

Supplemental file. A unifying framework for estimating generation time in age-structured populations: implications for phylogenetics and conservation biology: Supplement B.

Johan Jonasson¹, Tero Harkonen², Lisa Sundqvist³, Scott V. Edwards⁴ and Karin C. Harding^{5*}.

1 Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, Sweden. E-mail: jonasson@chalmers.se

2 Maritimas AB, Höga 160, 442 73 Kärna, Sweden. E-mail: tero.j.harkonen@gmail.com

3 Swedish Meteorological and Hydrological Institute, Gothenburg, Sweden. E-mail: lisa.sundqvist@smhi.se

4 Department of Organismic and Evolutionary Biology, Museum of Comparative Zoology, Harvard University, Cambridge, MA, United States of America. E-mail: sedwards@fas.harvard.edu

5 Department of Biological and Environmental Sciences, Gothenburg. Sweden. E-mail: karin.harding@bioenv.gu.se * Corresponding author.

Matlab files for generation of measures of generation times, population growth rates and the corresponding plots

In this section, the Matlab function and Matlab script that we used for producing figures are given. Instructions for how to use them are included in the files. We end the section with an example of how the script can be used to calculate generation times, population growth rates and the corresponding plots. The following Matlab function, `generationtime.m`, produces generation times and population growth rates for given life histories.

```

% This function computes the three measures A, T and mu and the
population
% growth rate. The inputs are b=(b_1,...,b_n), s=(s_1,...,s_n) and
% S=(S_1,...,S_n). Here b_j is the fecundity at age j, s_j is the
survival
% probability from age j-1 to age j and S_j=s_1*s_2*...s_j is the
survival
% probability from birth to age j. Note that you can also compute s
from S:
% s_1=S_1 and s_j=S_j/S_{j-1} for j=2,...,n. The number n is the upper
% limit for individual life length in the population under study. (The
% reason for requiring input of both s and S, even though one can be
% computed from the other, is to avoid the need for one function when
you
% want to input s and another when you want to input S.)

```

```

function [A,T,mu,lambda] = generationtime(b,S,s)

```

```

n=length(b);

```

```

f=b.*s; % computes fertility from fecundity and survival rate.

```

```

M=[diag(s(1:n-1)), zeros(n-1,1)];

```

```

L=[f;M]; %this row and the previous compute the Leslie matrix

```

```

lambda=eig(L);

```

```

lambda=lambda(1);

```

```

p(1)=f(1);

```

```

p(2:n)=S(1:n-1).*f(2:n); % computes the p-vector

```

```

R0=sum(p);

```

```

j=1:n;

```

```

v1=j.*p;

```

```

v2=lambda.^(-j);

```

```

A=v1*v2';

```

```

mu=j*p'/R0;

```

```

T=log(R0)/log(lambda);

```

The following Matlab script, `gentimegenerator.m`, allows the user to choose from the fecundity and survival functions used in this paper and output corresponding generation times and population growth rates.

```

% This script allows you to pick fecundity function, survival rate and
% corresponding according to the standard function given in this
paper.
% The choice is made by uncommenting the corresponding rows.
% The script then outputs the generation time measures A, T and mu and
% the population growth rate lambda.
% These are here given the names B, U, M and L respectively in order
for
% looping over a parameter of choice to produce vectors A, T, mu,
lambda,
% where each coefficient corresponds the corresponding value of the
% parameter that is varied.

j=1:n;

b=gamma*ones(1,n);
%b(1:4)=zeros(1,4);

%b=gamma*exp(-kappa*j);

%b=gamma*(1-exp(-kappa*j)).^3;

%b=zeros(1,n);, b(1)=c;, b(20)=c;

%s=exp(-alpha)*exp(-beta*exp(-rho*j)*(exp(rho)-1));
%S=exp(-alpha*j).*exp(-beta*(1-exp(-rho*j)));

%s=exp(-alpha*exp(beta*j)*(1-exp(-beta)));
%S=exp(-alpha*(exp(beta*j)-1));

s=exp(-alpha)*ones(1,n);
S=exp(-alpha*j);

[B,U,M,L]=generationtime(b,S,s);

```

Now, an example of how to use the Matlab files. Suppose that we want to use constant fecundity γ and exponentially decreasing survival function $e^{-\alpha j}$. Then uncomment the following rows in `gentimegenerator.m`:

```
b=gamma*ones(1,n);

s=exp(-alpha)*ones(1,n);

S=exp(-alpha*j);
```

Suppose now also that we want to study how A , T , μ and λ vary as γ varies between 0.1 and 0.5 with survival rate being fixed at 0.8 and age limit $n = 20$, i.e. we want to reproduce the plot in Figure ??(a). Recall that α is hazard rate, which is the negative logarithm of the survival rate. Hence we start by

```
alpha=-log(0.8);

n=20;
```

Then we may e.g. take

```
g = 0.1:0.001:0.5;
```

to create a γ -vector ranging from 0.1 to 0.5 with step size 0.001; note that we cannot call this vector `gamma` as that notation is occupied by the script. The generation times and population growth rates are now generated by

```
for i=1:length(g), gamma=g(i);, gentimegenerator, A(i)=B; T(i)=U;,
mu(i)=M;,, lambda(i)=L;,, end
```

Finally the plot is generated by

```
hold on
```

```
plot(lambda,A,'LineWidth',3), plot(lambda,T,'--','LineWidth',3),  
plot(lambda,mu,'-.','LineWidth',3)
```

which can then be edited to include legend and axes labels.