Identify Blueberry to *Arabidopsis* Orthologs

Scott J. Teresi *Edger Lab - Michigan State University*

This document contains my workflow, scripts and notes on generating *Vaccinium corymbosum* to *Arabidopsis thaliana* orthologs

Approach:

To identify orthologs, I am going to first identify orthologous genes using synteny. I will then follow up by using BLAST to identify any orthologs that we would've missed using the synteny-based approach (orthologous genes that may have translocated elsewhere, thus breaking synteny).

Rough Outline

1. Use **SynMap** on CoGe to compare syntenic blocks between *Arabidopsis thaliana* and *Vaccinium corymbosum*.
2. Supplement results with BLAST search to catch any non-syntenic orthologs (single-gene transpositions).
3. Clean up ortholog file, parse out gene-pairs that do not match our significance threshold. Finalize ortholog output.

Data Input and Genome Versions:

This section catalogs the versions of the genomes I used for this analysis and contains the methods I used to generate, missing/supplementary files.

Genome Versions Used:

Regular CoGE ID	Masked ID
Arabidopsis thaliana Col-0 (id1)	CNS PL2.0 Masked Repeats 50X (v10, id 16746)
Vaccinium corymbosum (id39928)	mask w/ RepeatMasker (v3, id 58746)

Running SynMap

This section describes the methods to run SynMap on CoGe. I will describe some of the options I have used here and why (in the following sections), but it would benefit the reader to go over SynMap's documentation and read more about E-values.

SynMap Analysis Options:

We are going to run SynMap with default options. Here is a link to the documentation. The default options at the time of writing are:

- DAGChainer

  Relative gene order

  Maximum distance between two matches: *20* genes

  Minimum number of aligned pairs: *5* genes

- Merge Syntenic Blocks

  Algorithm: Quota Align Merge

  Maximum distance between two blocks: *4:1*

- Syntenic Depth

  Algorithm ?????

- Fractionation Bias

  Test???

- CodeML

- Advanced Options

A Word on E-Values:

Briefly, E-values, which stands for expectation value, is a correction of the p-value for multiple testing (we are multiple testing when we search each gene for a match in the other genome, and by chance we could generate a significant p-value, so we must enforce a mathematical correction). In the context of database searches the E-value is the number of distinct alignments with a score equivalent to or better than S[1], that are expected to occur in the database search by chance.

SynMap Output:

It outputs a tab-separated text file. I would encourage the reader to check out the summary of the output format from this link under *Results* and *DAGChainer Output*.

[1] S: A score is the numeral value that describes the overall waulity of an alignment, higher numbers means higher similarity.

Filter the Output File:

The DAGChainer output file has a lot of extraneous information. We are going to distill it down into a 2-column tab-separated values (tsv) file. To do this we will use the Pandas module in Python to maniuplate the dataframe. If you have an aversion to Python you can always accomplish this task in R.

Set up Python Environment:

Please refer to the project README.md to ensure you have the correct Python packages.