

Paper 227-28

Applied Population Genetics Using SAS[®] Software

Kristan A. Schneider, University of Vienna, Austria

Georg T. Schneider, Institute for Mathematics, University of Vienna, Austria

Barbara G. Schneider, Institute for Medical Statistics, University of Vienna, Austria

Abstract

In applied biomathematics computer simulations are necessary for scientific investigations. This work demonstrates how SAS[®] Software products can be used as a powerful tool for various applications in population genetics. In the first step we use SAS/IML[®] along with Base SAS[®] Software to create data, reflecting a biological or genetic background. Afterwards we use SAS/GRAPH[®] to display data, and Base SAS[®] Software as well as SAS/STAT[®] to perform statistical evaluations. Efficient implementation of difference equations, expressing genetic models in SAS/IML[®], and data management on Base SAS[®] Software are discussed. Further, we show how SAS[®] Macro Language can be used to improve the efficiency of the simulations and to connect the various SAS[®] tools. We also make suggestions how to improve graphics using the Annotate facility and how to export graphics into external graphic files. The skill level of the audience is assumed to be advanced.

1 Introduction

One of the main subjects in population genetics is the investigation of mechanisms that generate and maintain genetic variability in populations, and the study of how this genetic variation leads to evolutionary change, adoption and specification under environmental influences. In particular, population genetics provides the basis for understanding the evolutionary processes that have led to the diversity of life. In population genetics biological situations are described as mathematical models. Usually these models are expressed through difference equation, differential equations or stochastic processes. Since even for simple models a global analysis of the dynamical behavior of such models including all biological relevant factors seems to be impossible, usually an approach using computer simulations is made, in order to get a better understanding of the biological background represented by the model. In this paper we show how SAS[®] software can be used for computer simulations and exemplify this for a basic selection-mutation model, that

can be expressed by difference equations. The main tool for the simulations is SAS/IML[®].

2 Population genetic background

Since, many known loci in natural populations are polymorphic, considerable discussions have been devoted to mechanisms responsible for maintaining such large amounts of genetic polymorphism. These mechanisms may be diverse and complex. However, the maintenance of single-locus polymorphism can be explained by a variety of selection models, covering situations as overdominance, frequency-dependent selection, density-dependent selection, and selection due to environmental heterogeneity in space or time. In the latter class of models, namely those in which temporal variation in selection intensities is assumed, environmental variables relevant to fitness may vary in a stochastic manner, or they may vary regularly.

Beside selection, mutation is one of the central factors guiding genetic evolution. It is the ultimate source of genetic variation, generating the genetic diversity upon which selection acts. In population genetics mathematical models describing the interaction of selection and mutation were first developed by Fisher, Haldane, and Wright.

In this work we will introduce a general discrete-time one-locus selection model in arbitrary changing environments. We will further treat an important special case of the general model, namely the case of cyclically changing environments, especially the case of two environments, which is called alternating selection. Such models fit, for example, species living in a seasonal environment where generation-time is short relative to the duration of a season.

3 Brief description of a basic population genetic model

In this section we will introduce the basic discrete, deterministic one-locus selection mutation model for arbitrary changing environments.

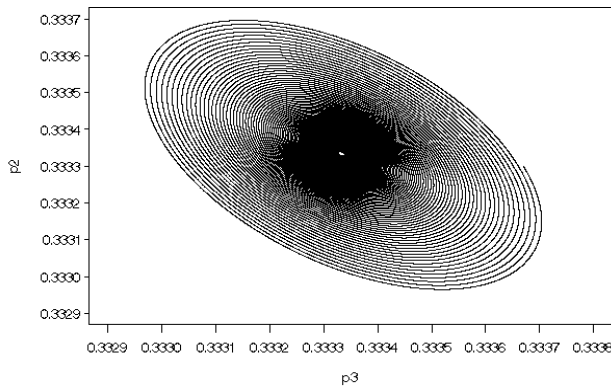


Figure 1: The Figure shows a trajectory in the (p_2, p_3) -plane converging towards a stable limiting cycle, starting at $(0.33303, 0.3333, 0.33367)$. There are 1000000 generations shown in the scatter plot. For the figure we used $\varepsilon = 0.001$ and $s = 0.0045$. The polymorphic equilibrium $\hat{p} = (1/3, 1/3, 1/3)$ is a source.

We suppose that the alleles $\mathcal{A}_1, \dots, \mathcal{A}_n$ can occur on a single autosomal locus of a sufficiently large, random mating population (so that random genetic drift can be neglected). Generations are assumed to be discrete and nonoverlapping. We further assume that genotype frequencies are the same in both sexes, that selection acts solely through differential viabilities and that mutation only occurs during reproduction. We denote the (relative) frequency of unordered $\mathcal{A}_i\mathcal{A}_j$ heterozygotes at the zygote stage of generation t by $2P_{ij} = 2P_{ij}(t)$ and that of $\mathcal{A}_i\mathcal{A}_i$ homozygotes by $P_{ii} = P_{ii}(t)$. Therefore, the frequency of the allele \mathcal{A}_i is

$$p_i = \sum_{j=1}^k P_{ij}. \quad (1)$$

Let μ_{ij} denote the mutation rate from \mathcal{A}_i to \mathcal{A}_j if $i \neq j$ and $\mu_{ii} = 0$. Since we assume random mating the P_{ij} are in Hardy-Weinberg proportions, i.e., $P_{ij} = p_i p_j$. We denote the fitness (viability) of $\mathcal{A}_i\mathcal{A}_j$ individuals by W_{ij} , satisfying $W_{ij} \geq 0$ and $W_{ij} = W_{ji}$. Hence, the frequency of $\mathcal{A}_i\mathcal{A}_j$ genotypes among adults that have survived selection is

$$P_{ij}^* = P_{ij}^*(t) = \frac{W_{ij}P_{ij}}{\bar{W}} = \frac{W_{ij}p_i p_j}{\bar{W}}, \quad (2)$$

where

$$\bar{W} = \bar{W}(t) = \sum_{i,j=1}^k W_{ij}(t)P_{ij}(t) \quad (3)$$

$$= \sum_{i,j=1}^k W_{ij}p_i p_j = \sum_{i=1}^k W_i p_i \quad (4)$$

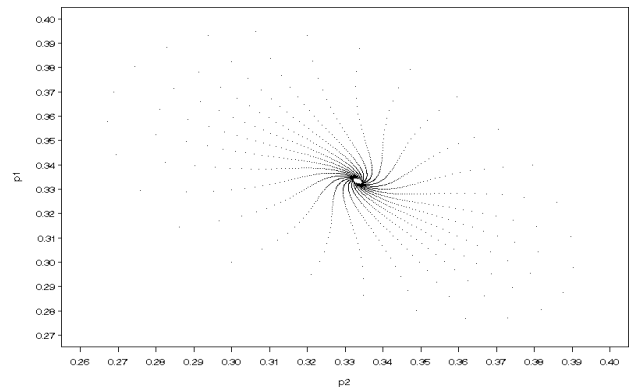


Figure 2: The figure shows a trajectory in the (p_1, p_2) -plane converging to a stable limit cycle. The polymorphic equilibrium $\hat{p} = (1/3, 1/3, 1/3)$ is a source. The figure shows a trajectory converging to a stable limit cycle. The starting point of the trajectory is $(0.3, 0.3, 0.4)$. For the figure we used $\varepsilon = 0.2$ and $s = 1.875$.

is the *mean fitness* of the population and

$$W_i = \sum_{j=1}^k W_{ij}p_j \quad (5)$$

is the *marginal fitness* of allele \mathcal{A}_i . Thus, the frequency of \mathcal{A}_i after selection is $p_i^* = \sum_{j=1}^k P_{ij}^* = p_i W_i / \bar{W}$. Then during recombination mutation occurs, thus the fraction of \mathcal{A}_i genes that do not mutate is $1 - \sum_{j=1}^n \mu_{ij}$, and \mathcal{A}_j genes give rise to a mutant \mathcal{A}_i with probability μ_{ji} . Therefore the frequency p_i' of \mathcal{A}_i in the next generation is

$$\begin{aligned} p_i' &= p_i^* \left(1 - \sum_{j=1}^n \mu_{ij}\right) + \sum_{j=1}^n p_j^* \mu_{ji} \\ &= p_i \frac{W_i}{\bar{W}} \left(1 - \sum_{j=1}^n \mu_{ij}\right) + \frac{1}{\bar{W}} \sum_{j=1}^n (p_j W_j) \mu_{ji}. \end{aligned}$$

Therefore we obtain the *selection-mutation equation*

$$p_i' = p_i \frac{W_i}{\bar{W}} + \frac{1}{\bar{W}} \sum_{j=1}^n (p_j W_j \mu_{ji} - p_i W_i \mu_{ij}). \quad (6)$$

In the following we will assume cyclical selection, i.e., fitnesses change periodically in time. We can express this mathematically as $W_{ij}(t) = W_{ij}(t + L) \forall t \in \mathbb{N}$, where L is the period length.

For given initial distributions of allele frequencies, fitnesses and mutation rates, we want to observe the the long time behavior of the system (6). The trajectories may converge to a fixed point or a limit cycle or may behave chaotically.

4 Implementation of the basic model on SAS/IML[®]

First we write the fitnesses in generation $s = t + kL$ ($k \in \mathbb{N}$, $t \in \{0, \dots, L-1\}$) as a symmetric $n \times n$ matrix

$$W(t) = \begin{pmatrix} W_{11}(t) & W_{12}(t) & \dots & W_{1n}(t) \\ W_{12}(t) & W_{22}(t) & \dots & W_{2n}(t) \\ \vdots & \vdots & \ddots & \vdots \\ W_{1n}(t) & W_{2n}(t) & \dots & W_{nn}(t) \end{pmatrix}. \quad (7)$$

Furthermore we successive append these matrixes and obtain a $n \times Ln$ matrix W . Further let

$$\text{one} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \quad (8)$$

and

$$K1 = \left(\sum_{j=1}^n \mu_{1j}, \dots, \sum_{j=1}^n \mu_{nj} \right). \quad (9)$$

We fix the number n of alleles and the length L of the environmental period. Then we create a SAS[®] data set containing the fitness matrices for several independent environmental cycles - be aware that we assume cyclical selection. Thus, each environmental cycle is represented by a $n \times Ln$ -matrix. For instance, suppose $n = 4$, $L = 15$ and we want to create 10 independent environmental cycles the data set containing the fitness matrices has 60 variables and 40 observations. Furthermore, we create a SAS[®] data set containing several mutation matrices, and a SAS[®] data set containing several initial distributions of allele frequencies as row vectors. For instance, let us assume we have 5 different initial distributions and the number of alleles is $n = 6$ then we have a data set containing six columns and five observations. For each combination of initial frequencies, mutation matrices and environmental cycles we want to observe the long time behavior of the trajectory. Therefore, we iterate the selection cycle several times and store the distribution of allele frequencies at the beginning of the cycles. We stop the iterations if an equilibrium is numerically reached, i.e., the euclidian distance between the allele vectors of allele frequencies of two consecutive cycles is less than a given size eps or if no equilibrium is reached within num generations. The frequencies are stored in a SAS[®] data set.

Now we assign in the names of the needed data sets and the other quantities to macro variables.

```
/*maximum number of iterations*/
%let num=50000000;
/*accuracy of convergence */
```

```
%let eps=1e-15;
/*length of environmental cycles*/
%let L=2;
/*number of alleles*/
%let n=3;
/*name of data set containing the initial
frequencies*/
%let pname=%str(cycles.initialfreq9);
/*name of data set containing the
fitnesses*/
%let wname=%str(cycles.fit6a);
/*name of data set containing the
mutation rates*/
%let mname=%str(cycles.M3);
/*name of output data set*/
%let simulationdata=%str(cycles.conv6a);
```

The output data set should contain a variable that indicates if an equilibrium is reached or not, it should group the data set for each combination of environmental cycles, mutationrates and initial frequencies. For each of this combination the numbering of observations is reset.

The following dataset creates the output data set:

```
data &simulationdata;
array x p1-p&n gneration iternum conv;
delete;
run;
```

Now we are able to implement the recursion relation (6) on SAS Iml, with the given parameters and initial conditions.

The programm begins by initializing the matrices and macro variables:

```
proc iml worksizes=400000000;
n=&n;
eps=&eps;
use &pname;
read all into p;
use &wname;
read all into wmat;
use &mname;
read all into m;
iteratio=0;
anzahl=ncol(p);
L=ncol(wmat)/n;
W=J(n,L*n,.);
aa=J(1,n+3,.);
one=J(n,1,1);
a=J(&num,n+3,.);
hy=J(n,1,.);
hz=J(n,1,.);
x=J(n,1,.);
K1=J(n,1,.);
```

In the next step the mutation rates, fitness matrices and initial frequencies are determined in iterative loops:

```
do mm=1 to nrow(m)/ncol(m);
m1=m[(1+(mm-1)*ncol(m)):mm*ncol(m),,];
K1=m1[,,];
do wm=1 to nrow(w)/n;
W=wmat[(1+(wm-1)*n):(wm*n),1:L*n];
do pm=1 to nrow(p);
p1=p[pm,];
iteratio=iteratio+1;
a[1,1:n]=p1[1,1:n];
a[1,n+1]=1;
a[1,n+2]=iteratio;
x[1:n,1]=t(a)[1:n,1];
xt=T(x);
knew=&num;
```

Now the main part of the loop is explained. In this part for a given environmental cycle, given mutation rates and given initial frequencies the selection-mutation cycle is iterated with respect to (6). Also the frequencies on the beginning of the environmental cycles are stored. The iteration stops if an equilibrium is numerically reached or if no equilibrium is established within num generations.

```
do k=2 to &num;
do t=0 to L-1;
w1=w[1:n,1+t*n:(t+1)*n];
hy=w1*x;
hz=t(m1)*(hy#x);
x=(1/(xt*hy))*(x#(hy#(one-K1))+hz);
xt=T(x);
end;
a[k,1:n]=xt[1,1:n];
a[k,n+1]=k;
a[k,n+2]=iteratio;
if ssq(a[k,1:n]-a[k-1,1:n])<eps**2
then do;
knew=k;
k=&num;
end;
end;
```

Finally the quantities of interest are written to the output data set.

```
if knew<&num then do;
a[1:knew,n+3]=0;
end;
else do;
a[1:knew,n+3]=1;
end;
b=J(knew,n+3,.);
b[1:knew,1:n+3]=a[1:knew,1:n+3];
edit &simulationdata;
```

```
append from b;
end;
end;
end;
quit;
```

5 Visualization

Now we show how SAS[®] procedures can be used to visualize the created data, and how to export the created graphic into an external file.

We can print the phase portraits of the system (6) at the beginning of the environmental cycles by using `proc gplot`. It is often desirable to label the trajectories occurring in phase portrait. We will show how this can be done using the Annotate facility. Since the starting points of the trajectories are known we can place the label near them. We consider a simple example how to generate an Annotate data set:

```
data anno;
length text $9. color style function $8.;
retain xsys '1' ysys '1' hsys '1' when 'a';
function='symbol'; size=4;
function='label';
size=4;
ftext=SWISS;
position='6';
x=29; y=35; text='a'; output;
x=33; y=41; text='b'; output;
x=38; y=33; text='c'; output;
x=35; y=63; text='d'; output;
x=60; y=8; text='e'; output;
x=7; y=35; text='f'; output;
x=5; y=99; text='g'; output;
x=95; y=8; text='h'; output;
x=3; y=5; text='i'; output;
run;
```

In order to obtain a graphic, appropriate for presentations, we set some graphical options and since we want to export the graphic we have to specify the output device, e.g., a bitmap file :

```
goptions ftext=SWISS ctext=BLACK htext=1
cells gsfname=grafout gsfmode=replace
device=bmp;
axis1 width=1 offset=(3 pct)
label=(a=90 r=0);
axis2 width=1 offset=(3 pct);
symbol1 c=black ci=black v=point height=1
cells interpol=NONE l=1 w=1;
symbol2 f=marker c=black ci=blue v='P'
height=1 cells interpol=NONE l=1 w=1;
symbol3 f=marker c=black ci=blue v='C'
height=1 cells
```

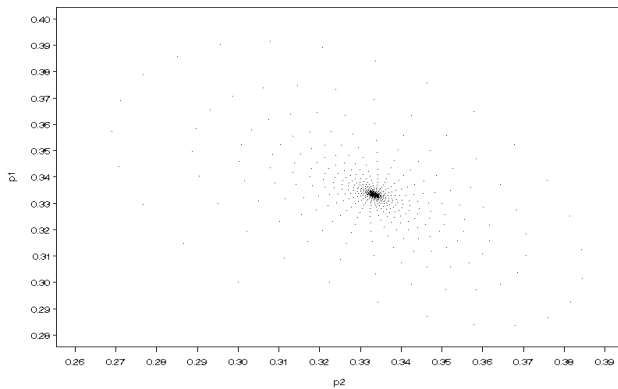


Figure 3: The figure shows a trajectory in the (p_1, p_2) -plane converging to the stable polymorphic equilibrium $\hat{p} = (1/3, 1/3, 1/3)$. For the figure we used $\varepsilon = 0.2$ and $s = 1.8$. The starting point of the trajectory is $(0.3, 0.3, 0.4)$.

```
interpol=NONE l=1 w=1;
goptions reset=f;
```

We then have to set a path where to set the graphic output:

```
filename grafout ' C:\ poster \ pict.bmp' ;
```

Finally we use the gplot procedure:

```
proc gplot data=&simulationdata ;
plot p2 * p3 =conv /
nolegend anno=anno
name='SCAT'
caxis = BLACK
ctext = BLACK
cframe =white
hminor = 0
vminor = 0
vaxis = axis1
haxis = axis2;
run;
quit;
```

6 Efficiency

In order to to achieve an efficient implementation on SAS/IML[®], certain knowledge in numerics, programming and SAS/IML[®] are necessary. We want to give some hints how to improve your SAS/IML[®] code. First, you should be aware of using loops. Often loops can be avoided, by using predefined functions and sub-routines. Using temporary variables can be useful as well, to avoid unnecessary calculation expand. It is recommendable to save data in smaller blocks in data intensive calculations. You can also store matrices. For data intensive calculations, the `worksize` and

`symsize` can be adjusted.

7 Conclusions

SAS/IML[®] can be used for numerical iterations of recurrence relations reflecting a biological background, as the above examples illustrate. Afterwards various SAS[®] tools can be used to handle the created data and to apply statistical analysis. Furthermore it turns out that SAS/GRAPH[®] is an appropriate tool to illustrate the created data and the applied statistics. However, the efficiency of your SAS/IML[®] code depends to a great extent on the implementation. Thus, for extensive numerical simulations it seems to be inevitable to focus on efficient programming in order to perform your simulations in adequate time. For data intensive simulation you data management is another important factor.

References

- [1] Bürger, R. 2000. The Mathematical Theory of Selection Recombination, and Mutation. Chichester: John Wiley & Sons.
- [2] Haldane, J. B. S. and Jayakar, S. D. 1963. Polymorphism due to selection of varying direction.
- [3] Hoeskstra, R. F. 1975. A deterministic model of cyclical selection. Genet. Res. 25, 1-15. Heredity 58, 237-242. (1974)
- [4] Hofbauer, J., and K. Sigmund. 1988. The Theory of Evolution and Dynamical Systems. Cambridge: University Press.
- [5] Hofbauer, J. 1985. The selection-mutation equation. J. Math. Biology (1985) 23: 41-53.
- [6] Karlin, S. and Lieberman, U. 1974. Random Temporal Variation in Selection Intensities: Case of Large Population Size. Theoretical Population Biology 6, 355-382.
- [7] Kirzhner, V.M., Korol, A.B., Ronin, Y.I., and Nevo, E. 1995. Genetic supercycles caused by cyclical selection. Proc. Natl. Acad. Sci. USA 92, 7130-7133.
- [8] Kirzhner, V.M., Korol, A.B., Ronin, Y.I., and Nevo, E. 1994. Cyclical behavior go genotype frequencies in a two-locus population under fluctuating haploid selection . Proc. Natl. Acad. Sci. USA 91, 11432-11436.
- [9] LaSalle, J. P. 1976, The stability of dynamical systems, Regional Conference Series in Applied Mathematics, SIAM, Philadelphia. PA.

- [10] Nagylaki, T. 1992. Introduction to theoretical population genetics. Berlin Heidelberg New York: Springer.
- [11] Nagylaki, T. 1975. Polymorphism in cyclically-varying environments. *Heredity* 35,67-74.
- [12] SAS/GRAPH® User's Guide, Version 8.
- [13] SAS/Iml® User's Guide, Version 8.
- [14] SAS® Language User's Guide, Version 8.

8 Contact Information

Kristan Schneider
address: Münichreiterstr.12 A-1130 Vienna Austria
Europe
e-mail: a9900273@ unet.univie.ac.at
phone: +43 1 0664 19 62 741
fax: +43 1 877 48 50