Zostera marina Gene PAV README

Script: parseSplitAlignments.py

Script reads a blat alignment file in an easier to read format (bestHit format) from standard in (stdin) and searches for broken alignments to determine whether transcript is present but fragmented within an Illumina short-read assembly.

bestHit example format- Designed to human readable and condense blat block hits into simple starts and ends

queryName queryLength queryAlignmentStart queryAlignmentEnd alignmentOrientation targetName targetLength targetAlignmentStart targetAlignmentEnd numberOfMatches percentIdentity percentCoverageOfQuery

Query is typically the transcript of interest and the target is the Illumina assembly.

Example output:

geneA 84 1 83 - Scaffold1 291911 129825 129743 83 100.00 98.81

parseSplitAlignments.py usage:

cat bestHit | python parseSplitAlignments.py presentGeneList.dat IDcutoff COVcutoff

presentGeneList.dat - text file that contains genes considered present based on single alignment hits from the bestHit file. The python script ignores these to only consider alignments that are possibly fragmented in the assembly

IDCutoff- minimum identity cutoff to consider when searching for fragmented transcript alignments. This should be set higher than cutoff used for single alignment hits to avoid spurious alignments

COVcutoff- minimum coverage cutoff to consider a transcript 'present' based on broken alignments. If 3 or fewer broken alignments with percent identities greater than the IDcutoff are found, and more than COVcutoff bases of the transcript length are covered, then the transcript is considered present.

Example, if COVcutoff = 80 and IDcutoff = 90, if 3 alignments with greater than 90% identity are found for geneB, covering 85% of the length of the gene, geneB is considered present ('1'). If 3 alignments with greater than 90% identity are found for geneC, covering only 65% of the length of the gene, geneC is considered absent ('0').

Each gene's presence or absence is encoded per illumina assembly, which is then combined into the transcript presence/absence matrix (zosteraTranscriptPAVMatrix.dat.bz2)

###########################

Script: clusterPAVgenes.R

R Script reads and parses the PAV transcript matrix file into *de novo* cluster genes in core, shell and cloud bins based on their observance across the population. Designed to avoid making hard cutoffs that are arbitrarily set.

core genes- present in nearly every considered sequenced individual in a population

shell genes- present in only some individuals in a population. Typically the midpoint for variation (30-70%).

cloud genes- present in a low number of individuals in a population.

Also included are a list of *Zostera* clones that are excluded from the final analysis (cloneLibIDs.dat), as well as genes that were inconsistently called among clonal genotypes. Example, if assemblies 1,2 and 3 were generated from a clone of the same genotype, any gene that showed presence/absence variation among those assemblies (poorlyCalledGenesInClones.dat) was excluded from the final analysis.