BSSM4PASIF code package

Michael E Sparks

July 17, 2007

1 Getting Started

This software facilitates generation of splice site probabilities for use with the `PASIF` program. For producing such distributions for use with `GeneSeqer` or `GenomeThreader`, please use the variant of this software called `BSSM4GSQ`. `BSSM4PASIF` can process either plain text `GeneSeqer` or gthXML v1.0 (or later) output. gthXML formatted output can be produced natively by the `gth` program of `GenomeThreader`, or generated from plain text `GeneSeqer` output using the `GSQ2XML.pl` script, available from either `http://www.genomethreader.org` or `http://www.public.iastate.edu/~mespar1/gthxml/`. The user is urged to study the following reports prior to using this software:

1. Brendel V, Xing L, Zhu W. (2004) Gene structure prediction from consensus spliced alignment of multiple ESTs matching the same genomic locus. *Bioinformatics.* **20**:1157-69.
2. Sparks ME and Brendel V. (2005) Incorporation of splice site probability models for non-canonical introns improves gene structure prediction in plants. *Bioinformatics.* **21**:iii20-iii30.

2 Directions

1. There are two subdirectories in the input directory, gsq/ and fas/. You absolutely must meet the following requirements to make the code work.
    (a) There must be a one-to-one correspondence between files in these two directories.

(b) Files in the fas/ directory must have an extension of ".fas" and those in the gsq/ directory must have one of either ".gsq" or ".xml", for plain text `GeneSeqer` or gthXML formatted data files, respectively. You cannot mix plain text and gthXML input files.

(c) The basename prefixes of cognate fas/gsq file pairs, i.e., the substrings prior to ".gsq" or ".fas", must be identical.

(d) All references made to a genomic template in the spliced alignment output file must be identical to the file's "file handle" mentioned above. This essentially mandates that each fas/gsq cognate file pair correspond to one genomic sequence and its spliced alignment annotation, respectively.

There are sample training data in the input/sample/ directory. These data will not generate any meaningful probabilities, and are only intended to demo the system.

2. Edit `Mktraindata.sh` such that the FORMAT variable is set correctly for the files placed in the input/gsq/ directory; this is described explicitly in the header of the script. Run `Mktraindata.sh`. This produces exon and intron data, sorted according to phase and placed in the output/exons_introns/ directory, and sampled, phase-sorted BSSM training data placed in the output/training_data/ directory. In each of these directories, data will be written to a subdirectory named according to the donor/acceptor dinucleotide termini trained for. If this is unclear, inspect the contents of the output directories after unpacking this code, run the script, and look at them again.

   `Mktraindata.sh` processes GT-AG introns, by default. For other types, tune the DON and ACC variables (set these in CAPITAL letters!) and run it again. This will not overwrite any existing output in the training_data/ or exons_introns/ directories so long as a different DON/ACC combination is used. Rerunning the script using a DON/ACC pair whose results were already recorded will cause that data to be overwritten.

   Splice site phase is indexed using the following nomenclature:

```
Phase 0 -> C O D I  (Intron falls between codons)
      1 -> C I O D  (Intron between 1st and 2nd codon pos)
      2 -> C O I D  (Intron between 2nd and 3rd codon pos)
```

3. Run `Mkbssmparm.sh`. The script will solicit some configuration information:
   - (a) Name of output file ("foo.bssm")
   - (b) Root directory of training data. (If you haven't done anything non-standard up to this point, it should be safe to just say "y" here.)
   - (c) Build GT model? For GT-AG parameterizations. (If you trained for these intron types, responding with "y" will put the probabilities in your *.bssm file. Else, say "n".)
   - (d) Build GC model? For GC-AG parameterizations. (If you trained for these intron types, responding with "y" will put the probabilities in your *.bssm file. Else, say "n".)
   - (e) File to write ascii data to ("foo.bssm.ascii")

   The `BSSM_print` utility can be used to produce an ascii representation of the trained splice site matrices. (Note that, if any among the seven training data files for each terminal (for either GT-AG or GC-AG introns) is empty, the matrix development code will emit NAN's. Since estimating probabilities from no data is certainly ridiculous, my code was not written to support it.) The *.bssm.ascii file presents the weight array matrices in the following order:

   ```
   for TERMINAL in (0-1):
     for HYPOTHESIS in (0-6):
       print transition probabilities
   ```

   where for TERMINAL, 0 and 1 index donor and acceptor sites, respectively; and for HYPOTHESIS, 0, 1, 2, 3, 4, 5, 6 index the T0, T1, T2, F0, F1, F2 and Fi hypotheses, respectively. You can verify that your parameter file contains valid probability mass distributions by using the `verify_pmf.pl` script in the src/plscripts/ directory, e.g.,

   ```
   $ cat something.bssm.ascii | ./verify_pmf.pl
   ```

   Please see the commentary in that file for more details; output of a row of zeros is expected, and does not signal an error.

3 Contact Info

If you have questions, concerns, etc., please email me at `mespar1@iastate.edu`.