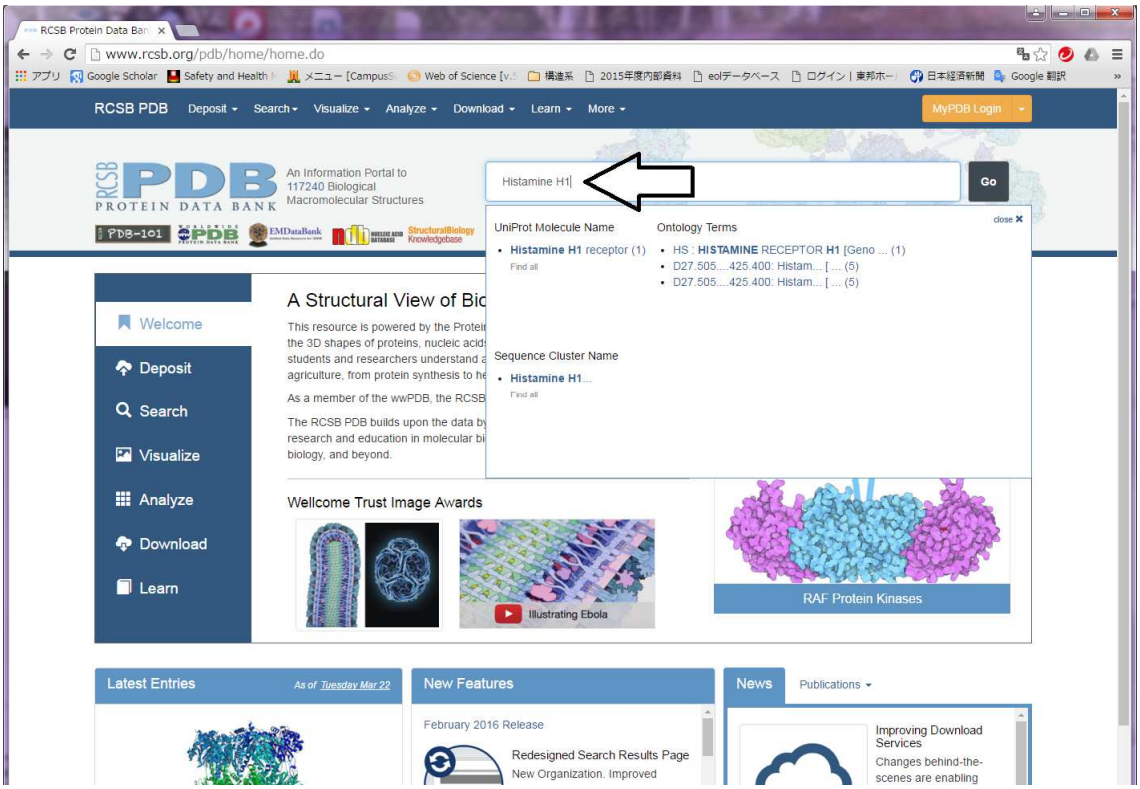
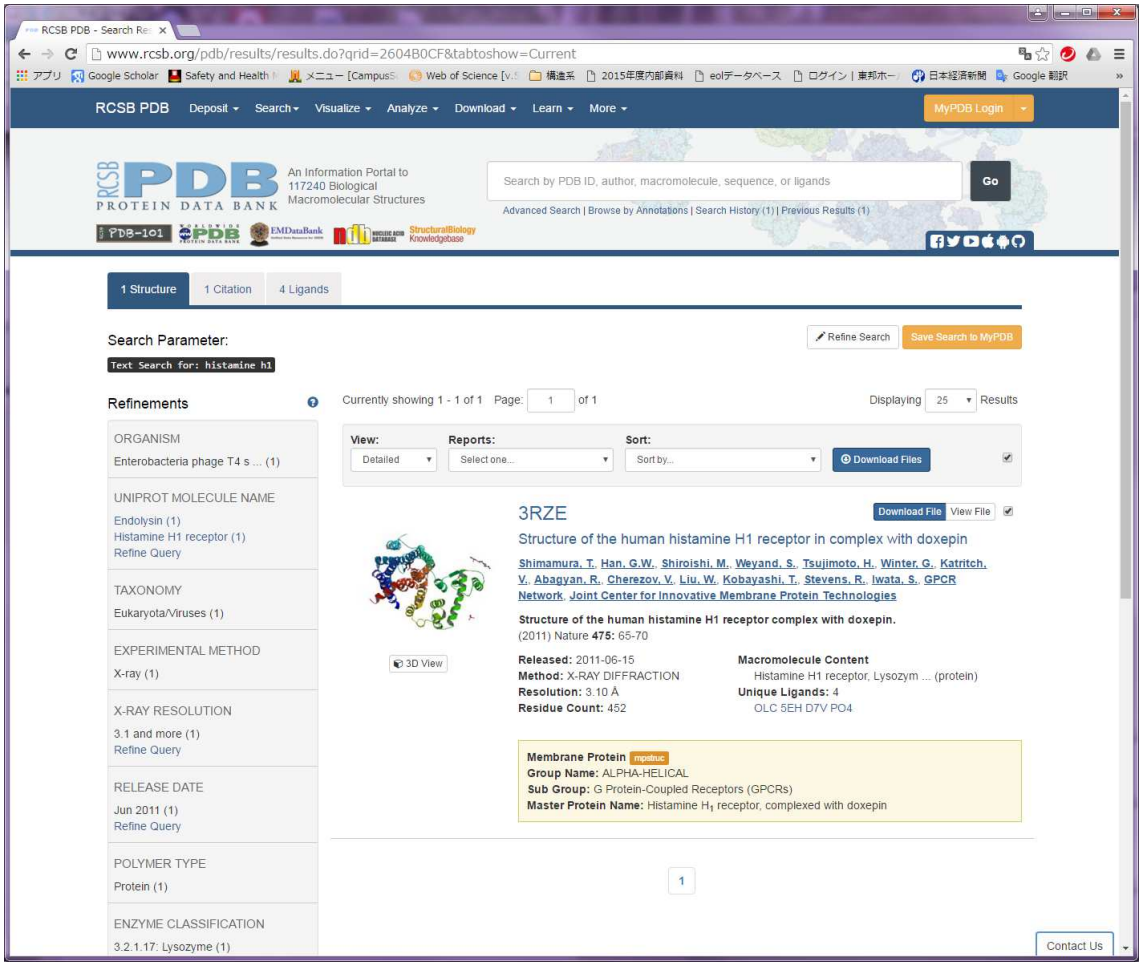
マニュアルはネット上にも他の人のがあるので探してみてください。

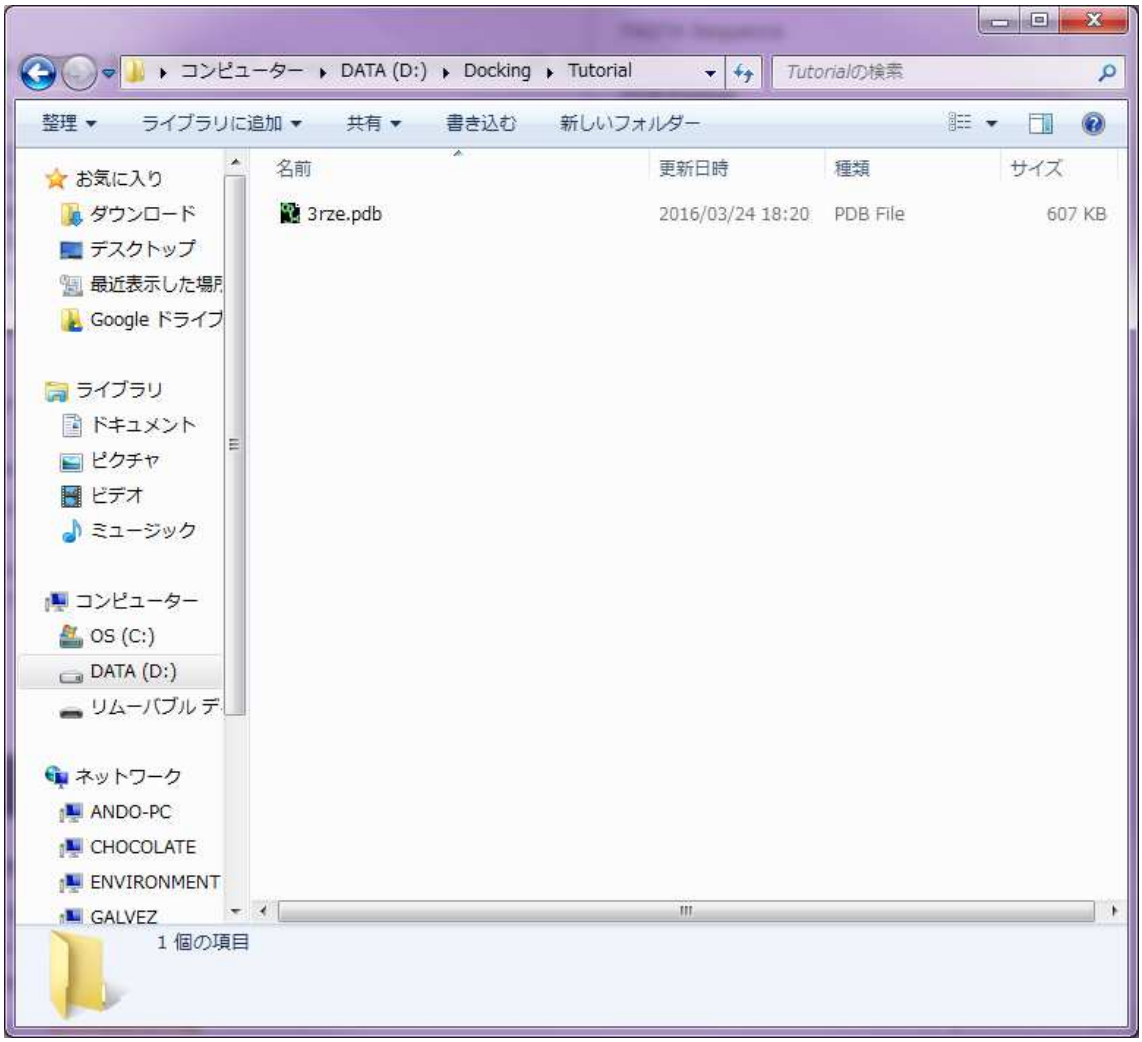
１．タンパク質構造データの取得

http://www.rcsb.org/pdb/home/home.do へ行き、目的のタンパク質構造で検索する。

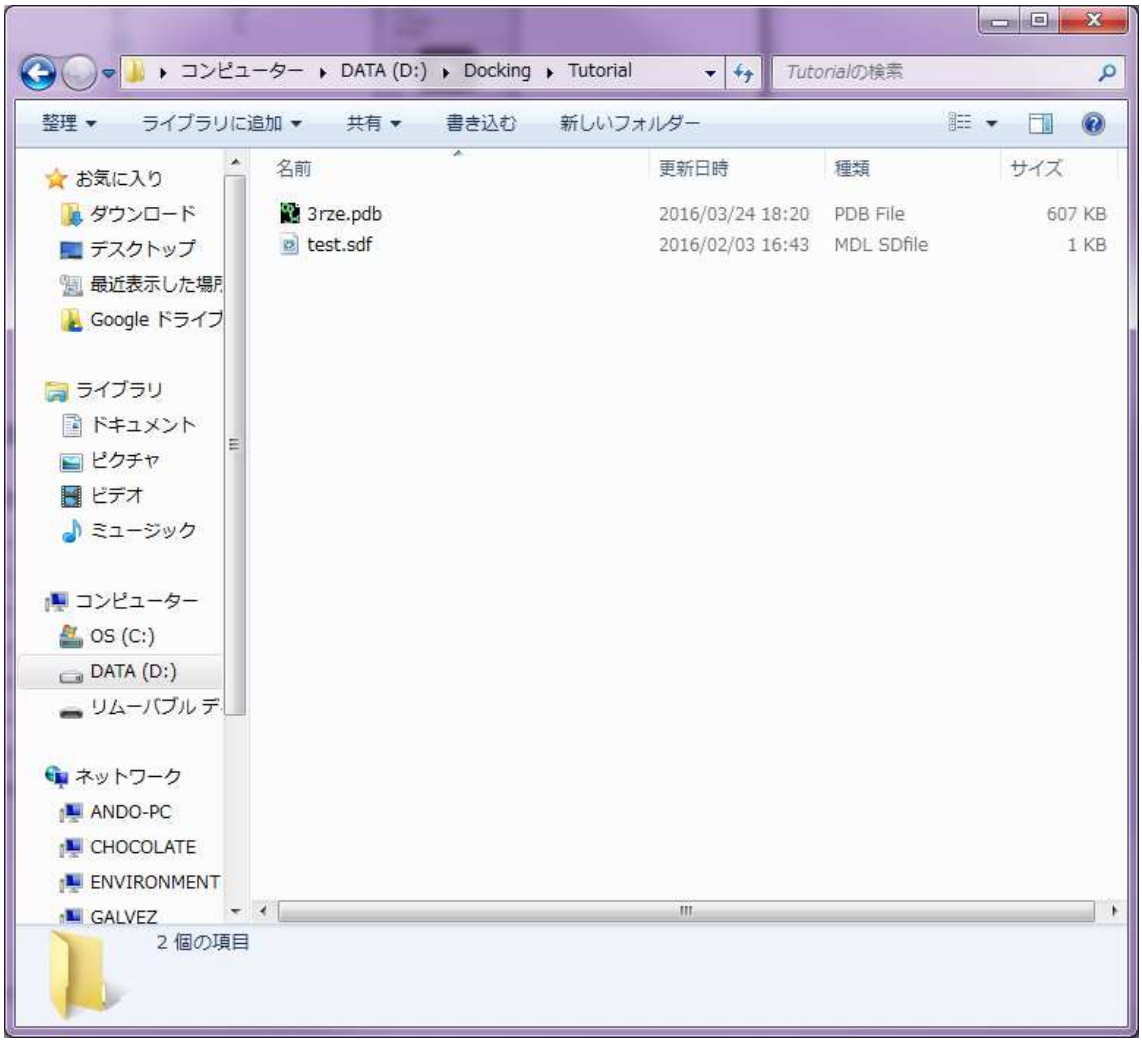




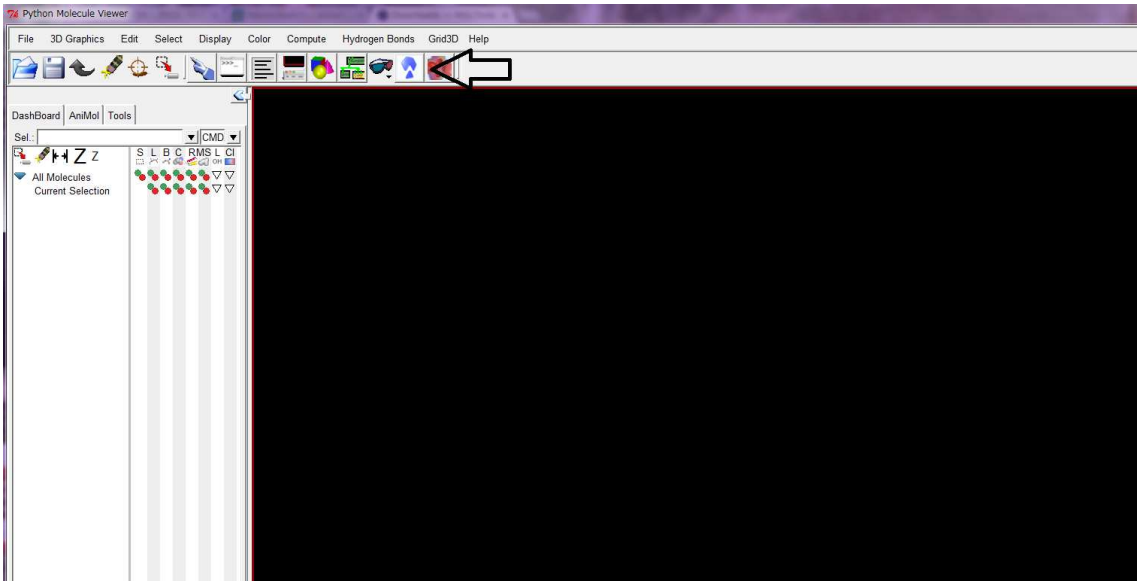
PDB Format を選択しダウンロードする。



２．ドッキングしたい構造を用意する。

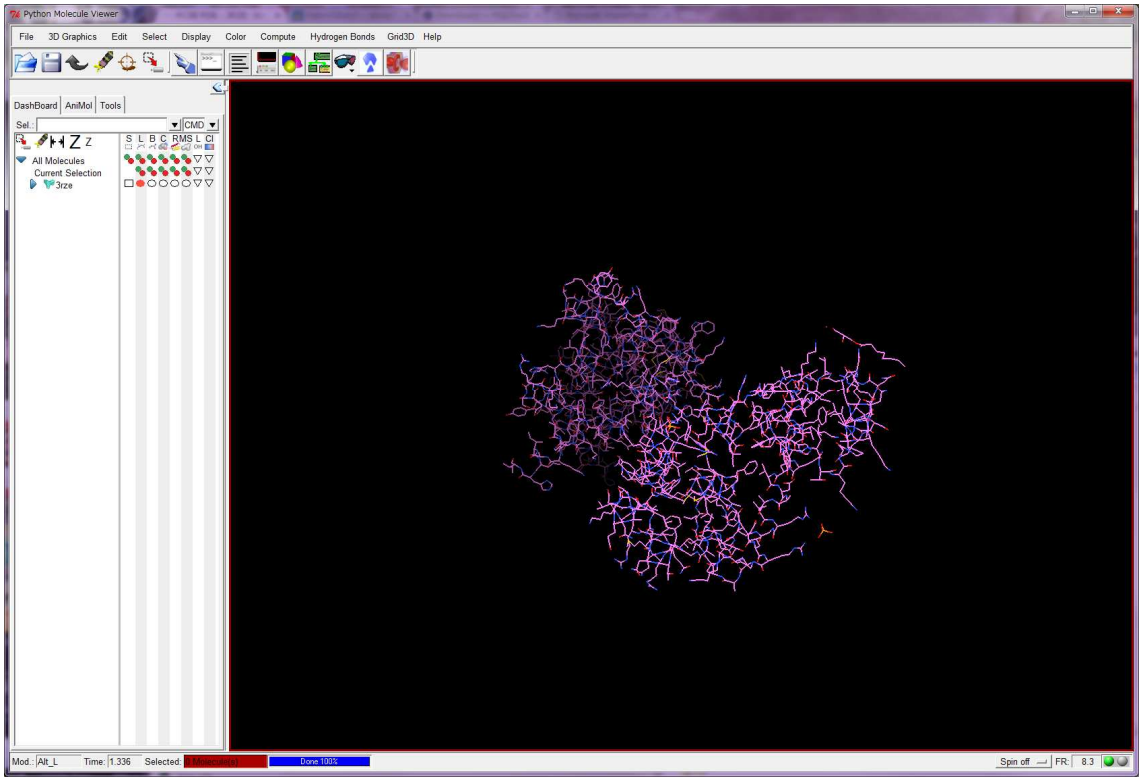


３．Python Molecule Viewer (http://mgltools.scripps.edu/downloads)を開く

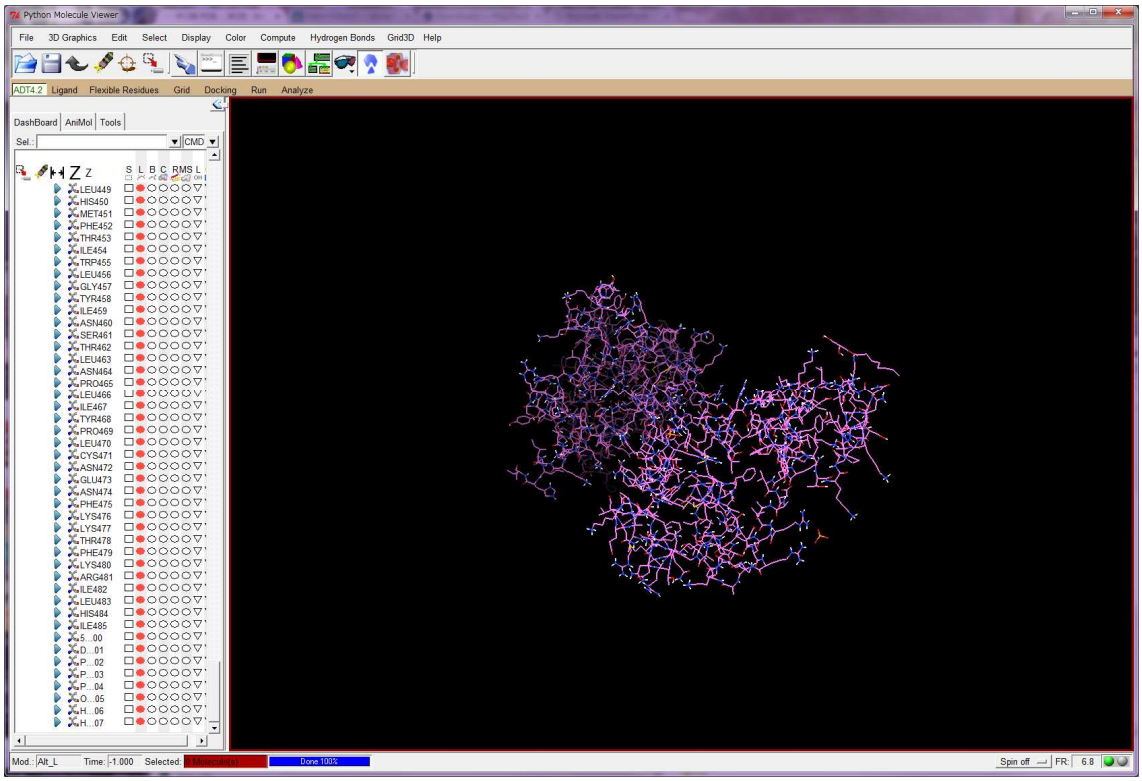


Auto Dock Tools(http://mgltools.scripps.edu/downloads)のボタンを押す

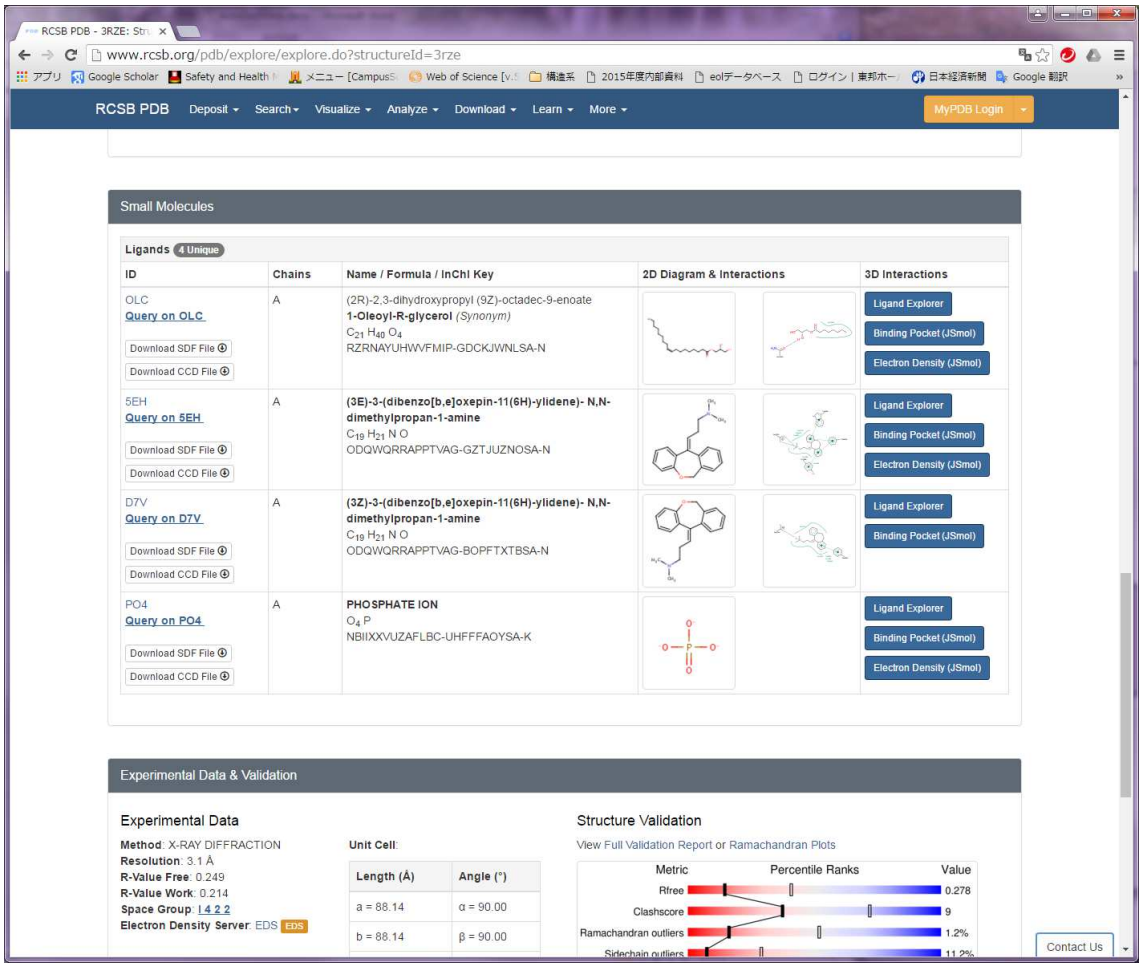
ダウンロードしたタンパク質を開く。



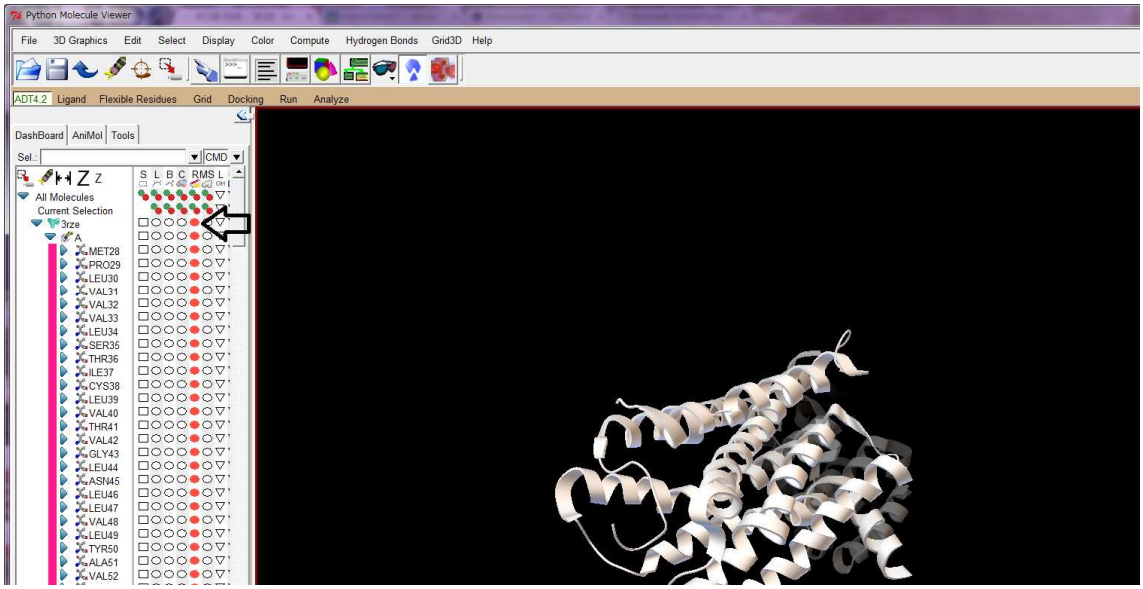
Edit>Hydrogens>add で水素を付加する。

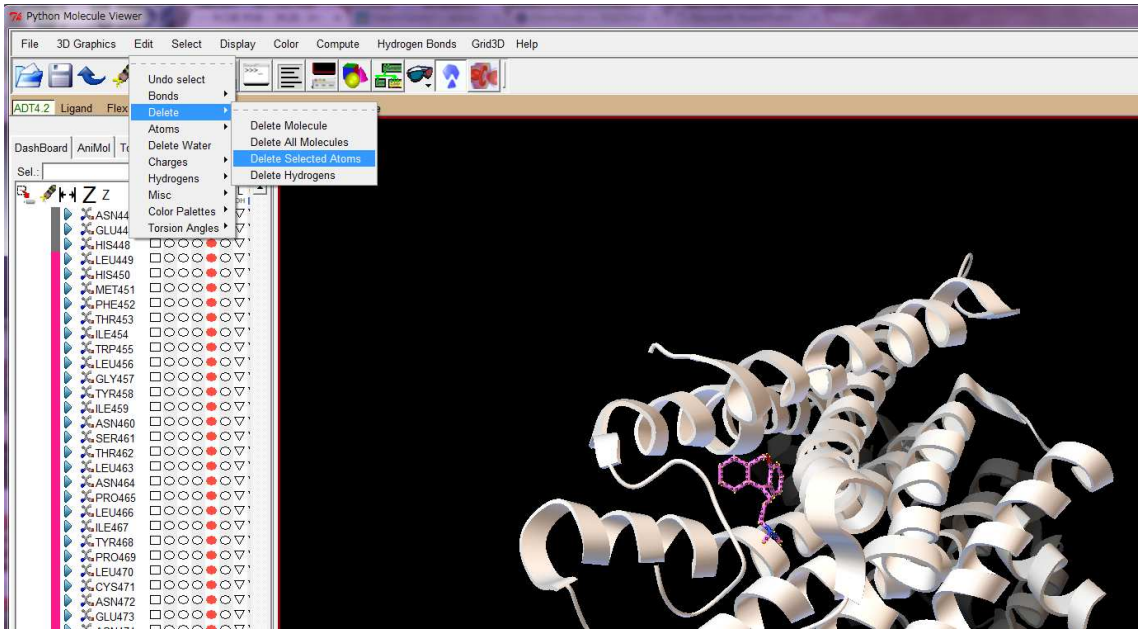


邪魔なリガンドは選択して削除する。この場合 5EH,D7V,PO4,OLC。タンパク質データをダウンロードした際のページで確認が可能。

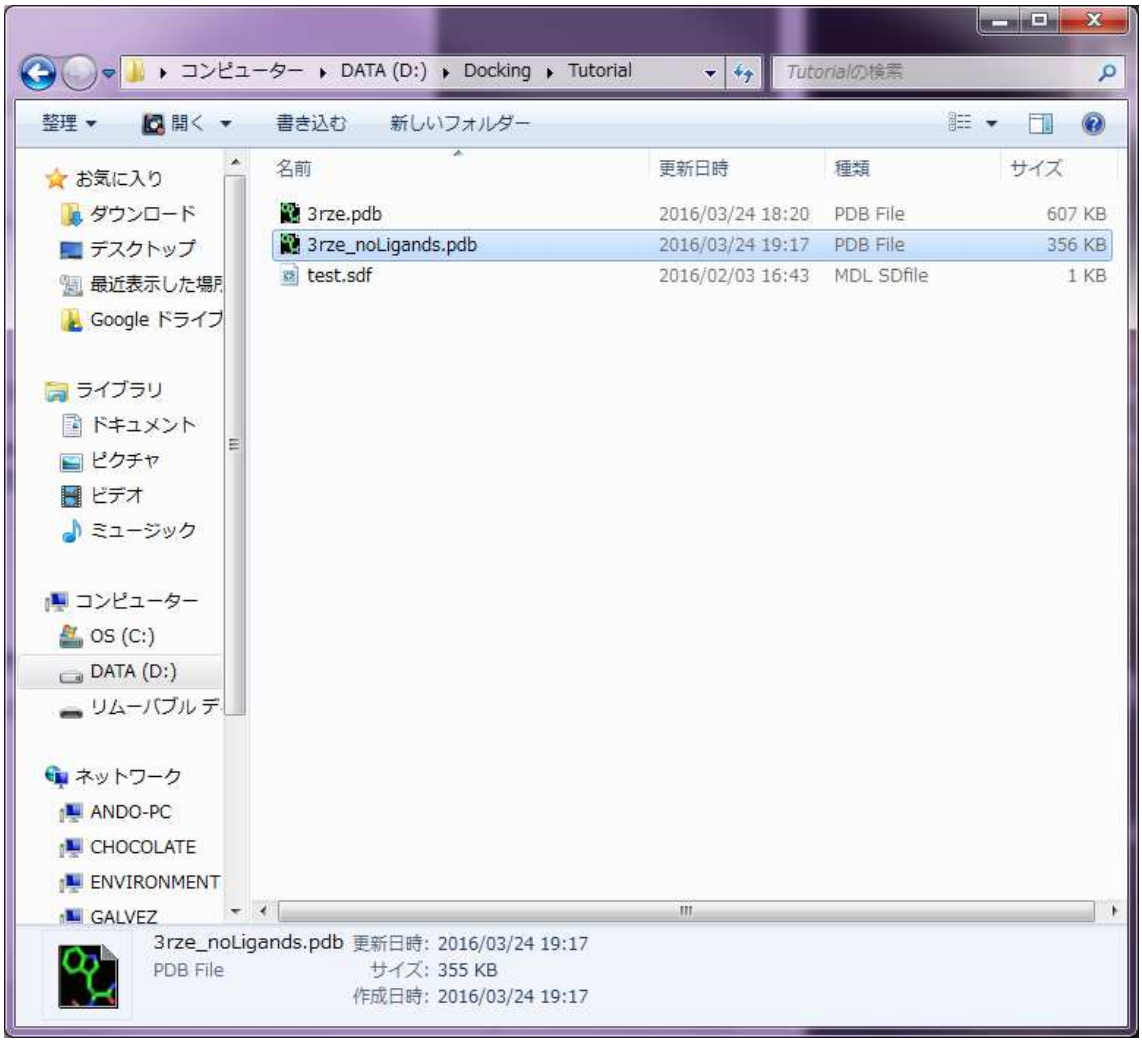


リボン表記にしておくと見やすい。

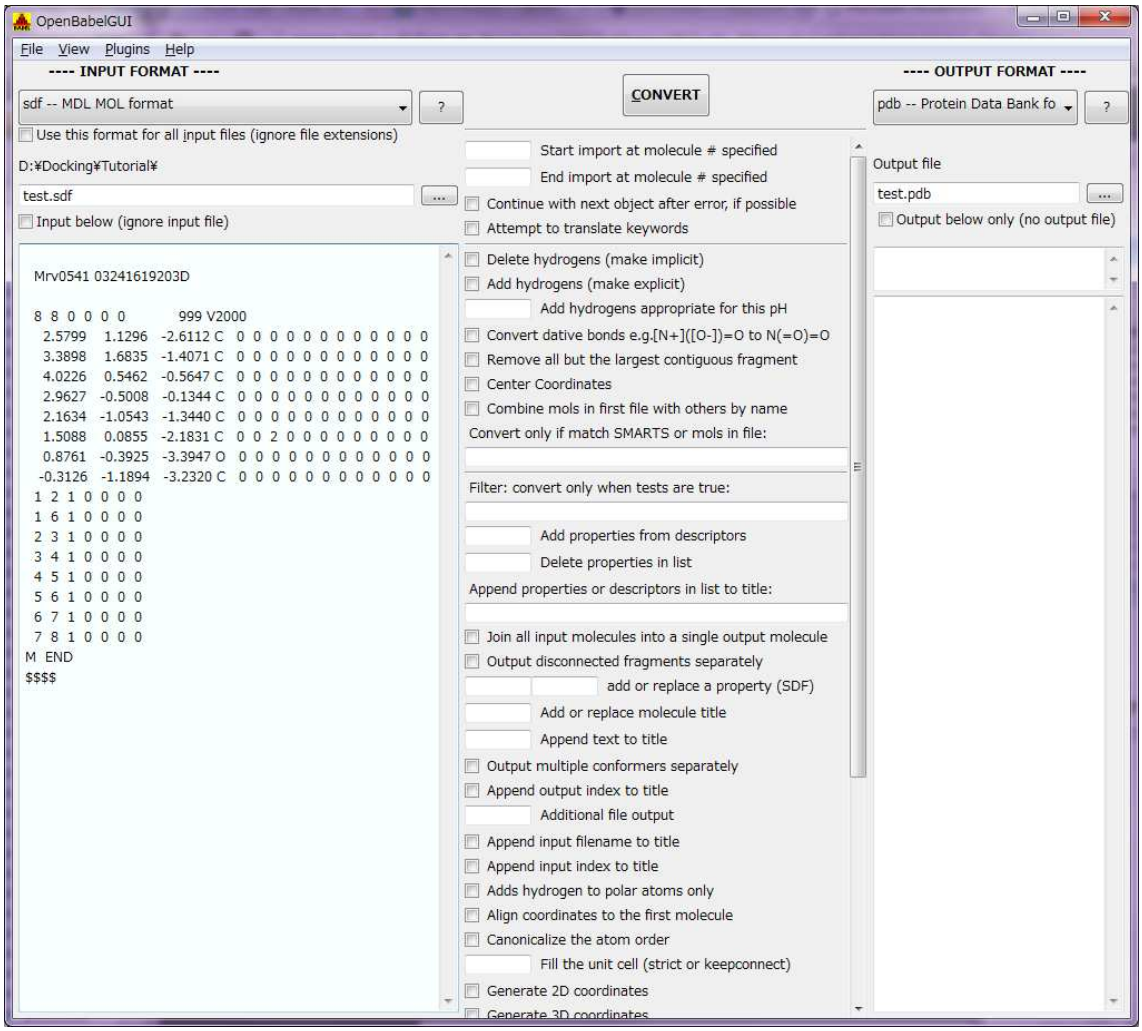


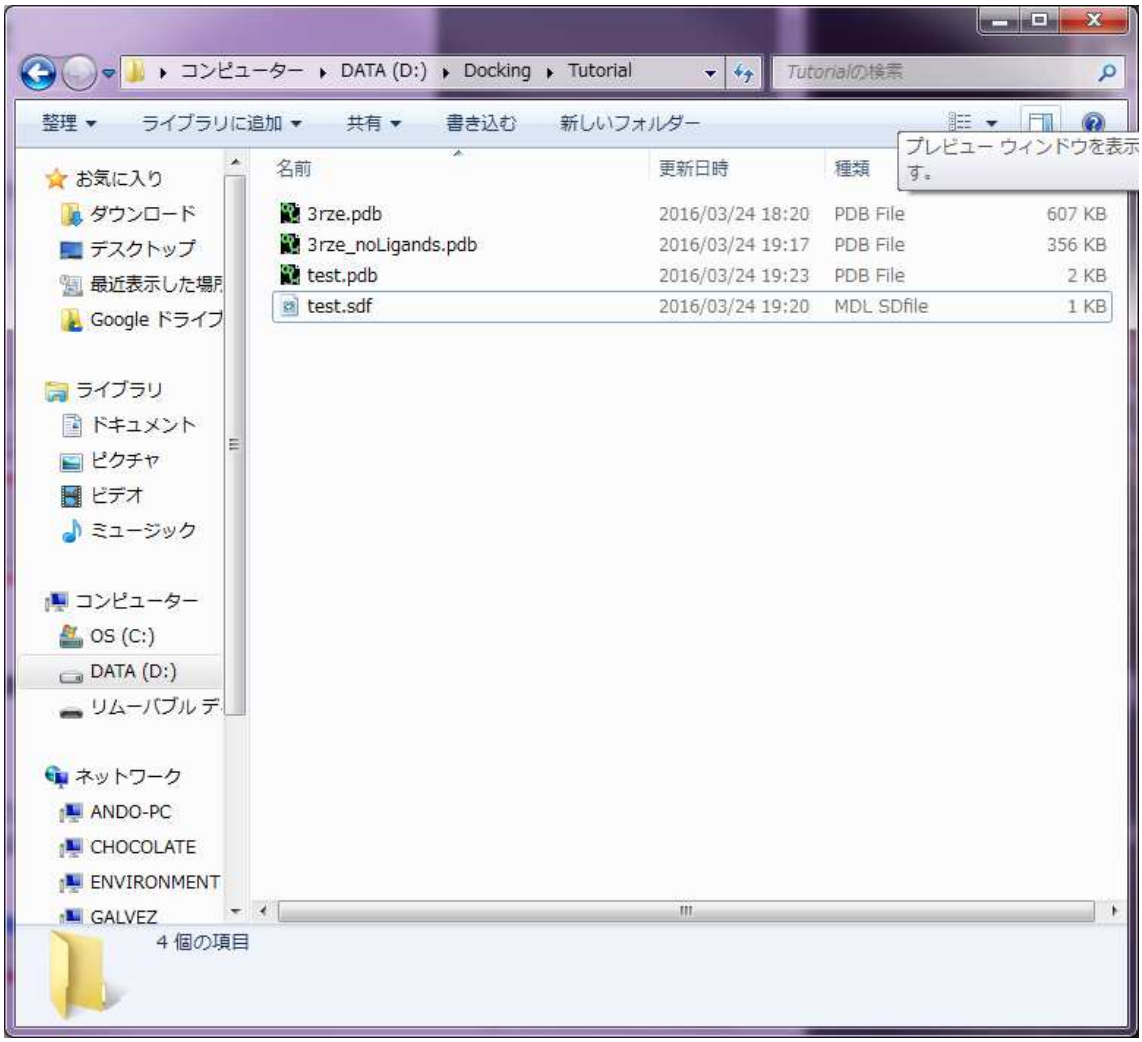

Python Molecule Viewer
File
3D Graphics
Edit
Select
Display
Color
Compute
Hydrogen Bonds
Grid3D
Help
Undo select
Bonds
Delete
Atoms
Delete Water
Charges
Hydrogens
Misc
Color Palettes
Torsion Angles
Delete Molecule
Delete All Molecules
Delete Selected Atoms
Delete Hydrogens
ADT4.2
Ligand
DashBoard
AniMol
Sel.:

削除したら保存する。


コンピューター
DATA (D:)
Docking
Tutorial
Tutorialの検索
整理
開く
書き込む
新しいフォルダー
お気に入り
ダウンロード
デスクトップ
最近表示した場所
Google ドライブ
ライブラリ
ドキュメント
ピクチャ
ビデオ
ミュージック
コンピューター
OS (C:)
DATA (D:)
リムーバブル ディスク
ネットワーク
ANDO-PC
CHOCOLATE
ENVIRONMENT
GALVEZ
名前
更新日時
種類
サイズ
3rze.pdb 2016/03/24 18:20 PDB File 607 KB
3rze_noLigands.pdb 2016/03/24 19:17 PDB File 356 KB
test.sdf 2016/02/03 16:43 MDL SDfile 1 KB
3rze_noLigands.pdb 更新日時: 2016/03/24 19:17
PDB File
サイズ: 355 KB
作成日時: 2016/03/24 19:17

4．ドッキングしたい構造を Marvin sketch (https://www.chemaxon.com/products/marvin/marvinsketch/)で三次元配座に変える。さらに OpenBabel(http://openbabel.org/wiki/Main_Page)で pdb 形式に変換する。



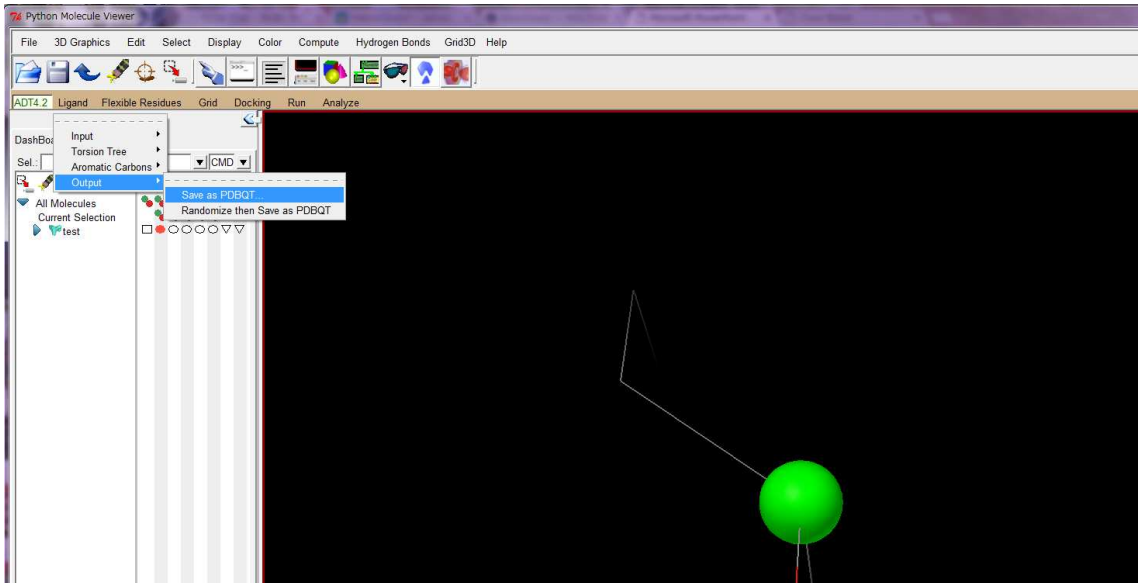


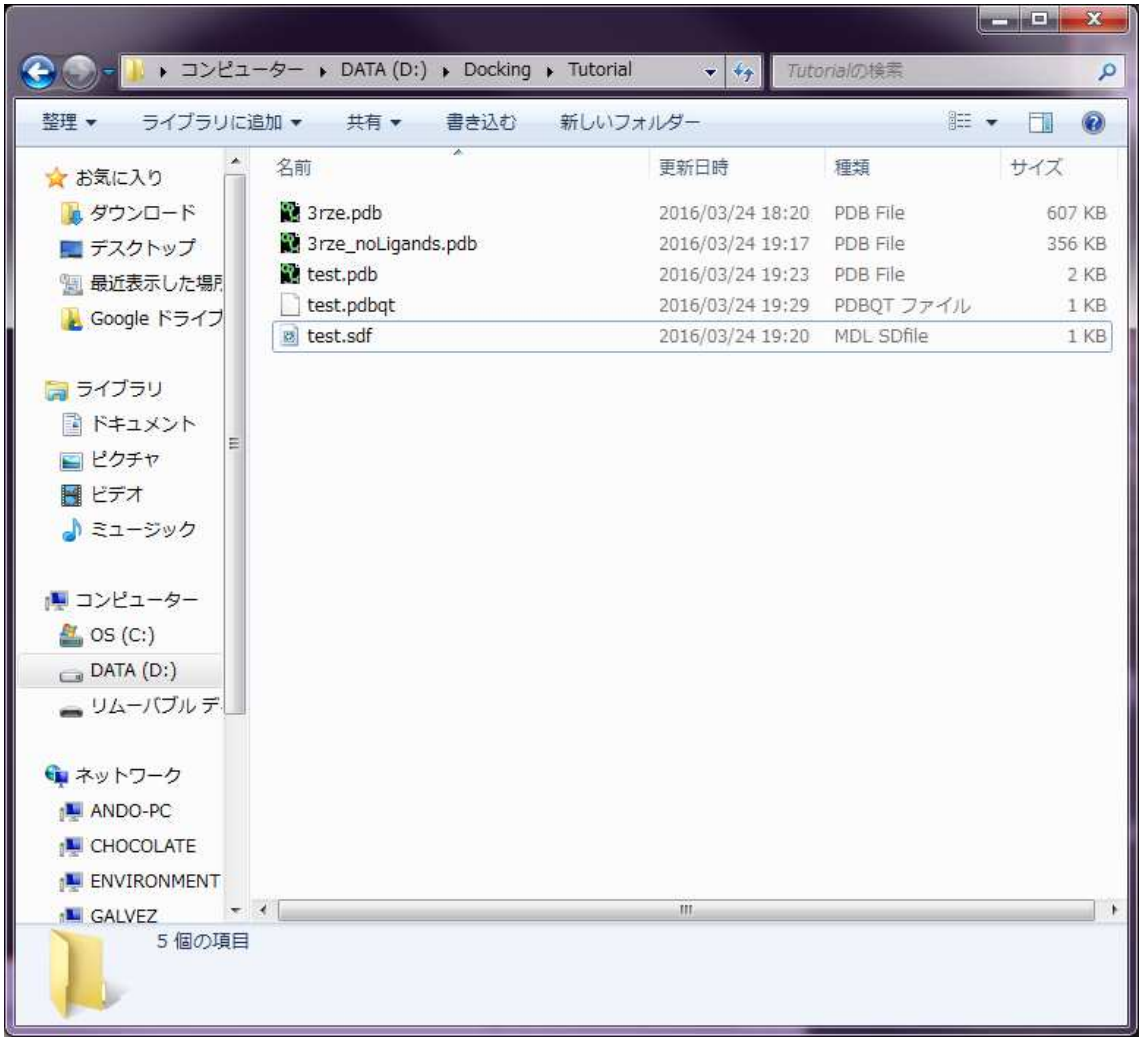
５．ドッキングしたい構造を Python Molecule Viewer で読み込み pdbqt 形式に変換する。

Ligand>Input>Open で読み込み

Ligand>TorsionTree>DetectRoot を選択

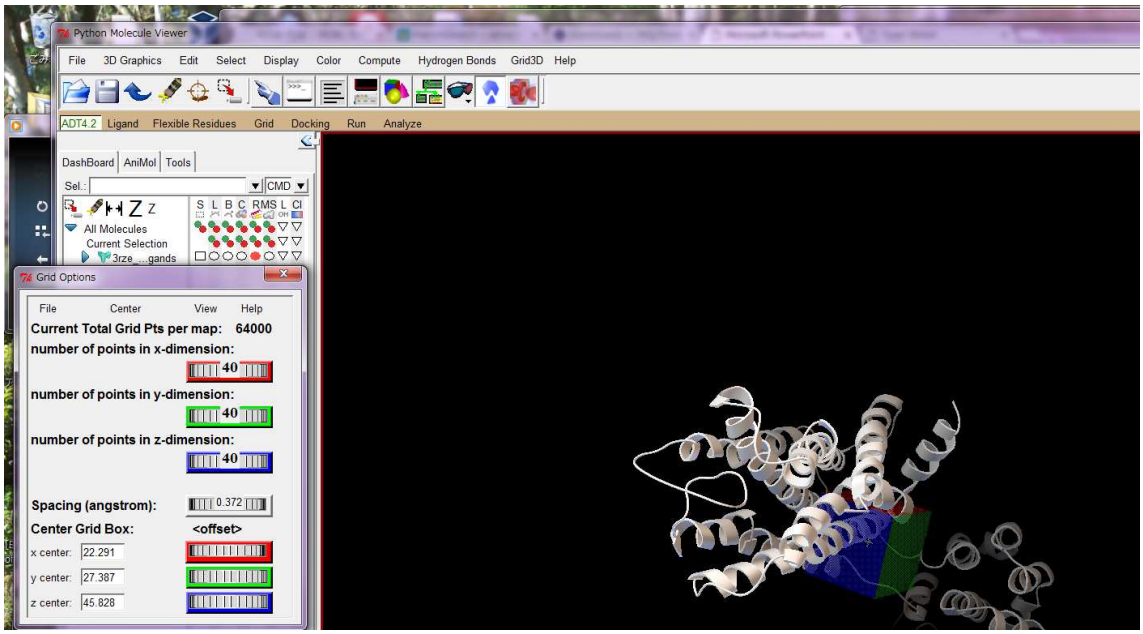
Pdbqt 形式で保存



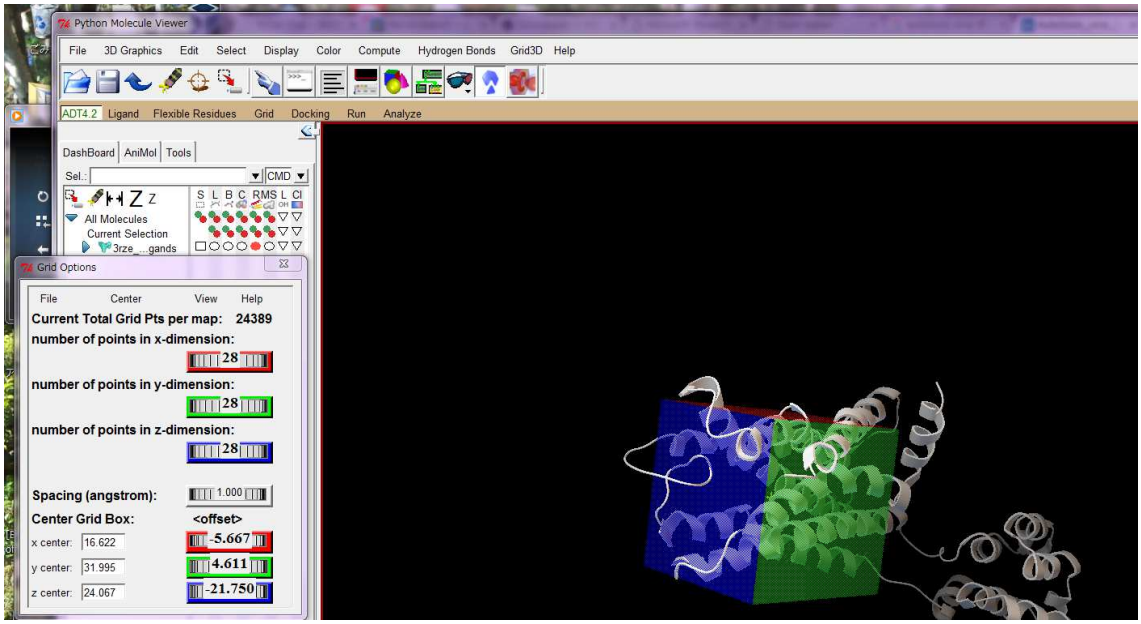


６．タンパク質データを読み込み pdbqt 形式で保存する。

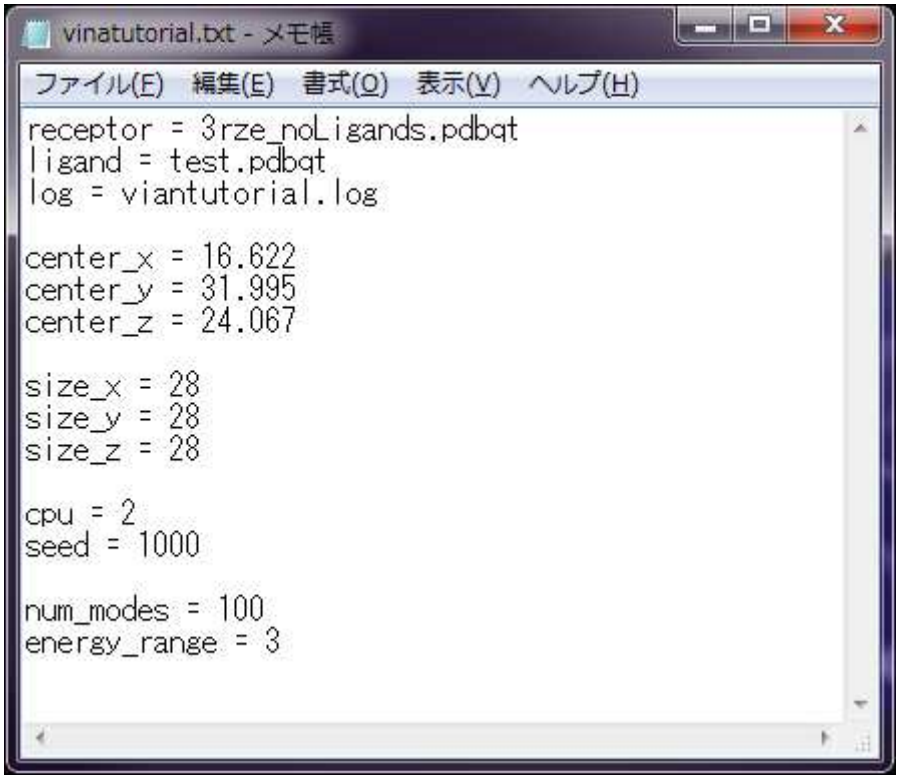
Grid>Macromolecule>Open で pdb ファイルを選択すると自動的に作成される。
表示をリボン形式にしてグリッドを設定する。設定範囲はもともとリガンドが存在したあたりにするといい。
Grid>GridBox を選択するとウィンドウが現れる。



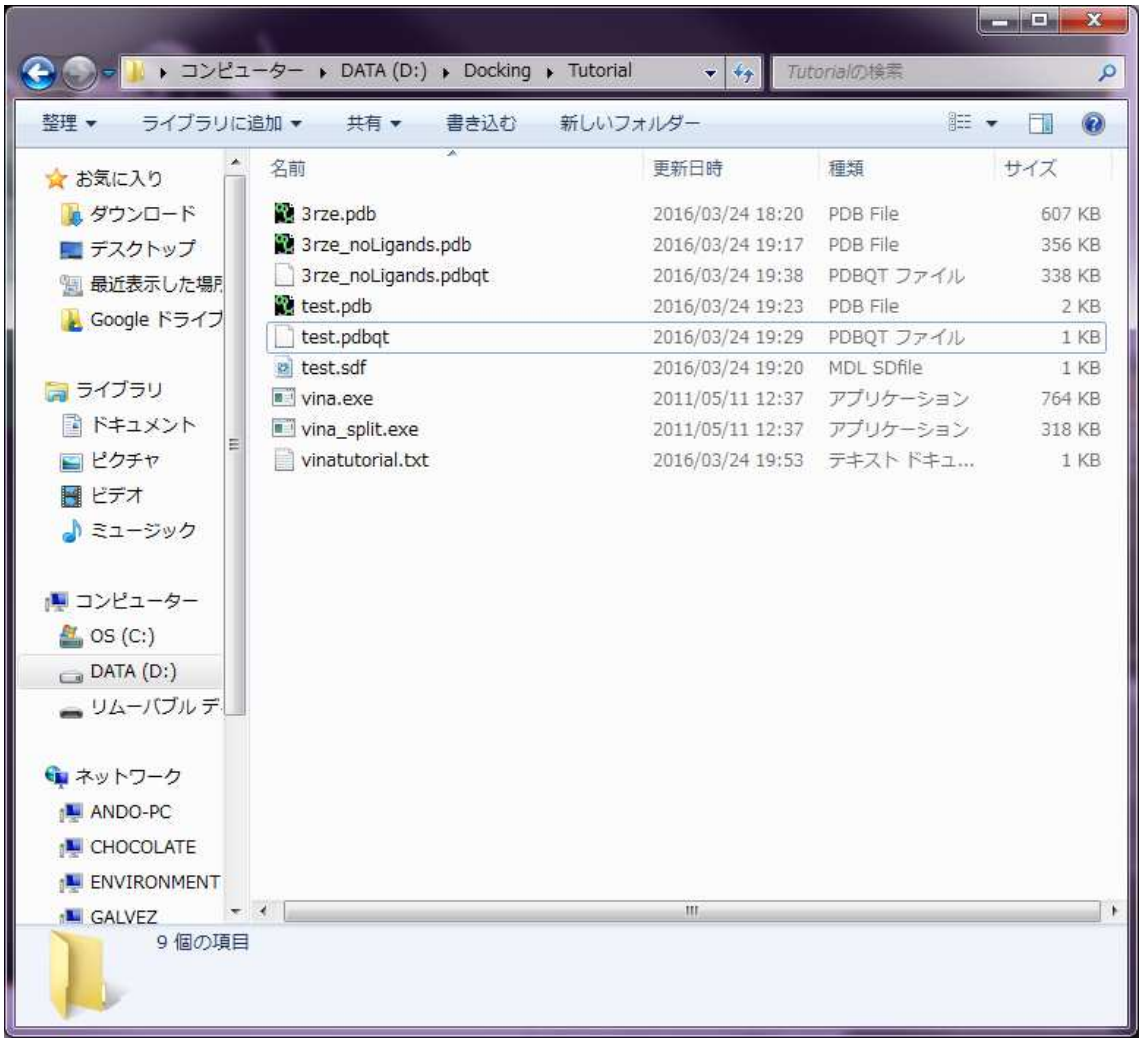
Spacing を 1 にし、グリッドサイズと位置を選択する。
大きさは 27000Angstrom^3 以下でなくてはいけない。



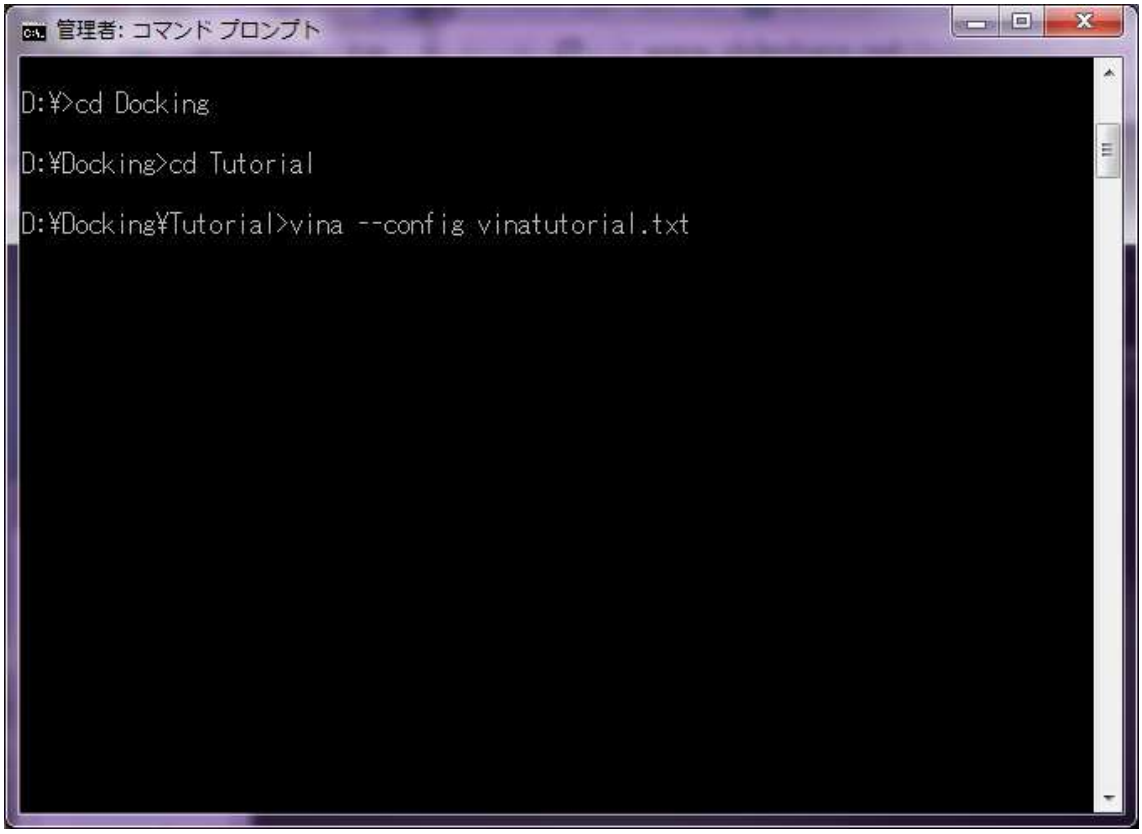
設定した値を自分でテキストファイルに書き込む。



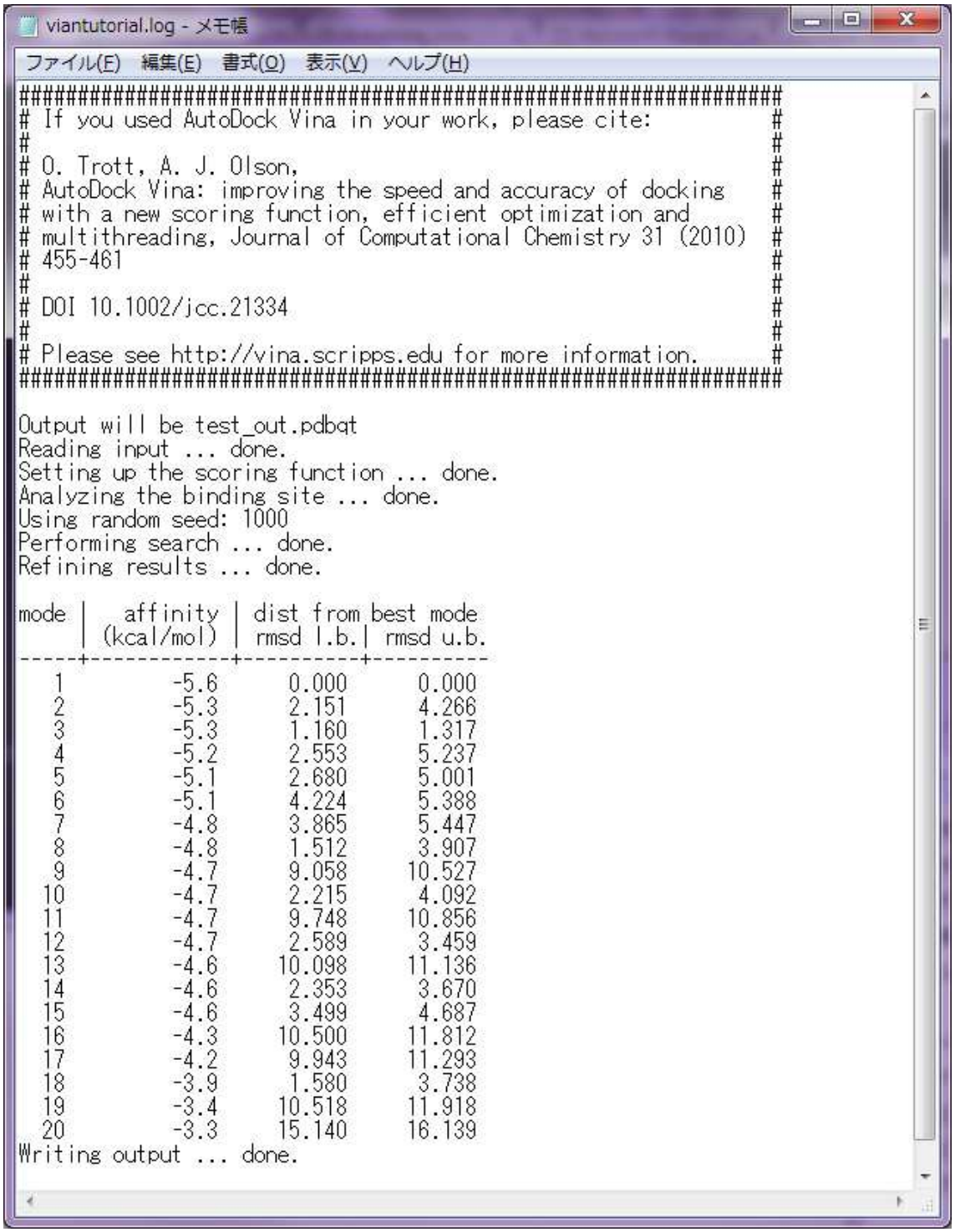
７．vina.exe と vina_split.exe(http://vina.scripps.edu/download.html)をフォルダにコピー



８．コマンドプロンプトを起動してファイルがあるディレクトリで
vina --config (作成した設定ファイル)と入力して実行



９．Log ファイルが作成される。Affinity が結合エネルギー


viantutorial.log - メモ帳

ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)

```
#################################################################
# If you used AutoDock Vina in your work, please cite:          #
#                                                               #
# O. Trott, A. J. Olson,                                        #
# AutoDock Vina: improving the speed and accuracy of docking    #
# with a new scoring function, efficient optimization and       #
# multithreading, Journal of Computational Chemistry 31 (2010)  #
# 455-461                                                       #
#                                                               #
# DOI 10.1002/jcc.21334                                         #
#                                                               #
# Please see http://vina.scripps.edu for more information.      #
#################################################################

Output will be test_out.pdbqt
Reading input ... done.
Setting up the scoring function ... done.
Analyzing the binding site ... done.
Using random seed: 1000
Performing search ... done.
Refining results ... done.

mode |   affinity | dist from best mode
     | (kcal/mol) | rmsd l.b.| rmsd u.b.
-----+------------+----------+----------
   1          -5.6      0.000      0.000
   2          -5.3      2.151      4.266
   3          -5.3      1.160      1.317
   4          -5.2      2.553      5.237
   5          -5.1      2.680      5.001
   6          -5.1      4.224      5.388
   7          -4.8      3.865      5.447
   8          -4.8      1.512      3.907
   9          -4.7      9.058     10.527
  10          -4.7      2.215      4.092
  11          -4.7      9.748     10.856
  12          -4.7      2.589      3.459
  13          -4.6     10.098     11.136
  14          -4.6      2.353      3.670
  15          -4.6      3.499      4.687
  16          -4.3     10.500     11.812
  17          -4.2      9.943     11.293
  18          -3.9      1.580      3.738
  19          -3.4     10.518     11.918
  20          -3.3     15.140     16.139
Writing output ... done.
```