017-036/index.html`. R package version 0.6.9.7.