使用 MolAICal 的深度学习模型和经典算法程序进行 GCGR 的从头药物设计

作者：MolAICal （update 2020-07-03）

更多教程（含英文教程）请见如下：
MolAICal 官方主页：https://molaical.github.io
MolAICal 文章介绍：https://doi.org/10.1093/bib/bbaa161
MolAICal 中文博客：https://molaical.github.io/cntutorial.html
MolAICal blogspot：https://qblab.blogspot.com

1. 简介

胰高血糖素受体（GCGR）是 2 型糖尿病的一个受体靶点。本教程使用人工智能的方法从头设计 GCGR 的药物分子，介绍了 MolAICal 的标准操作流程。本教程能够帮助药理学家、化学家以及其他科学家根据蛋白的 3D 活性口袋设计合理药物分子。

2. 工具

2.1. 所需软件下载地址

1) MolAICal: https://molaical.github.io
2) UCSF Chimera: https://www.cgl.ucsf.edu/chimera/

2.2. 操作示例文件

所有用到的操作教程均可在下面的网站下载：
https://github.com/MolAICal/tutorials/tree/master/001-AIGrow

3. 操作流程

3.1. 受体的处理

1. 下载胰高血糖素受体(GCGR)的 PDB 文件，PDB ID: 5EE7: http://www.rcsb.org/structure/5EE7
2. 启动 UCSF Chimera 软件: File→Open，载入 PDB 文件 5EE7.pdb（如图 1 所示）。

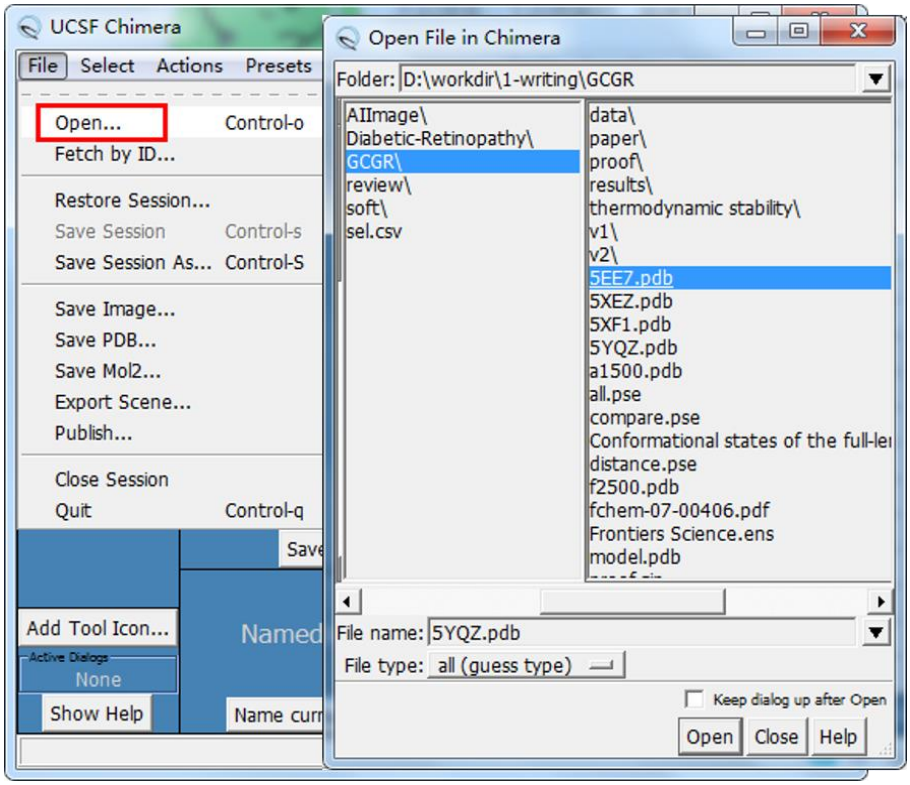


图 1. 载入分子

3. 选择并删除被选分子。

选中无用配体: Select→Residue→OLA (如图 2 所示).

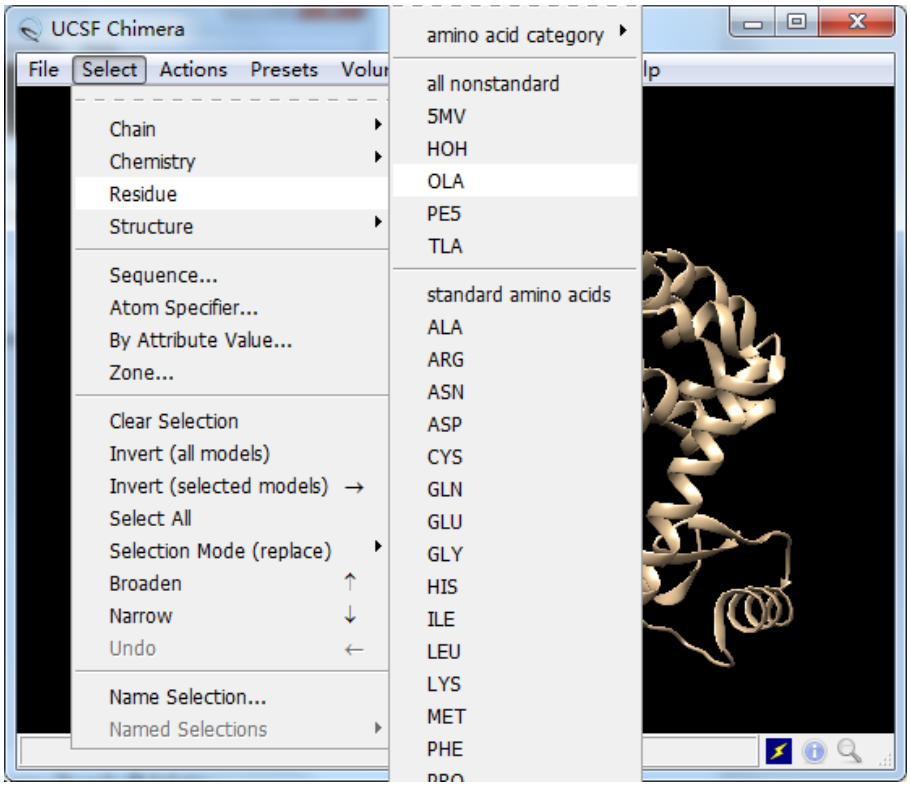


图 2. 选中某个分子

删除无用配体: Actions→Atoms/Bonds→delete (如图 3 所示)。

以上给出了一个简单示例，可以通过同样的操作继续删除 HOH、PE5 和 TLA 等。

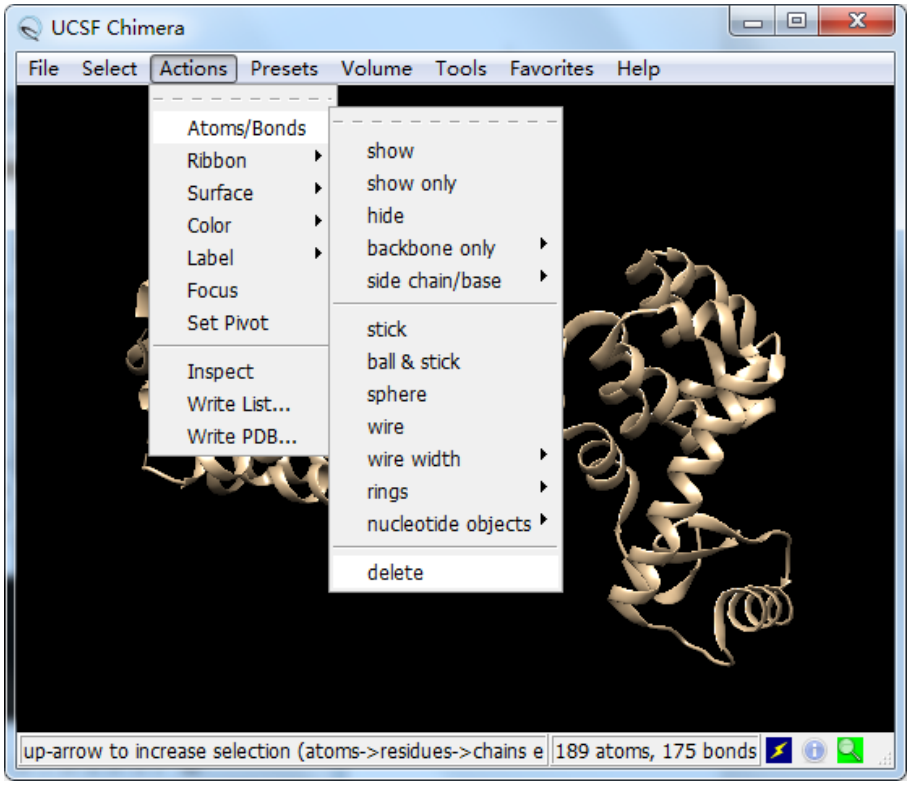


图 3. 删除选中的分子

4. 加氢。Tools→Structure Editing→AddH (如图 4 所示)。

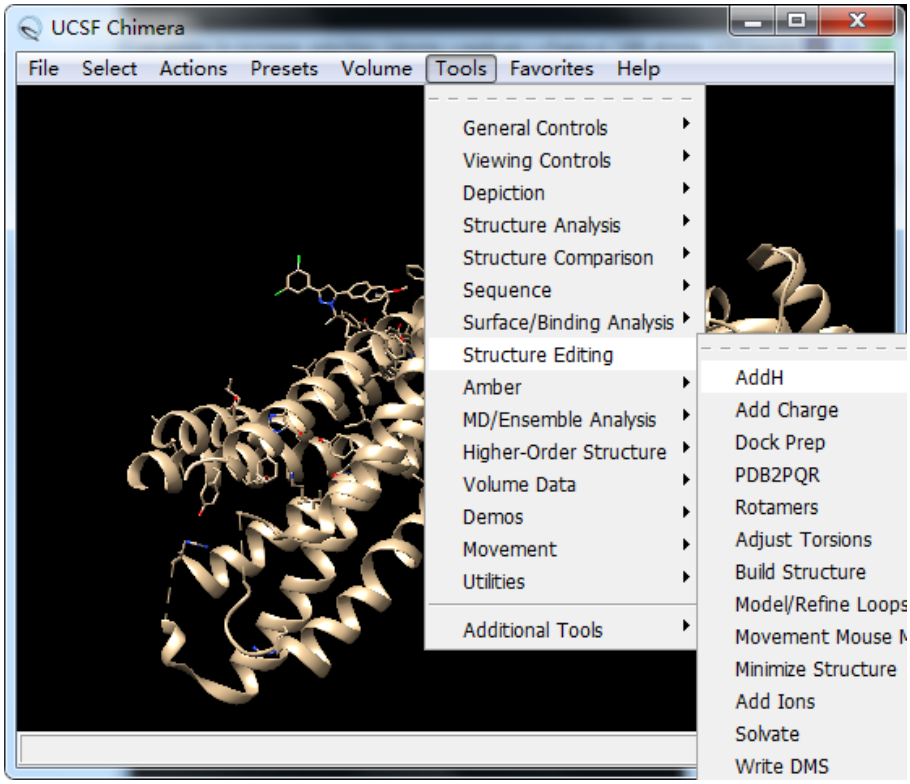


图 4. 加氢

注意事项: 由于 MolAICal 使用 AMBER 力场处理受体，在删除氢原子之前用 UCSF Chimera 加上氢原子。如果受体中含有冗余的氢原子，请先删除再加上。具体操作首先，选中氢原子：Select→Chemistry→element→H，然后删除所选氢原子：

Actions→Atoms/Bonds→delete (如图 3 所示)。因为 GCGR 晶体结构中不包含冗余氢原子，所以此处省略该步骤。

5. 保存坐标文件 GCGRH.pdb。保存 GCGRH.pdb 文件目的是可以方便后续的操作（如图 5 所示)。

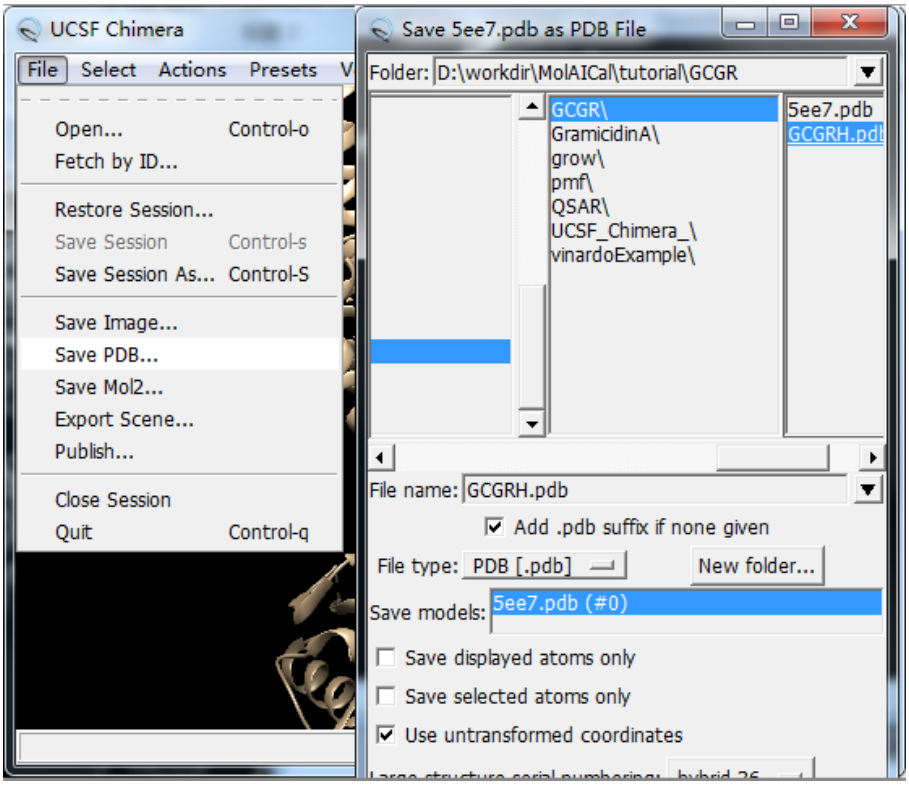


图 5. 保存 PDB 格式的复合物

6. 选中并删除配体 5MV(如图 6 和 7 所示)。将无配体结合的 GCGR 命名为 GCGRNoLigand.pdb 文件并保存 (如图 8 所示)。

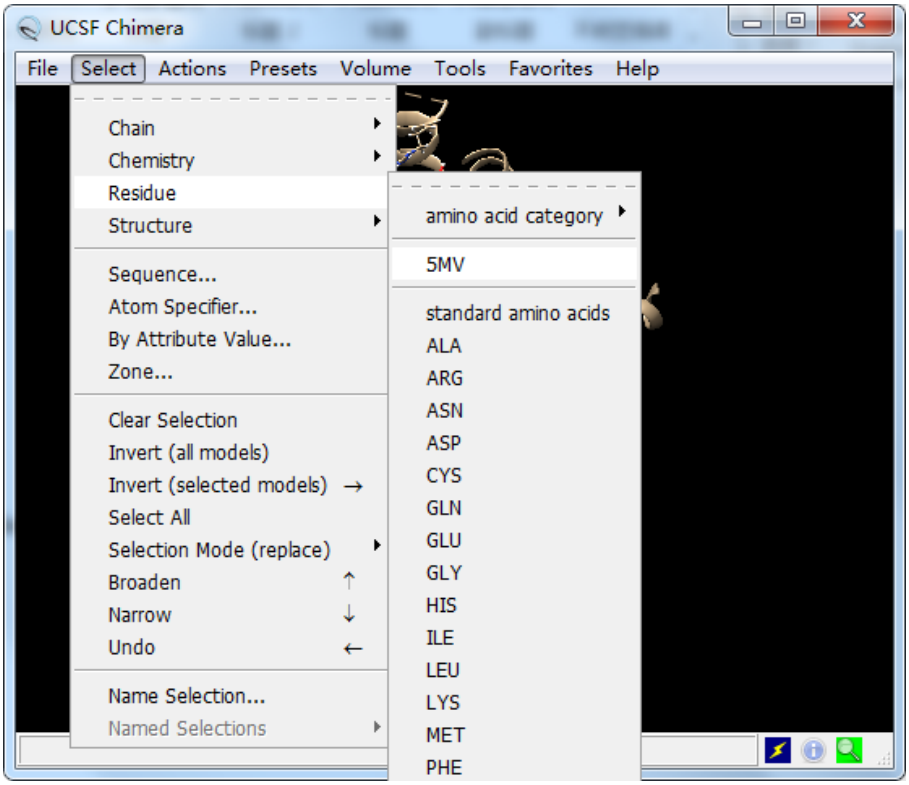


图 6. 选中 5MV 配体

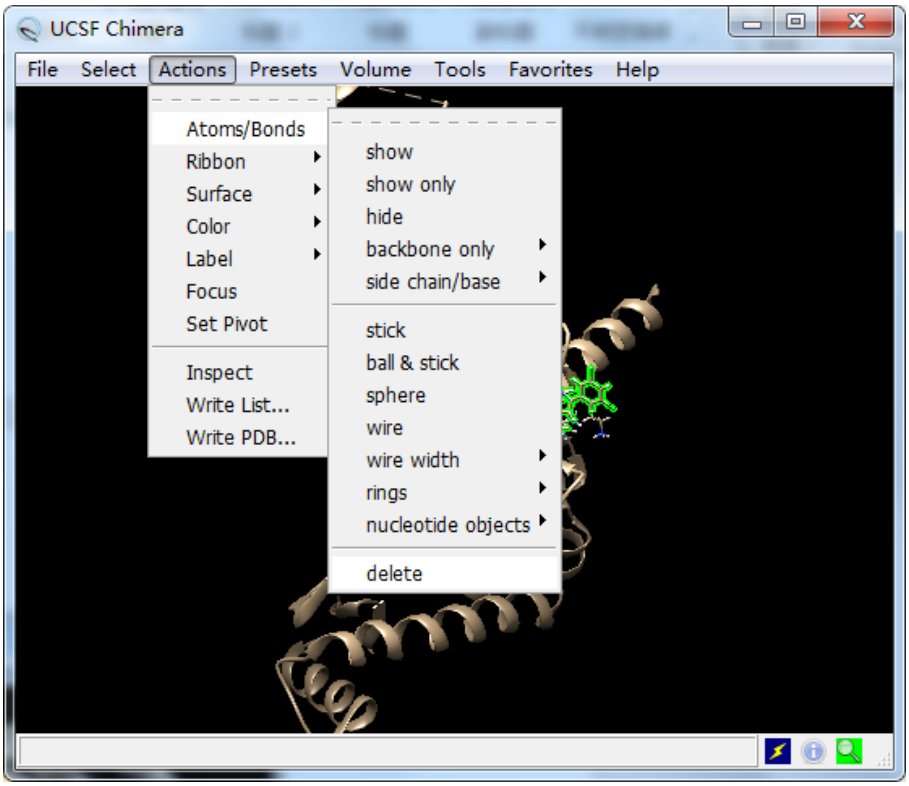


图 7. 删除 5MV 配体

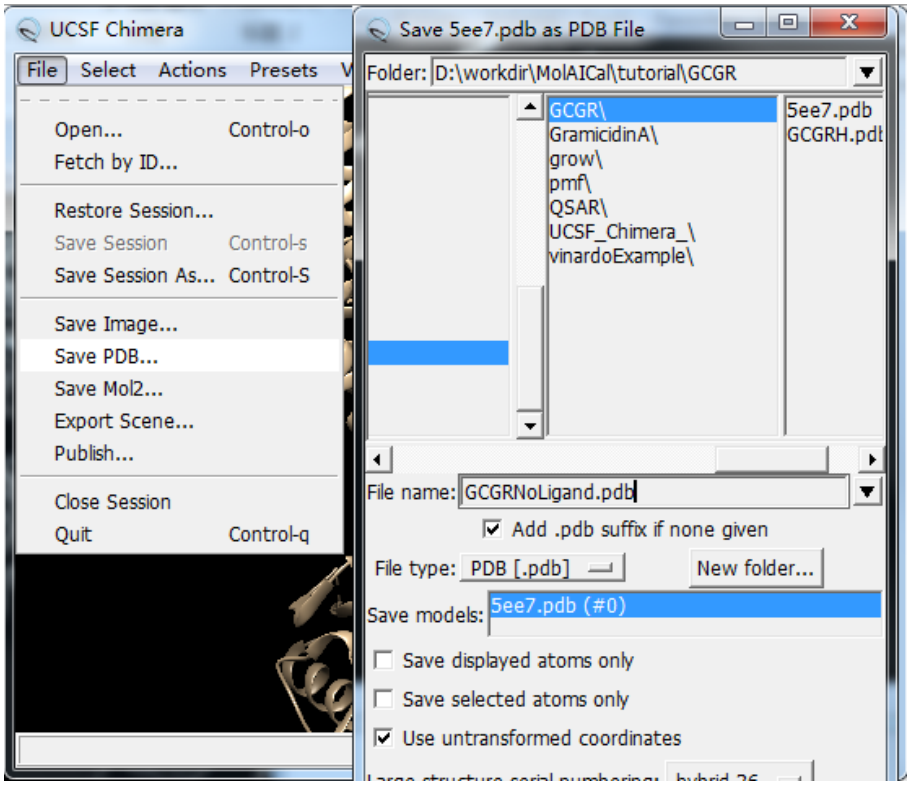


图 8. 保存 PDB 格式的受体坐标

3.2. 处理配体

1. 关闭会话 (File→Close Session)，重新载入 GCGRH.pdb 文件，选中配体 5MV (如图 9 和 10 所示):

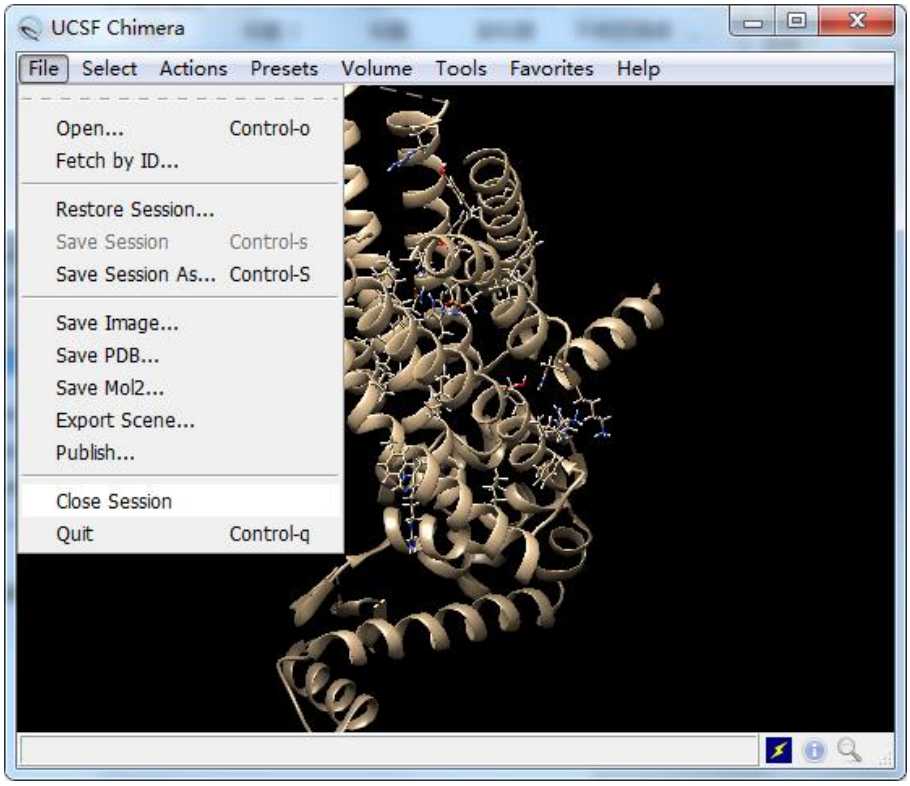


图 9. 关闭会话

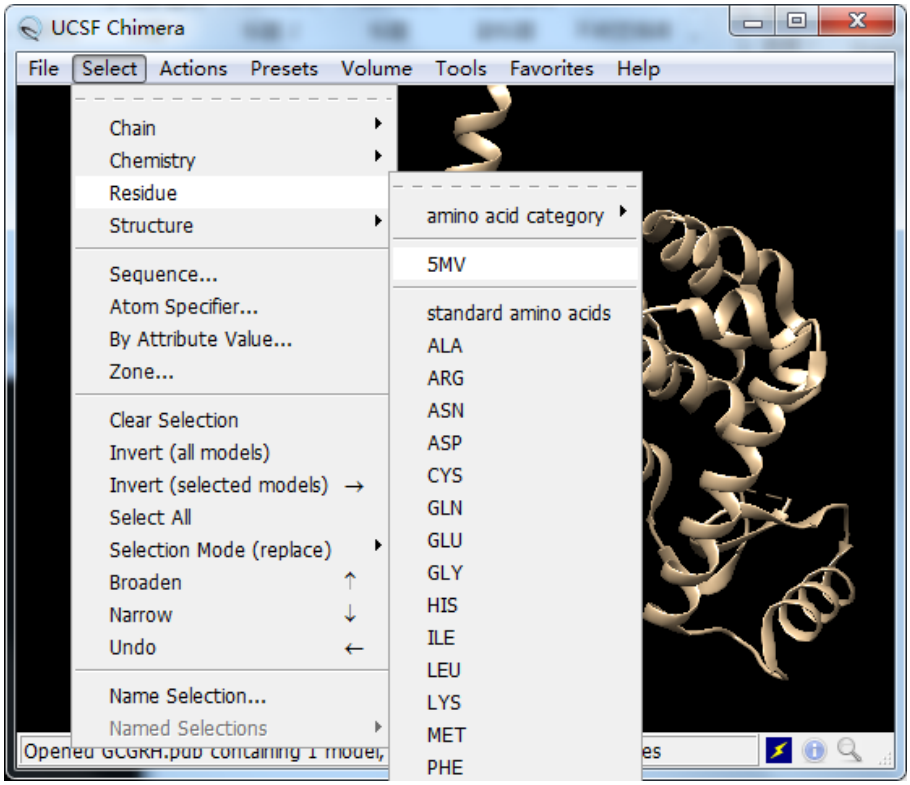


图 10. 选中 5MV

2. Invert (selected models): 反选配体 5MV (如图 11 所示):

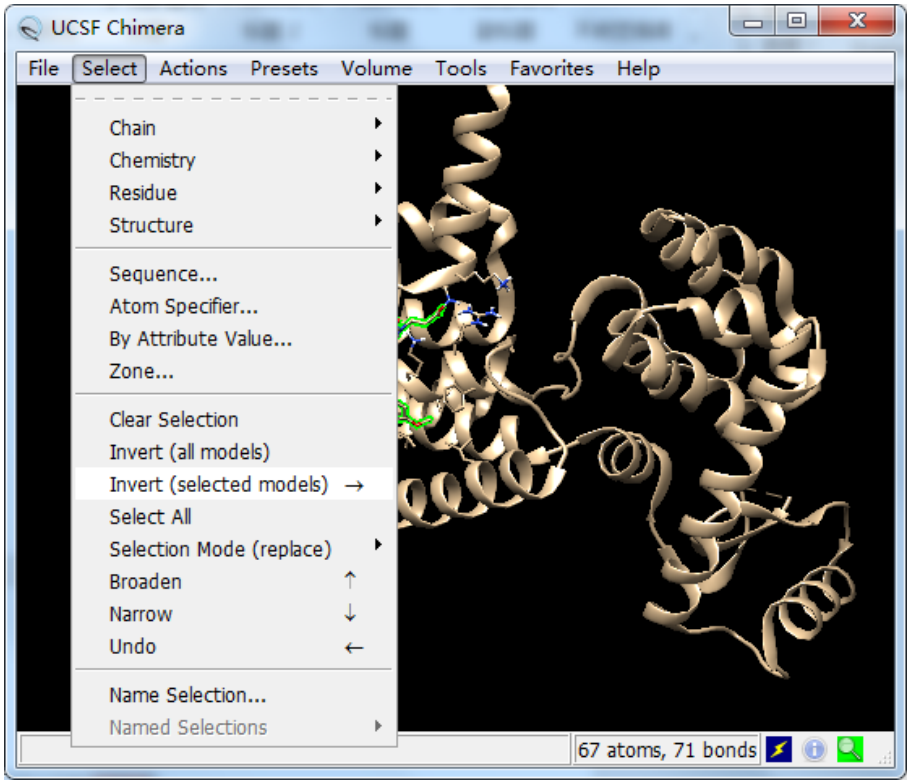


图 11. 反选配体 5MV

3. 删除反选 (如图 12 所示):

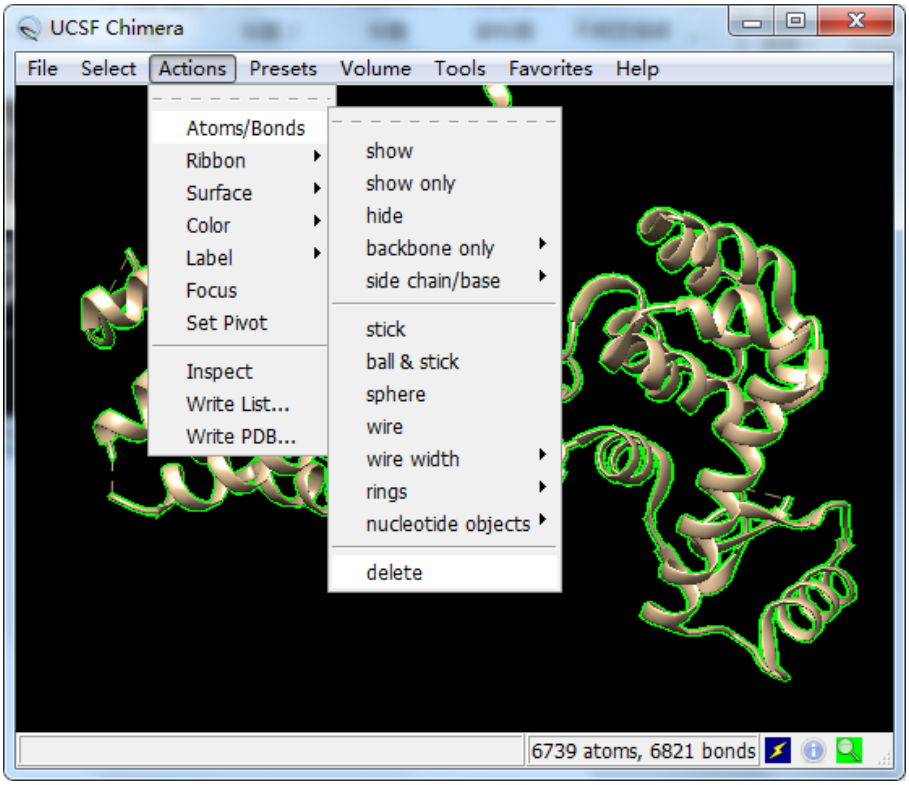


图 12. 删除反选

4. 将配体保存为 ligand.pdb。配体分子的片段可以作为药物生长的起始点（如图 13 所示）。

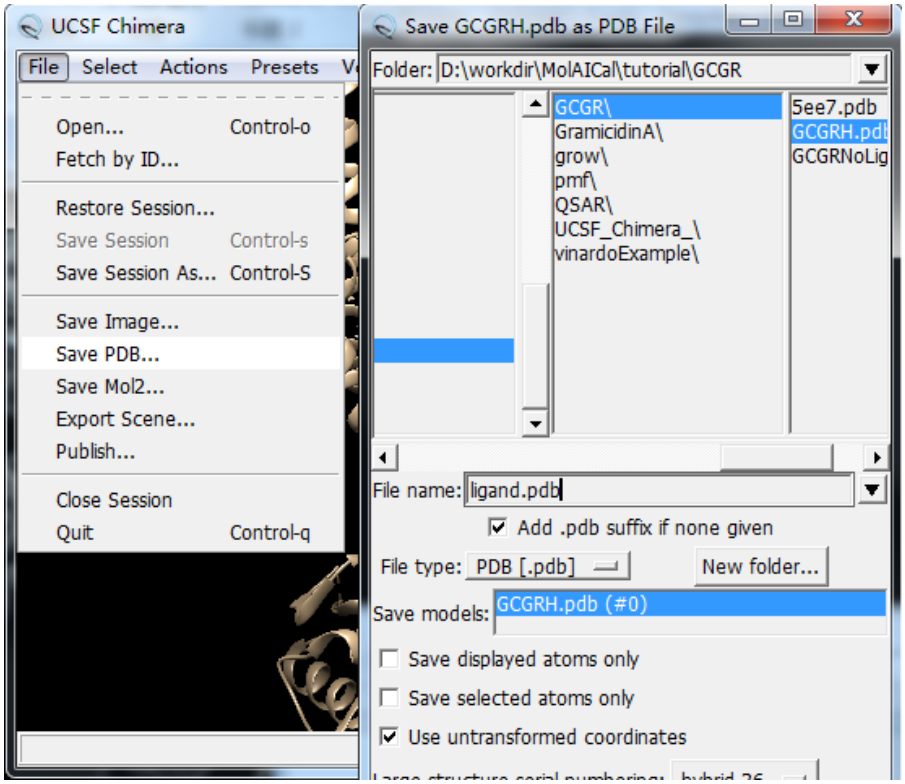


图 13. 保存配体坐标

3.3. 计算生长盒子中心

计算生长盒子的几何中心。(注意：如果没有配体，可以选择实验中已报道的关键残基作为起始的锚定点)。

1. 关闭会话 (File→Close Session)，载入 GCGRH.pdb，然后使用图 10 的方法选择配体（如图 14 所示):

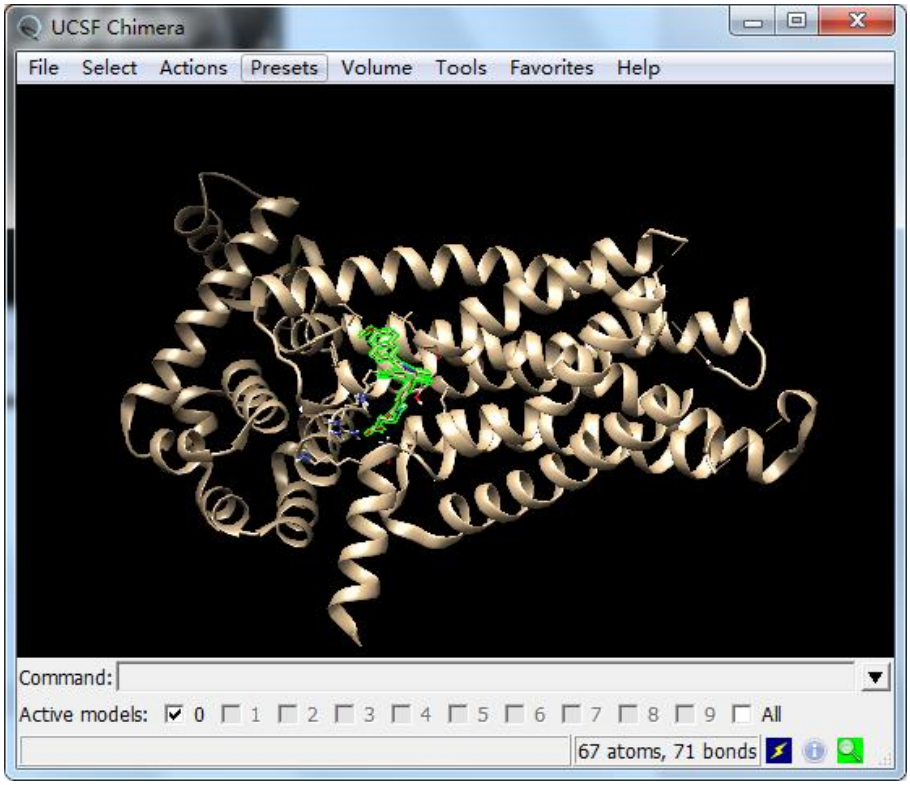


图 14. 选择配体

2. Tools→Structure Analysis→Distance (如图 15 所示):

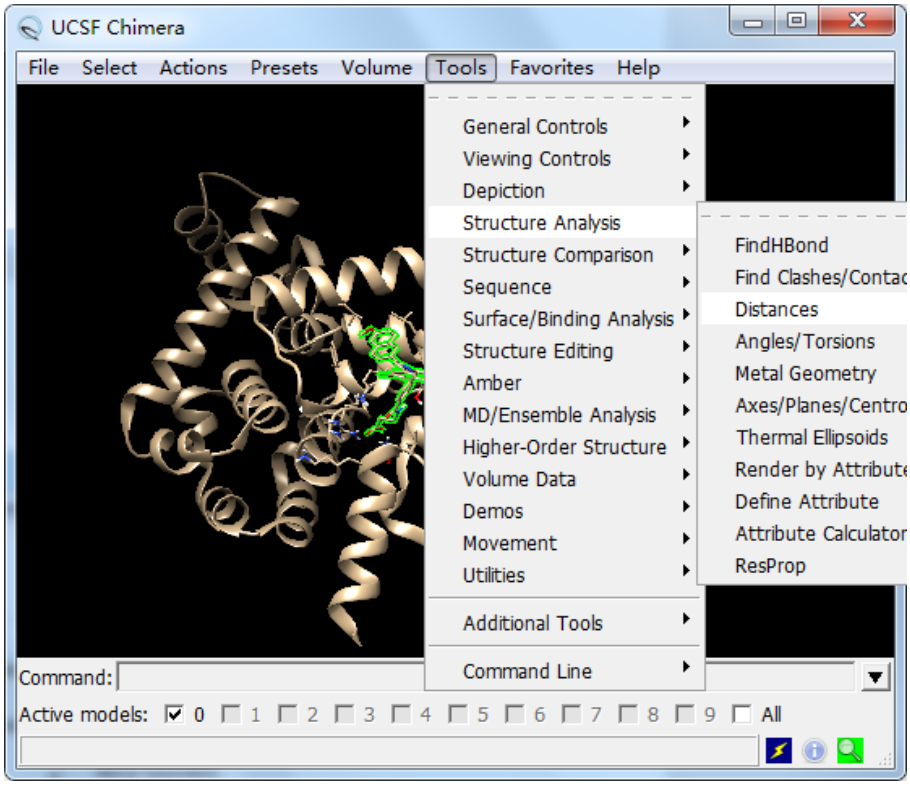


图 15. 选择距离工具

3. 点击 Axes/Planes/Centroids，然后点击 “Define centroid” (如图 16 所示):

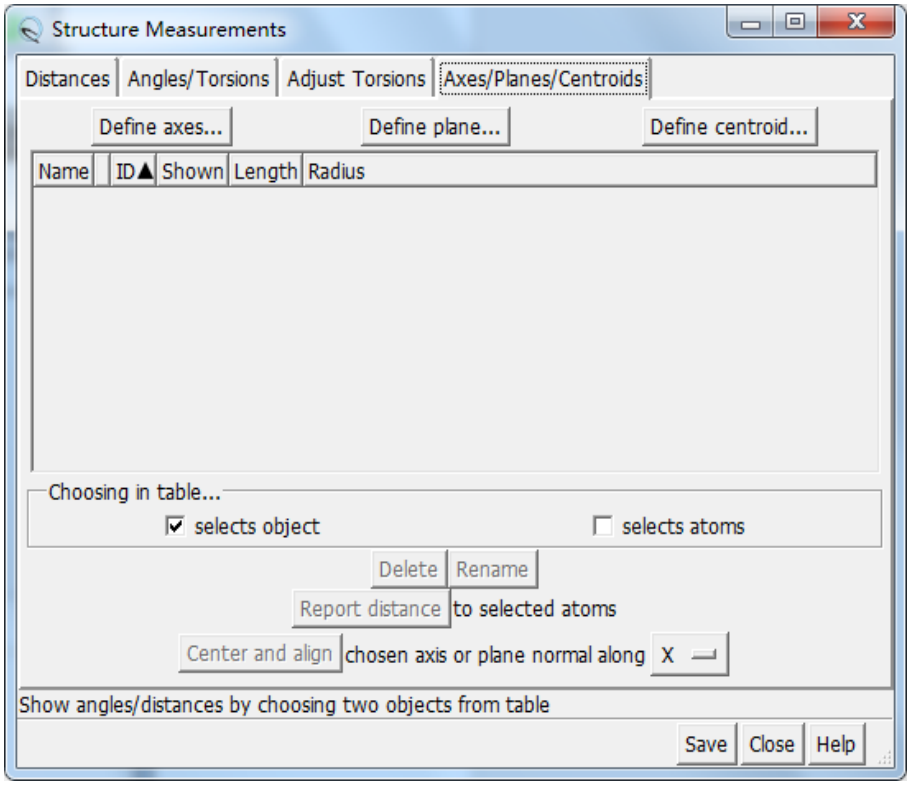


图 16. 定义质心

4. 选择弹出窗口中的 “OK” (如图 17 所示)。

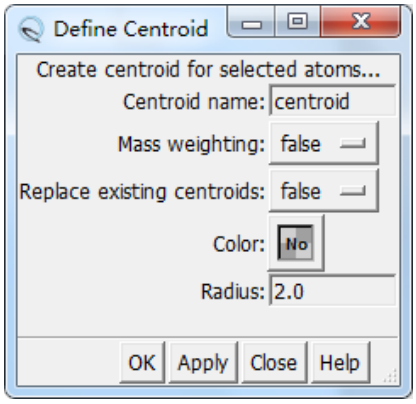


图 17. 定义质心盒子

5. 选择已被定义的质心 (蓝色所示), 然后点击“Report distance” (如图 18 所示)

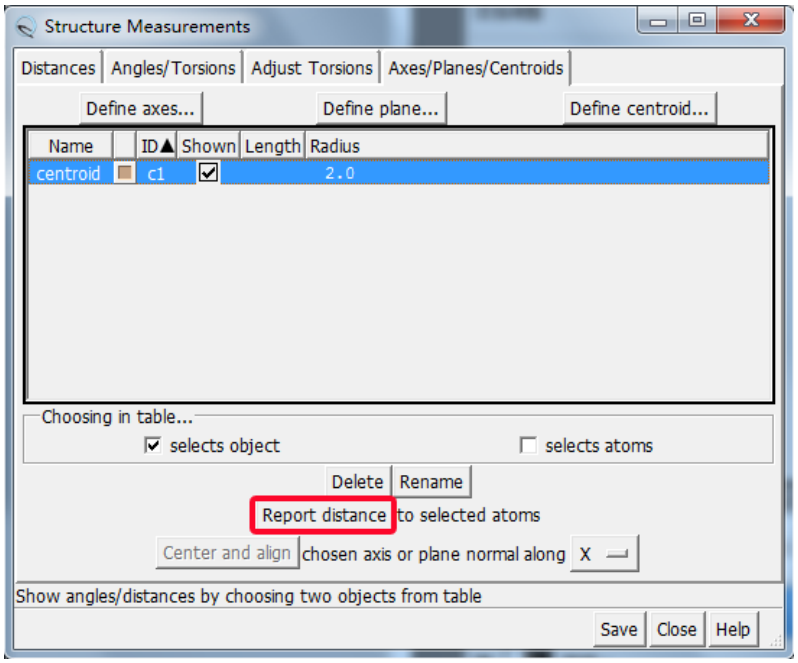


图 18. 点击 Report distance

6. 此时可以看到质心为(x, y, z): -30.011, 1.665, -36.581 (如图 19 所示):

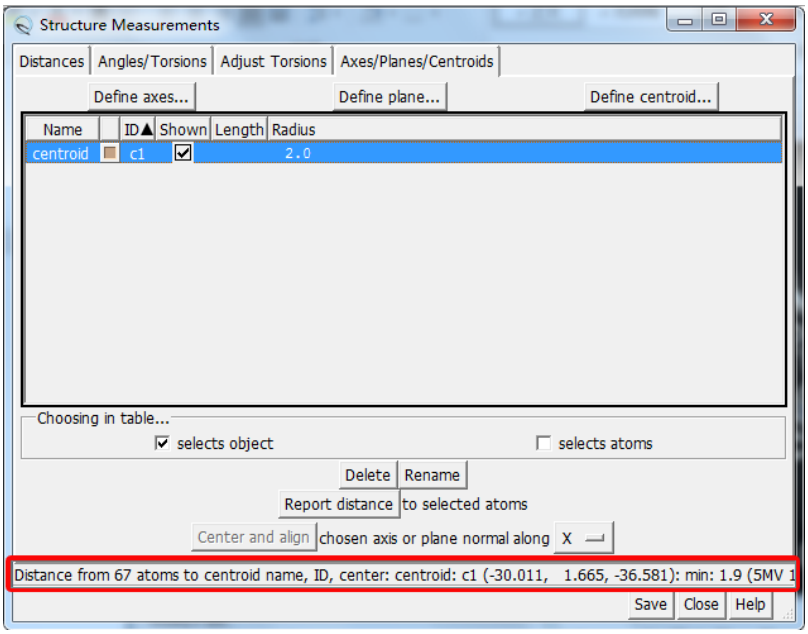


图 19. 展示盒子的质心坐标

7. 计算最终盒子的大小。你可以将 X, Y, Z 的长度调整为 30, 30, 30。用以下 MolAICal 命令生成 “box.bild”。(注意: **X, Y, Z 坐标中的双引号是必不可少的，X, Y, Z 坐标之间的间隔距离应该为一个空格。**)
   1) molaical.exe -tool box -i "-30.011 1.665 -36.581" -l "30.0 30.0 30.0" -o "D:\workdir\MolAICal\tutorial\GCGR\box.bild"
   2) File→Open,然后打开“**box.bild**”(如图 20 所示)，然后检查产生的盒子是否合适 (如图 21 所示)。

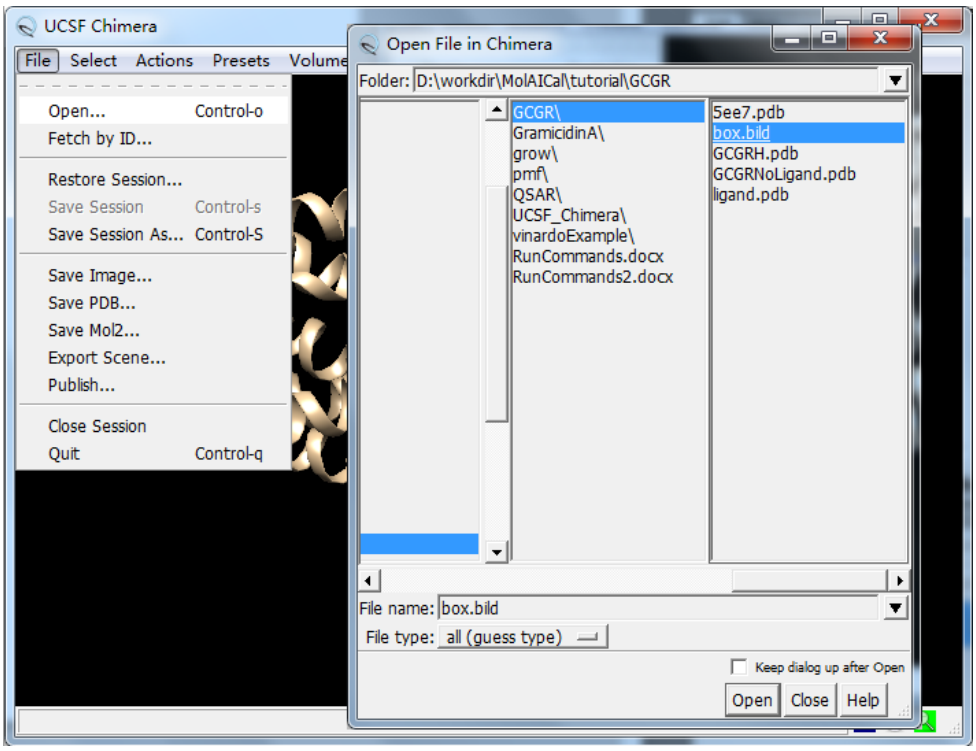


图 20. 打开 box.bild

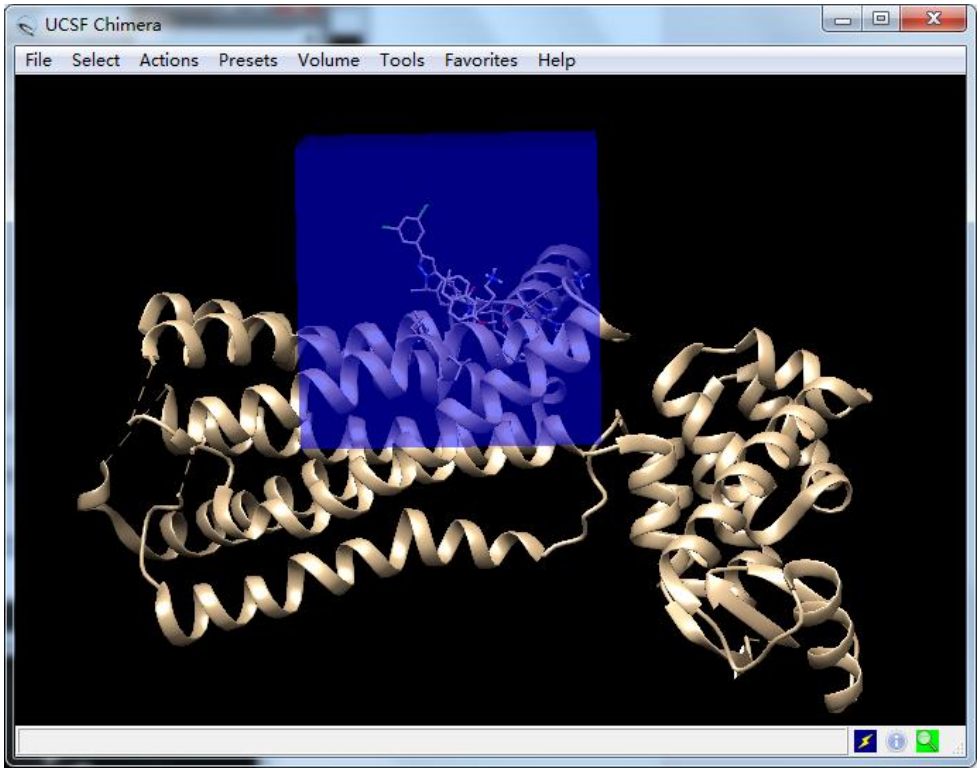


图 21. 在 GCGR 口袋中显示盒子

从图 21 可以看出，盒子的参数是合适的，因此最终盒子的质心参数为-30.011, 1.665, -36.581，盒子的长度 X, Y, Z 为 30.0, 30.0, 30.0。

3.4. 制作初始生长片段

本教程介绍了两种方法，你可以选择其中任何一种进行药物从头设计。如果受体活性口袋中配体的晶体结构已知的话，我们推荐第一种方法。

方法 1. 从配体的晶体结构中选择片段

在本方法中，初始片段是从 GCGR 活性口袋中配体的晶体结构中提取的。

1. 制作初始生长片段。打开上文保存的 ligand.pdb 文件。通过鼠标和键盘选择原子。
   1) Ctrl + 鼠标左键: 一次选择一个原子。
   2) Ctrl + Shift + 鼠标左键: 同时选择多个原子。

   本教程中，我们通过 Ctrl + 鼠标左键选择内层原子作为初始生长片段。(选择结果如图 22 所示)。

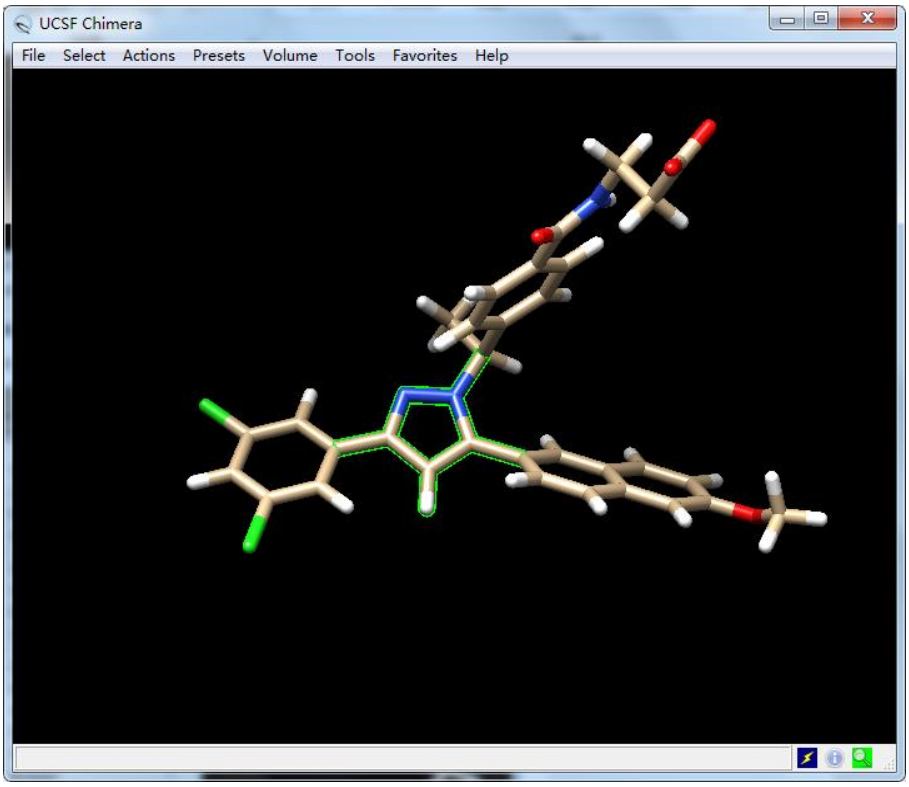


图 22. 选择初始生长部分

2. 反选（已选模型）如图 23 所示。

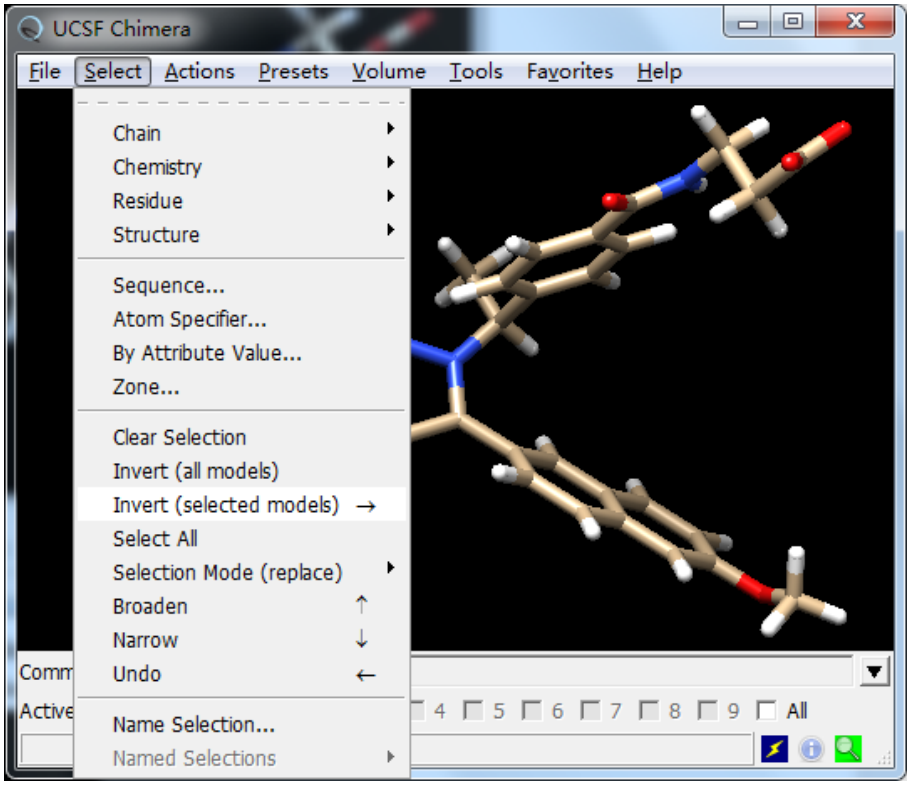


图 23. 选择 Invert (selected models)

3. 然后删除反选部分：Actions→Atoms/Bonds→delete.
   最终选择图 24 所示分子作为初始片段。

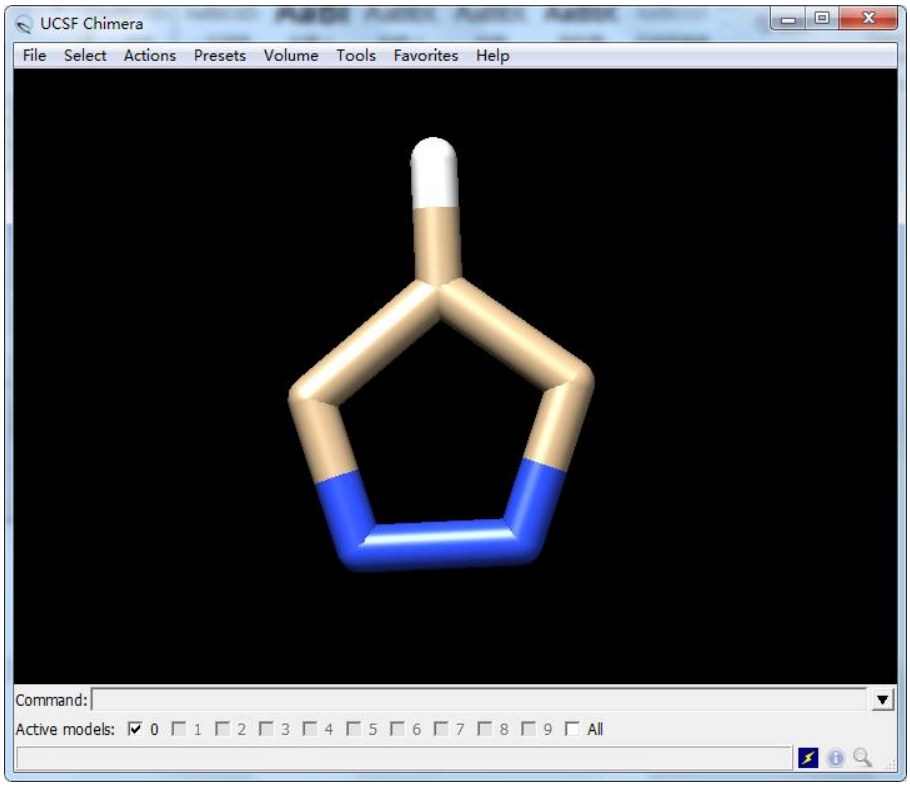


图 24. 初始生长片段

4. 为了使分子按照正确的方法生长，需要在初始生长片段上加氢，具体操作为："Tools→Structure Editing→AddH" (如图 25 所示)。

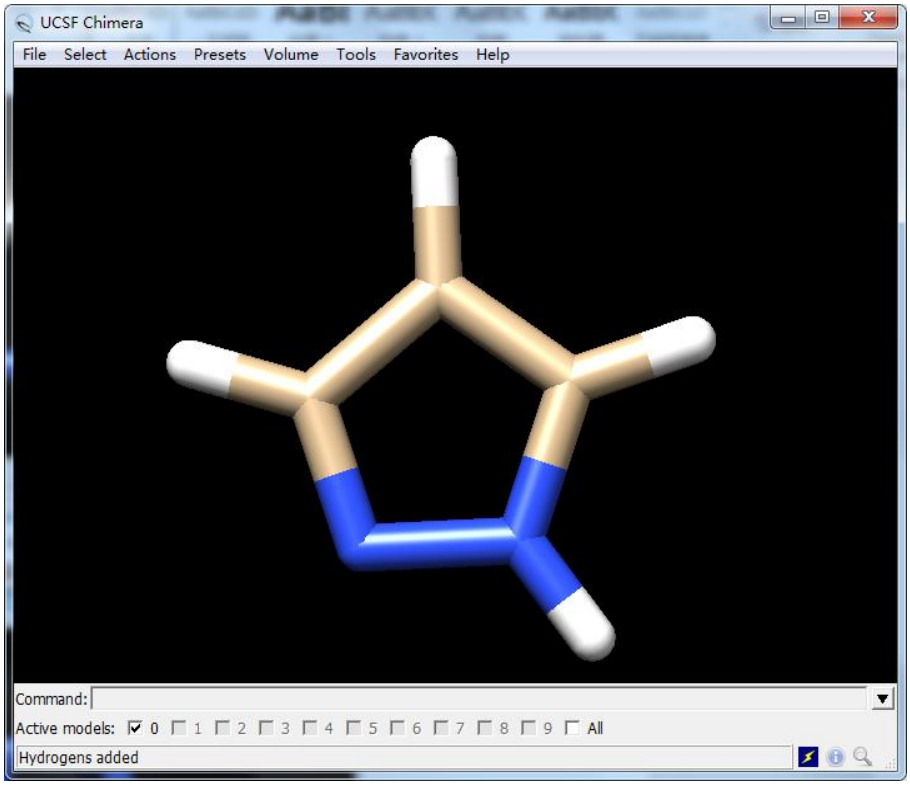


图 25. 初始生长片段加氢结果

5. 然后将片段保存为 sybyl Mol2 格式，命名为“startFrag.mol2” (如图 26 所示)。

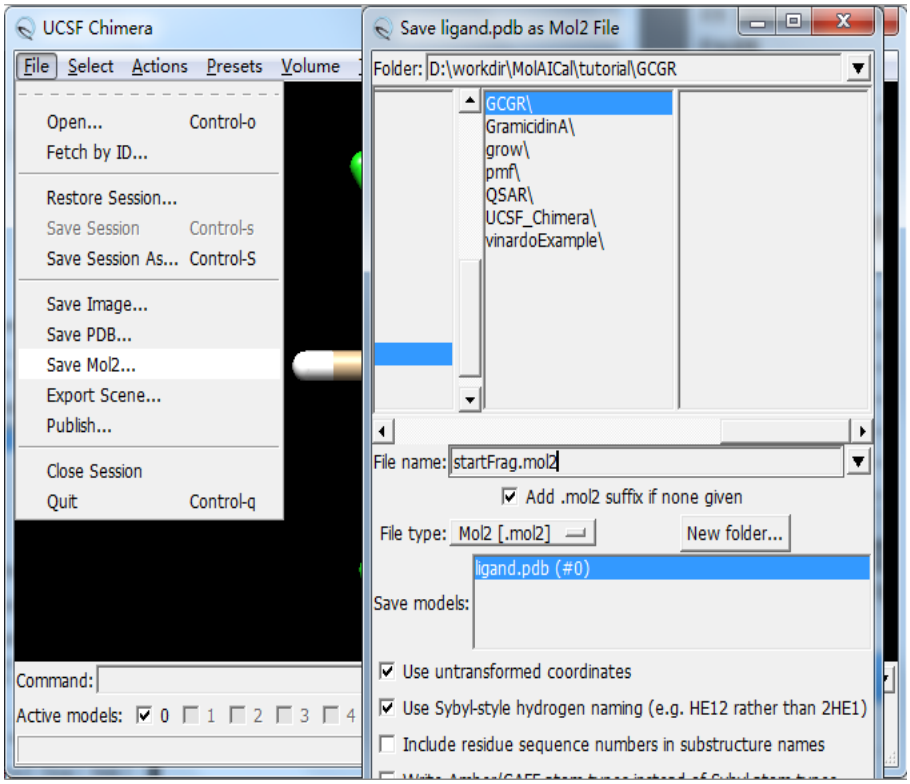


图 26. 保存初始生长片段的坐标

方法 2. 活性口袋中没有配体的晶体结构。

如果在受体活性口袋中没有配体的晶体结构，你可根据文献报到或者自己的经验选择一个

关键残基上的原子，操作同方法 1。将该原子做成独立文件，通过序列格式 SMILES 给出初始片段。MolAICal 将自动搜索 SMILES 格式片段在 GCGR 口袋中的最佳位置。
将“growMethod”设置为 randomFrag。然后添加“startAtomPosition”和“startSmiFrag”参数。“startAtomPosition”包含一个你所选残基中的原子。“startSmiFrag”为你指定的 SMILES 格式的初始片段文件。参数示例如下:

---

growMethod	randomFrag
startFragFile	D:/GCGR/genstartFrag.mol2
startAtomPosition	D:/GCGR/resPosition.pdb
startSmiFrag	C

注意：参数 startSmiFrag 决定起始片段，如果 startSmiFrag 的值是 SMILES，这个 SMILES 将会被转化成 startFragFile 参数指定的起始生长片段 D:/GCGR/genstartFrag.mol2；如果 startSmiFrag 的值是 null，那么 startFragFile 参数指定的片段“genstartFrag.mol2”必须要用户自行定义，所以，用户可以通过设置 startSmiFrag 为 null，实现自定义起始片段的构象，MolAICal 可以自动搜索此用户自定义片段在口袋的合适位置。

本教程选择方法 1 进行药物设计。

3.5. 运行药物从头设计程序

有两种选择：方法 1 利用 AI 模型及经典程序进行药物从头设计。方法 2 利用纯经典程序进行药物从头设计。为节省时间，你可以选择以下任何一种方法开始学习。

方法 1. 利用 AI 模型及经典程序的药物从头设计方法
将“libStyle”设置为 AIFrag。实例如下:

#定义可读文库路径：mol2, SMILES, AIFrag
libStyle AIFrag

注意：如果显示错误信息：如“*Warning: Atom 9 C.3 overlaps with protein!*”及“*Warning: some atoms overlap with protein.*”你可以用 MD 模拟工具或者 UCSF Chimera 最小化受体-配体复合物。只有 win64 或者 linux64 版本的 MolAICal 可以执行该命令。
按照教程中的描述，将控制台目录转换为 “001-AIGrow”
#> cd 001-AIGrow

最后在后台运行以下命令
linux 系统:
#> molaical.exe -denovo grow -i InputParFileAI.dat >& denovo.log &

windows 系统 (使用 PowerShell):
#> molaical.exe -denovo grow -i InputParFileAI.dat

如果想要在后台进行运算，你可以运行以下命令:
#> powershell -windowstyle hidden -command “molaical.exe -denovo grow -i InputParFileAI.dat”

方法 2. 使用经典程序进行药物从头设计
将“libStyle”设置为 mol2。MolAICal 将使用用户自定义的文库。实例如下:

```
# 定义可读文库路径: mol2, SMILES, AIFrag
libStyle                    mol2
```

Note: 如果显示错误信息：如“*Warning: Atom 9 C.3 overlaps with protein!*”及“*Warning: some atoms overlap with protein.*”你可以用 MD 模拟工具或者 UCSF Chimera 等软件最小化受体-配体复合物。任意版本的 MolAICal 可以执行该命令。

MolAICal 可以通过 JAVA 并行流调用多个 CPU 内核，你需要根据电脑的配置文件设置“coreNum”参数。
按照教程中的描述，将控制台目录转换为 “001-AIGrow”。

```
#> cd 001-AIGrow
```

在后台运行以下命令
linux 系统:

```
#> molaical.exe -denovo grow -i InputParFileCP.dat >& denovo.log &
```

windows 系统 (使用 PowerShell):

```
#> molaical.exe -denovo grow -i InputParFileCP.dat
```

如果想要在后台执行运算，你可以运行以下命令:

```
#> powershell -windowstyle hidden -command “molaical.exe -denovo grow -i InputParFileCP.dat”
```

4. 结果

当程序运行结束后你可以找到“results”目录，在本教程中，循环周期设置为 30，使用 30 个 CPU 核心运算 1-2 天完成整个计算。如果你想查看结果，打开 001-AIGrow 中的“results”目录，其中名为“AstatisticsFile.dat”的文件包含了药物设计的信息。该文件仅为示例文件并不包含所有结果。在文件“AstatisticsFile.dat”中将看到以下信息:

ID	Name	Cluster	Affinity(kcal/mol)	Formula	InChIKey	Synthetic_Accessibility
1	lig_1.mol2	[1]	-3.27	C9H13N6O14S2	BEQXRJFDXZCPLY-GRQBKTHUSA-O	77.41
2	lig_2.mol2	[1]	-7.67	C13H13N11O9S	IQOZHKOALGXOOC-WVXRZKCLSA-O	75.31

…….

“Affinity”表示结合能力打分。“Cluster”代表 k-means 聚类结果。你可以选择有代表性的配体应用到研究中。Synthetic_Accessibility 分值从 0 到 100，分值 100 表示该化合物是理论上最易合成的。例如，在 UCSF Chimera 中载入“GCGRNoLigand.pdb”, “ligand.mol2” 和“lig_2.mol2”文件 (如图 27)。其中红色的小分子是“lig_2.mol2”。结果表明 MolAICal 生成了与 GCGR 原始配体类似的小分子。

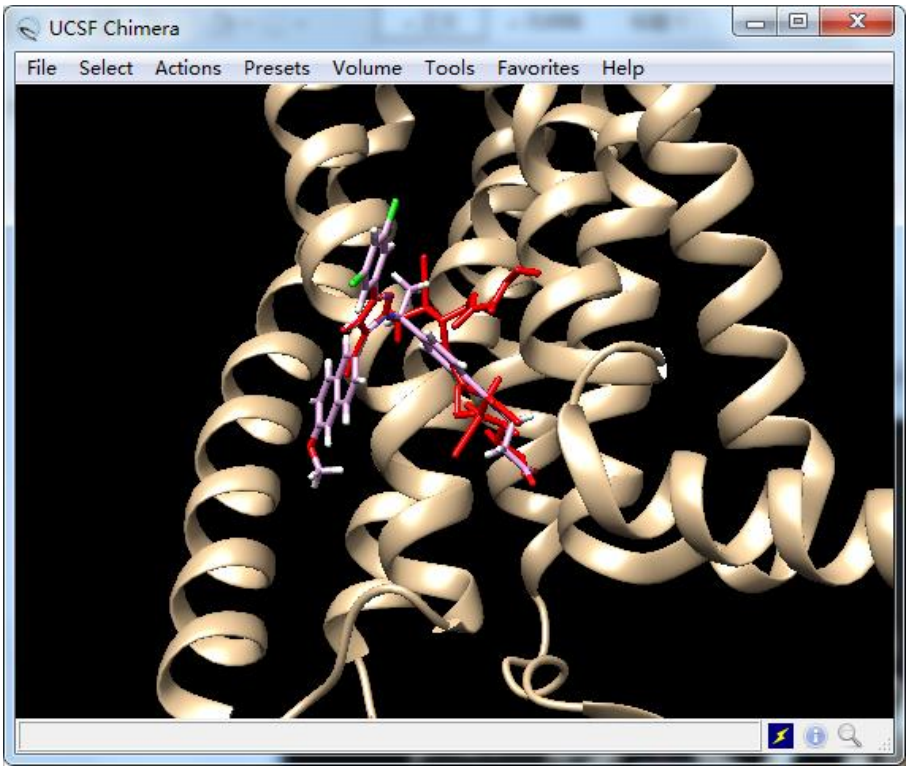
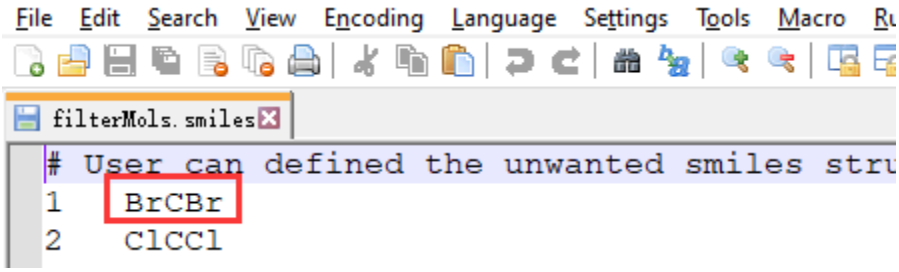


图 27. 药物设计的结果

注意：你可能发现羟基上可能有奇怪的氢原子。这可能是深度学习模型产生的不规则分子导致的。这个氢原子并不参与结合能力打分。你可以用 UCSF Chimera 删除该氢原子，然后再加氢。为了进一步精确地评估这些配体与 GCGR 受体的结合能力，推荐使用分子动力学模拟和 MM/GBSA 进行计算评估。

小技巧：假如用户不想让生成的分子包含不想要的片段（如：诡异的片段，或者不可能的片段等），用户可以在"MolAICal-xxx\filterMols\filterMols.smiles"文件中添加不想要片段的 SMILES 序列，一个 SMILES 占一行，如下图：



其中"MolAICal-xxx"使用户下载的 MolAICal 实际版本，SMILES 序列使用的是 OpenSMILES 格式(见：https://github.com/opensmiles/OpenSMILES)。用户可以用 MolAICal 将 mol2 格式的分子转化成 OpenSMILES 格式（更多内容请参考 MolAICal 的 manual）：

#> molaical.exe -tool mol2tosmi -i "D:/mol2tosmi/zinc_98180786.mol2"