Assembly evaluation using Mauve

Learning points:

- Using Mauve to re-order and orient contigs according to a reference genome
- Using Mauve to visually inspect assemblies for issues
- Using Mauve Assembly Metrics to compare several assemblies against a reference, and against each other

Re-order velvet contigs against a reference

We have provided a reference, annotated assembly in GenBank format in

```
/data/assembly/ref/NC_000913.gbk
```

Start Mauve.

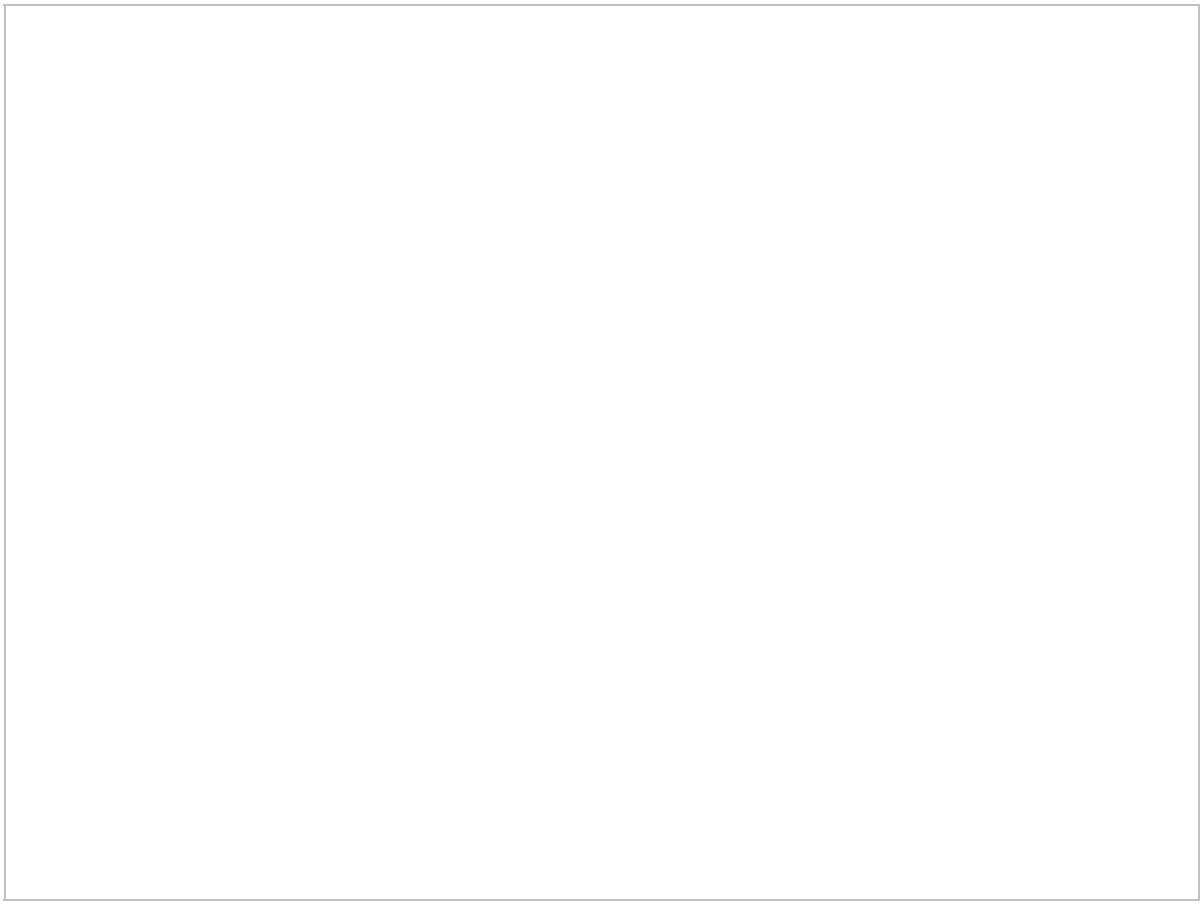
Choose "`Tools → Move Contigs`" from the menu. Choose as location for the output `~/assembly/mauve`, add a descriptive name for the output folder to the path. Add the `NC_000913.gbk` file as the *first* sequence and then an assembly of your choice as *second* sequence.

Wait for the alignments (there may be more than one round) to complete and then examine the final one (with the highest number) in the viewer. Zooming in is not too practical, you will need to use the < and > arrows to keep the region you want to focus on in view. From the Mauve manual:

> The alignment display is organized into one horizontal "panel" per input genome sequence. […] Each colored block (Locally Collinear Blocks or LCBs) surrounds a region of the genome sequence that aligned to part of the other genome, and is presumably homologous and internally free from genomic rearrangement. When a block lies above the center line the aligned region is in the forward orientation relative to the first genome sequence. Blocks below the center line indicate regions that align in the reverse complement (inverse) orientation.

Inside each block Mauve draws a similarity profile of the genome sequence. The height of the similarity profile corresponds to the average level of conservation in that region of the genome sequence.

More information can be found here: http://bit.ly/PqVfYB



Look for the following:

- What is the largest contiguous block of alignment?
- Can you spot misassemblies?
- Are there regions missing from the genome in your assembly? What gene features are in those places?

Continues on next page...

Using Mauve Assembly Metrics

Mauve Assembly Metrics is a small software built on top of Mauve. It uses the contig mover to order an assembly file towards the reference, and then extracts a set of metrics, for example completeness of the assembly, of the predicted genes etc. Once the program has been run on more than one assembly, a summary table and several summary plots can be produced.

As the name for each assembly is derived from the contig/scaffold file, renaming these files to short, but descriptive names is advisable. One could for instance collect copies of the relevant files in one folder and run the program from there. (hint for unix-geeks: using symlinks for this allows for both renaming, and prevents copying files).

Use Mauve Assembly Metrics as such:

- move into the folder where you want to gather the results
- for each assembly, write the following command

```
mauveAssemblyMetrics.sh \
-reference /data/assembly/ref/NC_000913.gbk \
-assembly your_file.fa \
-outputDir asm_name \
-reorder asm_name
```

Important: keep `asm_name` the same for both the `-reorder` and `-outputDir` parameters

This will run for a while, creating a folder called asm_name, with subfolders `alignment1, alignment2, ...`

- upon finishing, several `.txt` files are added to the folder, we'll ignore those for now
- NOTE if the analysis fails, and you want to retry it, remember to remove the folder that was created first (otherwise mauve will protest about a pre-existing folder, and quit)
- you can now open Mauve, choose `File → Open Alignment`, find the folder with the highest alignment number, and choose the `alignmentX` file. This alignment is the same as would have resulted from a separate contig

mover operation

- repeat the steps above for any other assembly you are interested in comparing
- after you have processed at least two assemblies, and while still in the same folder, type

```
mauveAssemblyMetrics.pl ./
```

- this will, using the `.txt` files in each folder, generate a `summaries.txt` file, and a set of PDFs.
- view the PDFs, and open the in a spreadsheet program (MS Excel, Open Office)

From the Mauve Assembly Metric paper (Darling et al, 2011):

> Given the following reference genome and assembled genome
>
> Reference: `AGGCTAGCGCGCGATTAGGATC`
> Assembly: `AGTAGCGGGCCGATTAAGANC`
>
> A genome alignment of the reference and assembly might look like:
> Reference: `AGGCTAGCGCG-CGATTAGGATC`
> Assembly: `AG--TAGCGGGCCGATTAAGANC`
>
> From this alignment, we would calculate the assembly scoring metrics as follows (not an exhaustive list of metrics):
>
> - Miscalled bases: 2 (C!G and G!A)
> - Uncalled bases: 1 (N)
> - Extra bases: 1 (Insertion of C in assembly)
> - Missing bases: 2 (Deletion of GC in assembly)
> - Number of extra segments: 1
> - Number of missing segments: 1
>
> In addition, the method produces a variety of other metrics. The location of miscalled bases, missing segments, and extra segments is exported to a

tab-delimited text file for subsequent analysis. GC content of the missing and extra regions is also exported. Misassemblies are identified as rearrangement breakpoints inside of contigs. The double cut and join (DCJ) distance between the assembly and reference is calculated. The DCJ distance is as a measure of the minimum number of rearrangement events involving two cuts and a join that would be required to transform the assembly's genome structure into that of the reference genome.
Finally each protein coding sequence in the reference genome is checked in the assembly for whether it yields an intact coding sequence, with types and location of substitution and frameshift errors reported.
Other metrics:

- breakpoints: In contig assemblies each breakpoint suggests a misassembly. In sca
  old assemblies a breakpoint could be either misassembly or failure to place a smaller scaffold within a larger scaffold (lack of assembly).
- Number of SNPs: the sum of uncalled bases (‘N’s) and miscalled bases.