Popmarker is implemented by Python and there is no need to install it. This pipeline includes the following five steps, where the root of directory is assumed in $HOME, the source codes are in the $HOME/source, and the example data used in this pipeline are given in the $HOME/Example.

Prerequisites packages include

Numpy (1.11.1 or later),

Scipy (0.17.1),

Matplotlib (1.5.1 or later),

Biopython (1.67 or later),

and an additional package StatsModels (0.6.1 or later) is recommended for calculating the False Discovery Rate (FDR).

**Step1. Finding Orthoship**

First, go to the folder of Step1

cd $HOME/Example/Step1/

and obtain 1-to-1 orthologous relationships among species (48 species in sp\_list.txt) based on the output of OrthoFinder (OrthologousGroups.txt)

python $HOME/source/orthoships.py -o $HOME/Example/InputFile/OrthologousGroups.txt -s $HOME/ Example /InputFile/sp\_list.txt -r orthoships\_result.txt

The orthoships\_result.txt will give 2287(/14265 in total) 1-to-1 orthologous groups. For the detailed options, -h (or --help) is available

python $HOME/source/orthoships.py -h

**Step2. Constructing Species Tree**

Go to the folder of Step2

cd $HOME/Example/Step2.SpeciesTree/

Construct a reference tree using above all orthologous groups

python $HOME/source/maketree.py --accuracy localpair --gap 10 --wag --gamma --orthogrp $HOME/Example/Step1/orthoships\_result.txt -s sp\_file\_path.txt

The sp\_file\_path.txt tells maketree.py where the files of the orthologous sequences are located (see $HOME/Example/sp\_file\_path.txt). This script will produce concatenated sequences for all species (concatenated\_seqs\_2206.aln) and a constructed tree (concatenated\_seqs\_2206.tree). For the detailed options, type

python $HOME/source/maketree.py -h

**Step3. Ranking gene (Rank1 and/or Rank2)**

Go to the folder of Step3

cd $HOME/Example/Step3.RankGene/

Copy (symbolically link) the species tree (concatenated\_seqs\_2206.tree) and 1-to-1 orthologous groups (orthoship\_result.txt) in the current folder by

cp -s $HOME/ Example/Step2.SpeciesTree/concatenated\_seqs\_2206.tree ref.sp.tree

cp -s $HOME/ Example/Step1/orthoships\_result.txt .

and copy the result of all-against-all blastp for each pair of orthologous protein sequences among species by

cp $HOME/Example/goodProteins.blast .

It has an option for comparing species in only a clade as the following format (see $HOME/ Example/Vca\_clade.txt)

Species\_name1

Species\_name2

Species\_name3

…

To rank genes with respect to the reference tree or a clade of species if –clade is given as following:

python $HOME/source/rankgene.py -b goodProteins.blast -t  ref.sp.tree -o orthoships\_result.txt --clade Vca\_clade.txt --corr 0 -r rankgene\_PCC\_result.txt

This command uses Pearson correlation coefficient (--corr 0) as the measurement and also tell program to rank genes in accord with the Vca clade (--clade Vca\_clade.txt). For the other options, type

python $HOME/source/ rankgene.py -h

**Step4. Constructing Gene Tree**

Go to the folder of Step4

cd $HOME/Example/Step4.GeneTree/

and copy three previously obtained files for later use

cp -s ../Step2.SpeciesTree/sp\_file\_path.txt .

cp -s ../Step3.RankGene/rankgene\_PCC\_result.txt .

cp -s ../Step3.RankGene/rankgene\_Kendall\_result.txt .

To construct a gene tree based on the top N genes of the ranking gene list

python $HOME/source/maketree.py --wag --gamma --accuracy localpair -g rankgene\_PCC\_result.txt -s sp\_file\_path.txt -t 1

This tells maketree.py to construct a gene tree using top one gene (-t 1) in the rank of rankgene\_PCC\_result.txt and to yield two outputs, ID000001.aln and ID000001.tree.

You might rename the output files to meaningful names as

mv ID000001.tree PCC.top1.tree

mv ID000001.aln PCC.top1.aln

Or, to construct a gene tree based on the concatenated multiple sequences in the top 2 genes (-t 2) as

python $HOME/source/maketree.py --wag --gamma --accuracy localpair -g rankgene\_PCC\_result.txt -s sp\_file\_path.txt -t 2 --con

It will produce concatenated\_seqs\_2.aln and concatenated\_seqs\_2.tree

And you might rename the output files by

mv concatenated\_seqs\_2.tree PCC.top2.tree

mv concatenated\_seqs\_2.aln PCC.top2.aln

The maketree.py can construct the gene tree using the concatenated gene sequences from N top genes in the Species Rank and also can specify more top N-th genes in order to include top genes un the Clade Rank. For example, using Top4 genes (from 1st gene to 4th gene) and additional top 15th genes, which is the top 12th gene in the Clade Rank, in the Species Rank to construct a gene tree by

python $HOME/source/maketree.py --wag --gamma --accuracy localpair -g rankgene\_PCC\_result.txt -s sp\_file\_path.txt -t 4 -n 15 --con

It will produce concatenated\_seqs\_5.aln and concatenated\_seqs\_5.tree

You might rename the output files by

mv concatenated\_seqs\_5.tree PCC.top4+15.tree

mv concatenated\_seqs\_5.aln PCC.top4+15.aln

The other example shows using the Top6 genes (1st gene to 6th gene) in the Species Rank and more top 8th gene, which is the 38th in Clade Rank, based on the correlation method of Kendall tau, as

python $HOME/source/maketree.py --wag --gamma --accuracy localpair -g rankgene\_Kendall\_result.txt -s sp\_file\_path.txt -t 6 -n 8 --con

It will produce concatenated\_seqs\_7.aln and concatenated\_seqs\_7.tree

You might rename the output files by

mv concatenated\_seqs\_7.tree Kendall.top6+8.tree

mv concatenated\_seqs\_7.aln Kendall.top6+8.aln

For the detailed options, type

python $HOME/source/maketree.py -h

**Step5. Calculating Distance between two trees**

Go to the folder of Step5

cd $HOME/Example/Step5.TreeTreeDist/

Copy the species tree and the gene trees to the current folder for the comparisons:

cp –s $HOME/Example/Step2.SpeciesTree/concatenated\_seqs\_2206.tree ref.sp.tree

cp -s $HOME/Example/Step4.GeneTree/PCC.top4+15.tree .

cp -s $HOME/Example/Step4.GeneTree/PCC.top1.tree .

cp -s $HOME/Example/Step4.GeneTree/PCC.top2.tree .

cp -s $HOME/Example/Step4.GeneTree/PCC.top4+15.tree .

For calculating the distance between species tree and gene tree, it is done by

python $HOME/source/treedist.py -r ref.sp.tree -q \*.tree -o whole\_tree\_distance.txt

For calculating the distance between only Vca clades in species tree and gene tree, it is

python $HOME/source/treedist.py -r ref.sp.tree -q \*.tree -t Vca051011E Vca051011F Vca051011G Vca1114GL Vca1116 Vca151112c Vca200612B VcaCCS02 VcaDS40M4 VcaHY01 VcaKC13 VcaNBRC15631 VcaUMTGB204 -o VcaClade\_tree\_distance.txt

For the detailed options, type

python $HOME/source/treedist.py -h