Programming for Bioinformatics
BIOL 8803 B
November 9th, 2015

This week we're going to do a bit of functional genomic analysis.

Commands and such from this week:

`sub` – Create a subroutine

passing scalars to subroutines

`@_` - Subroutine parameters

references and the `\` operator

passing arrays to subroutines

dereferencing – Go back from a reference

`goto` – go somewhere else in the script

`eval` – execute some code on the fly

`use` – use a module

`system` and `` ` `` - make a system call

`defined` – Check if a variable is defined

`//=` - defined or equals

`||=` - or equals

`given` – Start a `given/when` structure

`when` – Look at the data in a `given/when` structure

1.) Write a script to overlap two sets of genomic coordinates and find intersecting members.

This is a very common task in genome analysis. Often we want to find functional elements that overlap with other elements, e.g. transcription start sites contained in transposable elements or predicted transcription factor binding sites within DNase hypersensitivity sites. Or even some set of genes or horizontally transferred regions. Maybe you found some peaks in your ChIP-seq data and you want to see if those overlap with known enhancers. If you don't know what those words mean, go look them up.

Comparing all coordinates of the first set versus all coordinates of the second set will work for small sets only. That method scales with the multiplication of the sizes of the two sets. If you have 10^6 members in each set (not at all unrealistic) do you want to make 10^{12} comparisons? You need to find a better way. I did this very shortly after I started here, and I still use my modern version of this script.

Things your script should do:
1.) Take in two files that contain sets of coordinates in BED format. You can assume that the contents of both files are sorted by chromosome, start and stop. 15 bonus points if you detect input that is unsorted and alert the user. Having the data sorted allows you to greatly speed up the process of the overlap.
Definitions for BED data are here:
http://genome.ucsc.edu/goldenPath/help/customTrack.html#BED
2.) Take in an option for the minimum percent overlap, i.e. percent of bases of any given member of the first set that must be in a member of the second set to be counted as overlapping.
3.) Allow conditions to be passed, as in the example on slide 40 from Week 9 (where I use eval). My version has '$firstStrand" and "$secondStrand' because that's what I called the variables in my script. You can call them whatever you want.
4.) Print to STDOUT the members of the first set which overlap with the second set and meet the minimum overlap and other conditions specified. Each should only be printed once in this manner.
5.) Allow an option to print both the member of the first set, and the member of the second set that it overlaps with on the same line, in effect 'joining' the rows.

Two bed files can be found at http://jordan.biology.gatech.edu/biol8803b/

Example:
Overlap of the following two sets with a minimum of 100% overlap would yield one row.

Set 1:

```
chr1 1500 1750 1    0    +
chr1 4500 5000 2    0    +
chr1 8500 9500 3    0    +
```

Set 2:

```
chr1 1000 2000 a    0    -
chr1 3000 4000 b    0    -
chr1 9000 11000     c    0    -
```

Output:

```
chr1 1500 1750 1    0    +
```

If the ‘joining’ option (from #5) were given:

```
chr1 1500 1750 a    0    -    chr1 1000 2000 1    0    +
```

If the minimum overlap was 50% instead of 100% you would get two rows

```
chr1 1500 1750 a    0    -
chr1 8500 9500 3    0    +
```

If you were to condition the overlap on the first strand being equal to the second strand, you would get no output because the members of the two sets have opposite strands.

Note:
The overlap of two coordinates can be calculated as

```
MIN(stop_one, stop_two) - MAX(start_one, start_two)
```

Anything greater than 0 is the number overlapping bases, otherwise it indicates no overlap. Perl has min and max functions, or you can make your own.

If you are in the Jordan lab and you look at my version of this script…well, just don’t.

Deliverables:

- **Code:**
  - overlapBed.pl
- **A readme file:** this file should contain the number of lines in your output after you execute the command in wrapper11.sh