vital: Tidy data analysis for demography

Rob J Hyndman

9 July 2024

Demographic data structures in R packages

Package	Data class
demography	demogdata
StMoMo	StMoMoData (created by converting a demogdata object)
StanMoMo	Lists of matrices
lifecontingencies	data.frame
BayesMortalityPlus	tibble (that needs to be converted to a matrix for fitting)
MortalityLaws	individual vectors
HMDHFDplus	data.frame

Australian Deaths 1901–2020

```
# A tibble: 145,440 x 7
    Year   Age Sex    State Mortality Exposure Deaths
   <int> <int> <chr>  <chr>     <dbl>    <dbl>  <dbl>
 1  1901     0 female WA       0.129       2511    325
 2  1901     0 male   WA       0.158       2634    416
 3  1901     1 female WA       0.0275      2219     61
 4  1901     1 male   WA       0.0391      2175     85
 5  1901     2 female WA       0.00688     2180     15
 6  1901     2 male   WA       0.0131      2208     29
 7  1901     3 female WA       0.00584     1884     11
 8  1901     3 male   WA       0.00503     1988     10
 9  1901     4 female WA       0.00290     1722      5
10  1901     4 male   WA       0.00287     1743      5
# i 145,430 more rows
```

tsibble objects

Australian Deaths 1901–2020

```
# A tsibble: 145,440 x 7 [1Y]
# Key:       Age, Sex, State [1,212]
     Year   Age Sex    State Mortality Exposure Deaths
    <int> <int> <chr>  <chr>     <dbl>    <dbl>  <dbl>
  1  1901     0 female WA       0.129       2511    325
  2  1901     0 male   WA       0.158       2634    416
  3  1901     1 female WA       0.0275      2219     61
  4  1901     1 male   WA       0.0391      2175     85
  5  1901     2 female WA       0.00688     2180     15
  6  1901     2 male   WA       0.0131      2208     29
  7  1901     3 female WA       0.00584     1884     11
  8  1901     3 male   WA       0.00503     1988     10
  9  1901     4 female WA       0.00290     1722      5
 10  1901     4 male   WA       0.00287     1743      5
# i 145,430 more rows
```
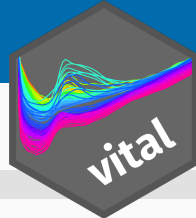
Variables

Index:
- Year

Keys:
- Age
- Sex
- State

Every row must have a unique combination of Index and Keys

vital objects

Australian Deaths 1901–2020
aus

```
# A vital: 145,440 x 7 [1Y]
# Key:       Age x (Sex, State) [101 x 12]
    Year   Age Sex    State Mortality Exposure Deaths
   <int> <int> <chr>  <chr>     <dbl>    <dbl>  <dbl>
 1  1901     0 female WA       0.129      2511    325
 2  1901     0 male   WA       0.158      2634    416
 3  1901     1 female WA       0.0275     2219     61
 4  1901     1 male   WA       0.0391     2175     85
 5  1901     2 female WA       0.00688    2180     15
 6  1901     2 male   WA       0.0131     2208     29
 7  1901     3 female WA       0.00584    1884     11
 8  1901     3 male   WA       0.00503    1988     10
 9  1901     4 female WA       0.00290    1722      5
10  1901     4 male   WA       0.00287    1743      5
# i 145,430 more rows
```

Variables
Index:
- Year

Keys:
- Age
- Sex
- State

Every row must have a unique combination of Index and Keys

Variables denoting age, sex, deaths, births and population can also be specified as attributes.

vital objects

```
index_var(aus)
```
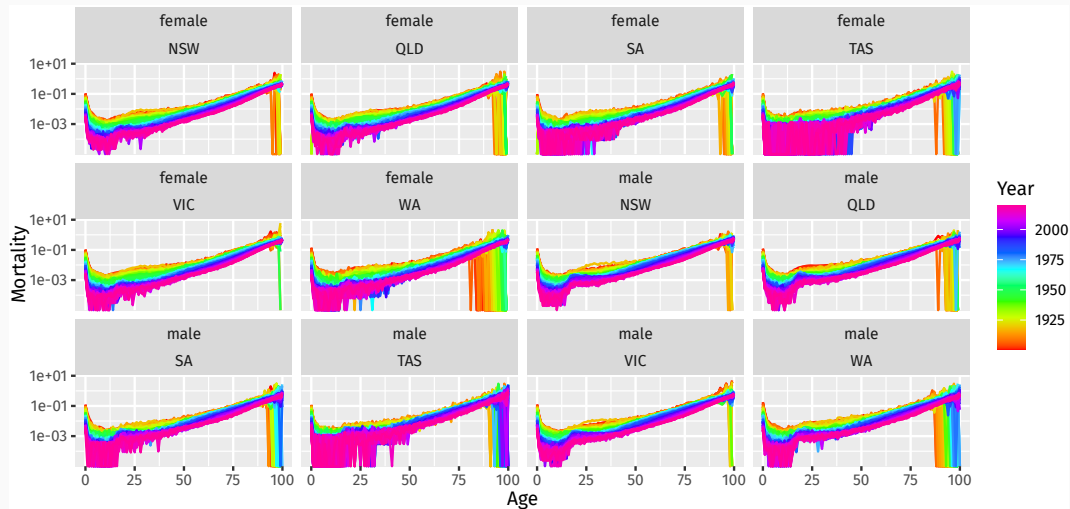
```
[1] "Year"
```

```
key_vars(aus)
```

```
[1] "Age"    "Sex"    "State"
```

```
vital_vars(aus)
```

```
      age       sex    deaths population
    "Age"     "Sex"  "Deaths" "Exposure"
```
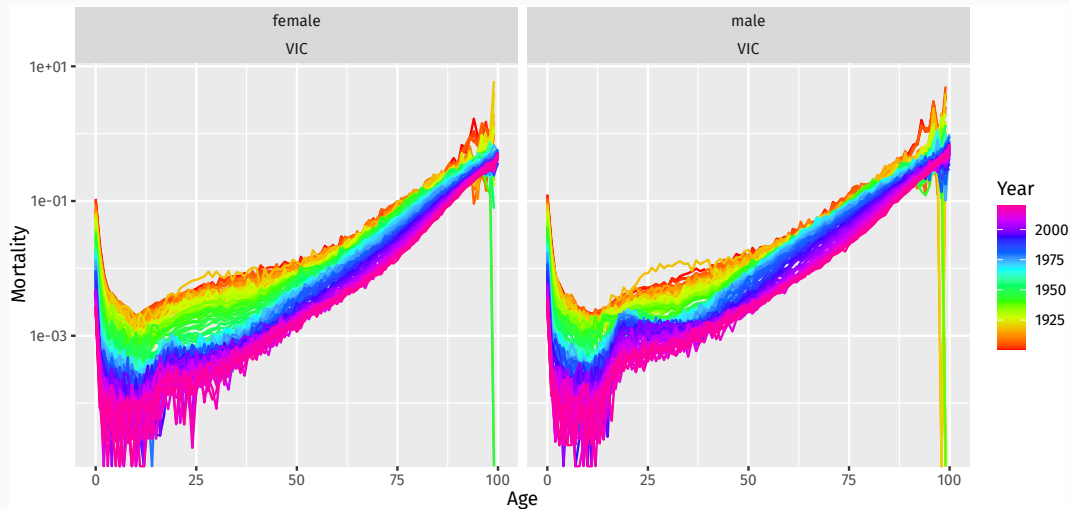
Rainbow plots

```
aus ▷ autoplot(Mortality) + scale_y_log10()
```
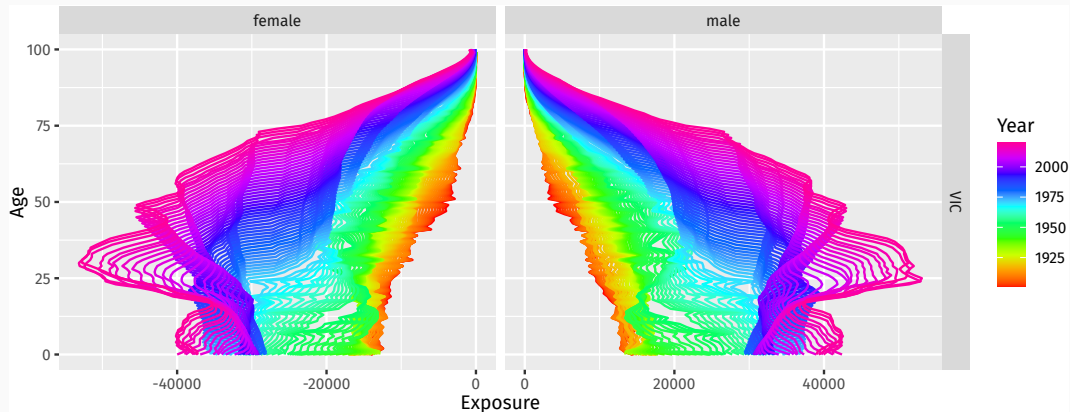
Rainbow plots

```
aus ▷ filter(State == "VIC") ▷ autoplot(Mortality) + scale_y_log10()
```

Rainbow plots

```
aus |> filter(State == "VIC") |>
  mutate(Exposure = if_else(Sex == "female", -Exposure, Exposure)) |>
  autoplot(Exposure) +
  facet_grid(State ~ Sex, scales = "free_x") + coord_flip()
```

Smoothing

```
sm_aus ← aus ▷ smooth_mortality(Mortality)
sm_aus

# A vital: 145,440 x 9 [1Y]
# Key:     Age x (Sex, State) [101 x 12]
    Year   Age Sex    State Mortality Exposure Deaths   .smooth .smooth_se
    <int> <dbl> <chr>  <chr>     <dbl>    <dbl>  <dbl> <dbl[1d]>  <dbl[1d]>
 1  1901     0 female NSW      0.107      17143   1833  0.107      0.00295
 2  1901     1 female NSW      0.0247     15071    373  0.0237     0.00141
 3  1901     2 female NSW      0.00686    15461    106  0.00804    0.000670
 4  1901     3 female NSW      0.00441    15629     69  0.00461    0.000405
 5  1901     4 female NSW      0.00374    15762     59  0.00341    0.000305
 6  1901     5 female NSW      0.00274    16030     44  0.00275    0.000251
 7  1901     6 female NSW      0.00252    16289     41  0.00230    0.000215
 8  1901     7 female NSW      0.00216    16639     36  0.00197    0.000189
 9  1901     8 female NSW      0.00169    16554     28  0.00175    0.000173
10  1901     9 female NSW      0.00109    16468     18  0.00162    0.000163
# i 145,430 more rows
```
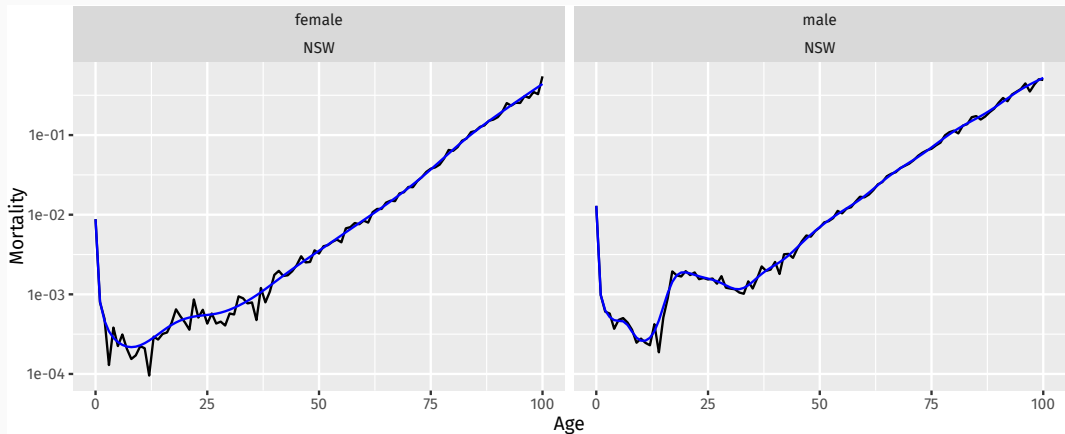
Smoothing

```
sm_aus ← aus ▷ smooth_mortality(Mortality)
sm_aus ▷ filter(State == "NSW", Year == 1980) ▷ autoplot(Mortality) +
  geom_line(aes(y = .smooth), col = "blue") + scale_y_log10()
```

Life tables

```
life_table(aus)
```

```
# A vital: 145,440 x 14 [1Y]
# Key:      Age x (Sex, State) [101 x 12]
    Year   Age Sex   State       mx      qx    lx       dx    Lx    Tx    ex    rx
   <int> <int> <chr> <chr>    <dbl>   <dbl> <dbl>    <dbl> <dbl> <dbl> <dbl> <dbl>
 1  1901     0 fema~ NSW     0.107   0.100  1     1.00e-1 0.935  56.2  56.2 0.935
 2  1901     1 fema~ NSW     0.0247  0.0244 0.900 2.20e-2 0.889  55.3  61.5 0.951
 3  1901     2 fema~ NSW     0.00686 0.00683 0.878 6.00e-3 0.875 54.4  62.0 0.984
 4  1901     3 fema~ NSW     0.00441 0.00441 0.872 3.84e-3 0.870 53.5  61.4 0.994
 5  1901     4 fema~ NSW     0.00374 0.00374 0.868 3.24e-3 0.867 52.7  60.7 0.996
 6  1901     5 fema~ NSW     0.00274 0.00274 0.865 2.37e-3 0.864 51.8  59.9 0.997
 7  1901     6 fema~ NSW     0.00252 0.00251 0.863 2.17e-3 0.861 50.9  59.1 0.997
 8  1901     7 fema~ NSW     0.00216 0.00216 0.860 1.86e-3 0.859 50.1  58.2 0.998
 9  1901     8 fema~ NSW     0.00169 0.00169 0.859 1.45e-3 0.858 49.2  57.3 0.998
10  1901     9 fema~ NSW     0.00109 0.00109 0.857 9.36e-4 0.857 48.4  56.4 0.999
# i 145,430 more rows
# i 2 more variables: nx <dbl>, ax <dbl>
```

Life expectancy

```
life_expectancy(aus)
```

```
# A vital: 1,440 x 8 [1Y]
# Key:      Age x (Sex, State) [1 x 12]
    Year   Age Sex    State    ex    rx    nx    ax
   <int> <int> <chr>  <chr> <dbl> <dbl> <dbl> <dbl>
 1  1901     0 female NSW    56.2 0.935     1 0.352
 2  1901     0 female QLD    56.8 0.937     1 0.338
 3  1901     0 female SA     58.1 0.939     1 0.324
 4  1901     0 female TAS    58.9 0.946     1 0.275
 5  1901     0 female VIC    55.8 0.937     1 0.334
 6  1901     0 female WA     53.1 0.922     1 0.35
 7  1901     0 male   NSW    52.6 0.925     1 0.33
 8  1901     0 male   QLD    50.6 0.924     1 0.33
 9  1901     0 male   SA     53.5 0.922     1 0.33
10  1901     0 male   TAS    57.3 0.930     1 0.33
# i 1,430 more rows
```
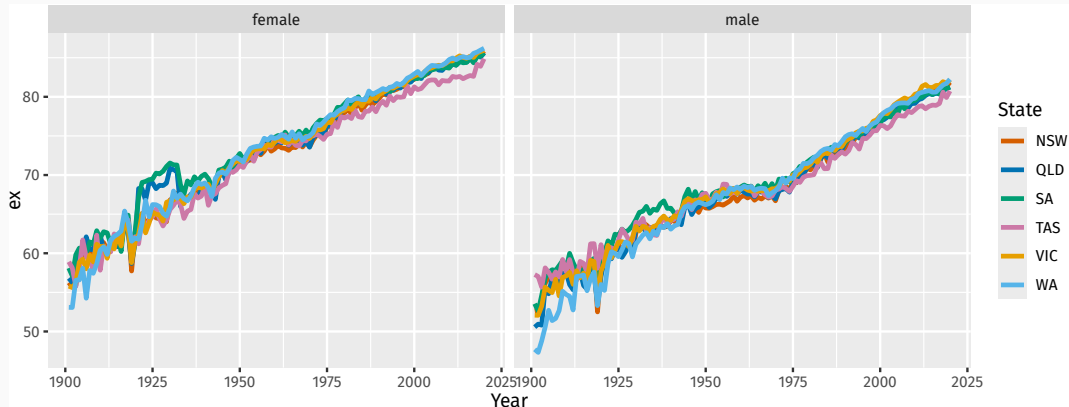
Life expectancy

```
life_expectancy(aus)  ▷
  ggplot(aes(x = Year, y = ex, colour = State)) +
  geom_line(linewidth = 1) +
  facet_grid(. ~ Sex)
```

Mortality models

$m_{x,t}$ = mortality rate at age x in year t.

Naive: $$m_{x,t} = m_{x,t-1} + \varepsilon_{x,t}$$

Lee-Carter: $$\log(m_{x,t}) = a_x + k_t b_x + \varepsilon_{x,t}$$

$\varepsilon_{x,t}$ = noise term with variance σ_x^2.

Mortality models

$m_{x,t}$ = mortality rate at age x in year t.

Naive: $\qquad\qquad m_{x,t} = m_{x,t-1} + \varepsilon_{x,t}$

Lee-Carter: $\log(m_{x,t}) = a_x + k_t b_x + \varepsilon_{x,t}$

$\varepsilon_{x,t}$ = noise term with variance σ_x^2.

```
fit ← aus ▷
  model(
    naive = FNAIVE(Mortality),
    lc = LC(log(Mortality))
  )
```

Mortality models

$m_{x,t}$ = mortality rate at age x in year t.

Naive: $\qquad m_{x,t} = m_{x,t-1} + \varepsilon_{x,t}$

Lee-Carter: $\log(m_{x,t}) = a_x + k_t b_x + \varepsilon_{x,t}$

$\varepsilon_{x,t}$ = noise term with variance σ_x^2.

```
fit ← aus ▷
  model(
    naive = FNAIVE(Mortality),
    lc = LC(log(Mortality))
  )
```

```
fit
```

```
# A mable: 12 x 4
# Key:     Sex, State [12]
   Sex    State    naive       lc
   <chr>  <chr>  <model>  <model>
 1 female NSW    <FNAIVE>    <LC>
 2 female QLD    <FNAIVE>    <LC>
 3 female SA     <FNAIVE>    <LC>
 4 female TAS    <FNAIVE>    <LC>
 5 female VIC    <FNAIVE>    <LC>
 6 female WA     <FNAIVE>    <LC>
 7 male   NSW    <FNAIVE>    <LC>
 8 male   QLD    <FNAIVE>    <LC>
 9 male   SA     <FNAIVE>    <LC>
10 male   TAS    <FNAIVE>    <LC>
11 male   VIC    <FNAIVE>    <LC>
12 male   WA     <FNAIVE>    <LC>
```

Lee-Carter models

$$\log(m_{x,t}) = a_x + k_t b_x + \varepsilon_{x,t}$$

```
fit ▷
  filter(Sex == "female",
         State == "NSW") ▷
  select(lc) ▷
  report()
```

Series: Mortality
Model: LC
Transformation: log(Mortality)

Options:
  Adjust method: dt
  Jump choice: fit

Age functions
A tibble: 101 × 3
    Age   ax   bx
  <int> <dbl> <dbl>
1   0 -4.07 0.0155
2   1 -6.20 0.0221
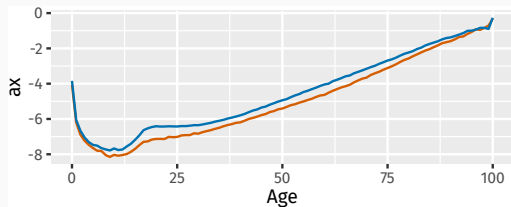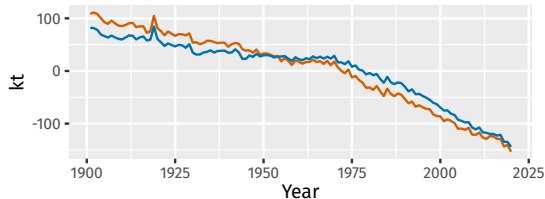3   2 -6.89 0.0199
i 98 more rows

Time coefficients
A tsibble: 120 x 2 [1Y]
   Year   kt
  <int> <dbl>
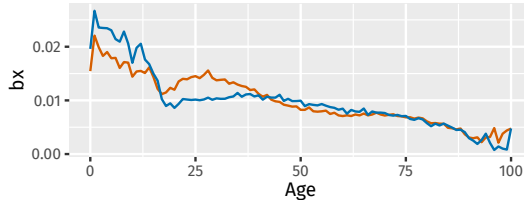1 1901 109.
2 1902 111.
3 1903 108.
i 117 more rows

Lee-Carter models

$$\log(m_{x,t}) = a_x + k_t b_x + \varepsilon_{x,t}$$

```
fit ▷
  filter(State == "NSW") ▷
  select(lc) ▷
  autoplot()
```

Lee-Carter models

$$\log(m_{x,t}) = a_x + k_t b_x + \varepsilon_{x,t}$$

```
fit ▷ select(lc) ▷ age_components()
```

```
# A tibble: 1,212 x 5
    Sex    State Age    ax      bx
   <chr>  <chr> <int> <dbl>   <dbl>
 1 female NSW      0  -4.07  0.0155
 2 female NSW      1  -6.20  0.0221
 3 female NSW      2  -6.89  0.0199
 4 female NSW      3  -7.24  0.0183
 5 female NSW      4  -7.47  0.0190
 6 female NSW      5  -7.65  0.0178
 7 female NSW      6  -7.80  0.0179
 8 female NSW      7  -7.81  0.0160
 9 female NSW      8  -8.05  0.0171
10 female NSW      9  -8.15  0.0170
# i 1,202 more rows
```

```
fit ▷ select(lc) ▷ time_components()
```

```
# A tsibble: 1,440 x 4 [1Y]
# Key:        Sex, State [12]
    Sex    State Year    kt
   <chr>  <chr> <int> <dbl>
 1 female NSW    1901 109.
 2 female NSW    1902 111.
 3 female NSW    1903 108.
 4 female NSW    1904 100.
 5 female NSW    1905  92.7
 6 female NSW    1906  89.5
 7 female NSW    1907  95.7
 8 female NSW    1908  90.5
 9 female NSW    1909  85.9
10 female NSW    1910  85.4
# i 1,430 more rows
```

Forecasts

```
fc ← fit ▷ forecast(h = 20)
fc
```

```
# A vital fable: 48,480 x 7 [1Y]
# Key:          Age x (Sex, State, .model) [101 x 24]
   Sex    State .model  Year  Age          Mortality    .mean
   <chr>  <chr> <chr>   <dbl> <int>            <dist>    <dbl>
 1 female NSW   naive   2021     0 N(0.0027, 1.8e-05) 0.00270
 2 female NSW   naive   2022     0 N(0.0027, 3.6e-05) 0.00270
 3 female NSW   naive   2023     0 N(0.0027, 5.4e-05) 0.00270
 4 female NSW   naive   2024     0 N(0.0027, 7.2e-05) 0.00270
 5 female NSW   naive   2025     0   N(0.0027, 9e-05) 0.00270
 6 female NSW   naive   2026     0 N(0.0027, 0.00011) 0.00270
 7 female NSW   naive   2027     0 N(0.0027, 0.00013) 0.00270
 8 female NSW   naive   2028     0 N(0.0027, 0.00014) 0.00270
 9 female NSW   naive   2029     0 N(0.0027, 0.00016) 0.00270
10 female NSW   naive   2030     0 N(0.0027, 0.00018) 0.00270
# i 48,470 more rows
```

19

NSW forecasts using Lee-Carter method

```
fc ▷ filter(State == "NSW", .model == "lc") ▷
  autoplot() + scale_y_log10()
```

Functional data models
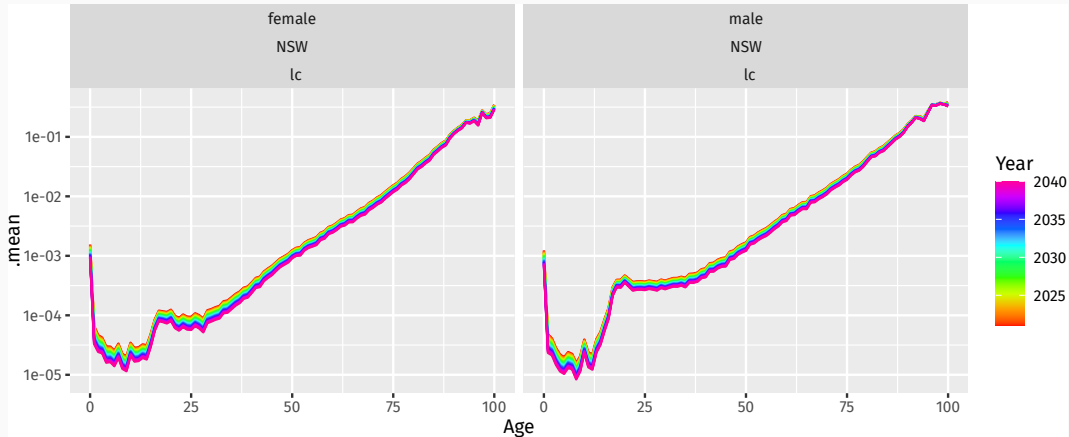
Let $m_{x,t}$ be the mortality rate at age x in year t.

$$\log(m_{t,x}) = s_t(x) + \sigma_t(x)\varepsilon_{t,x}$$

$$s_t(x) = \mu(x) + \sum_{j=1}^{J} \beta_{tj}\phi_j(x) + e_t(x)$$

- $s_t(x)$ = smoothed version of $y_t(x)$
- $\mu(x)$ = mean $s_t(x)$ across years.
- $\phi_j(x)$ and β_{tj} estimated using principal component analysis.
- $\beta_{1j}, \ldots, \beta_{Tj}$ modelled with ARIMA or ARFIMA processes.

Functional data models

```
fit ← aus ▷
  smooth_mortality(Mortality) ▷
  model(hu = FDM(log(.smooth)))
fit
```

```
# A mable: 12 x 3
# Key:      Sex, State [12]
   Sex     State       hu
   <chr>   <chr> <model>
 1 female  NSW     <FDM>
 2 female  QLD     <FDM>
 3 female  SA      <FDM>
 4 female  TAS     <FDM>
 5 female  VIC     <FDM>
 6 female  WA      <FDM>
 7 male    NSW     <FDM>
 8 male    QLD     <FDM>
 9 male    SA      <FDM>
10 male    TAS     <FDM>
```

Functional data models

$$s_t(x) = \mu(x) + \sum_{j=1}^{J} \beta_{tj} \phi_j(x) + e_t(x)$$

```
fit ▷
  filter(Sex == "female", State == "NSW") ▷
  report()
```

```
Series: .smooth
Model: FDM
Transformation: log(.smooth)

Basis functions
# A tibble: 101 x 8
    Age  mean  phi1     phi2     phi3     phi4      phi5     phi6
  <dbl> <dbl> <dbl>    <dbl>    <dbl>    <dbl>     <dbl>    <dbl>
1     0 -4.07 0.147   0.0625  -0.0270   0.0986   0.0112   -0.0624
2     1 -6.16 0.200  -0.0609  -0.194    0.116    0.0383   -0.238
3     2 -6.82 0.182  -0.0483  -0.157    0.0924   0.0443   -0.264
4     3 -7.17 0.170  -0.0368  -0.130    0.0362   0.000338 -0.321
5     4 -7.40 0.164  -0.0165  -0.114   -0.0154  -0.0303   -0.374
# i 96 more rows
```

Functional data models

$$s_t(x) = \mu(x) + \sum_{j=1}^{J} \beta_{tj} \phi_j(x) + e_t(x)$$

```
Coefficients
# A tsibble: 120 x 8 [1Y]
    Year  mean beta1  beta2   beta3    beta4   beta5   beta6
   <int> <dbl> <dbl>  <dbl>   <dbl>    <dbl>   <dbl>   <dbl>
1  1901     1  11.1 -0.522 -0.0553   0.207   0.358   0.0305
2  1902     1  11.8 -0.649  0.399    0.856   0.0319  0.422
3  1903     1  11.5 -0.930 -0.485    0.398   0.399  -0.376
4  1904     1  11.1 -0.827 -0.214   -0.000305 0.00125 -0.0783
5  1905     1  10.2 -0.563 -0.105    0.324   0.122   0.0478
# i 115 more rows
# i Use 'print(n = ...)' to see more rows

Time series models
   beta1 : ARIMA(0,1,1) w/ drift
   beta2 : ARIMA(0,2,2)
   beta3 : ARIMA(1,0,1)
   beta4 : ARIMA(0,0,2)
   beta5 : ARIMA(0,0,0)
   beta6 : ARIMA(2,0,2)
```
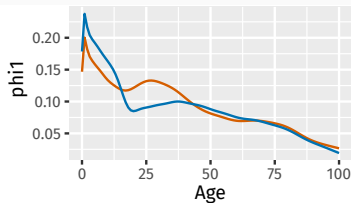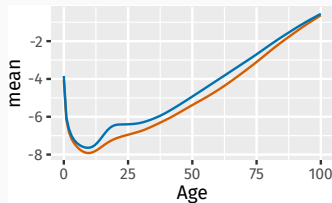
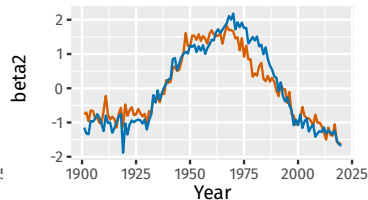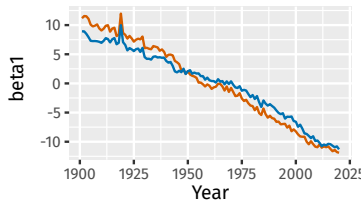# Functional data models

$$s_t(x) = \mu(x) + \sum_{j=1}^{J} \beta_{tj}\phi_j(x) + e_t(x)$$

```
fit ▷
  filter(State == "NSW") ▷
  autoplot()
```
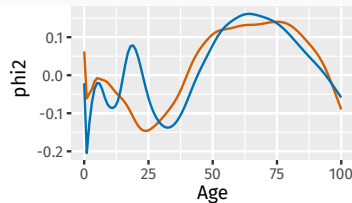
Functional data models

$$s_t(x) = \mu(x) + \sum_{j=1}^{J} \beta_{tj}\phi_j(x) + e_t(x)$$

```
fit ▷ age_components()
```

```
# A tibble: 1,212 x 10
    Sex     State   Age   mean   phi1    phi2      phi3      phi4       phi5     phi6
    <chr>   <chr> <dbl> <dbl>  <dbl>     <dbl>     <dbl>     <dbl>      <dbl>    <dbl>
 1 female   NSW      0  -4.07  0.147   0.0625   -0.0270    0.0986     0.0112  -0.0624
 2 female   NSW      1  -6.16  0.200  -0.0609   -0.194     0.116      0.0383  -0.238
 3 female   NSW      2  -6.82  0.182  -0.0483   -0.157     0.0924     0.0443  -0.264
 4 female   NSW      3  -7.17  0.170  -0.0368   -0.130     0.0362     0.000338 -0.321
 5 female   NSW      4  -7.40  0.164  -0.0165   -0.114    -0.0154    -0.0303  -0.374
 6 female   NSW      5  -7.57  0.158  -0.00759  -0.121    -0.0564     0.0247  -0.315
 7 female   NSW      6  -7.71  0.153  -0.00942  -0.133    -0.0976     0.112   -0.197
 8 female   NSW      7  -7.81  0.149  -0.0121   -0.143    -0.143      0.175   -0.0863
 9 female   NSW      8  -7.88  0.143  -0.0141   -0.148    -0.181      0.211    0.0131
10 female   NSW      9  -7.92  0.138  -0.0185   -0.142    -0.196      0.236    0.101
# i 1,202 more rows
```

Functional data models

$$s_t(x) = \mu(x) + \sum_{j=1}^{J} \beta_{tj}\phi_j(x) + e_t(x)$$

```
fit ▷ time_components()
```

```
# A tsibble: 1,440 x 10 [1Y]
# Key:        Sex, State [12]
    Sex    State  Year  mean beta1  beta2   beta3    beta4     beta5  beta6
    <chr>  <chr>  <int> <dbl> <dbl>  <dbl>   <dbl>    <dbl>     <dbl>  <dbl>
 1 female NSW    1901      1 11.2  -0.756 -0.0301  0.269   -0.155    0.409
 2 female NSW    1902      1 11.6  -0.708  0.0899  0.207    0.0282   0.507
 3 female NSW    1903      1 11.5  -0.962  0.169  -0.103    0.366    0.323
 4 female NSW    1904      1 11.1  -0.648  0.0985 -0.433    0.131    0.270
 5 female NSW    1905      1 10.1  -0.660  0.342  -0.0910   0.0862   0.612
 6 female NSW    1906      1  9.78 -0.865  0.496  -0.147   -0.101    0.306
 7 female NSW    1907      1  9.90 -0.861  0.0530  1.33     0.278    0.181
 8 female NSW    1908      1 10.1  -1.01   0.554  -0.0198  -0.00428  0.578
 9 female NSW    1909      1  9.42 -1.02   0.293  -0.365   -0.149    0.353
10 female NSW    1910      1  9.08 -0.650  0.172  -0.559   -0.253    0.0110
# i 1,430 more rows
```
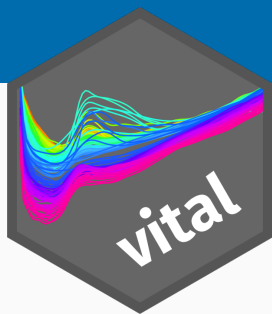
Other functionality

- Import data from Human Mortality Database and Human Fertility Database
- Convert demogdata, tsibble & data.frame objects to vital.
- Compute net migration from population, births and deaths.
- Compute total fertility rates from age-specific fertility rates.
- Various smoothing functions
- Coherent functional data models

Future plans

- Remaining tools from the `demography` package
- Stochastic population forecasting (as per Hyndman & Booth, *IJF*, 2008)
- All models handled by `StMoMo` package
- All methods from `MortalityLaws` package
- Suggestions from users
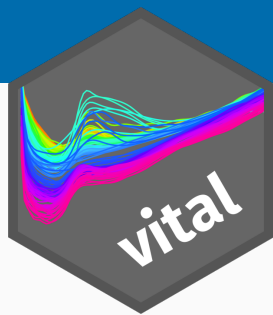
Future plans

- Remaining tools from the `demography` package
- Stochastic population forecasting (as per Hyndman & Booth, *IJF*, 2008)
- All models handled by `StMoMo` package
- All methods from `MortalityLaws` package
- Suggestions from users
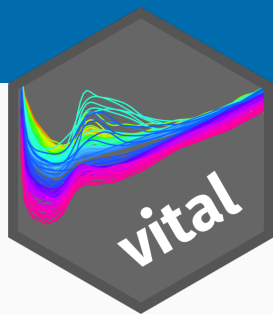
- **Slides**: robjhyndman.com/user2024
- **Package**: pkg.robjhyndman.com/vital/