SPPCR 2.0
The Final Frontier

Table of content

1) Intro

In the course of every program's life, it must be ported. To a new and better language, operating system, or platform. SPPCR 2.0 is a complete port, with bug fixes throughout, of Barry W. Brown's SPPCR 1.0.

2) Synopsis

The general use for this program is to calculate what we presume the actual ge, the mutation frequency, and the significance of a given data set, or pair of data set.

3) Input format and meanings

When using this program as a tool from Excel/Filemaker or other programs that export via applescript or what-have-you, you need to give it an initial argument of 4.The initial argument allows the program to be run several different ways, and allows me to give several types of specific, need-based output.
current initial arguments are:
0 - do nothing
2 - run hardcoded test data to test the program and make sure it is running
3 - interactively input data by hand responding to command prompts.

example:

```
Enter the number of runs (number of dna amounts):2
Enter the number of alleles seen:5
Enter the sizes of the 5 alleles:140 142 144 146 148
Enter the size of the 2 progenitor alleles. If the subject is homozygous, enter the size of the 1 progenitor twice.144
144
For each run, enter the expected genomes.
Expected ge for run 1:.8
Expected ge for run 2:.4
For each run, enter the number of replicates.
Replicates for run 1:
```

and so on. Terminology that the program used is all explained at the end of this section, as well as above the prompts.

4 - read in data in the following format:

numruns
numalleles
allele sizes (there needs to numallele of them)
progenitor allele's sizes, there needs to be 2 of them, if homozygous, repeat it twice
then, for each run/row
 a) observed/expected ge
 b) number of replicates
 c) number of alleles saw at each allele size

example:

```
4 2 19 144 146  148  150  152  154  156  158  160  162  164  166  168  170  172  174  176  178  180
```

```
154  156
0.81 96 0 0 0 1 1 52 53 0 0 0 0 0 0 0 0 0 0 0 0
0.58 32 0 0 0 1 2 15 11 0 0 0 0 0 0 0 0 0 0 0 0
```

Now lets look at this in details. 4 means we are using this format of input/output. The 2 means there are 2 different runs being looked at. The next number states that there at 19 possible allele sizes that have observed alleles in them. The next 19 numbers are of course the allele sizes. 154 and 156 are the 2 progenitors in this case.
After that, we have our first run, which has an expected ge of .81, 96 replicates, and the next 19 numbers are the observed alleles. Our second run has an expected ge of .58 and 32 replicates, and the next 19 numbers are of course the observed allele.

5 - This is the quiet version of option 6. It is used to calculate significance between 2 frequencies. The format to pipe your data in is:
Frequency 1
Frequency 2
Standard Error 1
Standard Error 2

6 - Like option 5, this will calculate the significance, but it is intended for interaction between the user and it. Just follow the prompt.

```
Enter the first mutant frequency: .046
Enter the second mutant frequency: .047
Enter the first standard error: .003
Enter the second standard error: .0042
```

7 - exactly like option 4, but has a verbose output

terminology:
run: A PCR experiment at one sample
allele size: PCR fragment size
replicates: number of wells examined
expected ge: what you think you put into the reaction
progenitors: parental alleles

3a) Assumptions about input

a) The first input to the program must be a single character, preferably of the numerical type from 0-6
b) At least one progenitor has been seen.

4) Output format and meanings

The computations are made for the whole. Meaning that if you do 4 runs, the ge and frequencies are calculated as if all the runs were one giant single run.
for mode:
2 - The output to the hardcoded data should be just a standard listing. It changes from build to build so that the developer may fine tune aspects and perhaps even discover bugs. It is not intended for the consumer's use.
3 -

```
d0 = 0.7106
```

The d0 is an antiquated statistical output used for legacy reason (hangover from sppcr 1.0 and previous incantations). In sppcr 2.0, the ge is already calculated for you.

```
95% CI (0.6417,0.7961)
```

This the the 95% Confidence Interval for the d0.
The 1/d0 and the confidence interval for that are exactly what they sound like.

```
estimated ge for run 0 = 1.0555
```

This is the statistically calculated estimate of what the ge is.

```
Mutant Frequency
estimate: 0.016160
bootsrap SE: 0.004675
```

This is gives the mutant frequency, and the resampled bootstrap error (to be used to determine the significance between 2 mutant frequencies)

4 - Since this is used strictly for connecting with outside programs via piping, this only outputs <# of runs> G.E.s, followed by the mutant frequency, followed by the standard error.

5 - Returns 1 number, that being your significance.

6 - The Z value is the statistical Z value used. If you wish to use a standard lookup table to confirm yourself, you can. If you are a normal person, and expect this program to do everything for you, it does. The calculated significance is provided on the next line.

```
Z = -0.193746
significance = 4.231874E-01
```

5) Known Problems

- Does not give proper results in the event of double progenitor loss. Single progenitor loss appears to have correct results, but it has not been thoroughly tested.

6) Frequently Asked Questions

Q) I have inputted several runs, each with the same expected ge, but different overall traits. Why do I get the same ge for all my runs?
A) the program calculates all "runs" as a single experiment. What you are seeing is the ge for all the runs together. If you wish to obtain a better estimate of the ge, do each rune individually.