Upstream to sppIDer:

combineGenomes.py

This script requires all the reference genomes to be in one location with a key between the file names and what the reference genome should be named in the combined genome. Additionally, a desired output name for the reference fasta is required and the optional trim threshold is allowed. The order the genomes are concatenated follow the order in the key file, the order of the chromosome/scaffolds/contigs will remain as they are in the given reference file, but will be renamed by the name in the key file and numbered sequentially. The key should be a text file with a list of the “desired reference name” and the actual reference fasta separated by a tab. The “desired reference name” cannot include any hyphens (-) an example is given as “SaccharomycesRefKey.txt”. The output will be a concatenated reference with the “desired reference name” and chromosome number (as an Arabic numeral) separated by a hyphen. This format is necessary for the plotting scripts which parse the chromosome names. Two steps in sppIDer require index files for the genome used, thus this custom script will also make these required files.

sppIDer.py

This script requires the combination reference genome made with combineGenomes.py and fastq formatted short-read sequence files for the test of interest. Additionally, you can choose to run it so that depth is analyzed by basepair (-byBP) or grouped by coverage (-byGroup). This script will create a file (output\_sppIDerRun.info) that will print all the argument choice and the time to run each step. Additionally, all the normal standard outputs for each step will be printed to screen along with the time to run each step.

This script will then run bwa –mem to map the reads to the combination reference genome and outputs a sam file. This same file is then passed to a custom python script (parseSamFile.py) to parse the data by which genome the reads map to and the mapping quality (MQ) the output of this is passed a Rscript (MQscores\_sumPlot.R) that will plot the percentage of reads that map to each genome and unmapped reads, the same plot without the unmapped bar, for those datasets where most the reads don’t map, and a violin plot showing the distribution of mapping qualities for each genome.

The sam output will also be used for samtools view which will only retain MQ >3 and then samtools sort that will order the reads to match the reference order. Next bedtools genomeCoverageBed is called to determine depth of coverage either by basepair (-d) or grouped by coverage (-bga). The –d option give is more accurate but takes longer for large genomes there for the –byGroup option is appropriate for larger genomes. This output is parsed by meanDepth\_sppIDer(-d or –bga).R. These scripts average depth by species, chromosome, and in 10,000 windows and prints these out to text files. The two scripts do the same thing but depend on the input from bedtools. Average depth by species is plotted by sppIDer\_depthPlot\_forSpc.R. The windowed average depth is plotted by sppIDer\_depthPlot(-d or –bga).R