SBEprimer Documentation

Purpose

SBEprimer will select compatible primers for minisequencing.

Usage

```
sbeprimer [inputfile] [outputfile] [tmbound] [homodimer-tm]
[groupsize] [minlen] [maxlen] [mintm] [maxtm] [fptemp] [fpfile]
[fast]
```

All parameters must be given in exactly this order. Calling SBEPrimer without any parameters or with insufficient or too many parameters will display on-screen help.

Parameters

[inputfile]	is the name of the file containing the snp data.
[outputfile]	is the name of the file that will contain the primers selected.
[tmbound]	is the temperature, for which heterodimer-interactions are calculated.
[homodimer-tm]	is the temperature, for which homodimer-interactions are calculated.
[groupsize]	is the maximum number of primers to be grouped together in one experiment.
[min/maxlen]	are the permissible lengths for the primers selected.
[min/maxtm]	specify a temperature range for the melting temperature of all primers.
[fptemp]	is the temperature for the false priming computation.
[fpfile]	Name of file containing the full sequence to check false priming against.
[fast]	Tells the program whether to use the fast (1) or the slow (0) false priming algorithm.

Inputfile

This parameter is the full path and filename (including any extensions) of the input file, containing the SNP data.
The input file has to be an ASCII text file. Each line should contain three words, separated by exactly one space character. A sample input file might look like this:

```
ID0 ATGAGAATGCTCTAACTTGTGTATGTGTCC GCTTTTTTGGCGTACGGACCCAGCCCTATT
ID1 GCCGTAGTAGTATCTGGCTCCTGCGGTCAT GACGAAAGATGACCCGCGGGCTGGTGTAAC
ID2 ACCATCGTTACTATGCGTATACTCCGTTAC GAAGCGTGCATTCTGAACGGTAAGACCTCA
ID3 GCTGTCGGGAACCTAACGATGTAGTAGTTA TGAGCTGTACTAGAAAGCCTTCGACGAATC
ID4 GGTTGGCAAATCGGCACCTCAATGATACAC CCGAGCCCTTGTTAACTCTGTGCCCGTATG
```

The first word in the input file is a unique SNP identifier. This can be any combination of characters, but no spaces are allowed in the identifier. This identifier will be written to the output file along with the primer. The maximum length for the identifier is 20 characters.
The second word is the sequence upstream of the SNP, in 5' to 3' direction. The only allowed characters are uppercase A, C, G and T. Note that N is NOT permitted.
The third word in each line is the sequence downstream of the SNP, also in 5' to 3' direction. Again, only uppercase A, C, G and T is allowed. The maximum length for both up- and downstream sequence is 50 characters each.
Make sure that the input file does not contain any additional spaces at the end of each line.
Note that the SNP itself is NOT contained in the input file. The program needs not know which bases may occur at the SNP position. Both the upstream and the downstream sequence MUST be given, they can be of length anywhere between 1 and 50 characters. They need not be the same length for all SNPs. If you do not want the program to choose a primer at, say, the upstream site for some SNP, simply put only one character at the corresponding position in the input file. This character will be ignored, as the program has a minimum length requirement for a primer (see the minlen parameter). Hence, if you did not want the program to choose the upstream primer for SNP ID1 in the example above, the inpufile should be modified like this:

```
ID0 ATGAGAATGCTCTAACTTGTGTATGTGTCC GCTTTTTTGGCGTACGGACCCAGCCCTATT
ID1 A GACGAAAGATGACCCGCGGGCTGGTGTAAC
ID2 ACCATCGTTACTATGCGTATACTCCGTTAC GAAGCGTGCATTCTGAACGGTAAGACCTCA
ID3 GCTGTCGGGAACCTAACGATGTAGTAGTTA TGAGCTGTACTAGAAAGCCTTCGACGAATC
ID4 GGTTGGCAAATCGGCACCTCAATGATACAC CCGAGCCCTTGTTAACTCTGTGCCCGTATG
```

Please also note that both the upstream and the downstream sequence must be given from the same strand, hence if the upstream sequence is from the + strand, so should the downstream sequence be. Hence, for ID0, for example, the template would read

```
5-ATGAGAATGCTCTAACTTGTGTATGTGTCCNGCTTTTTTGGCGTACGGACCCAGCCCTATT-3
```

where N marks the position of the SNP.

A problem arising frequently when using input files created on a PC or an Apple computer is the different handling of end-of-line symbols in these operating systems versus Unix. A PC, for example, uses two characters to mark the end of a line: Carriage return <CR> and line feed <LF>. Unix, on the other hand, uses only one. This difference will cause SBEprimer to stall, if <CR> and <LF> are used in the input file. If you do use a non-unix input file, it may be necessary to edit the file accordingly.

Outputfile

```
This filename will be used to write the output to. Any existing
file of that name will be overwritten. The filename can contain a
path.
A sample output file might look like this:

ID0 GGTCCGTACGCCAAAAAAGC 0
ID1 TATCTGGCTCCTGCGGTCAT 0
ID2 CCGTTCAGAATGCACGCTTC 0
ID3 AGGCTTTCTAGTACAGCTCA 0
ID4 CAGAGTTAACAAGGGCTCGG 0
```

Each line here again corresponds to one SNP. The first word in each line is the SNP ID you provided in the input file, the second word is the primer the program suggests in 5 to 3 direction, where the 3 end is adjacent to the SNP. The third word the number is the number of the experiment that the primer should be used in. If the program cannot find primers that will all work together or if you provided a maximum number of primers for one experiment, the program will suggest several experiments and indicate which primers will work together. In the example above, all primers can be used together in the same experiment.
In some cases, the program will not find a primer for each SNP. This may be due to the fact that all primers for a given site do false prime, or that no primer fulfills all criteria you supplied (for example, length and TM restrictions). If this is the case, the output file will simply not contain a primer for the SNP. If this happens, try rerunning the program with different parameters.

Tmbound

Tmbound is the temperature in degrees Celsius that is used to calculate primer dimer interactions. The formula

$$dG = dH - T\,dS$$

is used to calculate dG for interactions between two different primer candidates, and Tmbound specifies the temperature T to use in that calculation. Setting Tmbound to a temperature lower than the temperature used in the experiment will give an additional safety margin, whereas setting Tmbound to some high value (say, for example, 100 degrees) will assure that all interactions have positive dG, and hence no primer is excluded based on primer dimer formation. Note that every interaction with negative dG is assumed to form.

Homodimer-tm

Homodimer-tm is the temperature used to calculate dG for homodimer formation. Usually, you would set this temperature to the same value as Tmbound, however, in some cases it could make sense to use a different temperature, for example if SBEPrimer cannot find a primer for a given SNP, because all candidates form stable homodimers. In that case, it will help to choose a higher temperature for Homodimer-tm.
Note that checks for homodimer formation also cover hairpins.

Groupsize

The Groupsize parameter allows you to specify a maximum number of primers you want grouped together in one experiment. If you have only 100 beads, but 150 SNP sites, then SBEPrimer will try to find two primer sets for two experiments, each containing at most 100 primers, where all primers in each one groups will work together.

Min/maxlen

Minlen and maxlen specify the minimum and maximum length for primers. Only primers with length between (and including) minlen and maxlen will be considered further.

Min/maxtm

Mintm and maxtm specify the required melting temperatures for primers with their Watson Crick complements. Only primers with mintm<TM<maxtm will be considered further.

Fptemp

Fptemp is the temperature used for the false priming computation. The program will calculate all possible interactions of the 3'-end bases of each primer candidate with the template, and discard any primers that show a negative dG.
Set this to 0 to skip the false priming computation.

Fpfile

This is the name of file containing the full sequence to check false priming against. This file is a fasta-format file, i.e. its lines are alternating identifier and sequence lines. The identifier lines begin with a >, followed by a unique identifier for the following sequence. The sequence line contains the plain sequence. Degeneracies are allowed, and must be denoted by the N symbol. Hence, only A, C, G, T and N are allowed in the sequence lines of a fasta file.
Note that SBEPrimer does not automatically generate the negative strand for a given template sequence. Hence, make sure that both the plus and the minus strand of your template are explicitly given in the template file, both in 5 to 3 direction.
It is very important to be aware of the fact that the false priming module will assume a perfect match whenever a primer base pairs with N. Hence, if your template file contains sequences with stretches of several Ns, this will cause all primer candidates to falseprime at that position. Hence, all of them will be labeled infeasible, and the program will report it could not find any primers. A simple way around this problem is to replace longer stretches of N by just one N symbol.
The false priming module checks, for each primer, how many stable interactions a primer forms with the template sequence. The program requires the SNP sites to be included in the template as well, and hence assumes that there will be one stable

interaction. If it finds more or less for a given primer, it will discard that primer. Set filename to "null" to skip the false priming check (fptemp must be zero as well).
A sample template fill could look like this:

```
>template sequence
ATGAGAATGCTCTAACTTGTGTATGTGTCCGCTTTTTTGGCGTACGGACCCAGCCCTATTGCCGTA
GTAGTATCTGGCTCCTGCGGTCATGACGAAAGATGACCCGCGGGCTGGTGTAACACCATCGTTACT
ATGCGTATACTCCGTTACGAAGCGTGCATTCTGAACGGTAAGACCTCAGCTGTCGGGAACCTAACG
ATGTAGTAGTTATGAGCTGTACTAGAAAGCCTTCGACGAATCGGTTGGCAAATCGGCACCTCAATG
ATACACCCGAGCCCTTGTTAACTCTGTGCCCGTATG
>template complement
CATACGGGCACAGAGTTAACAAGGGCTCGGGTGTATCATTGAGGTGCCGATTTGCCAACCGAT.
>something else to check against
GGACACATACACAAGTTAGAGCATTCTCATGACGAAAGATGACCCGCGGGCTGGTGTAACGTAANG
GNGTATNNNCATAGTNANTACTACATCGTTAGGAAGGGCTCTGC
```

A problem arising frequently when using input files created on a PC or an Apple computer is the different handling of end-of-line symbols in these operating systems versus Unix. A PC, for example, uses two characters to mark the end of a line: Carriage return <CR> and line feed <LF>. Unix, on the other hand, uses only one. This difference will cause SBEPrimer to stall, if <CR> and <LF> are used in the input file. If you do use a non-unix input file, it may be necessary to edit the file.

Fast

Fast is a simple 0/1-switch to instruct the program which false priming routine to use. If set to zero, the program will calculate a full alignment of each template site with each primer, allowing for gaps. The maximum local alignment will then be chosen, and if dG for that alignment is negative, an interaction is assumed.
If set to 1, the program will execute the quick false priming check, which does not allow any gaps in the alignment. This calculation can be done faster, but is less accurate.

Return Values

```
SBEPrimer returns the primers it suggests to use in a file; see
the Outputfile paragraph in the Parameters section.
```

Functional Description

The program proceeds in the following steps:
Read the input file containing the different SNP sites and the template file.
False Priming Check:
Create a lookup-table for all possible 4-mer combinations (AAAA, AAAC, AAAT,, TTTT), storing a list of the SNP sites with 3 ends complementary to those 4mers in the lookup-table.
Go through the template sequence, looking at all 4mers in the templates. Check the corresponding 4mer in the lookup-table created in step 1a. If the table contains any SNP sites for the given 4mer, align the site and the following bases in the template

sequence using either the dgalign algorithm (fast command line parameter = 0) or by extending a base at a time (fast command line parameter = 1), not allowing any gaps in the latter case. In both cases, the maximum local alignment will be chosen.
If for a given SNP site the up- or downstream sequence shows negative free energy for mor than one position in the template sequence, mark the site as unusable due to false priming.
For the remaining up- and downstream sequences for all sites, generate a list of all primers adjacent to the SNP, that fulfill the length (between minlen and maxlen) and melting temperature (between mintm and maxtm) requirements.
Remove all primers from the list that form homodimers at the temperature specified by the homodimer-tm parameter.
For the remaining primer candidates, calculate all interactions with primer candidates for a different SNP site, at the temperature specified by the tmbound parameter. This is done using the dgalign algorithm.
For each SNP site, choose the primer with the fewest interactions.
Distribute the chosen primers over different experiments, using a graph coloring problem heuristic.
Write the output to the file specified.

Comments

The program is very sensitive to parameter choices. Usually, it is a good idea to start with very stringent conditions (tmbound and homodimer-tm around 20 degrees, fptemp maybe 30 degrees, using the slow false priming algorithm), and lower the bar in consecutive runs until a satisfactory result is achieved. Especially if lots of sites are excluded due to false priming, raising fptemp and going from the slow to the fast algorithm for false priming detection will help however, at the price of lower quality primers.