CYPminer (v1)

CYPminer is a Python-based program with a graphical interface, allowing users CYP identification/classification and downstream analyses from all kingdom protein sequences in a user-friendly manner. The program requires two external programs called USEARCH and RPSBLAST and two databases (*i.e.*, CYPdb_usearch and CYPdb_rpsblast). These programs and databases should be individually downloaded, and their locations should be provided to CYPminer. Users are able to freely download USEARCH (https://www.drive5.com/usearch/download.html), RPSBLAST (ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/), and the databases (https://github.com/Okweon/CYPminer). This manual will take you through the complete use of CYPminer.

1. **Begin by opening CYPminer.**
   1) On a PC, click the CYPminer_v1.exe (Fig. 1).

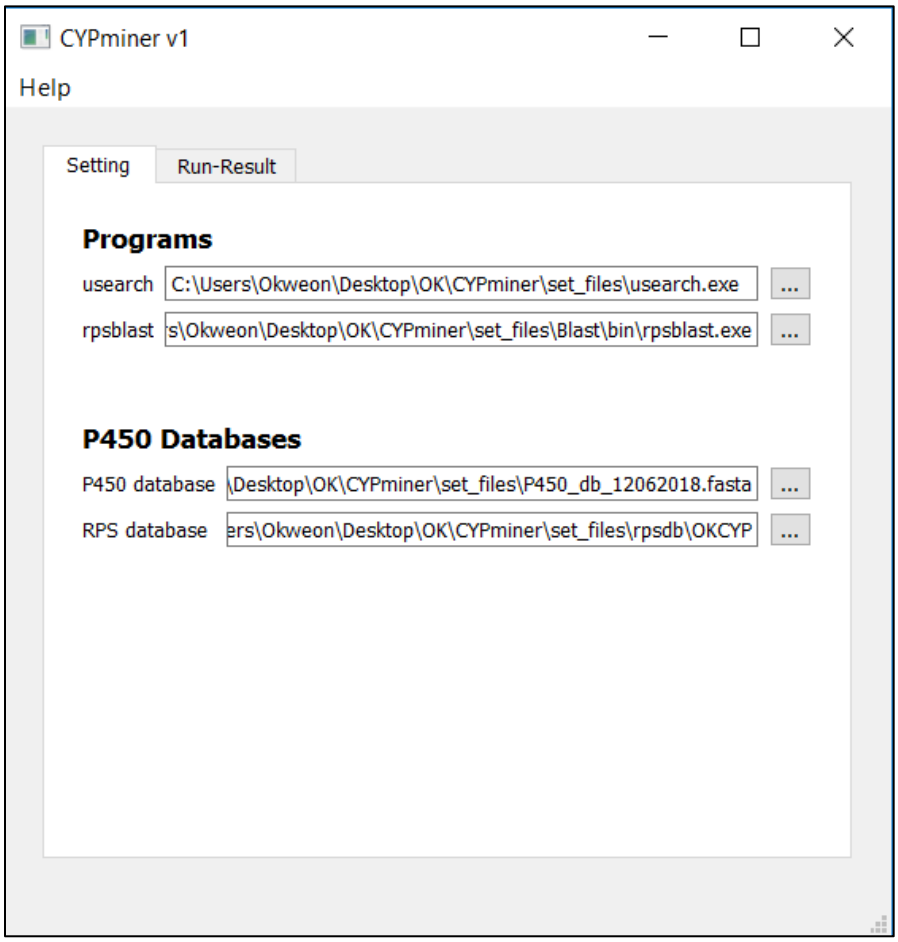

Figure 1. A screen view of the CYPminer.

2. **Set the paths of the two external programs and two local databases on the Setting window.**
The program requires two external programs called USEARCH and RPSBLAST and two databases (i.e., CYPdb_usearch and CYPdb_rpsblast). These programs and databases should be individually downloaded, and their locations should be provided to CYPminer. When click on the CYPminer_v1.exe, it leads to the setting window where you can enter the paths of the external programs and local databases into the corresponding fields. Here, the paths can be set in two ways, directly adding and using a file selector button (Fig. 2). Once all the paths have been specified, there is no need to set them up again. When needed, the paths can be edited anytime.

   1) To use a file selector button in Setting window (pictured below), click a file selector button, navigate through file system, and select the corresponding file; for usearch path, select usearch.exe; for rpsblast path, select rpsblast.exe; for P450 database, select P450 database fasta file; **for RPS database, select any file in the rpsdb directory and then remove the extension of the file (*e.g.*, OKCYP.rps → OKCYP)**.

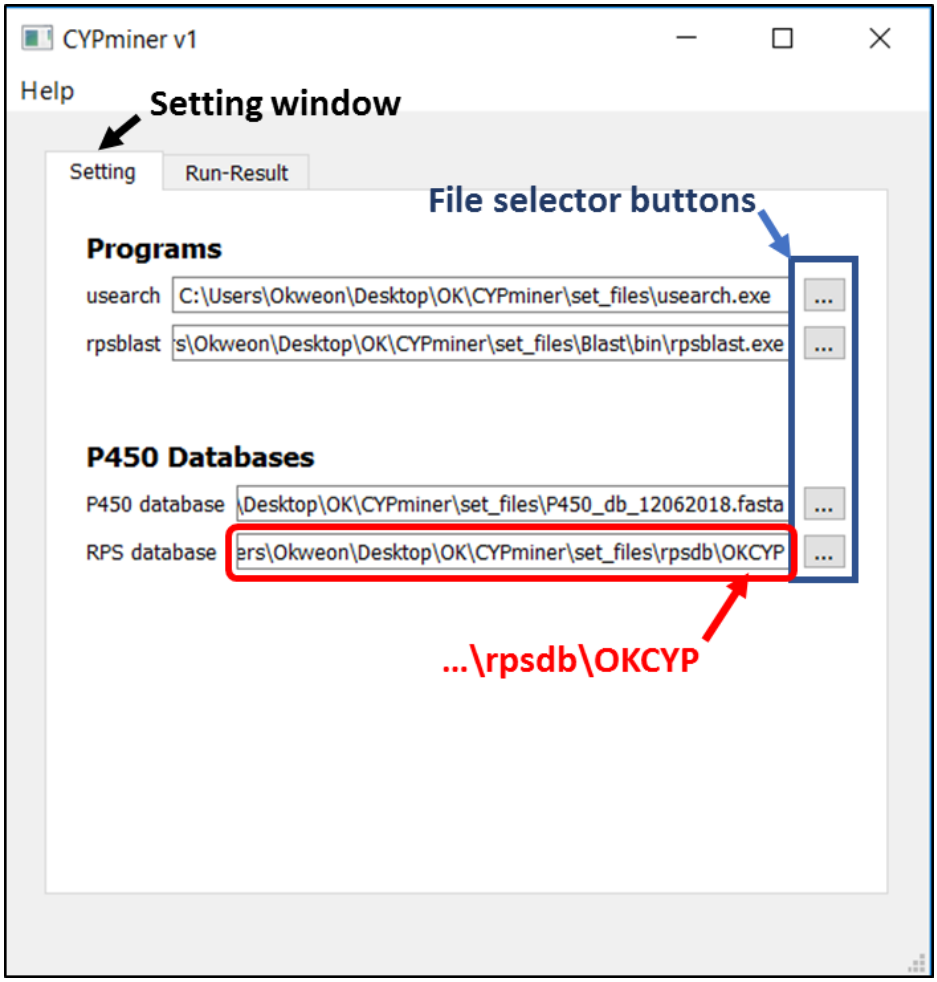

Figure 2. A screen view of the Setting window.

3. **Set the paths of the input directory and output directory on the Run-Result window.**
After clicking on the "Run-Result"-Button, you can provide the paths for your input directory that has protein fasta files and output directory where the results are saved. The paths can be set in two ways, directly adding and using the folder selector button (Fig. 3).

   1) Click on the “Run-Result”-tag (pictured below).
   2) To use the folder selector button in the Run-Result window (pictured below), click the folder selector button, navigate through the folder system, and select the corresponding directory.

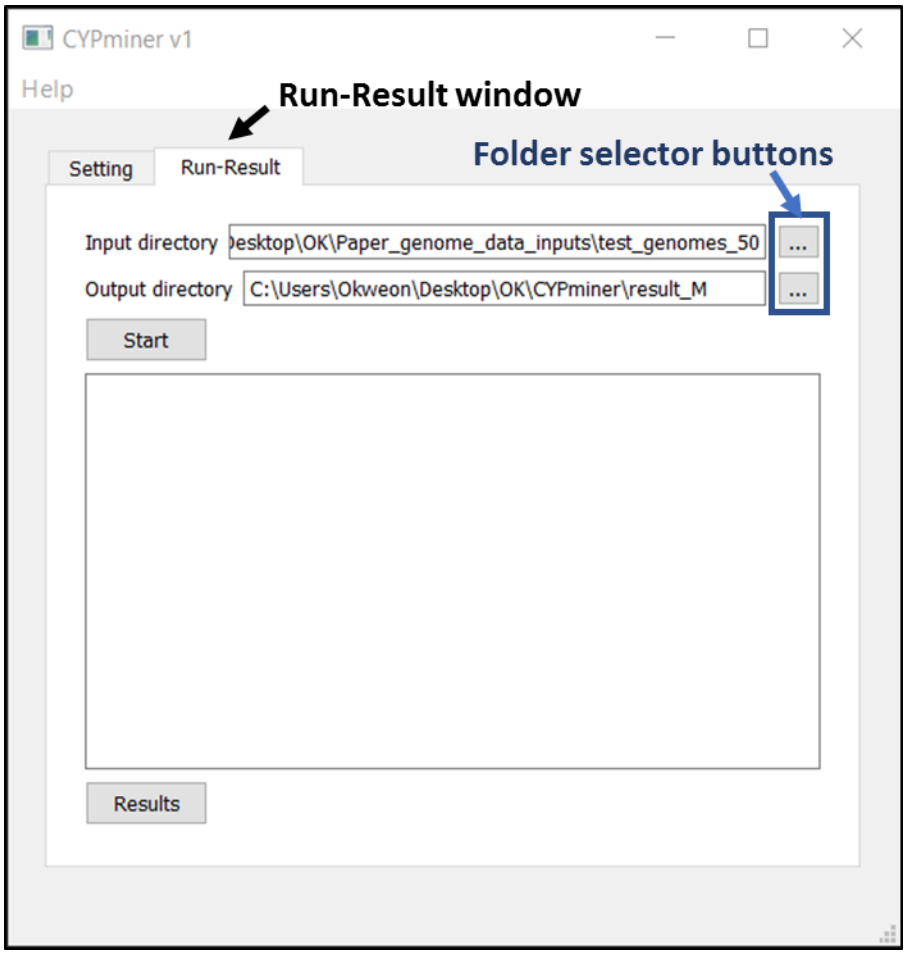

Figure 3. A screen view of the Run-Result window.

4. **Start analyzing and checking results on the Run-Result window.**
In general, we anticipate the increase in running time of the CYPminer as its input size increases. When running an analysis, user can see the progress of the CYPminer on the processing window or in the output directory (Fig. 4).

   1) Click Start button to start (pictured below).
   2) Click Results button to check the result files (pictured below).

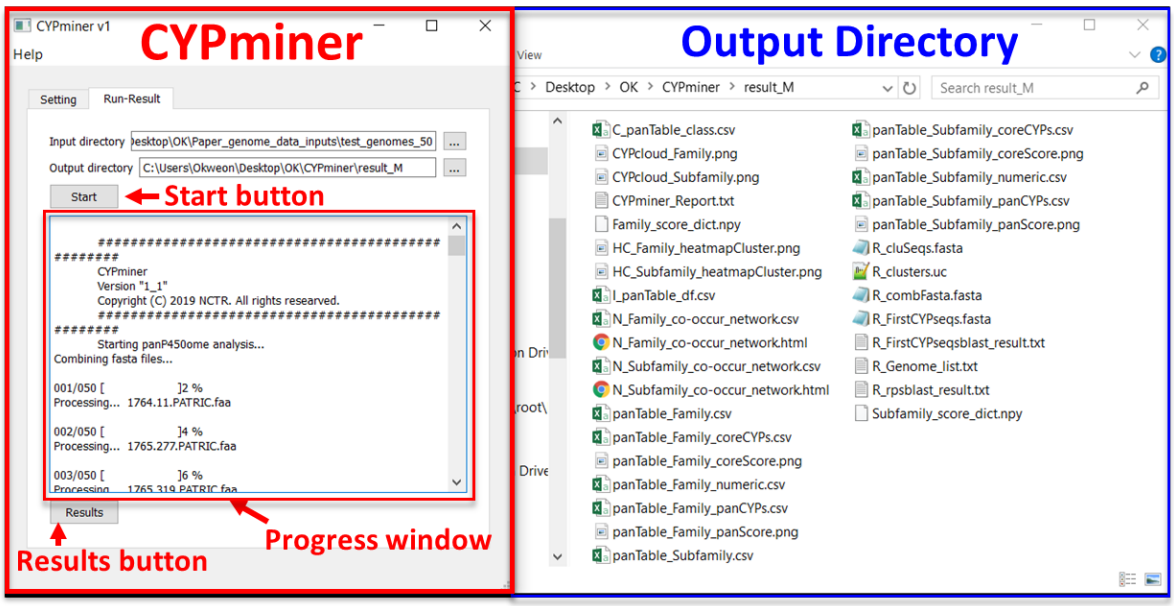

Figure 4. A screen view of Run-Result window and output directory.

5. Output files of the CYPminer.

CYPminer generates a maximum of 32 output files. The output files could be grouped into two groups, which consist of a raw data group, with 'R-' in the output filename and practical table, and a figure group, with a categorical initial name, such as 'I' (identification), 'C' (classification), and 'N' (network). The large table datasets (i.e., GCMs) were subjected to data visualization to generate four different categories of visual contexts (i.e., pan-CYPome, co-occurrence network, clustering, and CYP cloud), which make users easily understand the significance of table data in CYPminer.

1) **R_combFast.fasta**: a fasta file with all protein sequences in the input directory.
2) **R_clusters.uc**: an output file of the UCLUST.
3) **R_cluSeqs.fasta**: a Fasta file of the centroids (*i.e.*, representative sequences of the clusters).
4) **R_FirstCYPseqs.fasta**: a fasta file with the CYP sequences identified from the centroids.
5) **R_FirstCYPseqsblast_result.txt**: a blast output file of the R_FirstCYPseqs.fasta.
6) **R_Genome_list.txt**: a txt file with a complete genome list in the input directory.
7) **R_rpsblast_result.txt**: a txt file of the rpsblast output.
8) **I_panTable_df.csv**: a csv file of a GCM (Genome-CYP-Matrix) with no classification information.
9) **C_panTable_class.csv**: a csv file of the GCM with classification information of CYPs.
10) **N_family_co-occur_network.csv**: a csv file of a co-occurrence network (family level).
11) **N_family_co-occur_network.html**: a html file of a co-occurrence network (family level).
12) **N_subfamily_co-occur_network.csv**: a csv file of a co-occurrence network (subfamily level).

13) **N_subfamily_co-occur_network.html**: a html file of a co-occurrence network (subfamily level).
14) **HC_Family_heatmapCluster.png**: a png image file of a hierarchically-clustered heatmap of a GCM.
15) **HC_Subfamily_heatmapCluster.png**: a png image file of a hierarchically-clustered heatmap of a GCM.
16) **CYPcloud_Family.png**: a png image file of a CYP cloud (family level).
17) **CYPcloud_Subfamily.png**: a png image file of a CYP cloud (subfamily level).
18) **panTable_Family.csv**: a csv file of a GCM (classified at the family level and gene names as a value).
19) **panTalbe_Family_coreCYP.csv**: a csv file of a core CYPome (family level).
20) **panTable_Family_coreCYP.png**: a png image file of a core CYPome (family level).
21) **panTable_Family_numeric.csv**: a csv file of a GMC (classified at the family level and frequency data as a value).
22) **panTable_Family_panCYPs.csv**: a csv file of a pan CYPome (family level).
23) **panTable_Family_panScore.png**: a png image file of a pan CYPome (family level).
24) **panTable_Subfamily.csv**: a csv file of a GMC (classified at the subfamily level and CYP gene name[s] as a value).
25) **panTalbe_Subfamily_coreCYP.csv**: a csv file of a core CYPome (subfamily level).
26) **panTable_Subfamily_coreCYP.png**: a png image file of a pan CYPome (subfamily level).
27) **panTable_Subfamily_numeric.csv**: a csv file of a GMC (classified at the subfamily level and frequency data as a value).
28) **panTable_Subfamily_panCYPs.csv**: a csv file of a pan CYPome (subfamily level).
29) **panTable_Subfamily_panScore.png**: a png image file of a pan CYPome (subfamily level).
30) **Family_score_dict.npy**: a Numpy data file of the panCYPome score (family level).
31) **Subfamily_score_dict.npy**: a Numpy data file of the panCYPome score (subfamily level).
32) **CYPminer_Report.txt**: an analysis report file of the CYPminer.

6. Email support.

We provide technical support via email. Send a message to oh-gew.kweon@fda.hhs.gov