PL-search

Protein remote homology detection is a fundamental and important task in the analysis of protein structure and function. Many search methods have been proposed to improve the detection of remote homologues and the accuracy of ranking lists. The Position Specific Scoring Matrix (PSSM) profile and Hidden Markov Model (HMM) profile can contribute to improving the performance of state-of-the-art search methods.

In this study, we trace profile-link information used to construct the PSSM or HMM profiles in order to propose a Profile-Link-based search method (denoted PL-search). In PL-search, more robust profile links are constructed through the double-link and iterative extending strategies, and an accurate similarity score of sequence pairs is calculated from the two-level Jaccard distance for remote homologues. We tested our method on the classic and updated versions of the SCOP benchmark datasets. Our results show that whether HHblits, JackHMMER or PSI-BLAST are used, PL-search significantly improves the search performance in terms of ranking quality as well as the number of detected remote homologues.

1. Content

./query_example:	the testing examples in the format of Profile link construction
./ProfileLinkDB:	the constructed profile-link-db path for PL-search searching against
./DoubleLinkDB:	the contructed double-link-db path for PL-search searching
./Example_result:	saving the final result of query examples
./Query_original_result:	search result of query examples using original search method

2. Usage

inlink.pyc	**In-link construction**	
	-out_link_path	"The path of out-link from original search method"
		out_link format "flag(default 1) protein_name e-value"
	-out_path	"The path of in-link for writing"
double_link.pyc	**Double-link strategy**	
	-out_link_path	"The path of out-link with e-value=0.001 from original search method"
	-in_link_path	"The path of in-link with e-value=0.001"
	-double_link_path	"The path of double-link for writng"
iter_extend_link.pyc	**Iterative extending strategy**	
	-double_link_path	"The path of double-link"
	-out_path	"The path of extended profile link for writing"
	-beta1	"The parameter beta1 for controlling extending process"
	-beta2	"The parameter beta2 for controlling extending process"
plsearch.pyc	**Calculation of the similarity of sequence pairs by Jaccard distance and final ranking list**	
	-query_link_path	"The constructed profile link for query sequences"
	-out_path	"The path for writing search result"
	-double_link_path	"The path of constructed double-link"
	-profile_linkdb_path	"The path of constructed profile link db"
	-original_search_path	"The path of original search result for query sequences"

3. Example of PL-HHblits on SCOP1.59 database

python plsearch.pyc query_example/ example_result/ DoubleLinkDB/plhhblits/3CBLS_all159/ ProfileLinkDB/plhhblits/hhblits_link_db_080_110/ query_original_result/