High Throughput Ligand Interaction Profiler

Akshay Uttarkar[1], Shreyagirish09[1], Vidya Niranjan[1]

[1]R V College of Engineering

Vidya Niranjan

FEB 23, 2023

OPEN ACCESS

DOI:
dx.doi.org/10.17504/protocols.io.4r3l27njqg1y/v1

Protocol Citation: Akshay Uttarkar, Shreyagirish09, Vidya Niranjan 2023. High Throughput Ligand Interaction Profiler . **protocols.io** https://dx.doi.org/10.17504/protocols.io.4r3l27njqg1y/v1

Protocol status: Working
We use this protocol and it's working

Created: Jan 16, 2023

Last Modified: Feb 23, 2023

PROTOCOL integer ID: 75343

ABSTRACT

High Throughput Ligand Interaction Profiler (HT-LIP) is a web-based tool that runs on Google Colab, which allows users to predict ligand-protein interactions. HT-LIP can be used to screen large chemical libraries for potential drug candidates and can also provide insights into potential ligand-protein interactions and identifies the interacting residues. HT-LIP can be useful in identifying potential binding sites and key residues for ligand-protein interactions, which can aid in drug discovery and protein-ligand interaction studies.

Prepare the Google Drive

1 In your Google Drive home directory, create a new folder called “data”. Inside this folder, add 3

files.
(a) protein: in PDB format
(b) ligand: in SDF format
(c) reference ligand: in PDB format

For example, the data folder created here, have files with the following names :
(a) protein: 5Y15.pdb
(b) ligand: compounds_for_ml.sdf
(c) reference: reference_ligand.pdb

Connect to Google Drive

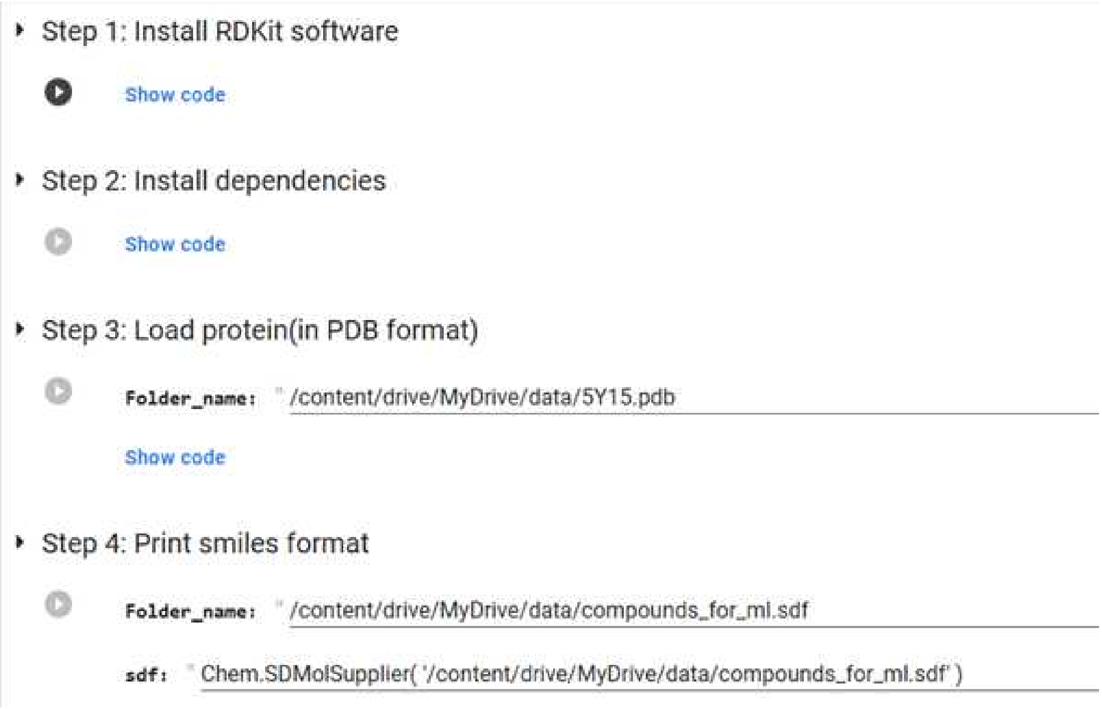
2 **Click on the URL provided below to get started.**

URL: Google Colab Notebook

Now, you will be directed towards the Google Colab Notebook. Towards the LHS side of the notebook, from the “files” section, click on the “mount drive” option. Then, a cell will be created, run the cell. Then click on “Run Anyway” and select “Connect to Google Drive” and give access. Now from the “drive” option, check whether the “data” folder created in the above step is seen in “MyDrive”.

Run the cells

3 Below are the seven steps that need to be followed in order to get the Protein-Ligands Interactions.



▸ Step 1: Install RDKit software

▶ Show code

▸ Step 2: Install dependencies

▶ Show code

▸ Step 3: Load protein(in PDB format)

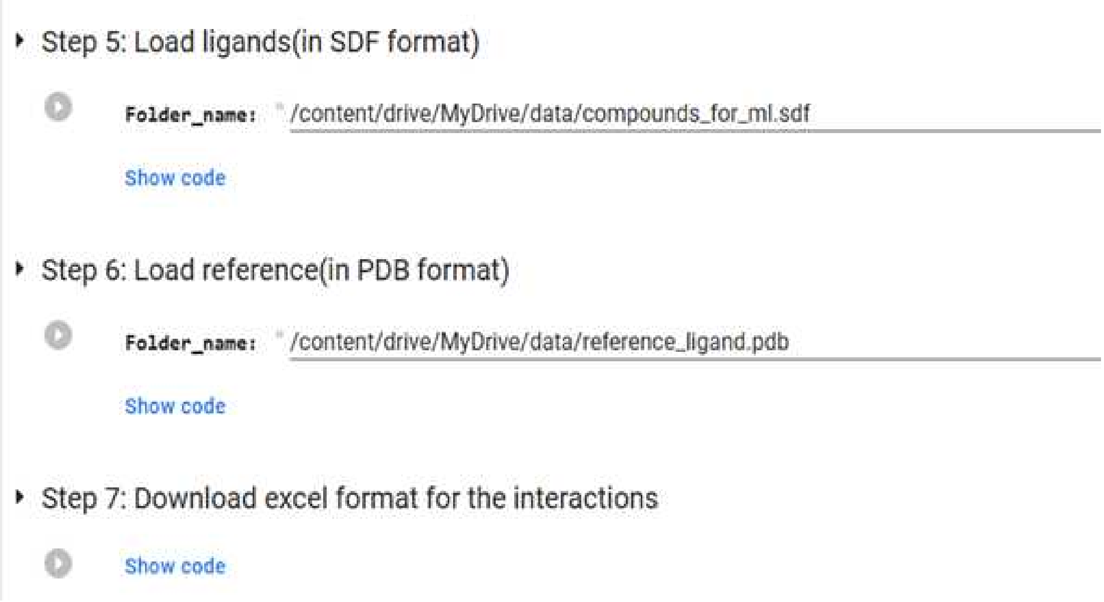
▶ `Folder_name:` " /content/drive/MyDrive/data/5Y15.pdb

Show code

▸ Step 4: Print smiles format

▶ `Folder_name:` " /content/drive/MyDrive/data/compounds_for_ml.sdf

`sdf:` " Chem.SDMolSupplier('/content/drive/MyDrive/data/compounds_for_ml.sdf')



▸ Step 5: Load ligands(in SDF format)

▶ `Folder_name:` " /content/drive/MyDrive/data/compounds_for_ml.sdf

Show code

▸ Step 6: Load reference(in PDB format)

▶ `Folder_name:` " /content/drive/MyDrive/data/reference_ligand.pdb

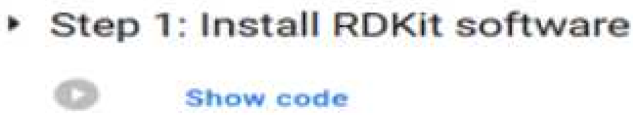
Show code

▸ Step 7: Download excel format for the interactions

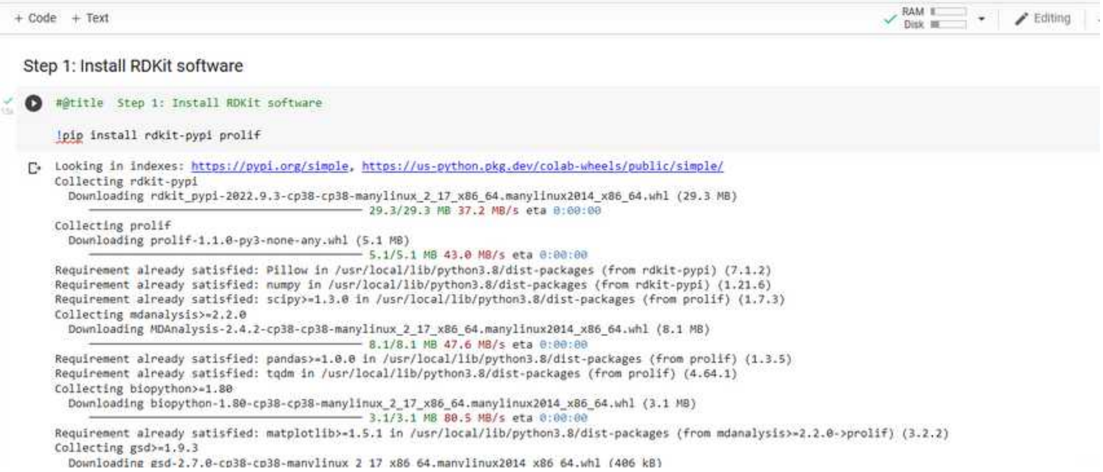
▶ Show code

Step 1: Install RDKit software

RDKit is an open source Chemoinformatics & Machine Learning Software. It provides tools for different kinds of similarity search. To install RDKit, click on the run cell option beside the show code(in blue).



In a few seconds RDKit will be installed.



Step 1: Install RDKit software

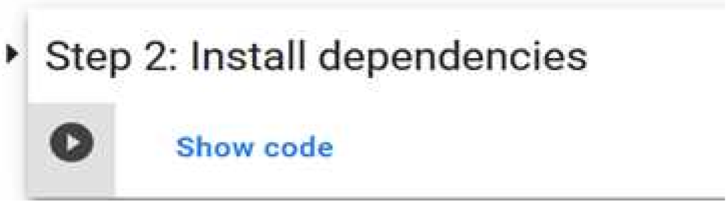
```
#@title  Step 1: Install RDKit software

!pip install rdkit-pypi prolif
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting rdkit-pypi
  Downloading rdkit_pypi-2022.9.3-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (29.3 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 29.3/29.3 MB 37.2 MB/s eta 0:00:00
Collecting prolif
  Downloading prolif-1.1.0-py3-none-any.whl (5.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 5.1/5.1 MB 43.0 MB/s eta 0:00:00
Requirement already satisfied: Pillow in /usr/local/lib/python3.8/dist-packages (from rdkit-pypi) (7.1.2)
Requirement already satisfied: numpy in /usr/local/lib/python3.8/dist-packages (from rdkit-pypi) (1.21.6)
Requirement already satisfied: scipy>=1.3.0 in /usr/local/lib/python3.8/dist-packages (from prolif) (1.7.3)
Collecting mdanalysis>=2.2.0
  Downloading MDAnalysis-2.4.2-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (8.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.1/8.1 MB 47.6 MB/s eta 0:00:00
Requirement already satisfied: pandas>=1.0.0 in /usr/local/lib/python3.8/dist-packages (from prolif) (1.3.5)
Requirement already satisfied: tqdm in /usr/local/lib/python3.8/dist-packages (from prolif) (4.64.1)
Collecting biopython>=1.80
  Downloading biopython-1.80-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.1/3.1 MB 80.5 MB/s eta 0:00:00
Requirement already satisfied: matplotlib>=1.5.1 in /usr/local/lib/python3.8/dist-packages (from mdanalysis>=2.2.0->prolif) (3.2.2)
Collecting gsd>=1.9.3
  Downloading gsd-2.7.0-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (406 kB)
```

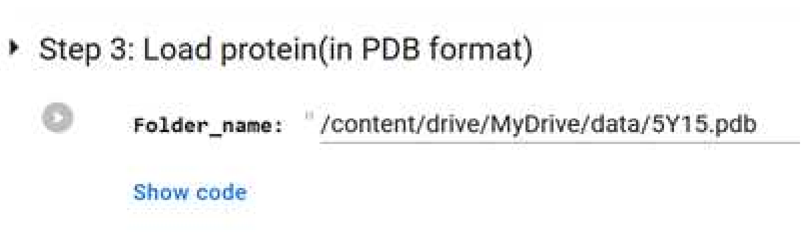
Step 2: Install dependencies

Click on the run cell option beside the show code(in blue) to install the dependencies(libraries). The required libraries will be installed in a few seconds.



Step 3: Load protein(in PDB format)

The protein used here is 5Y15. To upload your protein of interest, change “5Y15” to that protein name in the “folder_name” section. Now run the cell.



Step 3: Load protein(in PDB format)

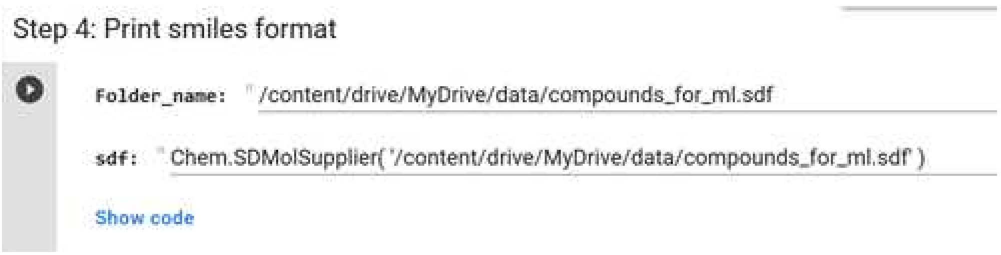
Folder_name: "/content/drive/MyDrive/data/5Y15.pdb

Show code

Step 4: Print smiles format

Change “compounds_for_ml” to your ligand name in the “folder_name” and "sdf" section. Now run the cell.

Here ligands are converted from SDF to SMILES format.



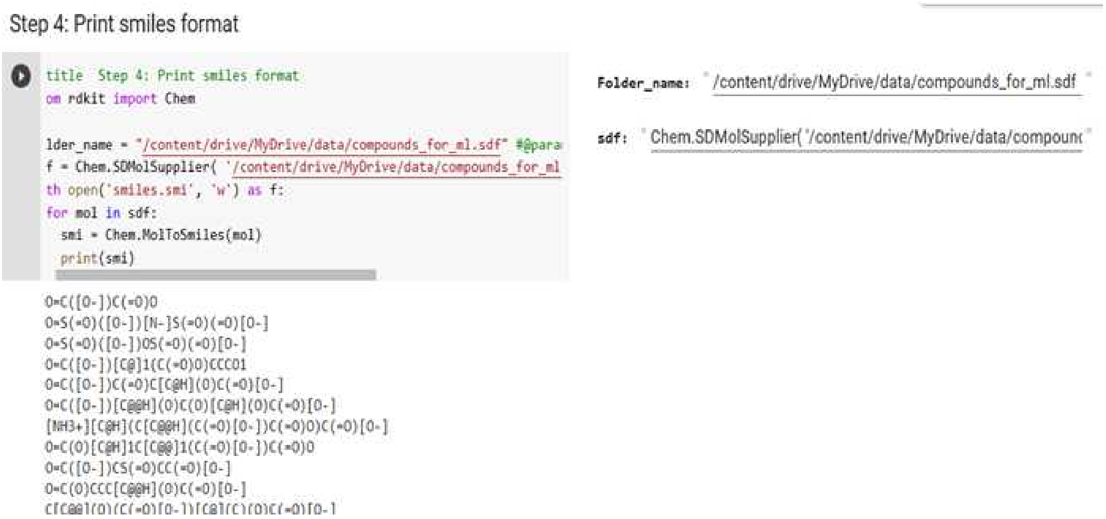
Step 4: Print smiles format

Folder_name: "/content/drive/MyDrive/data/compounds_for_ml.sdf

sdf: "Chem.SDMolSupplier('/content/drive/MyDrive/data/compounds_for_ml.sdf')

Show code

The smiles format will be printed.



Step 4: Print smiles format

```
title  Step 4: Print smiles format
om rdkit import Chem

lder_name = "/content/drive/MyDrive/data/compounds_for_ml.sdf" #@para
f = Chem.SDMolSupplier( '/content/drive/MyDrive/data/compounds_for_ml
th open('smiles.smi', 'w') as f:
for mol in sdf:
  smi = Chem.MolToSmiles(mol)
  print(smi)
```

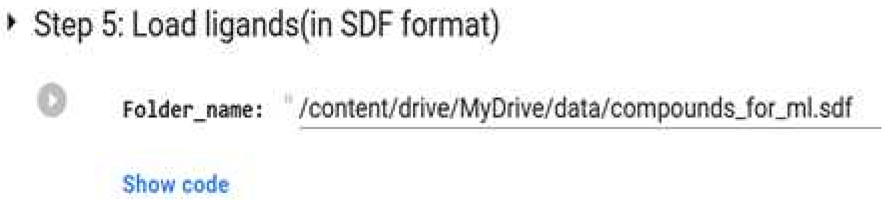
Folder_name: "/content/drive/MyDrive/data/compounds_for_ml.sdf "

sdf: "Chem.SDMolSupplier('/content/drive/MyDrive/data/compounc "

```
O=C([O-])C(=O)O
O=S(=O)([O-])[N-]S(=O)(=O)[O-]
O=S(=O)([O-])OS(=O)(=O)[O-]
O=C([O-])[C@]1(C(=O)O)CCCO1
O=C([O-])C(=O)C[C@H](O)C(=O)[O-]
O=C([O-])[C@@H](O)C(O)[C@H](O)C(=O)[O-]
[NH3+][C@H](C[C@@H](C(=O)[O-])C(=O)O)C(=O)[O-]
O=C(O)[C@H]1C[C@@]1(C(=O)[O-])C(=O)O
O=C([O-])CS(=O)CC(=O)[O-]
O=C(O)CCC[C@@H](O)C(=O)[O-]
C[C@@](O)(C(=O)[O-])[C@](C)(O)C(=O)[O-]
```

Step 5: Load ligands(in SDF format)

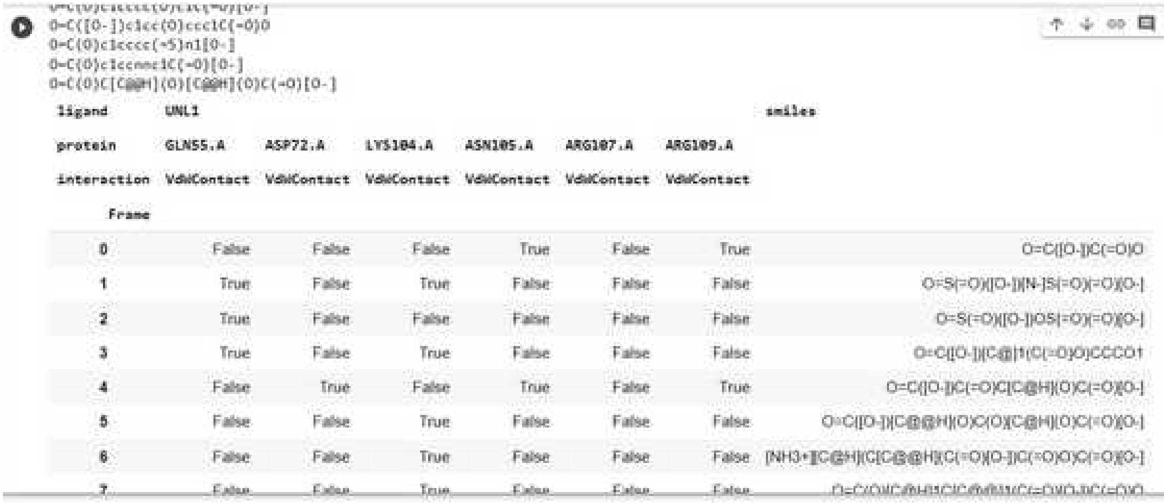
Change “compounds_for_ml” to your ligand name in the “folder_name” section(same as step 4). Run the cell.



▸ Step 5: Load ligands(in SDF format)

Folder_name: " /content/drive/MyDrive/data/compounds_for_ml.sdf

Following output will be obtained.

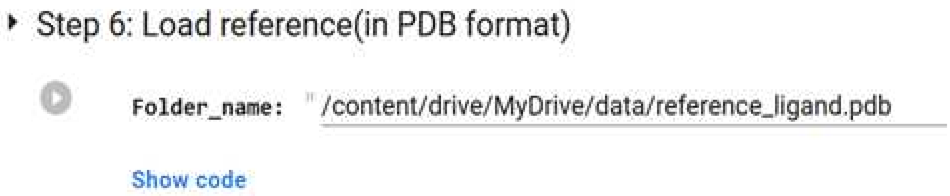


```
O=C([O-])c1cc(O)ccc1C(=O)O
O=C(O)c1cccc(=S)n1[O-]
O=C(O)c1ccnnc1C(=O)[O-]
O=C(O)C[C@@H](O)[C@@H](O)C(=O)[O-]
```

ligand	UNL1						smiles
protein	GLN55.A	ASP72.A	LYS104.A	ASN105.A	ARG107.A	ARG109.A	
interaction	VdWContact	VdWContact	VdWContact	VdWContact	VdWContact	VdWContact	
Frame							
0	False	False	False	True	False	True	O=C([O-])C(=O)O
1	True	False	True	False	False	False	O=S(=O)([O-])[N-]S(=O)(=O)[O-]
2	True	False	False	False	False	False	O=S(=O)([O-])OS(=O)(=O)[O-]
3	True	False	True	False	False	False	O=C([O-])[C@]1(C(=O)O)CCCO1
4	False	True	False	True	False	True	O=C([O-])C(=O)C[C@H](O)C(=O)[O-]
5	False	False	True	False	False	False	O=C([O-])[C@@H](O)C(O)[C@H](O)C(=O)[O-]
6	False	False	True	False	False	False	[NH3+][C@H](C[C@@H](C(=O)[O-])C(=O)O)C(=O)[O-]
7	False	False	True	False	False	False	O=C(O)[C@H]1C[C@@]1(C(=O)[O-])C(=O)O

Step 6: Load reference(in PDB format)

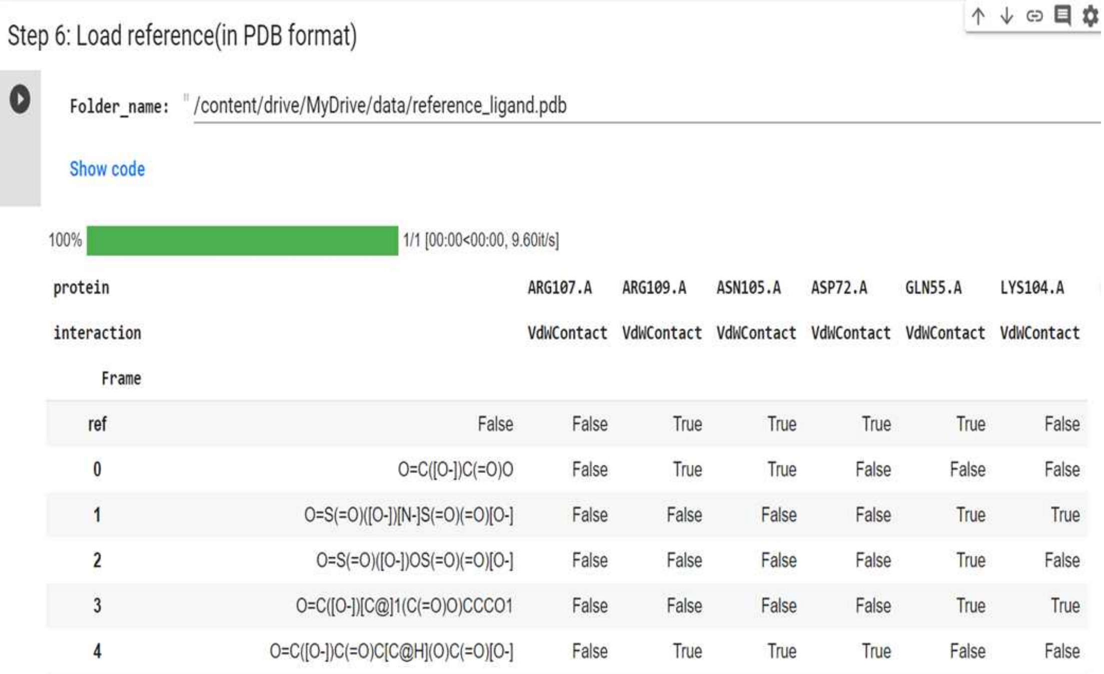
Change “reference_ligand” to your ligand name in the “folder_name” section. Now run the cell.



▸ Step 6: Load reference(in PDB format)

Folder_name: " /content/drive/MyDrive/data/reference_ligand.pdb

Following output will be obtained.



Step 6: Load reference(in PDB format)

Folder_name: "/content/drive/MyDrive/data/reference_ligand.pdb

100% 1/1 [00:00<00:00, 9.60it/s]

protein		ARG107.A	ARG109.A	ASN105.A	ASP72.A	GLN55.A	LYS104.A
interaction		VdWContact	VdWContact	VdWContact	VdWContact	VdWContact	VdWContact
Frame							
ref	False	False	True	True	True	True	False
0	O=C([O-])C(=O)O	False	True	True	False	False	False
1	O=S(=O)([O-])[N-]S(=O)(=O)[O-]	False	False	False	False	True	True
2	O=S(=O)([O-])OS(=O)(=O)[O-]	False	False	False	False	True	False
3	O=C([O-])[C@]1(C(=O)O)CCCO1	False	False	False	False	True	True
4	O=C([O-])C(=O)C[C@H](O)C(=O)[O-]	False	True	True	True	False	False

Step 7: Download excel format for the interactions

Download the excel file by running the cell. The downloaded excel file will be saved in the file section(LHS of colab notebook).

Step 7: Download excel format for the interactions

CITATION

Bouysset C, Fiorucci S (2021). ProLIF: a library to encode molecular interactions as fingerprints.. Journal of cheminformatics.

LINK

https://doi.org/10.1186/s13321-021-00548-6

. J Cheminform**13**, 72 (2021). https://doi.org/10.1186/s13321-021-00548-6