

Using_marinesurvival_with_marinelifehistdata

S Ellis

15/02/2023

The aim of this vignette is to provide an example of how to use this marinesurvival package- here using the data from the marinelifehistdata package.

Getting started

Load the packages

```
library(marinelifehistdata)
library(marinesurvival)
```

The models can be run in either rstan or cmdstanr. Here we use cmdstanr. A version of one of these functions must be installed and functioning for the models to run.

For cmdstanr installation. See <https://mc-stan.org/cmdstanr/articles/cmdstanr.html> for installation instructions (this is not an ordinary package)

```
library(cmdstanr)
library(rstan)
```

This guide also requires functions from the tidyverse suite

```
library(tidyverse)
```

Preparing the data

Here we use the Northern right-whale dolphin as an example.

```
eg.datasets = get_lifhist_data(data.type = "age-structure", species = "NorthernRightWhaleDolphin", sex = "F")
#> # A tibble: 4 x 6
#>   dataset species          sex reference `data source` data
#>   <dbl> <chr>          <chr>    <dbl> <chr>      <chr>
#> 1     90 NorthernRightWhaleDolphin F         41 bycatch age-structure
#> 2     91 NorthernRightWhaleDolphin M         41 bycatch age-structure
#> 3     92 NorthernRightWhaleDolphin F         45 bycatch age-structure
#> 4     93 NorthernRightWhaleDolphin M         45 bycatch age-structure
```

The model requires a named list with the elements (unless otherwise stated descriptions refer to all datasets combined):

1. *Nages* . The number of ages in the data (e.g. 0-10, *Nages* = 11)
2. *Nsamples*. The numbers of samples
3. *age* . Vector of all ages (e.g. seq(0, maximum age, by = 1)).
4. *sample_ages*. Vector of ages of all whales in the sample (datasets combined). Adjusted so that age at maturity = 0.
5. *Nspecies* . Number of species-sex in the datasets (so female+male of same species = 2).
6. *Ndatasets*. Number of datasets in sample (length(datasets))
7. *Npopulations*. Number of populations from which the datasets are drawn (i.e. F and M samples can come from the same population)
8. *species_vector*. Vector of length *Ndatasets* indicating which species-sex each dataset belongs to
9. *population_vector*. Vector of length *Ndatasets* indicating which population each dataset is drawn from.
10. *species_sex_vector* Vector of length *Nspecies* indicating if each species-sex is female (1) or male (2)
11. *species_maxages*. Vector of length *Nspecies* indicating maximum achieved age of each species-sex present in the data
12. *prior_a_s1*. Prior for alpha mortality parameter. Beta distribution shape parameter 1.
13. *prior_a_s2*. Prior for alpha mortality parameter. Beta distribution shape parameter 2.
14. *prior_b_s1*. Prior for beta mortality parameter. Beta distribution shape parameter 1.
15. *prior_b_s2*. Prior for beta mortality parameter. Beta distribution shape parameter 2.
16. *include_age_est_error*. Vector of length *Ndatasets* indicating whether to apply (1) or not (0) the age estimation error in the model.
17. *age_error_sd* Vector of length *Nsamples* indicating the standard deviation of age estimation around the reported sample to apply when accounting for age error.
18. *include_samplebias_error* Vector of length *Ndatasets* indicating whether to apply (1) or not apply (0) sampling bias estimations to each dataset.
19. *BiasMat*. A *Ndataset* x *Nages* matrix, with each cell indicating whether sampling bias is expected (1) or not expected (0) at a given age. This will be ignored if *include_samplebias_error* for that dataset is 0 (but still that column still needs to be included).
20. *direction_samplebias* Vector of length *Ndataset* indicating the direction of sampling bias (-1 under-sampled, 1 oversampled, 0 unknown) applied to each dataset. For a given dataset this will be ignored if *include_samplebias_error* for that dataset is 0 (but still needs to be included).
21. *include_popchange_error* vector of length *Ndataset* indicating whether to include population change error estimation for that dataset (1 include, 0 do not include).
22. *direction_popchange*. Vector of length *Ndataset* indicating direction of population change expected (-1 shrinking, 1 growing, 0 unknown/either). This is ignored if *include_popchange_error* for a given dataset = 0 but still needs to be included.

Note. That to interact with stan all vectors need to be included as arrays: simply apply *as.array()* to all vectors during input.

The data above can be derived ‘manually’ from the data in the *marine.lifehist.data* package (or other data). However, the *marine.lifehist.data* package provides a function to do the leg work:

```
mod.list = create_marinesurvival_modinput(eg.datasets)
mod.list
#> $Nages
#> [1] 100
#>
#> $Nsamples
#> [1] 155
#>
#> $age
#> [1] 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
#> [26] 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
#> [51] 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
```

[illegible]

```

#>
#> $age_error_sd
#> [1] 0.45 0.45 0.45 0.45 0.45 0.45 0.50 0.50 0.50 0.55 0.55 0.55 0.60 0.60 0.65
#> [16] 0.65 0.65 0.70 0.70 0.70 0.70 0.70 0.70 0.70 0.75 0.75 0.80 0.80 0.80 0.90
#> [31] 1.10 1.15 0.45 0.45 0.45 0.45 0.45 0.50 0.50 0.50 0.55 0.55 0.60 0.60 0.60
#> [46] 0.65 0.65 0.65 0.85 0.45 0.45 0.45 0.45 0.45 0.45 0.50 0.50 0.50 0.50 0.50
#> [61] 0.50 0.50 0.55 0.55 0.55 0.55 0.55 0.55 0.55 0.60 0.60 0.60 0.60 0.65 0.65 0.65
#> [76] 0.65 0.65 0.65 0.70 0.70 0.70 0.70 0.70 0.70 0.70 0.75 0.75 0.75 0.75 0.80
#> [91] 0.80 0.80 0.80 0.80 0.80 0.80 0.85 0.85 0.85 0.85 0.90 0.90 0.90 0.90 0.95
#> [106] 0.95 1.00 1.00 1.00 1.00 1.05 1.05 1.10 1.20 1.25 1.35 1.35 1.40 1.40 1.50
#> [121] 1.50 1.50 0.45 0.45 0.45 0.45 0.45 0.50 0.50 0.50 0.55 0.55 0.55 0.55 0.55
#> [136] 0.60 0.60 0.60 0.65 0.65 0.65 0.65 0.65 0.65 0.65 0.70 0.70 0.70 0.75 0.75 0.75
#> [151] 0.85 0.85 1.00 1.35 1.35
#>
#> $include_samplebias_error
#> [1] 0 0 0 0
#>
#> $BiasMat
#>      [,1] [,2] [,3] [,4]
#> [1,]    0    0    0    0
#> [2,]    0    0    0    0
#> [3,]    0    0    0    0
#> [4,]    0    0    0    0
#> [5,]    0    0    0    0
#> [6,]    0    0    0    0
#> [7,]    0    0    0    0
#> [8,]    0    0    0    0
#> [9,]    0    0    0    0
#> [10,]   0    0    0    0
#> [11,]   0    0    0    0
#> [12,]   0    0    0    0
#> [13,]   0    0    0    0
#> [14,]   0    0    0    0
#> [15,]   0    0    0    0
#> [16,]   0    0    0    0
#> [17,]   0    0    0    0
#> [18,]   0    0    0    0
#> [19,]   0    0    0    0
#> [20,]   0    0    0    0
#> [21,]   0    0    0    0
#> [22,]   0    0    0    0
#> [23,]   0    0    0    0
#> [24,]   0    0    0    0
#> [25,]   0    0    0    0
#> [26,]   0    0    0    0
#> [27,]   0    0    0    0
#> [28,]   0    0    0    0
#> [29,]   0    0    0    0
#> [30,]   0    0    0    0
#> [31,]   0    0    0    0
#> [32,]   0    0    0    0
#> [33,]   0    0    0    0
#> [34,]   0    0    0    0

```

```

#> [35,] 0 0 0 0
#> [36,] 0 0 0 0
#> [37,] 0 0 0 0
#> [38,] 0 0 0 0
#> [39,] 0 0 0 0
#> [40,] 0 0 0 0
#> [41,] 0 0 0 0
#> [42,] 0 0 0 0
#> [43,] 0 0 0 0
#> [44,] 0 0 0 0
#> [45,] 0 0 0 0
#> [46,] 0 0 0 0
#> [47,] 0 0 0 0
#> [48,] 0 0 0 0
#> [49,] 0 0 0 0
#> [50,] 0 0 0 0
#> [51,] 0 0 0 0
#> [52,] 0 0 0 0
#> [53,] 0 0 0 0
#> [54,] 0 0 0 0
#> [55,] 0 0 0 0
#> [56,] 0 0 0 0
#> [57,] 0 0 0 0
#> [58,] 0 0 0 0
#> [59,] 0 0 0 0
#> [60,] 0 0 0 0
#> [61,] 0 0 0 0
#> [62,] 0 0 0 0
#> [63,] 0 0 0 0
#> [64,] 0 0 0 0
#> [65,] 0 0 0 0
#> [66,] 0 0 0 0
#> [67,] 0 0 0 0
#> [68,] 0 0 0 0
#> [69,] 0 0 0 0
#> [70,] 0 0 0 0
#> [71,] 0 0 0 0
#> [72,] 0 0 0 0
#> [73,] 0 0 0 0
#> [74,] 0 0 0 0
#> [75,] 0 0 0 0
#> [76,] 0 0 0 0
#> [77,] 0 0 0 0
#> [78,] 0 0 0 0
#> [79,] 0 0 0 0
#> [80,] 0 0 0 0
#> [81,] 0 0 0 0
#> [82,] 0 0 0 0
#> [83,] 0 0 0 0
#> [84,] 0 0 0 0
#> [85,] 0 0 0 0
#> [86,] 0 0 0 0
#> [87,] 0 0 0 0

```

```

#> [88,] 0 0 0 0
#> [89,] 0 0 0 0
#> [90,] 0 0 0 0
#> [91,] 0 0 0 0
#> [92,] 0 0 0 0
#> [93,] 0 0 0 0
#> [94,] 0 0 0 0
#> [95,] 0 0 0 0
#> [96,] 0 0 0 0
#> [97,] 0 0 0 0
#> [98,] 0 0 0 0
#> [99,] 0 0 0 0
#> [100,] 0 0 0 0
#>
#> $direction_samplebias
#> [1] 0 0 -1 -1
#>
#> $include_popchange_error
#> [1] 1 1 1 1
#>
#> $direction_popchange
#> [1] -1 -1

```

Running the model

For simplicity, the stan code to run these models are stored as character vectors. There are several versions of the model for different variations of number of species and number of sexes. First job is to select the correct model code:

```

stancode = get_marinesurvival_stancode(mod.list)
#> [1] "One Species, Two Sexes"

```

The code and data can then be combined in either rstan or cmdstanr- here we use rstan:

```

modobj = stan_model(model_code = stancode)
mod = sampling(
  modobj,
  data = mod.list,
  chains = 4,
  cores = 4,
  iter = 2000,
  init = generate_inits(4, mod.list)
)

```

Ta da, model fit as an ordinary bayesian model. The summary data can be accessed but tends to be rather long because of all the estimates of unknown ages. The marinesurvival package has a function called `get_activeparameters` to identify the non-age parameters present in the model which can then be used to simplify the output.

```

#summary(mod)$summary
out = summary(mod, digits = 4, depth = 2)$summary
activepars = get_activepars(out = out, mod = mod, input.list = mod.list)

```

```

out[rownames(out) %in% activepars,]
#>           mean      se_mean      sd      2.5%      25%
#> b[1]  0.11487936 0.0002923607 0.02049866  0.07710312  0.10082619
#> b[2]  0.14401777 0.0003207941 0.02462987  0.09768453  0.12713185
#> a[1]  0.03988081 0.0001498718 0.01072915  0.02131439  0.03236906
#> a[2]  0.05813857 0.0001902233 0.01523161  0.03095696  0.04747159
#> r[1] -0.23704226 0.0017846737 0.12855788 -0.46390451 -0.32927559
#> r[2] -0.23733542 0.0016236065 0.12808770 -0.46320227 -0.33072843
#> lp__ -620.57934511 0.2435961401 9.18266812 -639.86904534 -626.50905716
#>           50%           75%           97.5%      n_eff      Rhat
#> b[1]  0.11416840  0.12811723  0.15717646 4916.010 0.9997966
#> b[2]  0.14361146  0.16046025  0.19394929 5894.831 0.9993080
#> a[1]  0.03924570  0.04654374  0.06317133 5124.964 0.9998636
#> a[2]  0.05685996  0.06769441  0.09120000 6411.565 1.0000201
#> r[1] -0.24020708 -0.15030862  0.02882835 5188.953 0.9997346
#> r[2] -0.24165170 -0.15110075  0.02494446 6223.766 0.9992538
#> lp__ -620.28632554 -614.22723671 -603.42518679 1421.009 1.0006391

```

And that is that.

Exploring the model output

The model is just a fitted Bayesian model so the posterior can be explored in any direction the user chooses. However, the `marinesurvival` package includes a couple of functions that automate some common functions.

1. Compare model to sample (there is a bit of fiddling first to get the inputs)

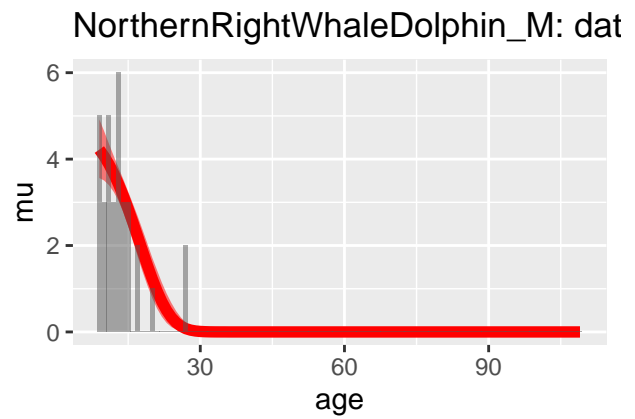
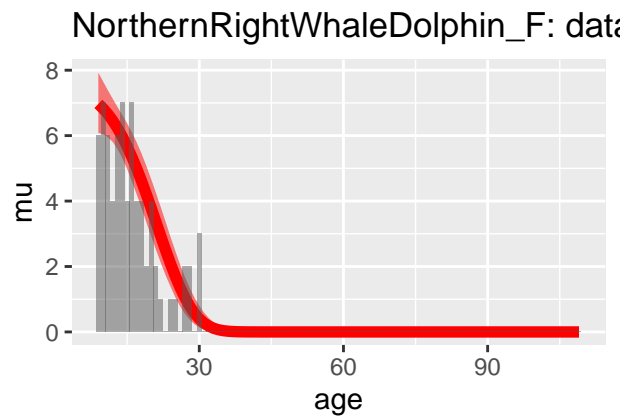
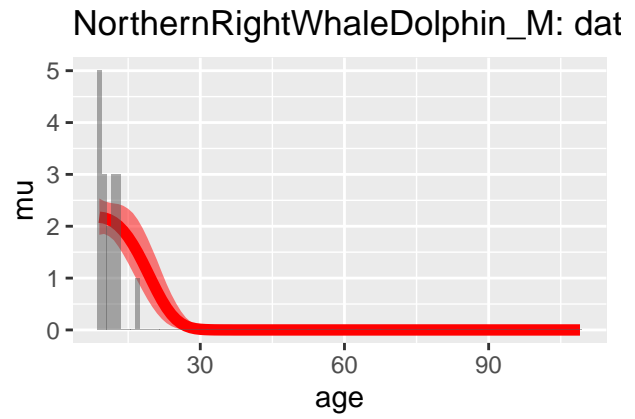
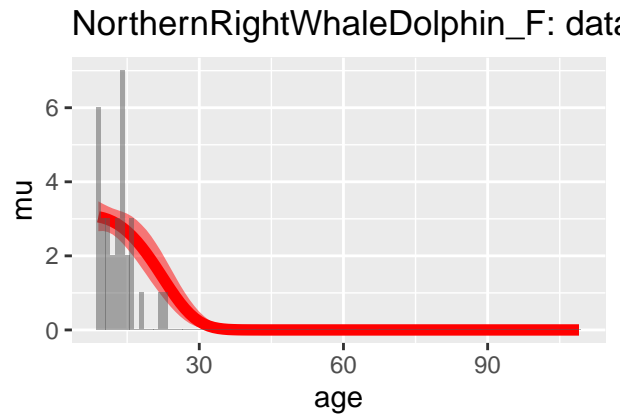
```

age.seq = seq(0, 100, 1)
post = rstan::extract(mod)
ages.at.mat =
  marine.lifhist.speciesdata$species_age.maturity %>%
  filter(species == "NorthernRightWhaleDolphin") %>%
  arrange(sex)
ages.at.mat = ages.at.mat$age.mat

datasets.key = get_lifhist_data(data.type = "age-structure", species = "NorthernRightWhaleDolphin", se
#> # A tibble: 4 x 6
#>   dataset species      sex reference `data source` data
#>   <dbl> <chr>      <chr>      <dbl> <chr>      <chr>
#> 1     90 NorthernRightWhaleDolphin F          41 bycatch age-structure
#> 2     91 NorthernRightWhaleDolphin M          41 bycatch age-structure
#> 3     92 NorthernRightWhaleDolphin F          45 bycatch age-structure
#> 4     93 NorthernRightWhaleDolphin M          45 bycatch age-structure
datasets.key$species.num = mod.list$species_vector
datasets.key$dataset.num = seq(1,4, 1)

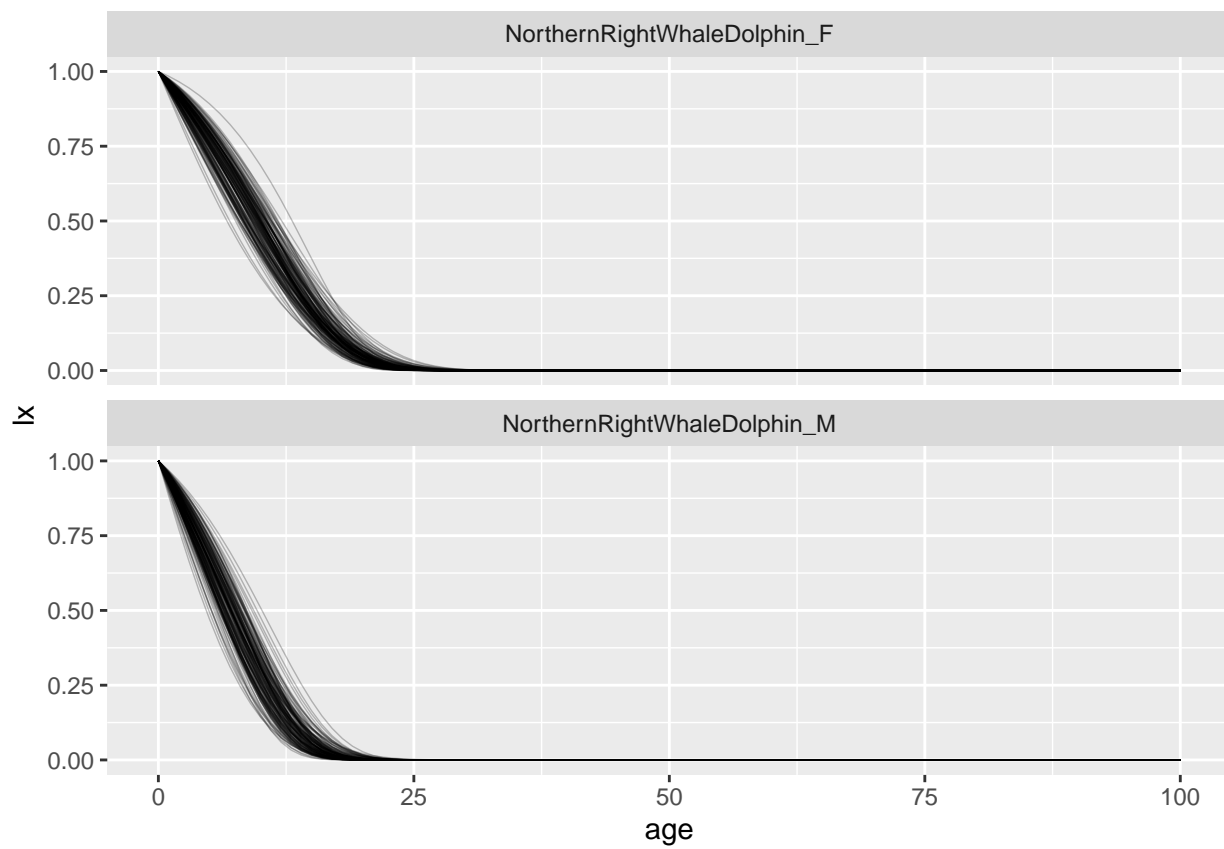
mod.to.sample.plot = plot_modtosample(age.seq = age.seq, post = post, input.list = mod.list, minages = a
#> [1] 1
#> [1] 2
#> [1] 3
#> [1] 4
mod.to.sample.plot

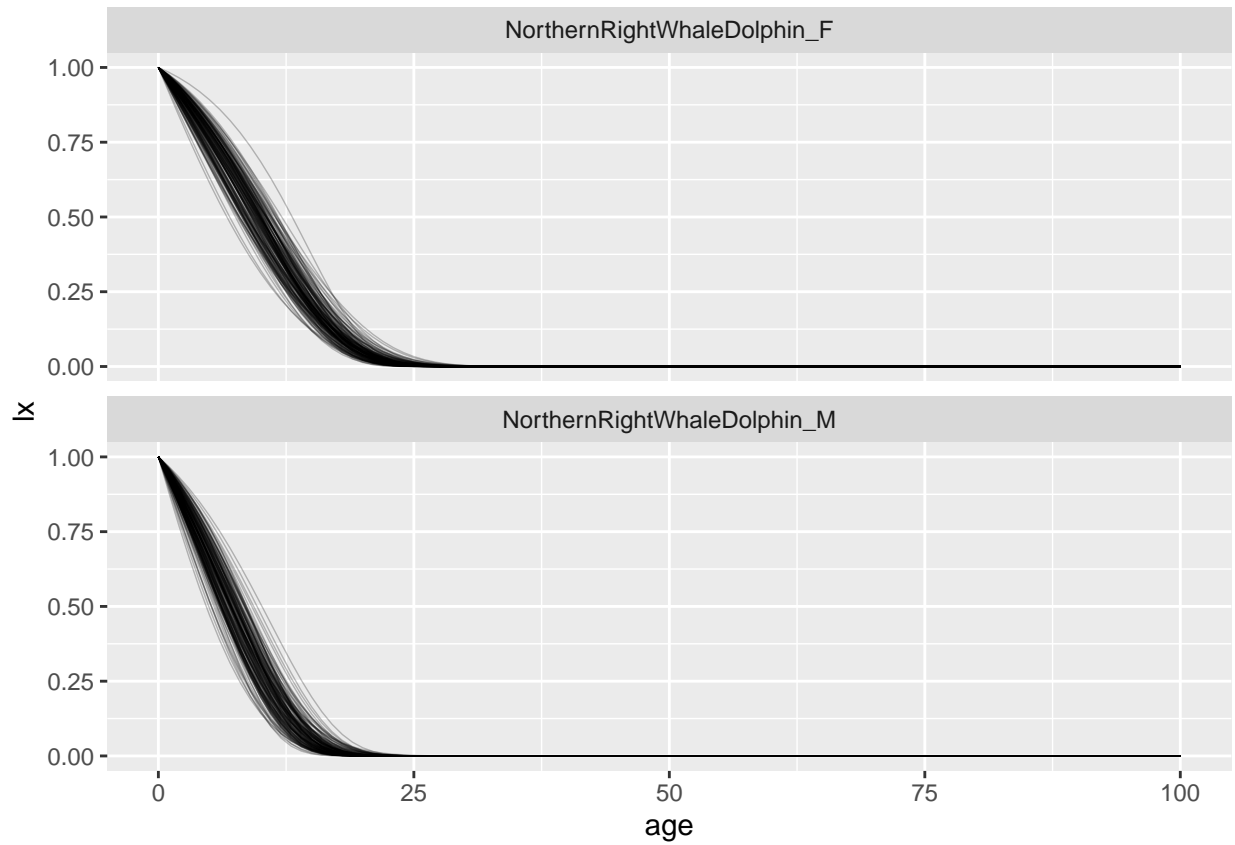
```



2. Plot survival curve

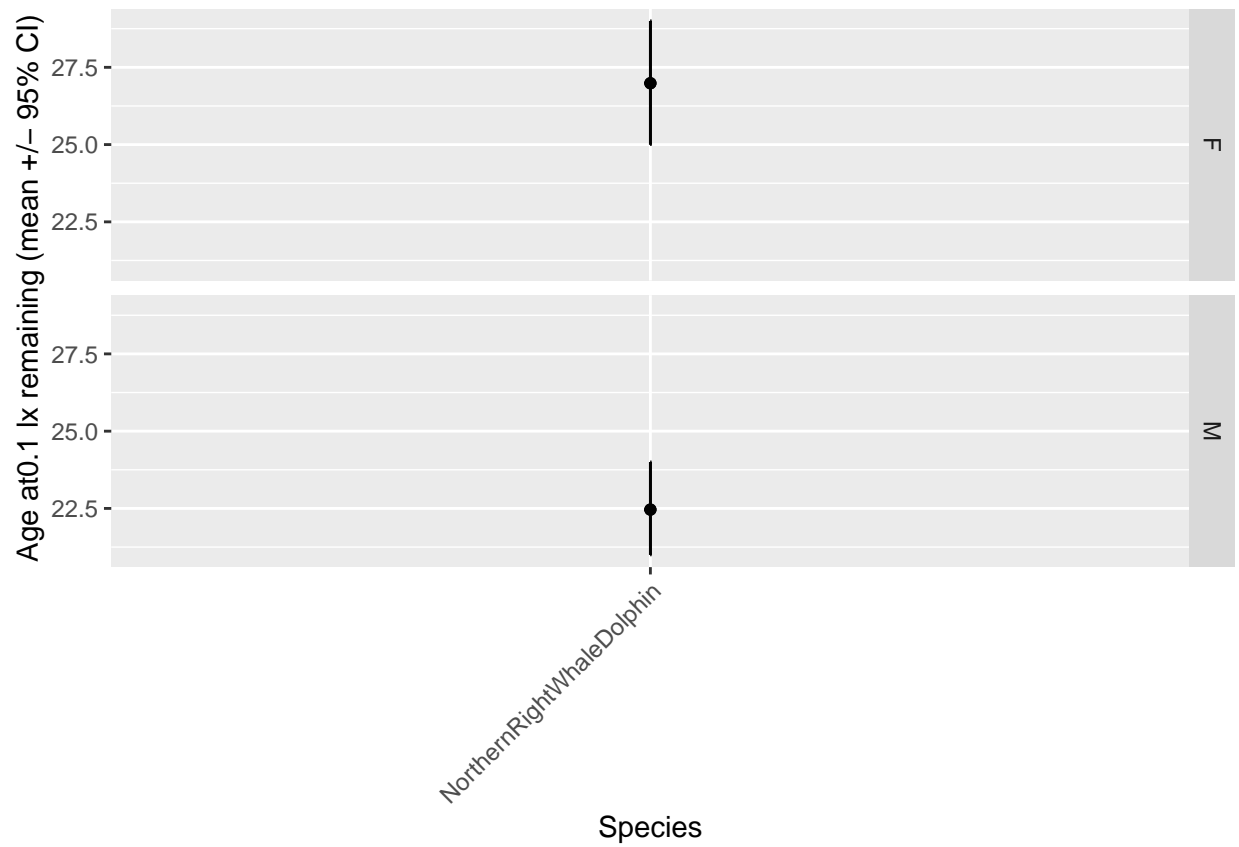
```
plot_posterior_survival(post = post, age.seq = age.seq, input.list = mod.list, names.key = datasets.key,
```

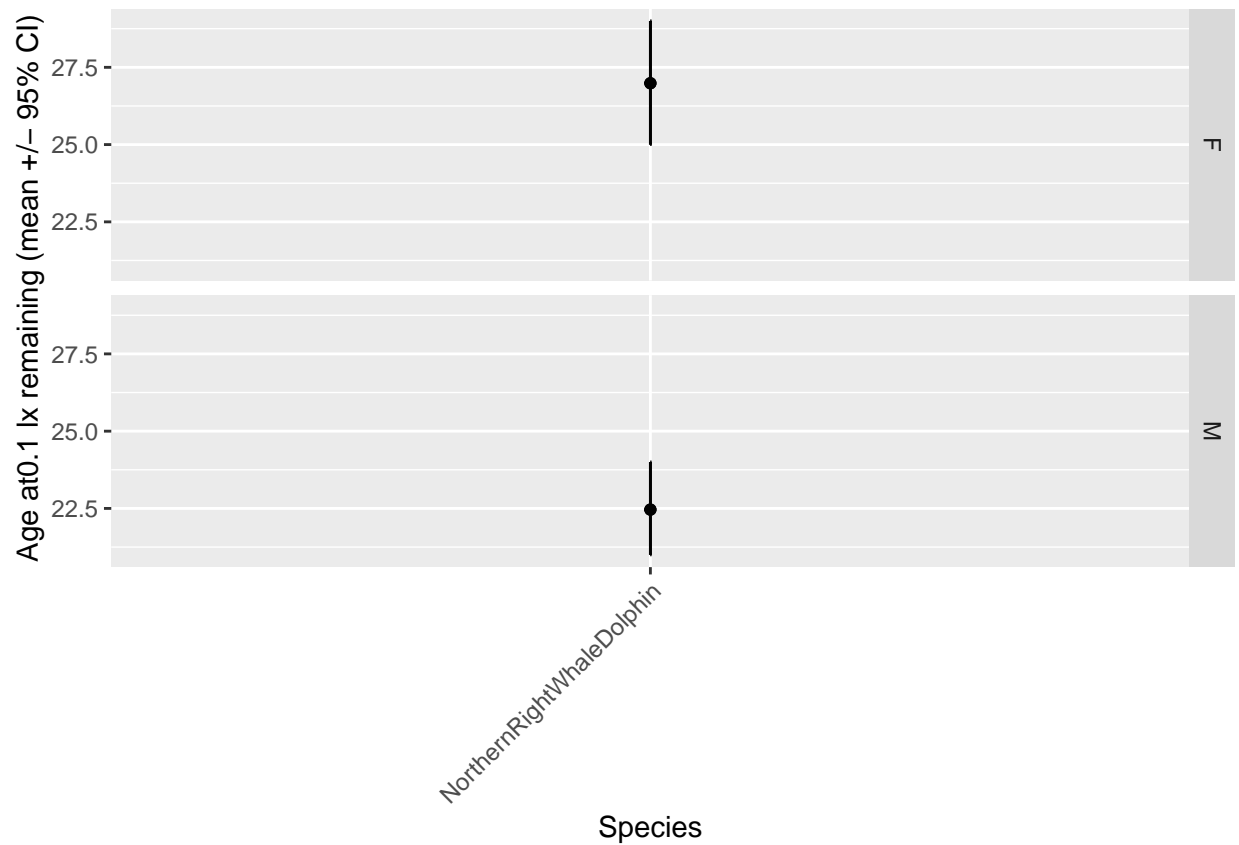





3. Get ordinary maximum lifespans (the age at which 90% of life years have been lived)

```
plot_ageX(X = 0.1, post = post, age.seq = age.seq, input.list = mod.list, minages = ages.at.mat, names.l
```





```
ageX.df = plot_ageX(X = 0.1, post = post, age.seq = age.seq, input.list = mod.list, minages = ages.at.m)
ageX.df
#>           species    mean lCI uCI      sd sex species.num
#> 1 NorthernRightWhaleDolphin 26.9850 25 29 1.131413  F          1
#> 2 NorthernRightWhaleDolphin 26.9850 25 29 1.131413  F          1
#> 3 NorthernRightWhaleDolphin 22.4615 21 24 1.200366  M          2
#> 4 NorthernRightWhaleDolphin 22.4615 21 24 1.200366  M          2
```

Here endeth the lesson.