Vignette: Calculating contemporary Ne

Christine Ewers-Saucedo, George Sheriff

June 14, 2015

Effective population size is an important aspect of the evolutionary trajectory of a population. Its calculation, however, remains challenging, and has triggered the development of diverse calculations. We can distinguish between two fundamentally different ways to estimate Ne. On the one hand, coalsescent theory provides a means to estimate Ne over evolutionary times using either summary statistics, such as the number of segregating sites (Watterson 1975), number of alleles (Ewens 1972), heterozygosity (Zouros 1979, Kimmel et al 1998), variance of the number of microsatellite repeats (Kimmel et al 1998), or using the shape of the genealogy itself (Kingman 1982). The parameter calculated is theta, the mutation-rate scaled effective population size. Knowing the mutation rate allows us then to convert theta into Ne. Several of these theta calculators are available in the R package pegas (Paradis 2010). The coalescent theory also allows the estimation of changes in Ne over time based on changes in the branching pattern of the genealogy, which can be visualized in so-called scyline plots.
On the other hand, Ne can be estimated by observing and quantifying deviations from expections of infinitely large populations. We expect, for example, the neither linkage disequilibrium (Weir 1979) and drift (Waples) in infinitely large populations. The presence of either of these phenomena thus indicates a population smaller than infinity, and the magnitude of the phenomena scales with population size. These estimators calculate Ne of a population over the past few generations, whereas coalescent estimators generally integrate Ne over the past Ne to 4*Ne generations, depending of the mode of inheritance of the genetic marker employed.

This vignette will show the functionality of the functions LDNe and VarNe, which calculate effective population size based on levels of linkage disequilibrium and drift, repectively. First, you need to install the package multiNe like so:

```
install.packages("multiNe")
```

To test if multiNe was correctly installed, type:

```
library(multiNe)
```

If no errors appeared, the package is installed properly.

Linkage disequilibrium effective population size

To calculate Ne based on linkage disequilibrium, we need genotypic data of at least two loci from a panmictic population, stored as a genind object. The dataset "simG" are simulated microsatellite data of three loci and 50 individuals, generated in rmetasim (Strand and Niehaus 2014). We will load the dataset like so:

```
data(simG)
```

Before we can calculate Ne, we need to determine two parameters: "crit" and "mating". Waples (2006) showed that alleles at low frequencies will bias the Ne estimate. Thus, we should remove low frequency alleles. This is accomplished by the paramter "crit", which specifies the lowest allele frequency retained in the data set. Alleles with lower allele frequencies will be removed, as well as heterozygous individuals which have a low frequency allele. In our case, we chose 0.01 as the lowest allowed allele frequency. Note, however, that this value depends on your dataset: If this value is too high, most data will be removed, and we reduce the

power for estimating Ne. Leaving too many low frequency alleles in the data, on the other hand, will bias our results. Waples and Do (2008) suggest critical values between 0.05 and 0.01, but systematic tests remain to be undertaken. Lastly, we need to determine whether our study system can be better characterized as mating randomly, or monogamously. In most cases, random mating may be more appropriate, as it refers to the lifetime mating pattern. Now we are ready to calculate LD Ne:

```
LDNe(simG, crit=0.01, mating="random")
```

The output consists of a named vector. The first value is the point estimate for the complete dataset. The following values summarize the jackknifing output.

Drift effective population size

Drift effective population size is also called temporal or variance effective population size. It is based on the premise that finite populations experience drift, and this drift can be quantified in changes in allele frequencies from one generation to the next. Thus we need genotypic data of at least one locus (but more loci are always better) from two or more generations. The dataset "simG3" are simulated microsatellite data of three loci and 50 individuals per generations, generated in rmetasim (Strand and Niehaus 2014). It is important to note that each generation has to be stored as its own population within the genind object. We will load the dataset like so:

```
data(simG3)
```

If each generation is its own genind object, you can combine those objects into a single genind object using the function repool of the {adegenet} package. The function calculates one Ne value for each possible comparison between between generations, and jackknifes the result. You can run the function varNe by simply typing:

```
varNe(simG3)
```

The output consists of a matrix, each row summarizing the results for a generation pair.