This is a simple tool for similarity-based virtual screening written in Python. Given a set of known positive molecules, it can return a set of property matched neighbors from the virtual screening library.

Features

- Descriptor and Fingerprint generation for distance calculation.
- Multiple metrics for similarity assessment.
- Customized number output for each target compound.

Dependency

- Python3
- RDKit
- Scipy
- Numpy

Miniconda is recommended to config the environment. Miniconda is an open-source, cross-platform, software package manager. It supports the packaging and distribution of software components, and manages their installation inside isolated execution environments. Miniconda can be download from https://docs.conda.io/en/latest/miniconda.html.

Easy Install via Conda

Install RDKit:

conda install -c rdkit rdkit

Install Scipy:

conda install scipy

Install Numpy:

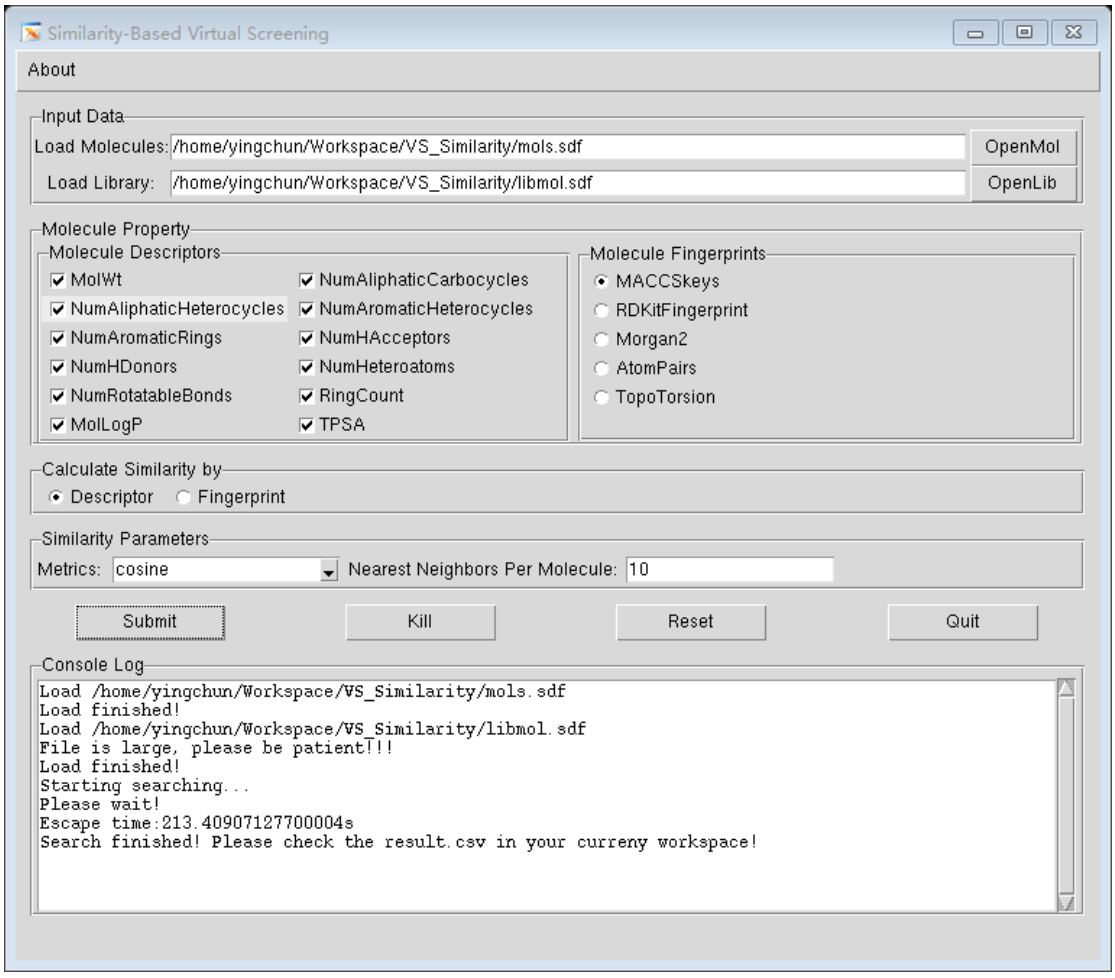
conda install Numpy

When all the dependency installed, just run python VS.py in shell in Linux or python VS.py in cmd in Windows.

To do

- Parallel calculation to accelerate screening and reduce time-consuming.

The following is a test in Linux:



The input mols.sdf has about 2000 molecules while the libmol.sdf has about 160K molecules. Because loading molecular library take 2 minute, the total time cost is about 7 minutes.

The output result was shown below:

	A	B	C	D	E	F	G	H	I	J	K
1	target	mol0	mol1	mol2	mol3	mol4	mol5	mol6	mol7	mol8	mol9
2	CC[C@H](C	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N
3	CC(C)C[C@	CC(C)CC(N	CC(C)CC(N	CC(C)CC(N	CC(C)CC(N	CC(C)CC(N	CC(C)CC(N	CC(C)CC(N	CC(C)CC(N	CC(C)CC(N	CC(C)CC(N
4	CC(C)C[C@	CCNC(=NC	CCNC(=NC	CCNC(=NC	CCNC(=NC	CCNC(=NC	CCNC(=NC	CCNC(=NC	CCNC(=NC	CCNC(=NC	CCNC(=NC
5	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC
6	CC(=O)N[C	CSCCC(NC	CSCCC(NC	CSCCC(NC	CSCCC(NC	CSCCC(NC	CSCCC(NC	CSCCC(NC	CSCCC(NC	CSCCC(NC	CSCCC(NC
7	CCCCCCCC	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N	CCC(C)C(N
8	C/C=C/C[C	C/C=C/CC	C/C=C/CC	C/C=C/CC	C/C=C/CC	C/C=C/CC	C/C=C/CC	C/C=C/CC	C/C=C/CC	C/C=C/CC	C/C=C/CC
9	C[C@@H]((	CC(O)C(CO	CC(O)C(CO	CC(O)C(CO	CC(O)C(CO	CC(O)C(CO	CC(O)C(CO	CC(O)C(CO	CC(O)C(CO	CC(O)C(CO	CC(O)C(CO
10	CC(=O)N[C	CCOC(=O)(	CCOC(=O)(	CCOC(=O)(	CCOC(=O)(	CCOC(=O)(	CCOC(=O)(	CCOC(=O)(	CCOC(=O)(	CCOC(=O)(	CCOC(=O)(
11	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO	Cc1ncc(CO
12	CC1=C2[N	CC1=C2[N	CC1=C2[N	CC1=C2[N	CC1=C2[N	CC1=C2[N	CC1=C2[N	CC1=C2[N	CC1=C2[N	CC1=C2[N	CC1=C2[N
13	N[C@@H]((	NC(Cc1cnc	NC(Cc1cnc	NC(Cc1cnc	NC(Cc1cnc	NC(Cc1cnc	NC(Cc1cnc	NC(Cc1cnc	NC(Cc1cnc	NC(Cc1cnc	NC(Cc1cnc
14	C[S+](CC[C	C[S+](CCC(	C[S+](CCC(	C[S+](CCC(	C[S+](CCC(	C[S+](CCC(	C[S+](CCC(	C[S+](CCC(	C[S+](CCC(	C[S+](CCC(	C[S+](CCC(
15	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=	CC(=O)C(=
16	N[C@@H]((	NC(Cc1ccc	NC(Cc1ccc	NC(Cc1ccc	NC(Cc1ccc	NC(Cc1ccc	NC(Cc1ccc	NC(Cc1ccc	NC(Cc1ccc	NC(Cc1ccc	NC(Cc1ccc
17	O=C(O)CC	O=C(O)CC	O=C(O)CC	O=C(O)CC	O=C(O)CC	O=C(O)CC	O=C(O)CC	O=C(O)CC	O=C(O)CC	O=C(O)CC	O=C(O)CC
18	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C	C[N+](C)(C
19	NCCCC[C@	NCCCCC(N	NCCCCC(N	NCCCCC(N	NCCCCC(N	NCCCCC(N	NCCCCC(N	NCCCCC(N	NCCCCC(N	NCCCCC(N	NCCCCC(N
20	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC	N=C(N)NC

The first column is the target molecules. The columns mol0 ~ mol9 were screened similar molecules sorted by similarity from high to low.