

# CMPyMOL User Guide<sup>1</sup>

BY VENKATRAMANAN KRISHNAMANI

January 11, 2016

<sup>1</sup><https://goo.gl/VyMXPd>



# Contents

<b>1 Installation Instructions</b>	<b>3</b>
1.1 Mac OS X . . . . .	4
1.1.1 Prerequisites and Initial Setup . . . . .	4
1.1.2 Running CMPyMOL . . . . .	5
1.2 Linux . . . . .	5
1.2.1 Prerequisites and Initial Setup . . . . .	5
1.2.2 Running CMPyMOL . . . . .	6
1.3 Windows . . . . .	6
<b>2 Load PDB or Precalculated Contact Map</b>	<b>7</b>
2.0.1 Command for Launching CMPyMOL . . . . .	8
2.0.2 PDB file format . . . . .	8
2.0.3 Load PDB file . . . . .	9
2.0.4 Calculate Contact Map . . . . .	9
2.0.5 CMPyMOL & PyMOL Window . . . . .	10
<b>3 Functionality</b>	<b>11</b>
3.1 CMPyMOL Main Window . . . . .	12
3.2 Selections . . . . .	13
3.3 Overlays . . . . .	13
3.3.1 Secondary Structure . . . . .	14
3.3.2 Charged Interactions . . . . .	15
3.3.3 Hydrophobic Interactions . . . . .	16
3.3.4 B-factor Overlay . . . . .	16
3.3.5 User-defined Pairwise Aminoacid Interactions . . . . .	17
3.4 Plots . . . . .	17
3.4.1 Pairwise Heatmap . . . . .	18
3.4.2 Contacts Histogram . . . . .	19
3.4.3 Variance Contact Map . . . . .	20



## Companion website

The website<sup>1</sup> contains the source code and brief instructions for running CMPyMOL:

- This software is provided under The MIT License (MIT).

## Acknowledgements

- A special word of thanks goes to Professor Markus Deserno<sup>2</sup> for his useful and insightful discussions about this software in particular and science in general.

---

<sup>1</sup><https://goo.gl/VyMXPd>

<sup>2</sup><https://www.cmu.edu/physics/people/faculty/deserno.html>



# 1

## Installation Instructions

*“Only Mac OS X and Linux operating systems are fully supported at this time.”*

—

## 1.1 Mac OS X

### 1.1.1 Prerequisites and Initial Setup

#### PyMOL

PyMOL binaries can be installed using HomeBrew by following the below instructions.

---

```
# Open Terminal.app and enter the following commands

# Installing HomeBrew
ruby -e "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/master/install)"

# Installing PyMOL
brew tap homebrew/science
brew install python --with-brewed-tk --enable-threads --with-x11
brew install pymol
brew link pymol

# While installing PyMOL you might have to enter your superuser
# password to complete installation sucessfully.
```

---

#### Python Dependencies

Installing python dependencies wxpython, matplotlib, python imaging library and numpy.

---

```
# Open Terminal.app and enter the following commands
brew install wxmac
brew install wxpython
pip install matplotlib
pip install pillow
pip install numpy
```

---

#### Installing Stride (*optional*)

Skipping this step will automatically disable certain features in CMPyMOL.

---

```
# Download Stride
wget http://webclu.bio.wzw.tum.de/stride/stride.tar.gz

# Unzip the downloaded file
tar xvfz stride.tar.gz

# Compile Stride
cd stride
```

```
make

# Open the .profile file from your “home” directory with your
# favorite text editor
# Add the following line at the end of the .profile file by
# appropriately replacing the <directory stride> to the
# downloaded stride package path.
export PATH=$PATH:<directory stride>

# source the .profile file changes
source ~/.profile
```

---

### 1.1.2 Running CMPyMOL

Download git package from the CMPyMOL repository and run it.

```
# Download the git repository
git clone https://github.com/emptyewer/CMPyMOL
cd CMPyMOL

# Executing the below command will automatically launch PyMOL. Then
# follow on screen instructions.
python CMPyMOL1.5.py
```

---

## 1.2 Linux

### 1.2.1 Prerequisites and Initial Setup

PyMOL binaries can be installed using `aptitude` that is provided by default in Ubuntu by following the below instructions.

```
# Installing PyMOL
sudo apt-get install pymol

# While installing PyMOL you might have to enter your superuser
# password to complete installation sucessfully.
```

---

### Python Dependencies

Installing python dependencies `wxpython`, `matplotlib`, `python imaging library` and `numpy`.

```
sudo apt-get install python-wxgtk2.8
sudo pip install matplotlib
```

---

```
sudo pip install pillow
sudo pip install numpy
```

---

### Installing Stride (*optional*)

Skipping this step will automatically disable certain features in CMPyMOL.

---

```
# Download Stride
wget http://webclu.bio.wzw.tum.de/stride/stride.tar.gz

# Unzip the downloaded file
tar xvfz stride.tar.gz

# Compile Stride
cd stride
make

# Open the .bashrc file from your “home” directory with your
# favorite text editor
# Add the following line at the end of the .bashrc file by
# appropriately replacing the <directory stride> to the
# downloaded stride package path.
export PATH=$PATH:<directory stride>

# source the .bashrc file changes
source ~/.bashrc
```

---

#### 1.2.2 Running CMPyMOL

Download git package from the CMPyMOL repository and run it.

---

```
# Download the git repository
git clone https://github.com/emptyewer/CMPyMOL
cd CMPyMOL

# Executing the below command will automatically launch PyMOL. Then
# follow on screen instructions.
python CMPyMOL1.5.py
```

---

### 1.3 Windows

Currently not supported.

## 2

# Load PDB or Precalculated Contact Map

*“Single frame and multi-frame PDB files are currently supported by CMPyMOL”*

---

### 2.0.1 Command for Launching CMPyMOL

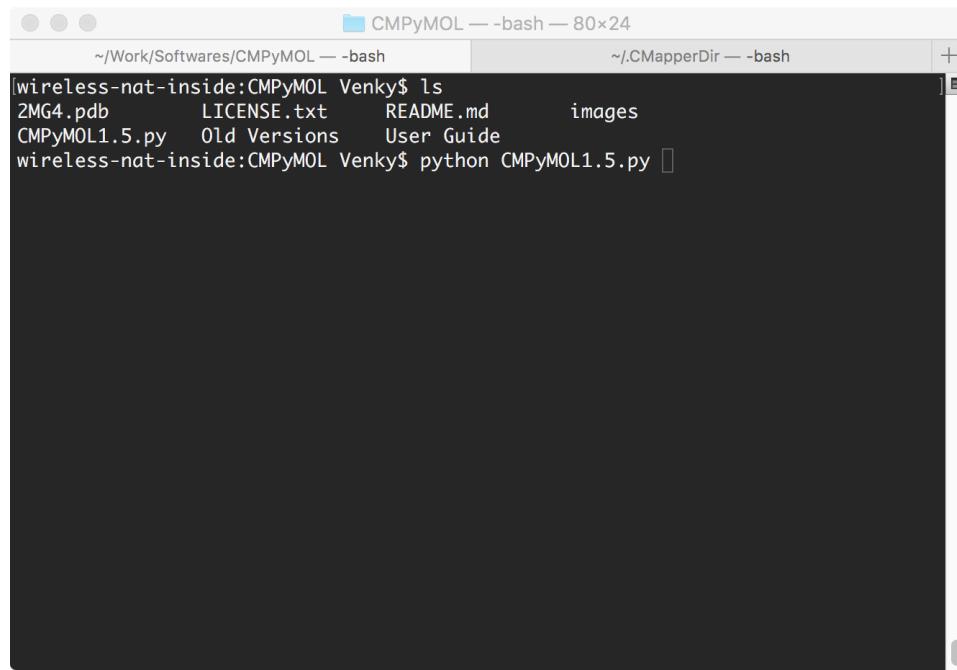
A screenshot of a terminal window titled "CMPyMOL — -bash — 80x24". The window shows a file listing command being run: "ls". The output of the command is displayed, showing files like "2MG4.pdb", "LICENSE.txt", "README.md", and "images". Below the listing, the command "python CMPyMOL1.5.py" is entered and followed by a right arrow indicating execution.

Figure 2.1: Command to invoke CMPyMOL from Terminal.

If the install instructions from the previous chapter was successful, invoking CMPyMOL will launch up PyMOL and start CMPyMOL.

### 2.0.2 PDB file format

---

```
MODEL X
.
.
.
ATOM ...
ATOM ...
ATOM ...
.
.
.
ENDMDL
```

---

### 2.0.3 Load PDB file

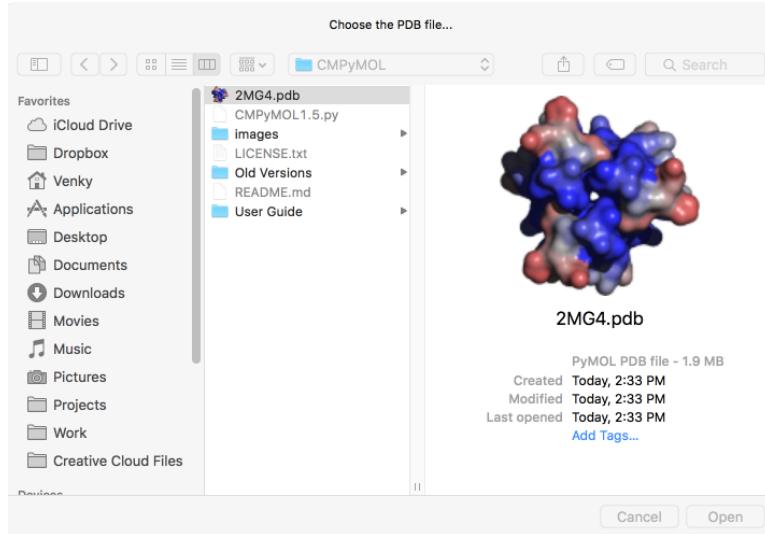


Figure 2.2: Pop-up dialog for file selection.

Either a single frame PDB file or a multi-frame PDB from MD simulation trajectory or NMR trajectory can be loaded at this step. The correct format of the PDB file is provided in Section 2.0.2 for reference. Other structure formats are currently not supported.

### 2.0.4 Calculate Contact Map

Figure 2.3: Calculate contact map or load from a precalculated file.

Figure 2.4: Cut-off distance and the basis atom to generate contact map.

A protein contact map represents the pairwise distance between all possible amino acid residue pairs of a three-dimensional protein structure. These dis-

tances are typically calculated between either  $\text{C}\alpha$  or  $\text{C}\beta$  atom. The cut-off distance eliminates all the pairwise distances that are larger than that value.

The user has the choice to calculate the contact map afresh or load it from a precalculated contact map file (Figure 2.3). If the choice was to calculate contact map, a prompt for a choice of  $\text{C}\alpha$  or  $\text{C}\beta$  atom as the basis atom and the cut-off distance (defaults to 8Å) for calculating the contact map will be presented (Figure 2.4).

### 2.0.5 CMPyMOL & PyMOL Window

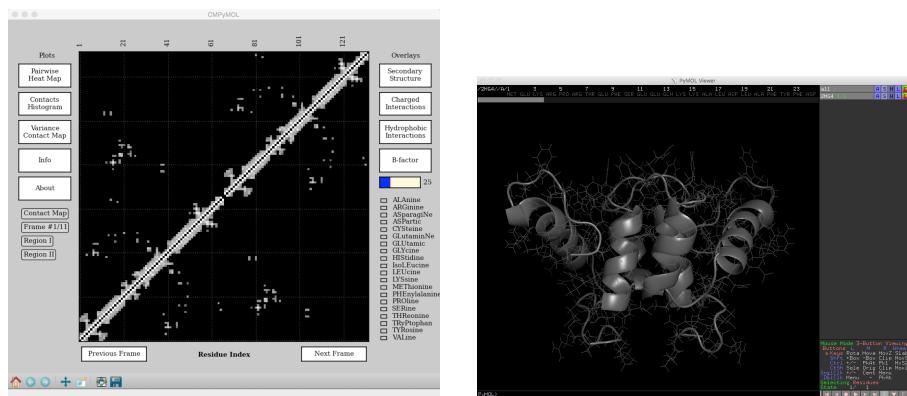


Figure 2.5: CMPyMOL window.

Figure 2.6: PyMOL window

Following the above steps will result in launching of PyMOL visualization window, where the protein is loaded, and CMPyMOL window that displays the contact map. By default the protein is represented as a cartoon. The next chapter will go into functional details of CMPyMOL extension for PyMOL.

# 3

## Functionality

*“The overlays in CMPyMOL enhances the interpretation of contact maps.”*

—

### 3.1 CMPyMOL Main Window

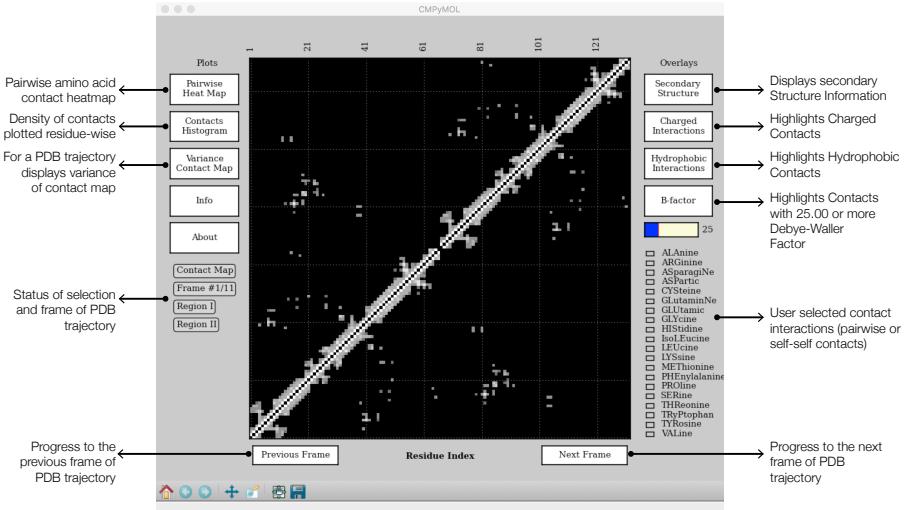


Figure 3.1: Main CMPyMOL Window and Its Functionality

The main window of CMPyMOL (Figure 3.1) provides controls for all the selection, overlay and plots to analyze contact maps. The overlays (the toggle buttons on the right of the contact map) superpose chemical and structural information on top of the contact map when activated. The plots (buttons on the left side of the contact map) pops open a new window that provides an overview of the nature of contacts.

## 3.2 Selections

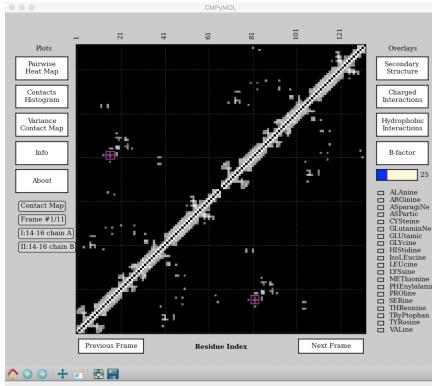


Figure 3.2: Selection in CMPyMOL window.

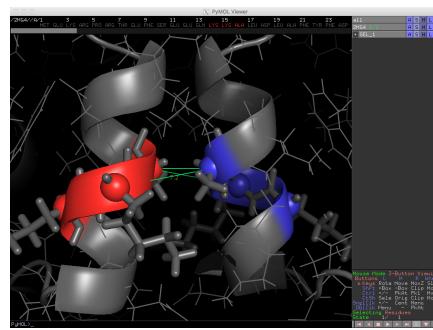


Figure 3.3: Focus on the selected region in PyMOL window.

Selecting a particular region by clicking-n-dragging on a portion of the CMPyMOL main window (shown as a purple selection box Figure 3.13) highlights the corresponding region in the PyMOL window. Since contact map is a pairwise interaction matrix, the two interacting regions are colored **red** and **blue**. The side chains of the interacting residues are shown in gray as “sticks”.

## 3.3 Overlays

Overlays in CMPyMOL provide a intuitive superposing of chemical and structural information on top of the contact map. This allows of correlating interacting residues with its type (chemical or structural).

### 3.3.1 Secondary Structure

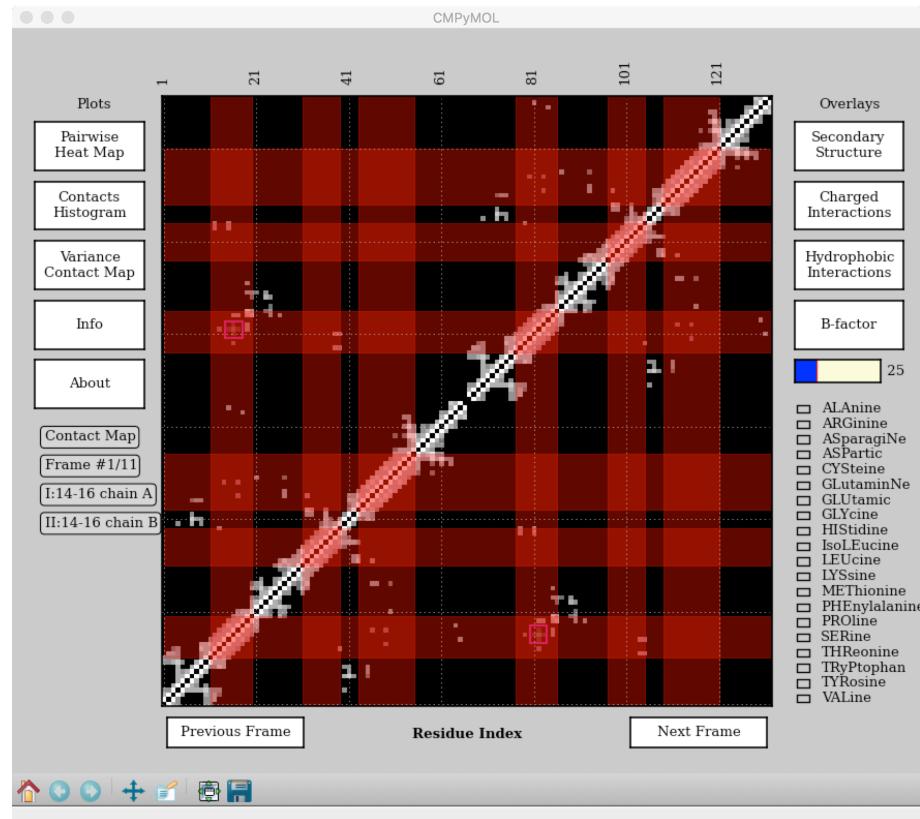


Figure 3.4: Secondary structure overlay on top of contact map.

The secondary structure overlay superposes  $\alpha$ -helical and  $\beta$ -sheet as red and green translucent rectangles, respectively. Notedly, the selection from the last section (Section 3.2) is in a  $\alpha$ -helix –  $\alpha$ -helix interaction, which can be confirmed by the selection highlighted in PyMOL window (Figure 3.13).

### 3.3.2 Charged Interactions

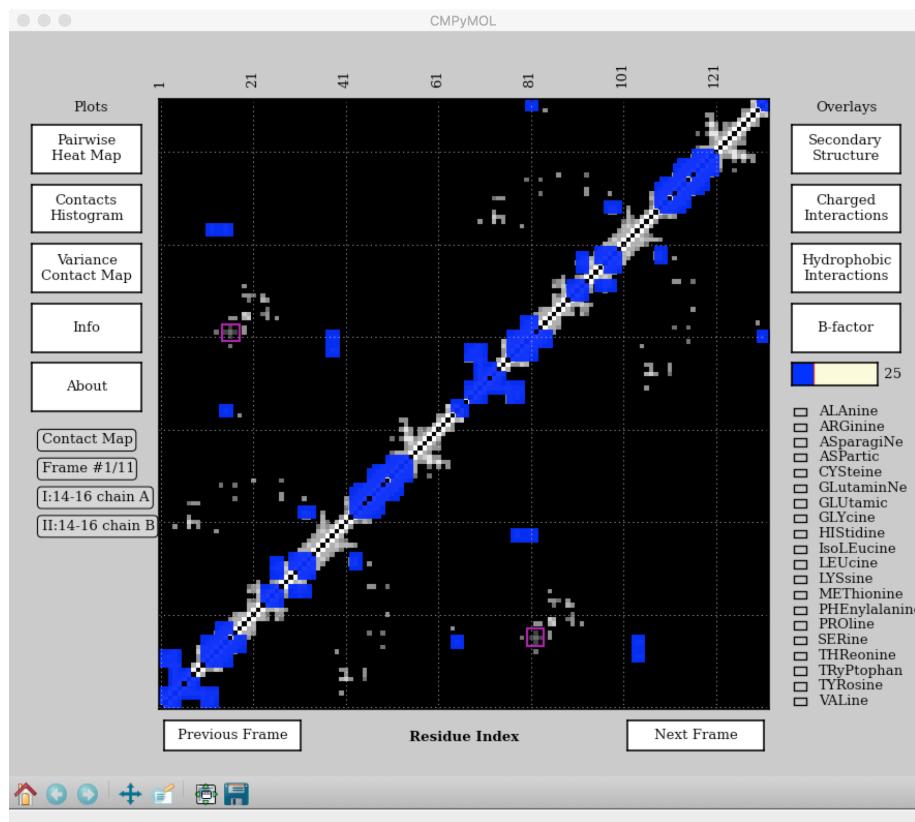


Figure 3.5: Charge-charge interactions highlighted on top of contact map.

Similar to secondary structure overlay, charged interaction overlay highlights contact points where two charged residues are located within the cut-off distance specified in Section 2.0.4. These charged interaction points are highlighted as blue pixels on top of the contact map.

### 3.3.3 Hydrophobic Interactions

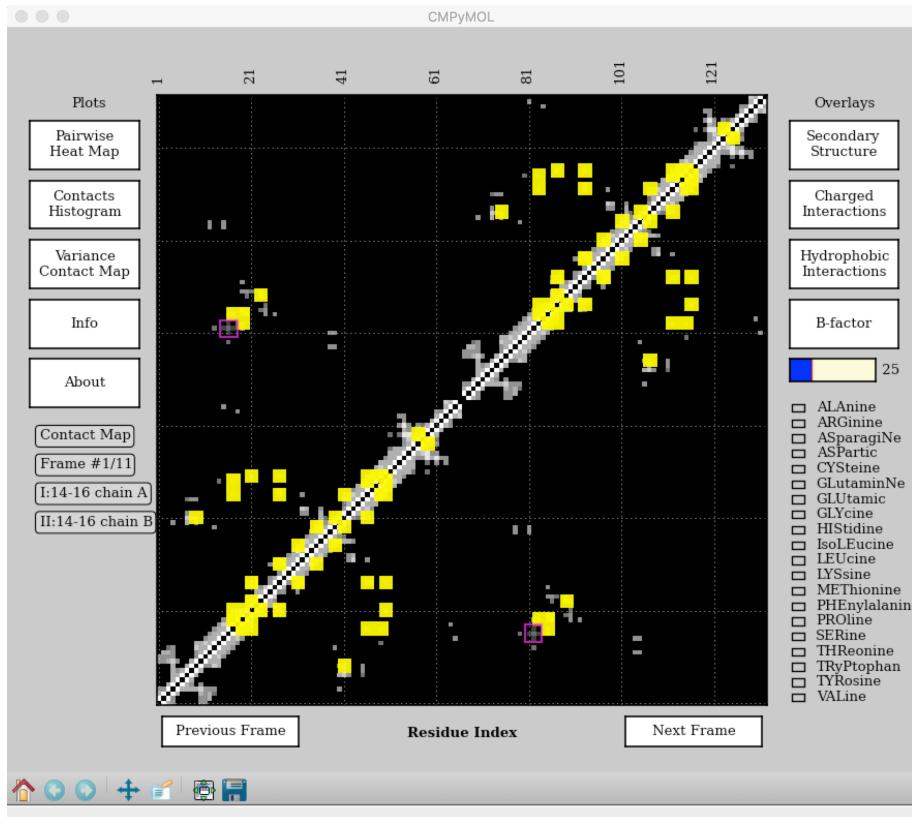


Figure 3.6: Hydrophobic interactions highlighted on top of contact map.

Hydrophobic interactions overlay highlights the points of contact where two hydrophobic residues are coming in contact. These interaction points are highlighted as **yellow** over the contact map.

### 3.3.4 B-factor Overlay

In a similar fashion B-factor (if available from the PDB) highlights the points with a larger value than set with the slider below the B-factor button (in the case shown Figure 3.6, the highlighted contact points have a b-factor larger than 25). These interaction points are highlighted as **cyan** over the contact map.

### 3.3.5 User-defined Pairwise Aminoacid Interactions

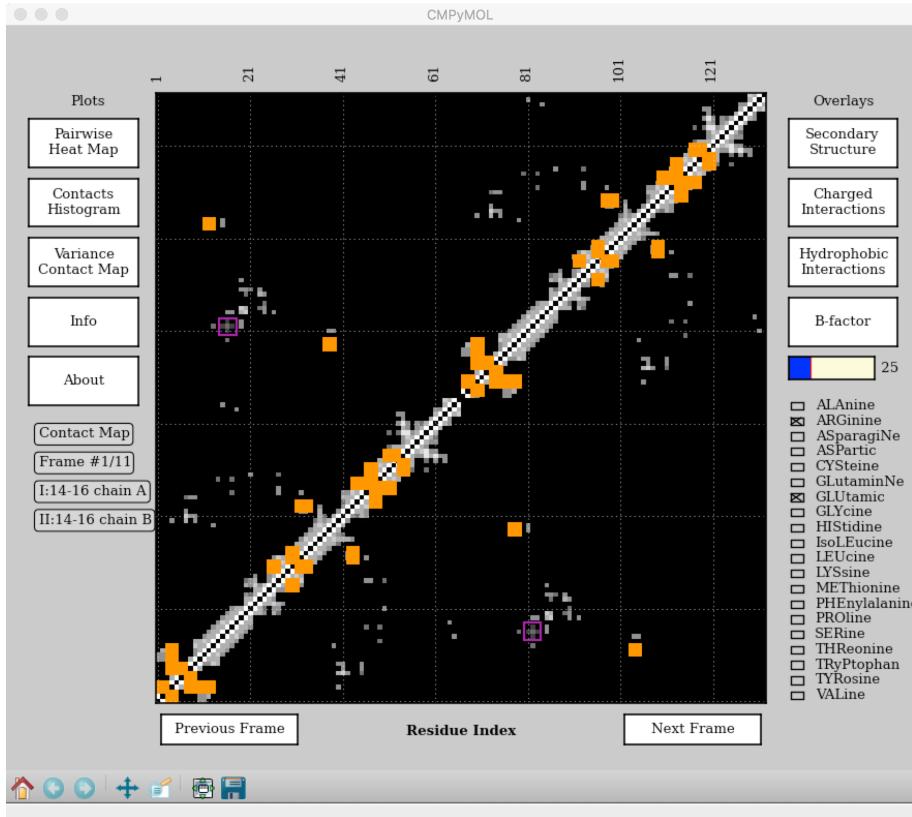


Figure 3.7: User selected interaction between Arginine and Glutamic Acid residues highlighted on top of contact map.

In addition to the above overlays, the user can choose to highlight interaction points of two specific aminoacids. In Figure 3.7, the interaction points of Arginine and Glutamic acid residues are shown in orange. Note that these highlighted residues are subset of charged interactions (Figure 3.5) as expected.

## 3.4 Plots

CMPyMOLPlots provide statistical analysis of the contact points.

### 3.4.1 Pairwise Heatmap

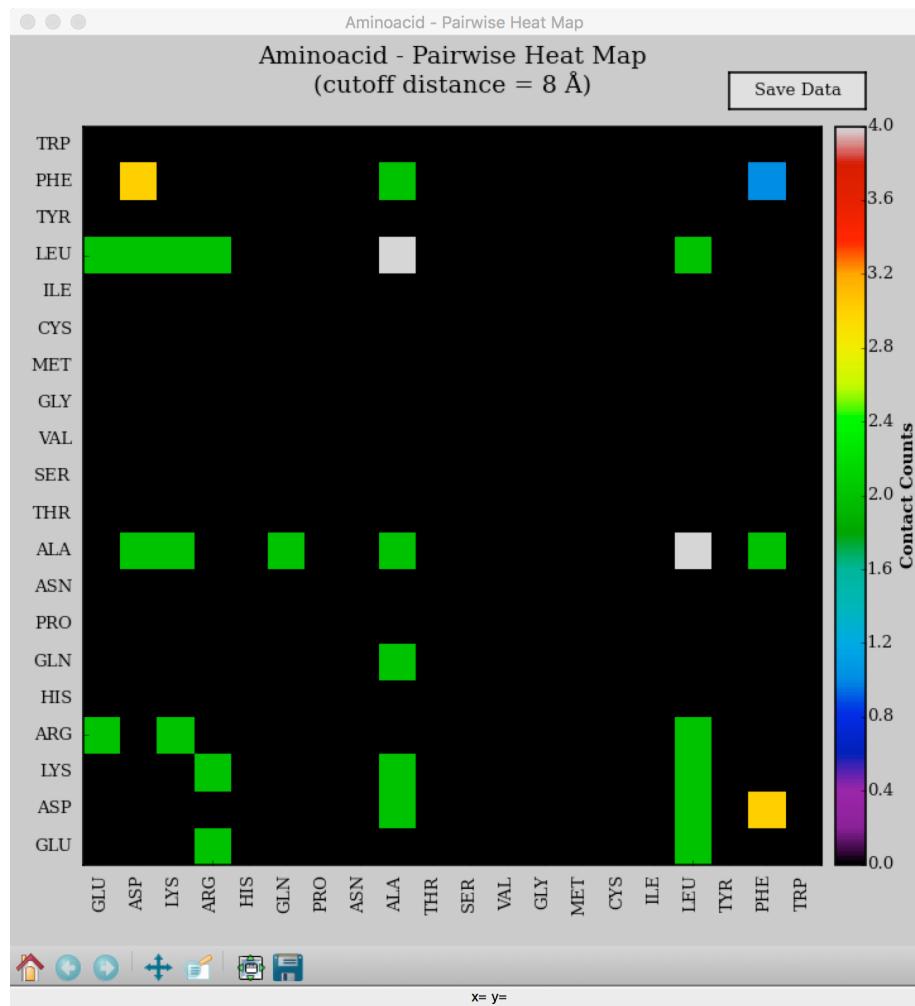


Figure 3.8: Heatmap of pairwise residue-residue interaction map. The residues are listed according to increasing hydrophobicity.

This heatmap counts the number of pairwise contacts of a given aminoacid to the rest of the other aminoacids. The order of aminoacids in this plot are arranged according to their hydrophobicity. The color scale shows the number of each pairwise contacts in the protein.

### 3.4.2 Contacts Histogram

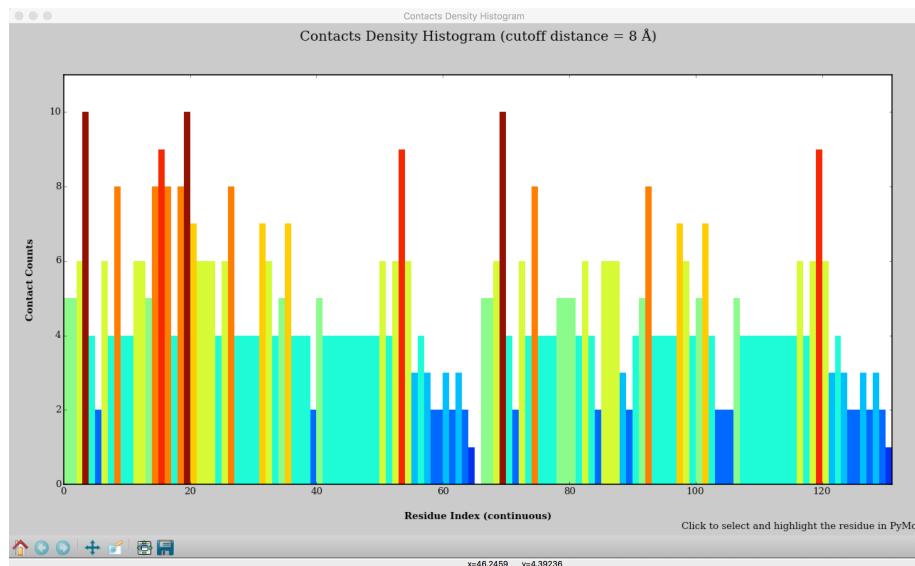


Figure 3.9: The residue-wise density of contacts in the PDB.

The contact histogram plot, graphs the density of contacts with respect to residue position. This plot is interactive and clicking on a particular bar selects the corresponding residues and the surrounding contacts in the PyMOL window.

### 3.4.3 Variance Contact Map

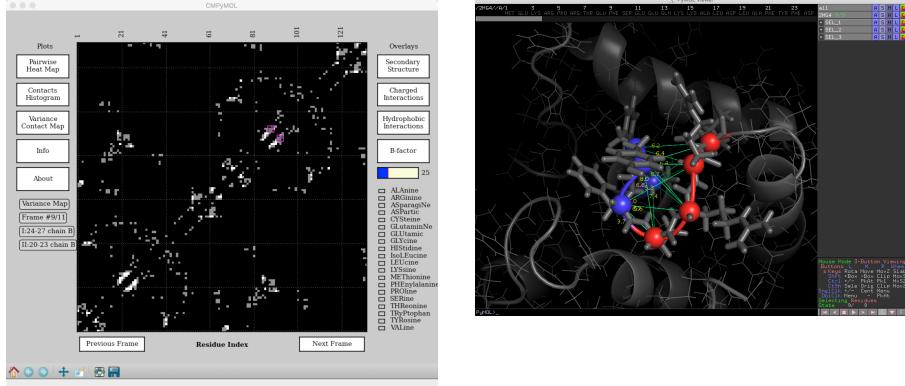


Figure 3.10: Variance Contact Map. Showing the regions of highest flexibility.

Figure 3.11: Selecting a high variance region in the intra-subunit quadrant reveals in the PyMOL window that the region is a flexible loop.

The PDB loaded as an example here is a NMR structure with multiple-frames (10 models). When a multi-frame PDB is loaded, the “Variance Contact Map” button is enabled and it allows for calculating the variance in contact distance for a residue pair. The contacts in a “Variance Contact Map” here are colored in grayscale with the color white having the most variance and gray shades with less variance.

This allows for readily identifying flexible regions of the protein. In figure 3.10 a selection highlights the intra-protein flexible region with high variance. In the PyMOL window it can be verified that indeed the region of high variance is on a loop. Clicking on the “Next Frame” or “Previous Frame” progresses the PDB to the next PDB model and correspondingly displays the “Variance Contact Map” and the change in the PDB structure.

In a similar manner, selecting a inter-protein interaction site of high-variance (Figure 3.12 also reveals that the regions are in a loop-loop interaction.

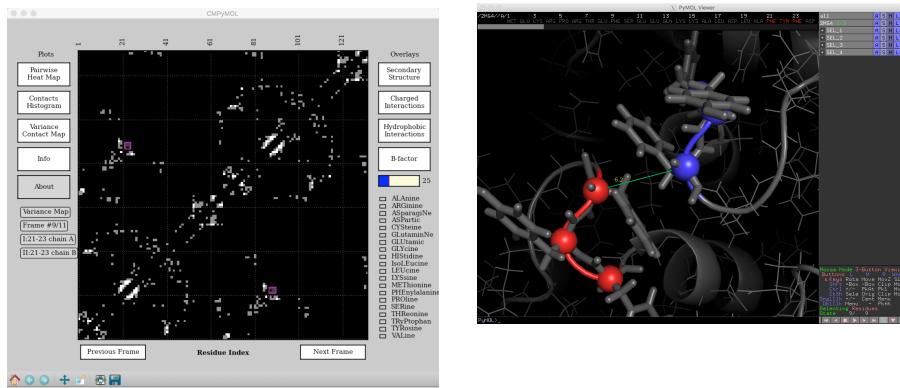


Figure 3.12: Variance Contact Map. Showing the regions of highest flexibility.

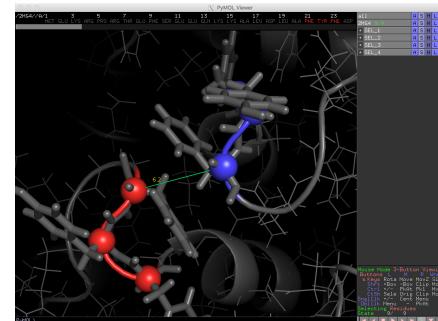


Figure 3.13: Selecting a high variance region in the inter-subunit quadrant reveals that it represents an interaction between two tyrosine residues on loops.