

ChimeraX Tutorial

Dr. S. Eagon
California Polytechnic State University

Important Note: There is both a Chimera and ChimeraX program out there. Chimera is an older program that is being phased out for ChimeraX. This tutorial is only for ChimeraX, and many of the functions and commands listed below will not work for the older Chimera program. So, be sure to download the correct program!

Installing ChimeraX

ChimeraX is available for free download for all operating systems at the official website here:

<https://www.cgl.ucsf.edu/chimerax/download.html>

Scroll down to find the Production Releases to get the most recent, lightly tested version (the “Daily Builds” might contain bugs, so it’s best to go with the Production Releases). Download the highest version number that you see (i.e. version 0.91 is better than 0.9).

Getting PDB files and basic information about a protein

Crystal structures of proteins can be found in the PDB (protein data bank) database here:

<https://www.rcsb.org/>

The PDB is a public resource maintained by the National Institutes of Health (NIH) and is supported by public funds (your taxes).

To download a crystal structure, you need to find the appropriate PDB (protein data bank) file on the website. You can do this using search strings, or you can input a protein database identification (PDB ID) number.

Example Instructions for Downloading a Protein

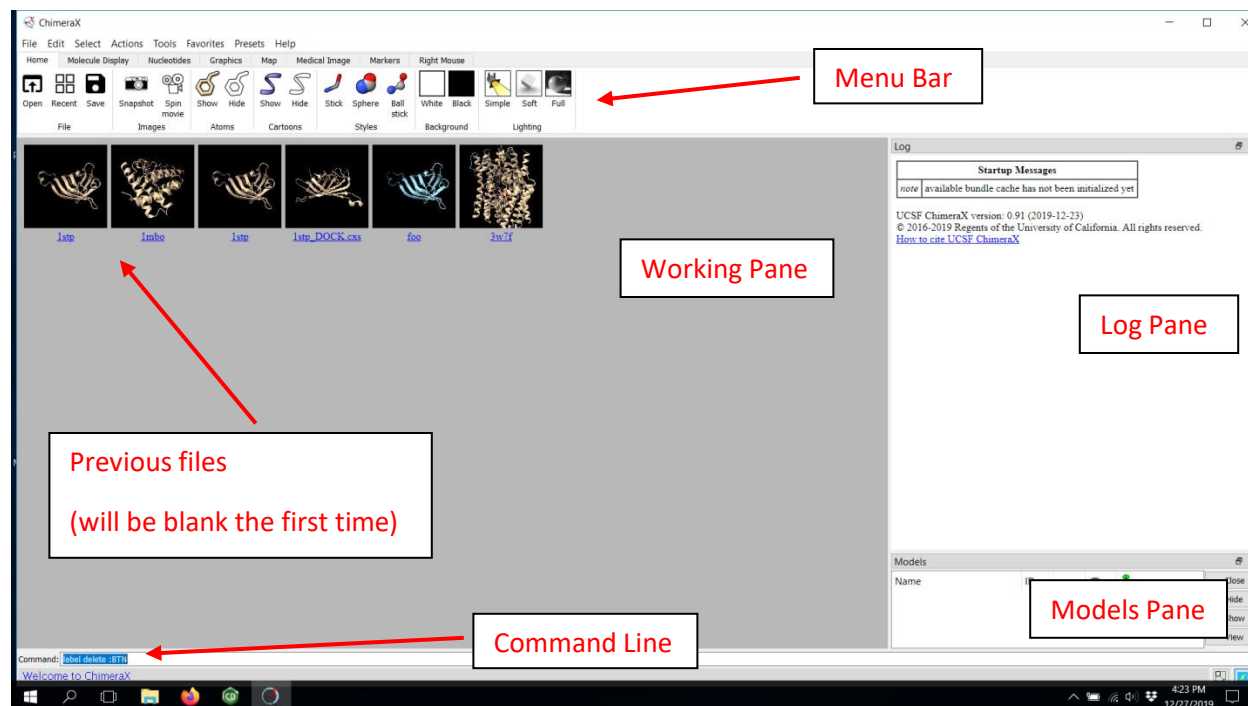
Navigate to <https://www.rcsb.org/>

- Enter “1stp” in the search bar (without the parentheses marks) and hit “enter.”
- At the upper right, you will see a “download” button. Click this to get a drop-down menu and select the “PDB Format” option (not the one with (gz) tag, which is a compressed version)
- This will download the file through your browser. It will be file ending with .pdb
- Note that the download page contains much information about the structure, including:
 - The source article for the structure, with additional background info
 - A picture of the overall structure
 - A list of sub-units and ligands
 - Estimated structures of bound ligands and interacting residues
 - Source organism (what life form the protein came from)
 - X-ray data and date it was deposited
 - Genetic sequence info

ChimeraX Functionality Overview: Basic Stuff

Note: The following pictures are from the Windows10, ChimeraX 0.91 version. Different versions and different OS environments will have a slightly different layout.

- When you first open the program, you should see something like this:



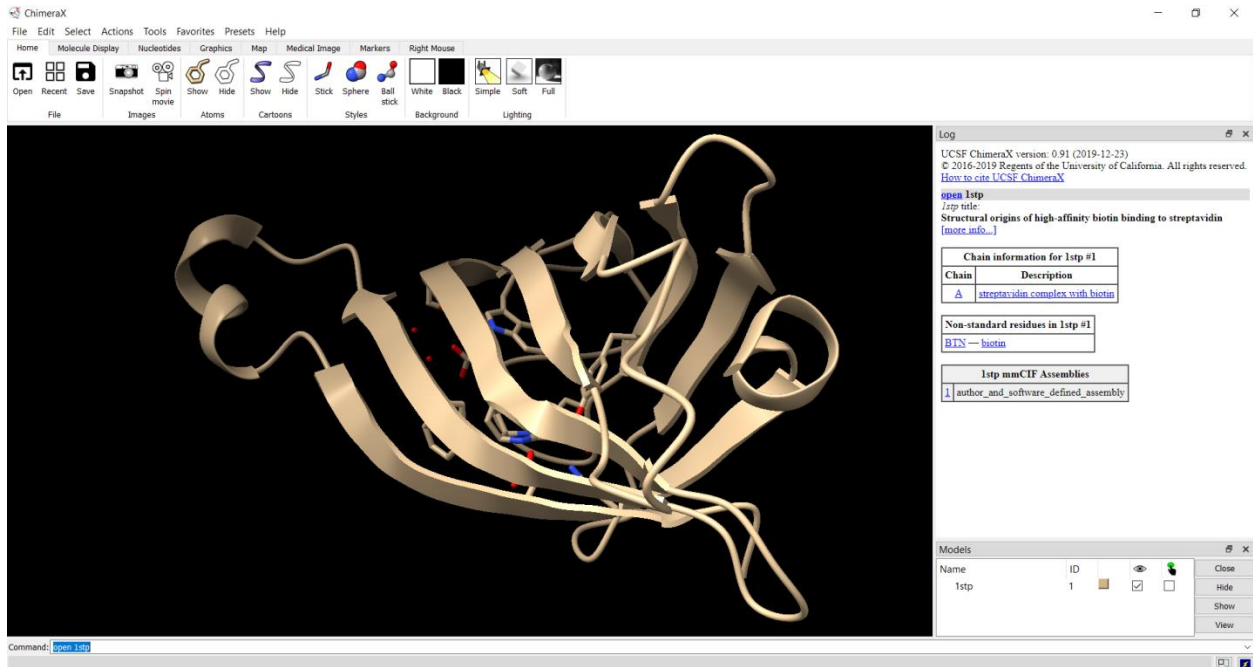
Note that the ChimeraX window is laid out as a series of individual panes (like the **Log** pane and **Models** pane on the right). You can move these around, re-size them, or remove them. We'll play with this more later. The grey area is the working pane where the protein will be once we load it.

- Let's first load our practice protein 1stp. There are two ways to do this:
 - If you've downloaded the protein, you can use the "Open" button on the "Home" tab, then select the 1stp.pdb file.
 - If you are connected to the internet, you can instead call the file directly by typing the following command into the command line:
 - `$ open 1stp`

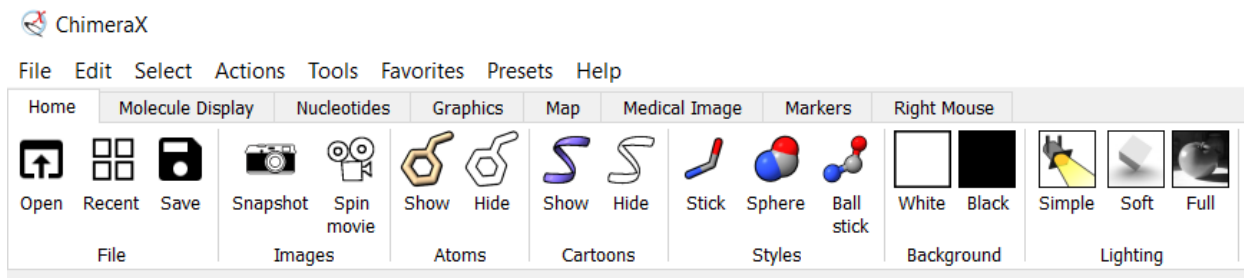
Note: From here on out, I will be indicating all line commands with the "\$" notation. You don't actually type the "\$," just the text that follows. For those of you not familiar with programming, the "\$" is the standard prompt in the Unix environment, and I just use it as a convention.

Another Note: Whenever you are using line commands, you must (in most cases) use it exactly as written. Any mistakes and the command will fail. While some people don't like command line interfaces, they become much faster with practice as opposed to clicking through menus to find the right button. I suggest learning to master the command line functions! Also note the **Log** pane lists all commands used during the session. There are also some commands that can only be done by command line, so it's best to at least get comfortable with them.

- If you correctly loaded the 1stp file, it should look something like this:



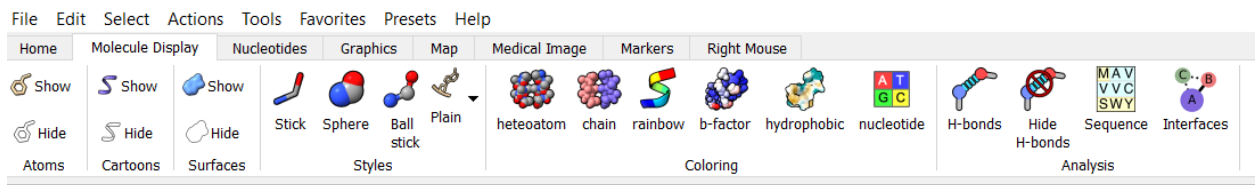
- Let's start to play with the mouse and move this puppy around:
 - Left-click-holding on the protein and allows you move it about the vertical and horizontal axis
 - Left-click-holding in the black region allows you to rotate the protein clockwise or counterclockwise
 - Spinning the middle wheel allows you to zoom in and out
 - Right-click-hold allows you to move the center of the protein about the screen
- Next, play around with the different buttons on the **Home Tab** and see what happens:



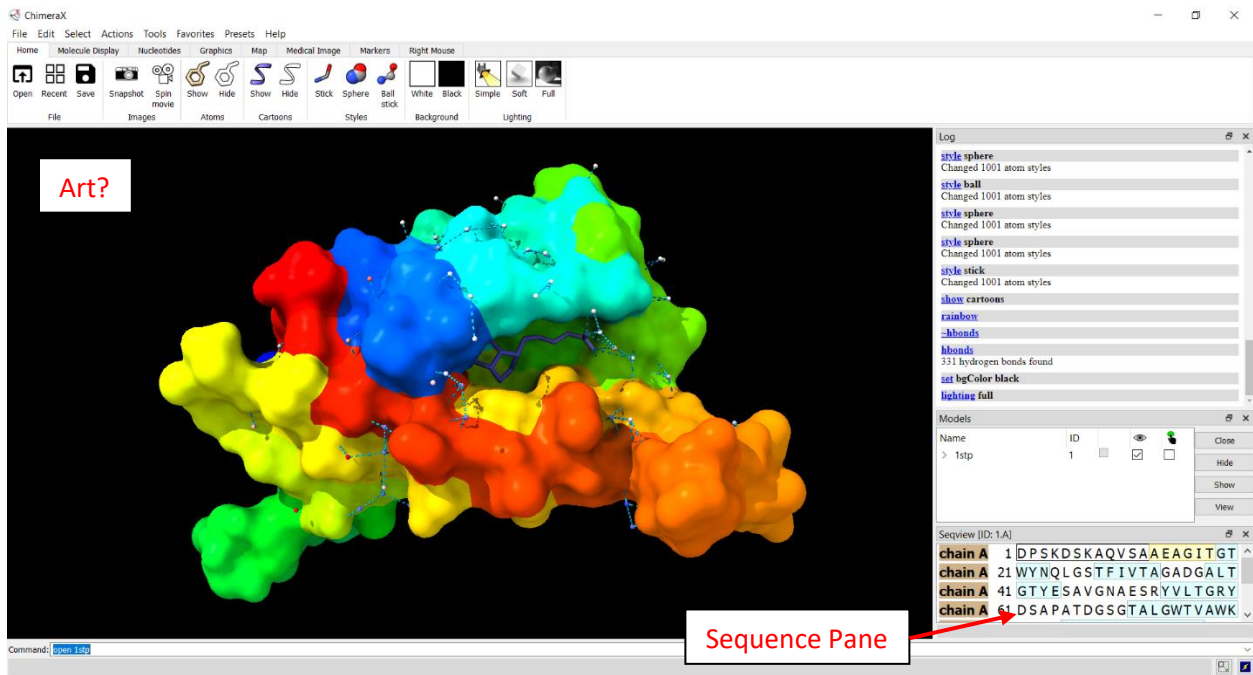
- As you play with the buttons, note a few things:
 - The **Log** pane will list what equivalent command is being typed. For example, when you hit the “White” button in the “Background” section, you’ll see the following popup in the log which is the equivalent line command entry:
 - `set bgColor white`

Note: If you click on the blue underlined text (like “set” above), you will get the help menu for that command. Paying attention to the log is also a great way to learn the line commands!

- You'll also notice as you play around, that the "Styles" only applies to the molecules, not to the ribbon backbone.
- Feel free to take a snapshot or make a movie of your little protein.
 - By default, these will be saved to your desktop as a PNG (pictures) or MP4 (movies). If you want to save it elsewhere or in a different format, you'll have to do it by the command line interface.
- Next, hit the **Molecule Display** tab and start to play around with the buttons here as well



- Note that nucleotide related functions won't work here, as there are no nucleotides in the structure of 1stp
- Also note that hitting the **Sequence** button brings up a new window pane on the right.
- If you keep mashing those buttons, you'll end up with some pretty abstract looking art, maybe even something like this:



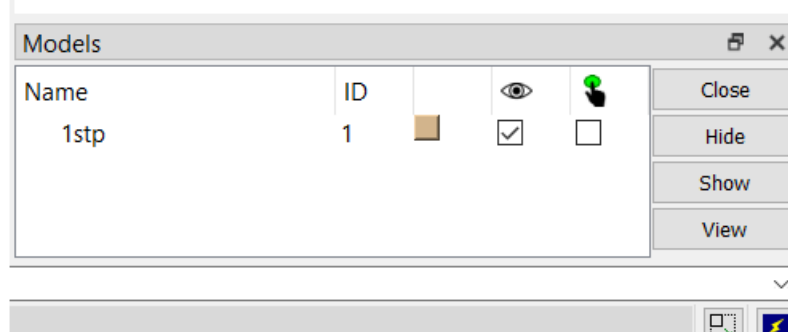
- While this may be "art" to some people, you've probably gotten to a mishmash that isn't too helpful.
- To return us to the initial pre-set, use the following menu selections:
 - **Presets → Original Look**

Note: From here on, I will always indicate drop-down menu selections using the same bold purple text as above, with the "→" indicating the next subsequent selection

- Continue to play around with the different **Presets** that are available.
 - Also experiment with combining different presets with different buttons found on the “Home” and “Molecule Display” tabs as before
 - Remember that you may not see changes for the nucleoside selections, since there are none in this particular protein.

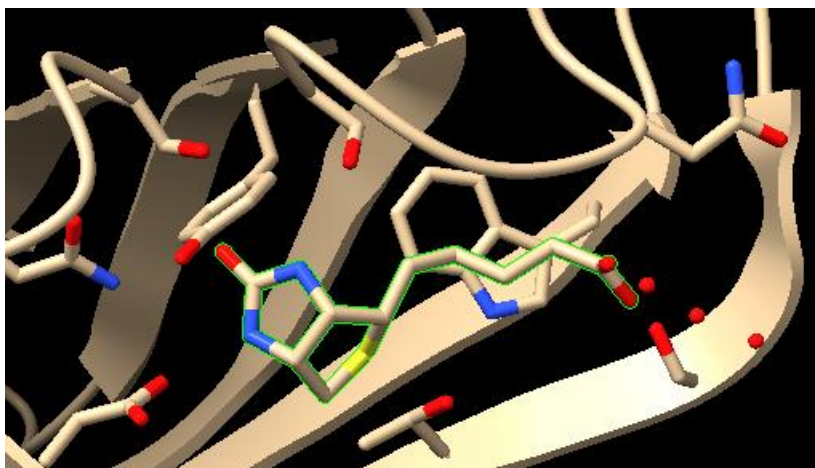
Processing a Protein Part 1: Removing Stuff & Colors

- Restart the ChimeraX program and load the 1stp protein as before
 - You'll note that you can also call the protein from the start up screen. ChimeraX keeps a list of the most recent proteins you've worked with, and you can quickly re-load them this way.
- Let's start changing some colors here. First, we want to remove all of the small molecules and leave only the protein. You can do this by clicking the "Hide" atoms in the "Home" tab.
 - You can also accomplish the same in the command line by typing:
 - `$ hide atoms`
 - Now that you have just the protein ribbon, note the small tan-colored box sitting in the **Model** pane

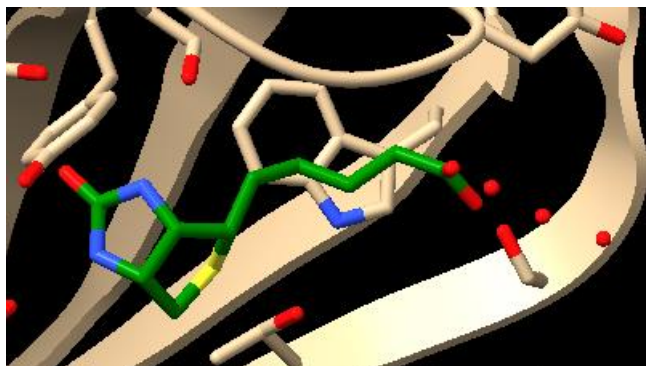


- You can click the tan colored box to bring up a list of available colors. Color your protein green (any shade is fine). Note that the colored box will change to match what you have selected
- Let's also change colors by the command line. You can get the full list of colors to appear in the **Log** pane by typing the following:
 - `$ color list`
- Each color is next to the name required in the command line. For example, we see the color "lightskyblue." You can change to this color with the following:
 - `$ color lightskyblue`
- Let's return to the original coloring scheme. Do this using the menu again:
 - **Presets → Original Look**
- Now, select the color red, either by choosing it from the box in the **Models** pane or using the line command:
 - `$ color red`
- You'll note that when you change color this way, everything turns the color you selected—protein, ligands, everything! Now we need to select individual parts so we can tell the program to only color what we select.
- Return again to the original coloring scheme.
 - **Presets → Original Look**
- Let's start by selecting the sulfur atom of the ligand, which will appear yellow. If you hover your mouse over the atom for a second, a small dialog box will appear that says something like this:
 - `/A BTN 300 S1`

- This means chain A, molecule code BTN, atom code #300, sulfur #1
- Hold CTL (control) and left click to select the sulfur atom. If you did this correctly, it should be outlined in green.
- You can expand your selection by hitting the “up” arrow on the keyboard. This should highlight the entire ligand.



- Now, let’s change just the ligand green. The only way to change a selection is using the command line:
 - `$ color sel green`
 - The word “sel” is short for “selection.” You are telling the program to change what you’ve got highlighted to the color green
- Now that the ligand is only green, it’s difficult to see the non-carbon atoms. You can add standard heteroatom coloring by next typing the following:
 - `$ color sel byhet`
 - Note that hitting the “Heteroatom” button on the “Molecule Display” tab will also do the same thing.
- To deselect your current selection, you need to CTL + left-click in a blank space, or type:
 - `$ sel clear`

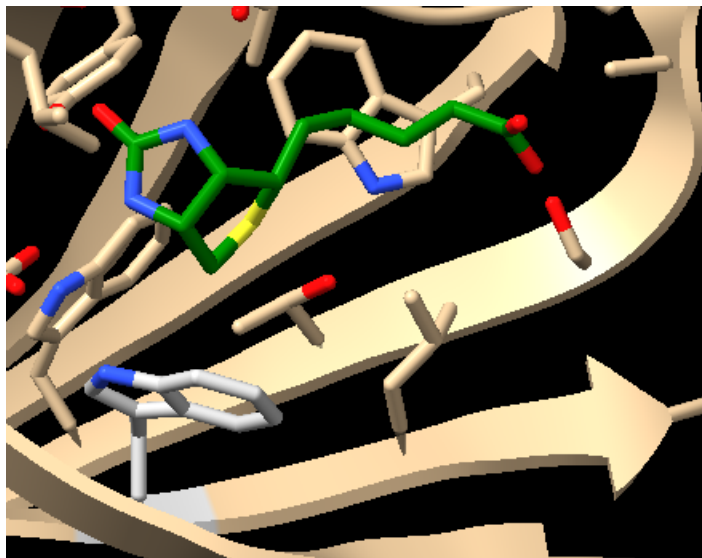


- You’ll notice little floating red dots near the ligand. Those are water molecules that are near the ligand, and ChimeraX will display these by default. To see all of the water molecules in the crystal, type the following command:
 - `$ show solvent`

- All these water molecules don't usually matter (unless a water molecule is specifically required in the mechanism). We can remove all of the water molecules with the command:
 - `$ delete solvent`

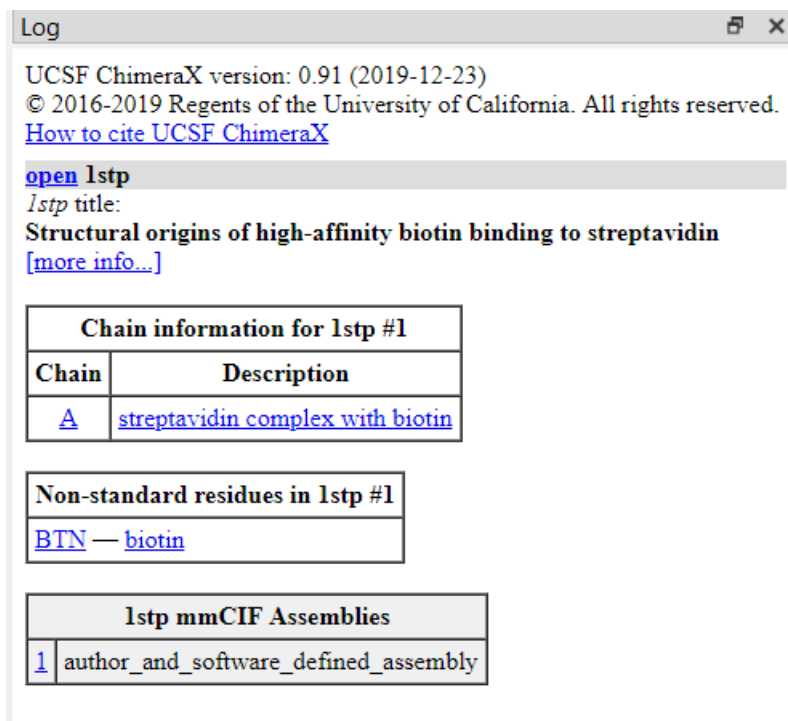
Note: Although ChimeraX has an "Undo" button, deleting things is always permanent and cannot be undone. Use the "delete" command carefully! The "hide" command can often be used instead.

- Since we have made a lot of changes, let's save our progress so far. You can save your session using the **File → Save** command. This will save your session as a .csx file. Name it something like "1stp part1"
- Next, let's say we want to select and color a nearby residue. You'll notice a tryptophan residue at the "bottom" of the structure, "south" of the sulfur atom. Let's turn this guy silver. There are several ways to do this:
 - You can CTL + left-click one atom on the residue, then hit the "up" arrow until the entire residue is selected
 - If you hover over the mouse over the residue, you'll see a pop-up up that says the residue is "A/ TRP 108 CD2" which tells us that this is residue #108 in the protein. You can select by residue using the command:
 - `$ sel :108`
 - The ":" in the command here means "residue or atom"
- Once you've selected just this tryptophan, color it silver and then add the heteroatom using the following commands:
 - `$ color sel silver`
 - `$ color sel byhetero`



Processing a Protein Part 2: How to Show, Hide & use Labels

- Restart the program and load the starting 1stp PDB file to reset everything
- Pay special attention to the initial output in the **Log** pane:



- There are a few important things to take note of here:
 - There is only one chain, called “A” by default
 - The ligand is in the “Non-standard” residues, and has the identifier BTN
 - Note that this BTN identifier is also on the PDB web page!
- First, remove the water molecules with the command:
 - \$ delete solvent
- Next, let’s add labels to the ligand and the residues which are shown. Use the command:
 - \$ label @@display
- This should cause labels to pop up in the default color. You can make changes to the labels such as:
 - \$ label height 1.0
 - Putting different numbers in the “1.0” changes the size of the font
 - \$ label color red
 - Makes the label font the input color (red here).
- You’ll note that the ligand is marked “BTN 300.” This is not a useful label, since it’s not a residue. You can remove this label using the command:
 - \$ label delete :BTN
 - Note you can use the word “ligand” instead of “:BTN” here
- If you make a mistake or you want to remove all labels, just use:
 - \$ label delete

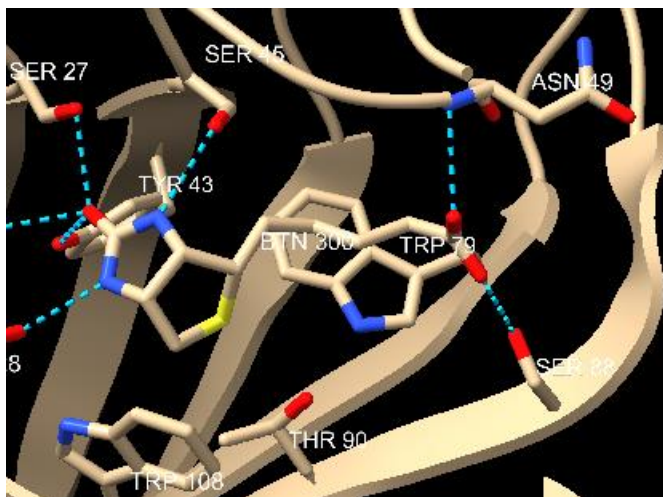
Processing a Protein Part 3: Finding H-Bond Contacts

- Restart the program and load the starting 1stp PDB file to reset everything. Again, the easiest way is usually to do it by line command:
 - \$ open 1stp
- Remove the water molecules again with:
 - \$ delete solvent
- First, let's find the H-bonds in the structure. You can do this by either hitting the "H-bonds" button on the "Molecule Display" tab, or you can type the command:
 - \$ hbonds
- You'll notice that this will show all estimated H-bonds in the protein structure

Note: The H-bonds that ChimeraX shows are estimates based on an internal algorithm. We don't know for sure if these are correct without additional data. Still, the algorithm that ChimeraX has a good history of predicting these interactions, so it's considered a reasonable approximation. Remember, however, that we don't know anything for sure without experimental validation.

- Since we're only interested in the H-bonds in the molecule, let's remove the current H-bonds by either hitting the "Hide H-bonds" button in the "Molecule Display" tab, or use the command:
 - \$ hide hbonds
 - Note that you can also use the following command to hide things:
 - \$ ~hbonds
- To only find contacts between the ligand and nearby residues, choose one of the following options:
 - Select the ligand using CTL+left-click, pressing "up" on the keyboard to highlight the full ligand structure, then hit the "H-bonds" button on the "Molecule Display" tab
 - Use the command
 - \$ hbonds :BTN
 - Use the command
 - \$ hbonds ligand
- To better keep track of what's going on here, let's add some labels with:
 - \$ label @@display
- We can make a few other changes to the H-bonds. Try these:
 - \$ color hbonds magenta
 - Should turn all of the H-bonds magenta. You can see the list of colors with:
 - \$ color list
 - \$ hbonds :BTN showdist true
 - Should remake the H-bonds in the default color, but also have the distances marked in Angstroms
 - \$ hbonds :BTN showdist true color magenta
 - This will combine the above commands.
- For now, let's just use the default coloring, best achieved with the command:
 - \$ hbonds :BTN
- Since we've done a lot of work here, let's save our session using the **File → Save** command, naming the file H-bonds.

- You'll notice that there is an H-bond that appears to contact the ribbon backbone near Asn49. We need to know whether or not we're making contact with the backbone N or O in the amide. To do this, we need to show this part of a protein in stick form. There are a couple of ways to do this:
 - Select an atom in Asn49, then hit the up arrow until the entire residue is selected. You can then hit the "Hide Cartoon" button in the "Home" tab to hide that part of the ribbon, leaving the backbone exposed.
 - You can use the command:
 - `$ hide :49 cartoon`



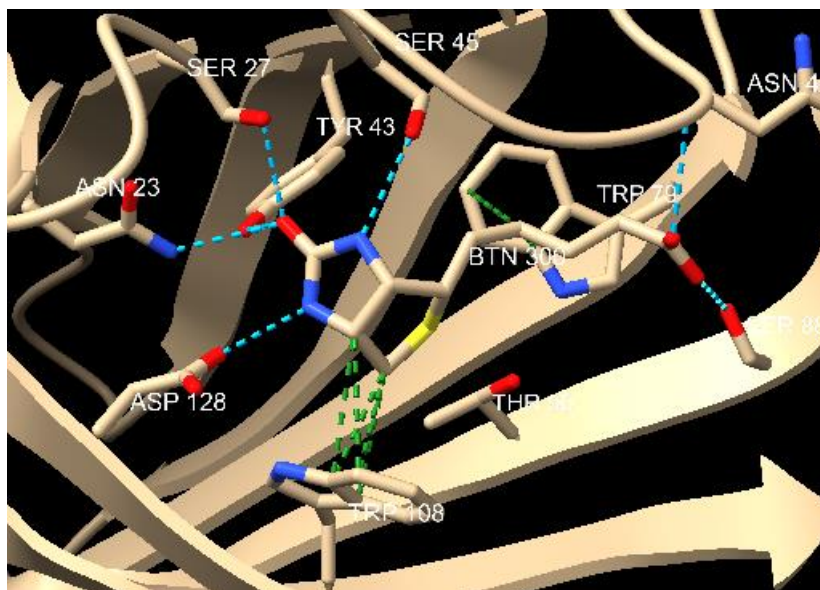
- You can see that the ligand is forming an H-bond to the N atom. This information will be useful when we make a 2D pharmacophore later. For now, you can leave it looking like this, or you can show the cartoon again, either by clicking the "Show Cartoon" in the "Home" tab, or using the command:
 - `$ show :49 cartoon`

Processing a Protein Part 4: Finding Non-Polar (vdW) Contacts

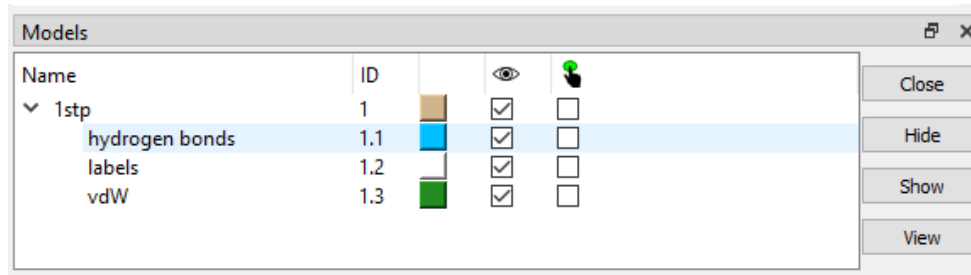
- Open the “H-Bond” session that you saved earlier. We want the protein to have labels and H-bonds shown.

Note: There are some commands we will use here that can't be fixed with the “Undo” command. It's best to have a saved session ready to go so you can reload if you get stuck!

- Unfortunately, there is no easy way for ChimeraX to determine a non-polar interaction, so we'll have to do it manually. Most vdW interactions range from 2-4 Å, depending on the size of the atoms involved. So, first let's display every connection between the ligand and the protein at a distance of 3.8 Å using the command:
 - \$ contacts /A@C* restrict :BTN@C* distance 3.8 reveal true name vdW
 - The 3.8 Å number is arbitrary here, and you can change it. However, 3.8 Å should give you all reasonable interactions (though there could be more).
 - You can repeat the command with different numbers (instead of 3.8) to see what happens if you want. Try adding a bigger number if you don't get any results using 3.8. Numbers like 4.0 or 4.2 are also reasonable.
 - The “C*” means “all carbon atoms.” So, the translation of the command is something like “show all contacts between carbons on chain A (/A@C*) and ligand BTN carbons (:BTN@C*) at a distance of 3.8 Angstroms.”



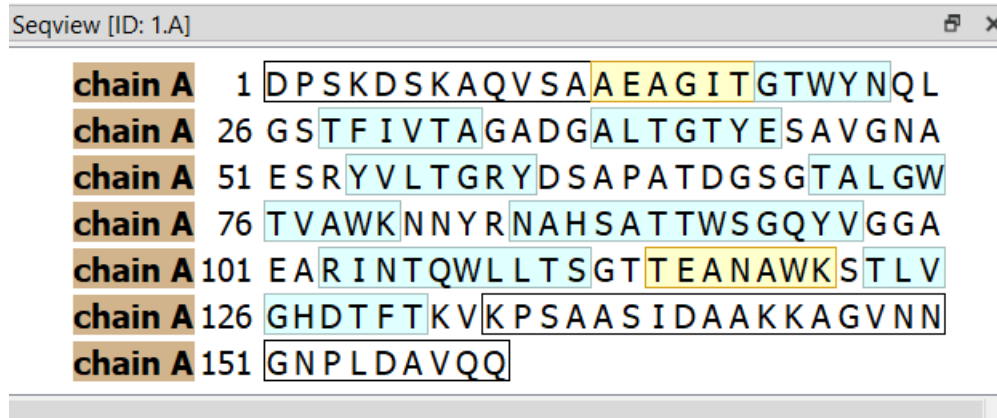
- ChimeraX indicates that our ligand is making several contacts with Trp108 and Trp79.
 - If you want to change colors, the easiest way is to click the colored box in the **Models** pane on the vdW line (which is what we named these contacts). Be sure that the arrow next to the 1stp is clicked to show the list of all associated objects



- If you don't have a **Models** pane open, you can bring it up using **Tools → General → Model Panel**
- Lastly, you'll notice that Asn49 and Thr90 are shown, but they aren't making any contacts. Let's hide them to clean up the picture with the following commands:
 - `$ hide :90 :49`
- We can remove the leftover labels we don't want with:
 - `$ label delete :90 :49 :BTN`
- Save this session as "Ligand Contacts" for future reference.

Processing a Protein Part 5: Using the Sequence Pane

- Open a fresh 1stp file as before
- Bring up the **Sequence** pane by pressing the “Sequence” button on the “Molecule Display” tab.



- Some important things to note:
 - The first chain is labeled “A” by default. Since there is only one chain for this protein, you only see chain A.
 - The yellow regions correspond to α -helices
 - The blue regions correspond to β -pleated sheets
 - The white regions correspond to sections with no secondary structure
 - The white regions with a black box around them correspond to disordered sections which were not determined by X-ray (you can’t see them!)
- Let’s turn the α -helices red. Double-left-click on one of the yellow regions to select them. Then enter the command
 - \$ color sel red
- Next, let’s select the second β -sheet, starting on residue 28. Highlight this sequence by holding-left-click and dragging over the TFIVTA sequence. Then, color it blue with:
 - \$ color sel blue
- You can use the **Sequence** pane to easily highlight sections of a chain, or find specific sections/residues.

Processing a Protein Part 6: Working with Surfaces

- Open a fresh 1stp file as before
- Get rid of all stick atoms one of the following ways:
 - Click the “Hide” atoms button on the Home or Molecule Display tab
 - Type the command:
 - \$ hide atoms
- Let’s now look at the polarity of the protein surface. You can calculate the molecular lipophilicity potential surface one of the following ways:
 - Click the “Hydrophobic” button on the coloring section of the Molecule Display tab or instead you can type the command:
 - \$ mlp
- You will get the protein surface mapped by lipophilicity, with dark cyan being the most hydrophilic and goldenrod being the most lipophilic surface (with white being in between).
 - An easy way to remember is that the oily spots are the color of cooking oil!
- Let’s see where our ligand sits on the surface. You can show the ligand using the command:
 - \$ show ligand
 - You’ll notice the aliphatic chain of the ligand fits into a greasy region!
- Since our surface is solid, we can’t see the residues or protein structure underneath. You can display both by adding transparency with the following command:
 - \$ trans 50
 - The “50” here is a percent. Feel free to put in different numbers and see what you like.
- Try the following commands to play with your surface and see what happens:
 - \$ surface hide
 - \$ surface color silver
 - \$ surface style mesh
 - \$ surface style dot
 - \$ surface style solid
 - \$ surface ligand @<5 visible 1
 - \$ mlp
 - \$ surface ligand @<5 visible 1
- Remember you can always revert to the original style with Presets → Original Look
- Note that the b-factor button (next to the hydrophobic button) is for the **Debye–Waller factor**. In terms of proteins, this usually means how “floppy” or “disordered” that part of the protein is.
 - The default blue color means highly ordered/rigid, while the red color means disordered/floppy.
 - This color scheme isn’t particularly useful for our purposes, but is interesting to those that study protein structure and X-ray crystallography.

Processing a Protein Part 7: Working with a Metalloprotein

- Let's play just a minute with a metalloprotein so we can learn how to deal with a metal atom
- Load the protein 3hku with the following command:
 - \$ open 3hku
 - This protein is carbonic anhydrase II with the drug topirimate bound, which is used to treat seizures.
- Depending on your ability to see color, it can be difficult to see the ligand versus the residues. Using starkly contrasting colors can help identify things more easily, especially for the colorblind or partially colorblind. Try the series of commands below to modify the color scheme and get rid of the water molecules:
 - \$ color ligand green
 - \$ color byhet
 - \$ delete solvent

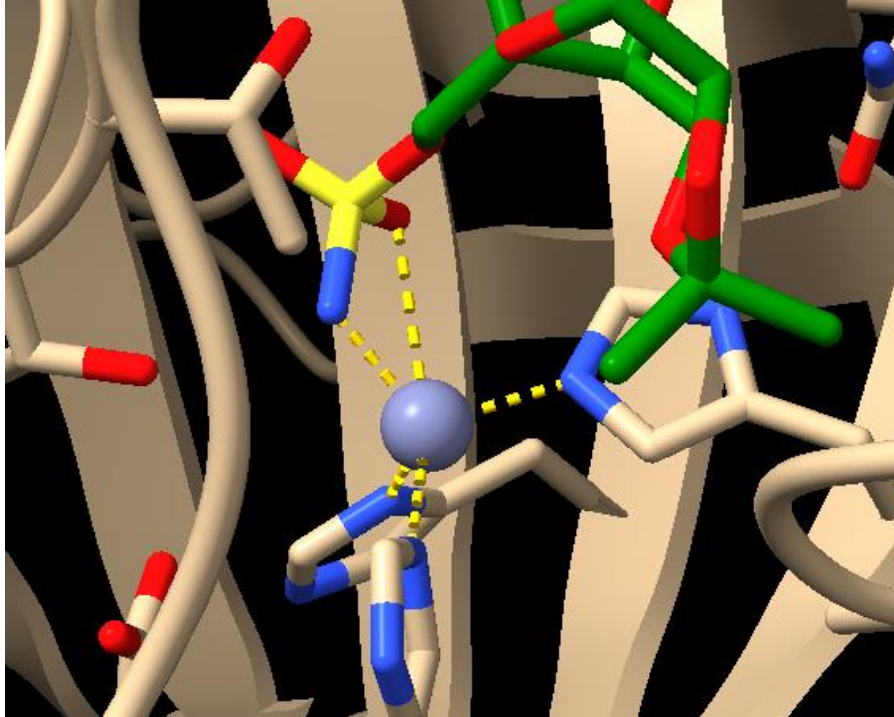


- Also note the presence of the zinc metal here, which appears a grey sphere! Many proteins have metals, and they appear as colored spheres in ChimeraX

Note: You should notice purple dotted lines between the zinc atom and nearby residues. These are expected metallic bonds calculated by ChimeraX. By default, ChimeraX will mark these when you open a PDB file containing a metal. Recall that because metals have a positive charge, they will coordinate with electronegative atoms—most usually N, O and S.

- Let's try to manually look for atoms that are interacting with the metal. We do this because ChimeraX doesn't always recognize likely metal-ligand bonds, although it's usually pretty good with metal-protein bonds
- First, let's get rid of the purple metal bonds ChimeraX has marked with the following command:
 - \$ hide pseudo

- The “pseudo” is short for pseudobonds, which is what ChimeraX calls the purple lines.
- Next, we’re going to use a fancy “contacts” command to identify any N or O atom within 3.5Å of the zinc atom:
 - \$ contacts :ZN restrict #1@N*,O* distance 3.5 reveal true name metalbonds color gold
 - The “#1@N*,O*” means any nitrogen (N*) or oxygen atom (O*)
 - The 3.5 Å is arbitrary, and you can change it, Note that 3.5 works well for most metal-ligand and metal-protein bonds. Any number from 3-4 is usually good.



- Note that we have a new bond to the oxygen of the ligand that ChimeraX didn’t have marked before. This means the zinc is likely coordinating to this atom, too!
 - The most common ligand groups to coordinate to metals are amines and carbonyls.

Note: Metals can coordinate to many, many atoms. It’s not uncommon to have 4-8 interactions. While this might sound like it would cause steric crowding, metal atoms are much, much larger than carbon atoms and can therefore interact with lots of nearby electronegative atoms.

- Check out the 2D pharmacophore at the end of this tutorial for an example of a 2D model with a zinc atom in a different protein making multiple contacts to protein residues and a ligand.

Processing a Protein Part 8: Deleting Extra Chains and Cofactors

It's pretty common for proteins to be crystallized as dimers, trimers, or larger "mers." They are also often crystallized with salts, additives, and co-factors. Most of these molecules just add to the clutter and can be removed to help focus on the ligand/drug of interest.

- Let's practice with a protein crystal structure tetramer. Load the 1hw1 protein structure with the following command:
 - \$ open 1hw9
 - This is the crystal structure of HMG CoA Reductase bound with the drug simvastatin, a drug designed to prevent cardiovascular disease.
- The log pane shows up that there are four chains and two ligands present

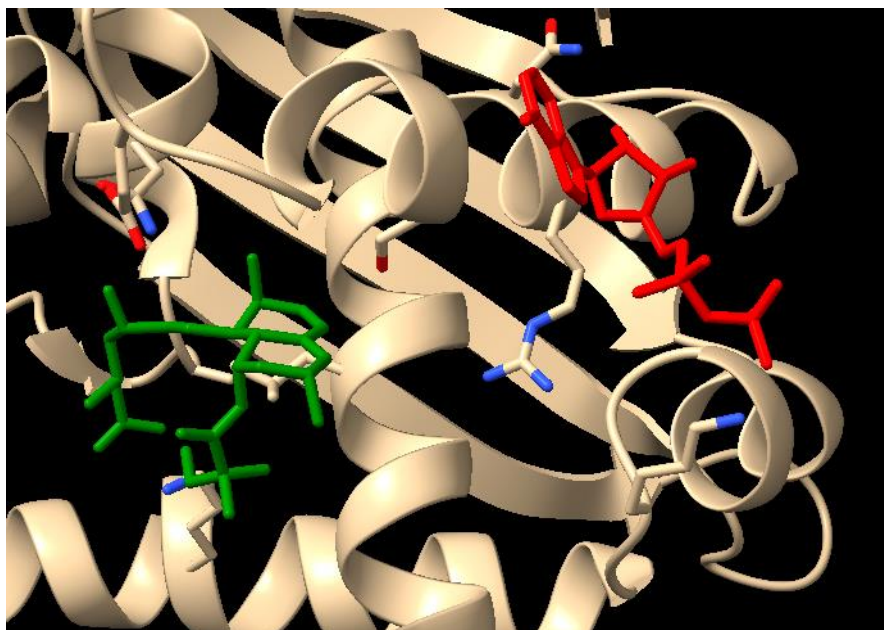
Chain information for 1hw9 #1	
Chain	Description
A B C D	HMG-coa reductase

Non-standard residues in 1hw9 #1	
ADP	— adenosine-5'-diphosphate
SIM	— Simvastatin acid (dimethyl-compactin)

- There are 4 chains here, named A, B, C and D. If you rotate the structure, you'll notice that it's symmetrical and all 4 chains are the same.
 - It is possible to have a structure with different chains, but it's far more common to have them be duplicates. It's generally pretty easy to see the symmetry, as different chains have very different structures.
- Let's get rid of the other chains with the following command:
 - \$ delete /b-d
 - Here's we're telling it to delete chains b through d.

Note: You should notice purple dotted lines between the zinc atom and nearby residues. These are expected metallic bonds calculated by ChimeraX. By default, ChimeraX will mark these when you open a PDB file containing a metal. Recall that because metals have a positive charge, they will coordinate with electronegative atoms—most usually N, O and S.

- You'll note that there are 2 small molecules here, ADP and SIM. Let's color them so we can more easily identify them:
 - \$ color :SIM green
 - \$ color :ADP red



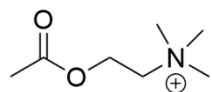
- These molecules are not close enough to interact with each other, so we can go ahead and delete the ADP co-factor
 - \$ delete :ADP

Note: Most co-factors bind allosterically and do not interact with the ligand, so we can usually delete them. Common co-factors and additives include cations (sodium and calcium most commonly), anions (usually chloride), sugars, ADP, ATP, orotic acid, steroids, NAD, NADP, FAD, SAH, SAM and vitamins. It's pretty common for proteins to have multiple co-factors around.

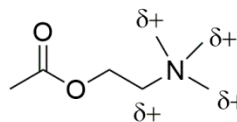
- It is possible for a drug to interact with a cofactor if it's close enough. If you think your ligand/drug might be interacting with a cofactor, then leave it in there and look for H-bonds and vdW contacts. If it turns out your drug/ligand doesn't interact with the cofactor, you can always delete the cofactor later.

Processing a Protein Part 9: Recognizing Cation-Pi Interactions

Before we start talking about cation-pi interactions, we first need to deal with the issue of where the positive charge actually resides. In general chemistry and organic chemistry classes, we use formal charge as a way to account for electrons by putting positive/negative charges on certain atoms. The problem with this formalism is that it gives the wrong impression as to which atoms are actually cationic. For example, consider the structure of acetylcholine with a formal charge versus where the charge resides:

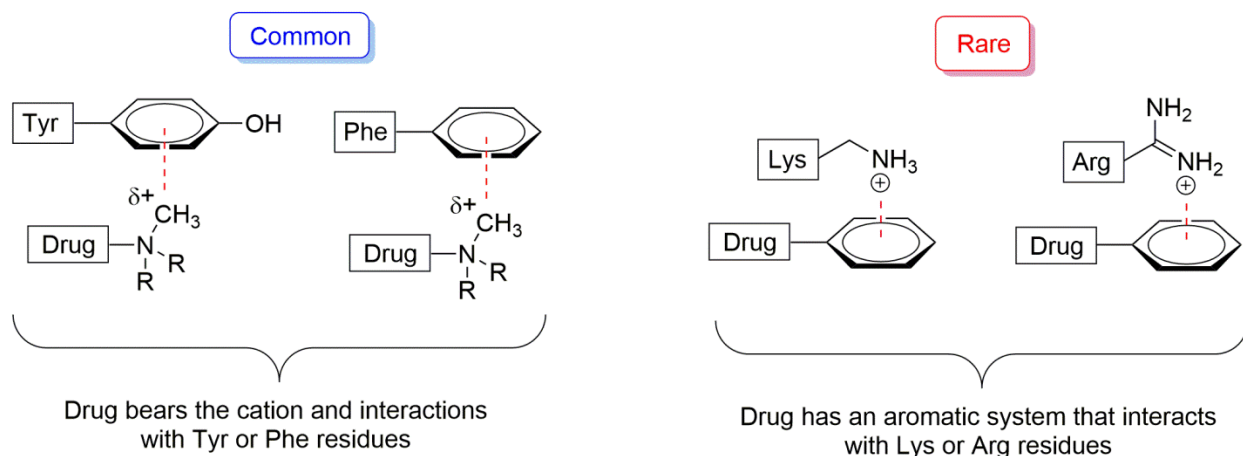


"Formal" charge showing cation on N atom



Where the positive charge actually resides

If you have unhindered carbon groups (particularly methyl groups) attached to a quaternary nitrogen, then the positive charge on this unhindered methyl group will sit on the pi system. In contrast, a primary amine ($R-NH_3^+$) will leave significant charge near the nitrogen, so the nitrogen atom can sit on an aromatic system. This leads to 2 possible ways that you'll see cation-pi interactions:

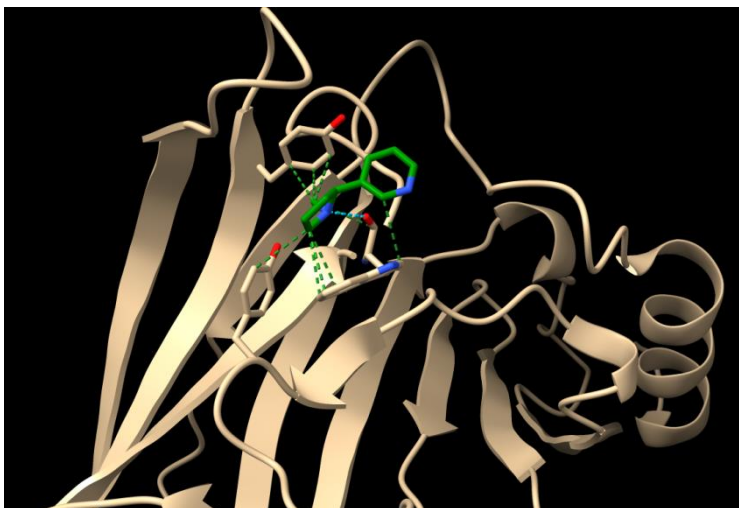


Note: When the cation is on the drug (the "common" example above), interactions are common with unhindered carbons (especially methyl groups) bound to the nitrogen atom. Just look for nearby Tyr or Phe rings. Tryptophan and Histidine rings can serve as a pi-system for a cationic drug, but they are less common. If the drug has a phenyl system, then it is possible for a nearby Lys or Arg to form a cation-pi interaction, but this is pretty rare. An example protein with a cation-pi interaction is shown on the next page.

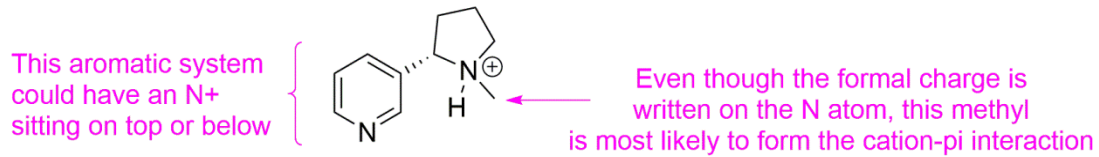
Example of a Cationic Drug Interacting with an Aromatic Protein Residue

Note: These are the most common and easiest to find, as our normal processing will reveal these when we're looking for non-polar contacts as you'll see below. Remember, however, that many ligands won't have a cation-pi interaction

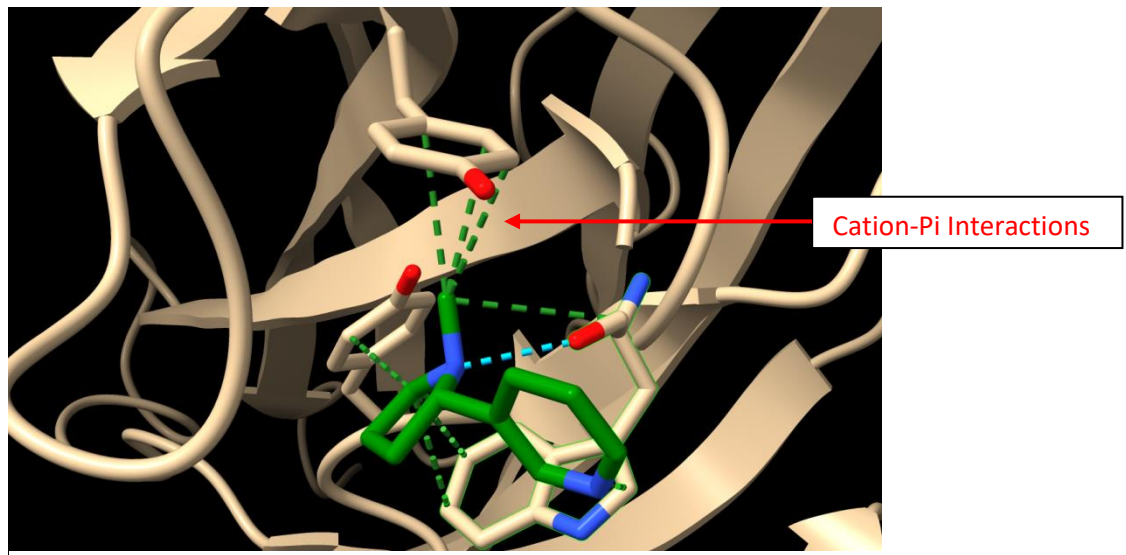
- Load the crystal 1uw6 with the following command:
 - `$ open 1uw6`
- Look at the size of this puppy! You'll note that there are multiple chains here, named A through T. The easiest way to delete everything but just chain A is to do the following:
 - `$ delete /b-t`
 - Here we're telling the program to delete chains B through T.
- Change the settings to something more familiar with:
 - **Presets → Original Look**
- The ligand here is nicotine, and the code (in the log pane) is NCT. Let's first find the H-bonds and vdW interactions with the following commands:
 - `$ delete solvent`
 - `$ hide atoms`
 - `$ hbonds ligand reveal true`
 - `$ contacts /A@C* restrict :NCT@C* distance 3.8 reveal true name vdW`
- You'll note that there appears to be some contacts between the ligand and the backbone of the nearby tryptophan residue. You can show the backbone residues by selecting the residue by holding down CTL and left-clicking the residue, or by selecting it with the command:
 - `$ select :143`
 - After you select it, hit the "Hide" on the cartoon section of the "Molecule Display Tab"
- Lastly, let's add a bit of color to the ligand with the following:
 - `$ color ligand green; color byhet`
 - The semicolon here is a way to combine commands together, and gives the same result as if you did them one at a time



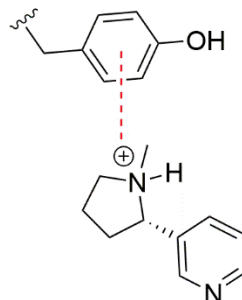
- Nicotine has both an aromatic system (the pyridine) and an ammonium cation at physiological pH (7.4). This means nicotine has 2 potential places that it could form a cation-pi interaction with the protein



- Looking at the pyridine system, there doesn't appear to be any cationic residues nearby, so nothing to look for here (as expected because these interactions are rare). You can use the following command to look for these rare interactions:
 - \$ contacts /A@N* restrict :NCT@Car distance 5 reveal true name catpi
 - This command looks for any nitrogen atoms within 5 Å of an aromatic carbon (Car) in your ligand (NCT here). We get a "No distances" result because there are no such interactions here. Most ligands won't have this kind of interaction.
- Looking at the pyrrolidine ring, however, you'll notice that the N-Methyl has several contacts with a nearby Tyr residue. This is a cation-pi interaction!

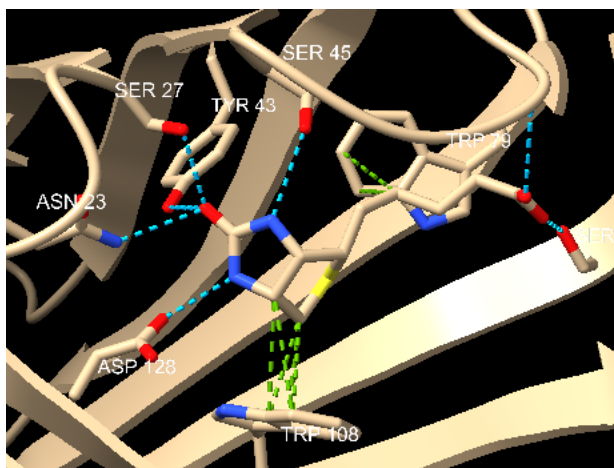


- When you show these interactions, the standard way to do this is to draw a dotted line from the positive sign to the center of the aromatic ring like this:

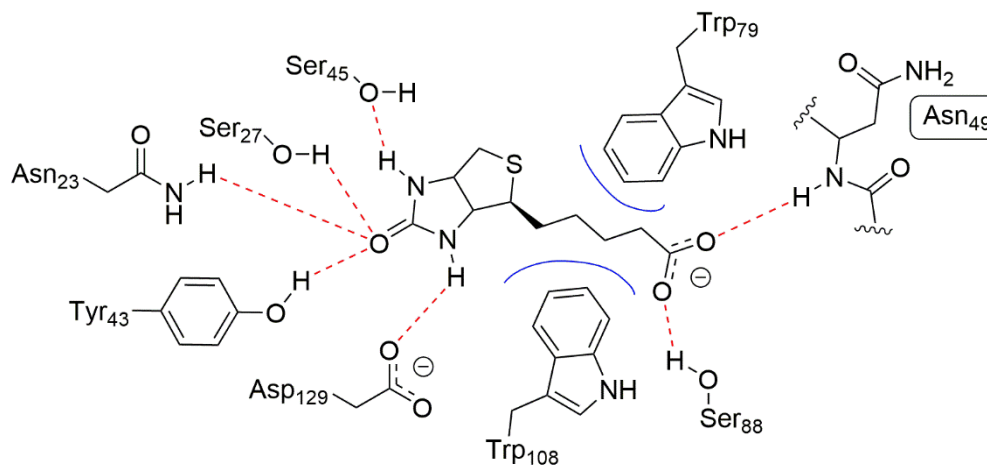


Processing a Protein Part 10: Building a 2D Pharmacophore

- It's common to publish 2D pharmacophores to simplify a binding region. While no 2D picture can fully capture a 3D space, you can communicate the general idea to an audience without using complicated software like ChimeraX
- Load the "Ligand Contacts" session that you processed before with the 1st structure



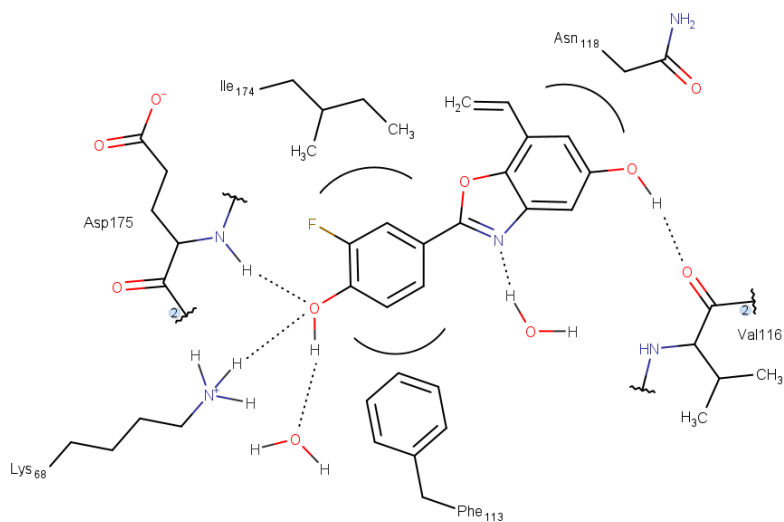
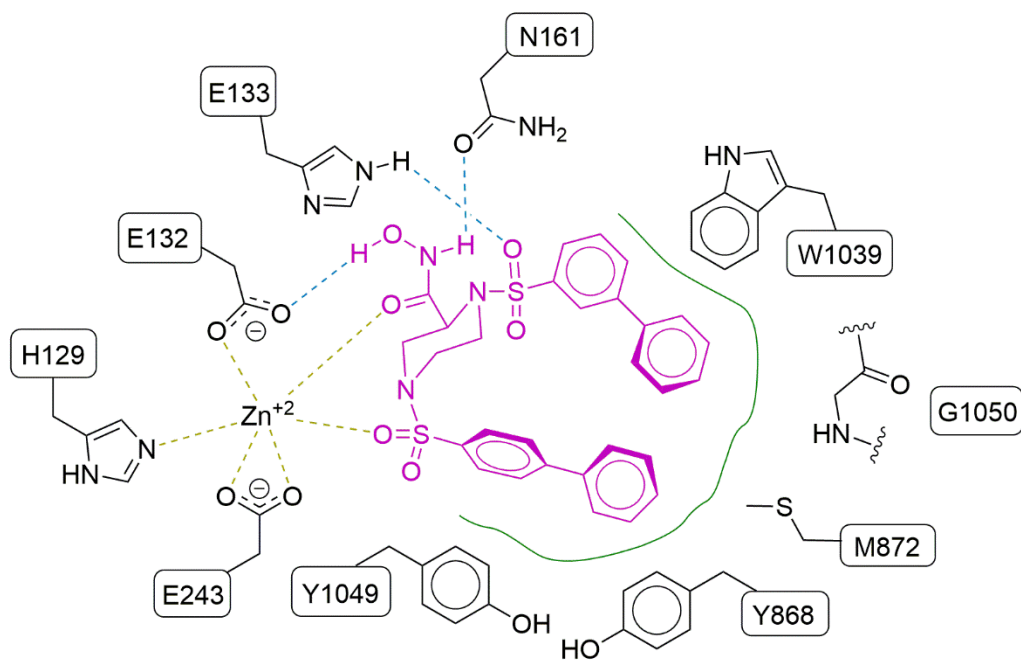
- Using ChemDraw or a similar program, do your best to make a 2D representation.
 - Note that by convention, vdW interactions are usually shown as surfaces (curved lines), not contacts (like the green above). You can still show them as contacts if you want.



- You can see an example above, where the polar contacts are shown as red dashed lines, and the vdW regions are shown as blue lines. Not all ligands have h-bonds, but they will have vdWs.

Note: There is no single "right" answer about possible interactions. For example, you could make the argument that the sulfur atom of Biotin is H-bonded to the Thr90 residue, as it's within the range for a reasonable H-bond. Also remember, **just because a protein crystallizes in one form doesn't mean that the crystallized form reflects a biologically relevant one!** As long as your interactions are reasonable, then your pharmacophore will be reasonable. We'll discuss how to experimentally test predicted pharmacophore in later chapters, particularly chapter 13. Also, note the protonation states here! You cannot see H atoms in a crystal, so you'll have to infer their relative positions.

More 2D Pharmacophore Examples from Past Research Projects:



Additional Resources for the Curious:

You can find an extensive guide and several different tutorials on the ChimeraX website here:

<https://www.cgl.ucsf.edu/chimerax/tutorials.html>

ChimeraX can do much more than what was covered here. If you will be doing anything in the biochemical space, this is a really cool program to know, and something you can add to your resume. Protein movies are really cool to add to any slideshow, too!