

Class10_bggn13

AUTHOR

Lisanne Stouthart (PID A69036187)

Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see <https://quarto.org>.

He said uploading the class lecture is sufficient. But we did not go over all the questions. Please let me know if you want me to do them. 

PDB Intro

```
# Go to PDB
# Enter 1IEP
# Click on structure
# Display Files – PDB Format
# Search for atom, this data is used for the viewer of protein structure
# Meaning we dont have information for the first 224 residus.
```

Little exercise

```
# Download a CSV file from the PDB site (accessible from "Analyze" > "PDB Statist
# Save your input data file into your Project directory
pdbstat <- "~/Downloads/data_export_summary.csv"
pdbstats <- read.csv(pdbstat, row.names=1)
head(pdbstats)
```

	X.ray	EM	NMR	Multiple.methods	Neutron	Other
Protein (only)	167,317	15,698	12,534	208	77	32
Protein/Oligosaccharide	9,645	2,639	34	8	2	0
Protein/NA	8,735	4,718	286	7	0	0
Nucleic acid (only)	2,869	138	1,507	14	3	1
Other	170	10	33	0	0	0
Oligosaccharide (only)	11	0	6	1	0	4
Total						
Protein (only)	195,866					
Protein/Oligosaccharide	12,328					
Protein/NA	13,746					

Nucleic acid (only)	4,532
Other	213
Oligosaccharide (only)	22

```
x <- pdbstats$Total
x
```

```
[1] "195,866" "12,328" "13,746" "4,532" "213" "22"
```

```
#as.numeric(x) #Warning: NAs introduced by coercion, the comma's are messing this
gsub(',', '', x)
```

```
[1] "195866" "12328" "13746" "4532" "213" "22"
```

```
convert_comma_numbers <- function(x) {
  #remove comma
  x <- gsub(',', '', x)
  #convert to numeric
  x <- as.numeric(x)

  return(x)
}
```

```
convert_comma_numbers(pdbstats$Total)
```

```
[1] 195866 12328 13746 4532 213 22
```

The 'apply()' function is very useful as it can take any function and "apply" it over either the ROWS or COLs of a data.frame

```
n.tot <- sum(convert_comma_numbers(pdbstats$Total))
n.tot
```

```
[1] 226707
```

```
195866 / 248838887 * 100
```

```
[1] 0.07871197
```

```
colSums(apply(pdbstats, 2, convert_comma_numbers))/n.tot
```

	X.ray	EM	NMR	Multiple.methods
0.8325592064	0.1023479646	0.0635181093	0.0010498132	
Neutron	Other	Total		
0.0003617003	0.0001632063	1.0000000000		

Q1 - What percentage of structures in the PDB are solved by X-Ray and Electron Microscopy.

```
n.xray <- sum(convert_comma_numbers(pdbstats$X.ray))
n.em <- sum(convert_comma_numbers(pdbstats$EM))

n.xray/n.tot * 100
```

[1] 83.25592

```
n.em/n.tot * 100
```

[1] 10.2348

Q2 - What proportion of structures in the PDB are protein?

```
protein <- (n.xray+n.em)/n.tot * 100
protein
```

[1] 93.49072

Q3 - Type HIV in the PDB website search box on the home page and determine how many HIV-1 protease structures are in the current PDB?

```
# Typing in "HIV-1 protease" gives 25,309 structures. This seems a bit much.
```

Visualizing the HIV-1 protease structure

```
# open Molstar (https://molstar.org/viewer/)
# 1HSG open -> apply
# Save your input data file into your Project directory
# Copy Paste it in the right folder

library(bio3d)
hsg_try <- read.pdb("~/Downloads/1hsg.pdb")
print(hsg_try)
```

Call: read.pdb(file = "~/Downloads/1hsg.pdb")

Total Models#: 1

```
Total Atoms#: 1686, XYZs#: 5058 Chains#: 2 (values: A B)
```

```
Protein Atoms#: 1514 (residues/Calpha atoms#: 198)
```

```
Nucleic acid Atoms#: 0 (residues/phosphate atoms#: 0)
```

```
Non-protein/nucleic Atoms#: 172 (residues: 128)
```

```
Non-protein/nucleic resid values: [ HOH (127), MK1 (1) ]
```

Protein sequence:

```
PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGGFIKVRQYD  
QILIEICGHKAIGTVLVGPTPVNIIGRNLLTQIGCTLNFPQITLWQRPLVTIKIGGQLKE  
ALLDTGADDTVLEEMSLPGRWKPKMIGGIGGGFIKVRQYDQILIEICGHKAIGTVLVGPTP  
VNIIGRNLLTQIGCTLNF
```

```
+ attr: atom, xyz, seqres, helix, sheet,  
       calpha, remark, call
```

```
summary(hsg_try)
```

```
Call: read.pdb(file = "~/Downloads/1hsg.pdb")
```

Total Models#: 1

```
Total Atoms#: 1686, XYZs#: 5058 Chains#: 2 (values: A B)
```

```
Protein Atoms#: 1514 (residues/Calpha atoms#: 198)
```

```
Nucleic acid Atoms#: 0 (residues/phosphate atoms#: 0)
```

```
Non-protein/nucleic Atoms#: 172 (residues: 128)
```

```
Non-protein/nucleic resid values: [ HOH (127), MK1 (1) ]
```

```
+ attr: atom, xyz, seqres, helix, sheet,  
       calpha, remark, call
```

```
# Extract water molecules
water_indices <- which(hsg_try$atom$resid == "HOH" | hsg_try$atom$resid == "WAT")
water_molecules <- hsg_try$atom[water_indices, ]

# View water molecules
print(water_molecules)
```

	type	eleno	elety	alt	resid	chain	resno	insert	x	y	z	o
1560	HETATM	1562	0	<NA>	HOH	A	305	<NA>	20.857	43.192	21.450	1
1561	HETATM	1563	0	<NA>	HOH	A	307	<NA>	14.076	19.789	19.440	1
1562	HETATM	1564	0	<NA>	HOH	A	309	<NA>	28.075	21.177	7.222	1
1563	HETATM	1565	0	<NA>	HOH	A	314	<NA>	16.759	40.274	1.287	1
1564	HETATM	1566	0	<NA>	HOH	A	315	<NA>	13.997	22.233	21.468	1
1565	HETATM	1567	0	<NA>	HOH	A	324	<NA>	11.282	30.738	1.625	1
1566	HETATM	1568	0	<NA>	HOH	A	325	<NA>	16.774	42.740	2.296	1

1567	HETATM	1569	0 <NA>	HOH	A	327	<NA>	14.623	29.552	28.235	1
1568	HETATM	1570	0 <NA>	HOH	A	328	<NA>	1.651	36.463	19.459	1
1569	HETATM	1571	0 <NA>	HOH	A	329	<NA>	14.435	44.966	11.913	1
1570	HETATM	1572	0 <NA>	HOH	A	330	<NA>	19.877	40.160	21.917	1
1571	HETATM	1573	0 <NA>	HOH	A	331	<NA>	17.126	45.934	10.801	1
1572	HETATM	1574	0 <NA>	HOH	A	332	<NA>	8.840	28.026	4.860	1
1573	HETATM	1575	0 <NA>	HOH	A	335	<NA>	10.341	18.480	14.477	1
1574	HETATM	1576	0 <NA>	HOH	A	341	<NA>	19.233	16.711	9.027	1
1575	HETATM	1577	0 <NA>	HOH	A	342	<NA>	23.799	21.928	12.391	1
1576	HETATM	1578	0 <NA>	HOH	A	344	<NA>	9.953	37.934	4.548	1
1577	HETATM	1579	0 <NA>	HOH	A	345	<NA>	8.478	35.995	5.789	1
1578	HETATM	1580	0 <NA>	HOH	A	357	<NA>	3.960	19.389	17.384	1
1579	HETATM	1581	0 <NA>	HOH	A	373	<NA>	27.561	43.155	19.015	1
1580	HETATM	1582	0 <NA>	HOH	A	384	<NA>	1.245	19.292	18.124	1
1581	HETATM	1583	0 <NA>	HOH	A	386	<NA>	31.402	27.051	3.335	1
1582	HETATM	1584	0 <NA>	HOH	A	389	<NA>	32.446	31.200	4.417	1
1583	HETATM	1585	0 <NA>	HOH	A	391	<NA>	25.480	38.468	17.938	1
1584	HETATM	1586	0 <NA>	HOH	A	394	<NA>	23.940	41.721	0.346	1
1585	HETATM	1587	0 <NA>	HOH	A	401	<NA>	5.912	15.727	3.369	1
1586	HETATM	1588	0 <NA>	HOH	A	406	<NA>	9.272	33.891	12.681	1
1587	HETATM	1589	0 <NA>	HOH	A	408	<NA>	21.185	25.233	16.048	1
1588	HETATM	1590	0 <NA>	HOH	A	416	<NA>	18.474	26.012	21.664	1
1589	HETATM	1591	0 <NA>	HOH	A	420	<NA>	9.469	16.910	17.371	1
1590	HETATM	1592	0 <NA>	HOH	A	422	<NA>	13.074	17.786	16.615	1
1591	HETATM	1593	0 <NA>	HOH	A	439	<NA>	28.821	29.338	7.342	1
1592	HETATM	1594	0 <NA>	HOH	A	457	<NA>	23.284	23.107	15.132	1
1593	HETATM	1595	0 <NA>	HOH	A	468	<NA>	3.114	26.260	6.773	1
1594	HETATM	1596	0 <NA>	HOH	A	501	<NA>	6.382	26.424	5.893	1
1595	HETATM	1597	0 <NA>	HOH	A	503	<NA>	35.293	43.006	5.212	1
1596	HETATM	1598	0 <NA>	HOH	A	510	<NA>	21.891	49.715	7.192	1
1597	HETATM	1599	0 <NA>	HOH	A	524	<NA>	34.085	32.735	2.849	1
1598	HETATM	1600	0 <NA>	HOH	A	529	<NA>	31.491	41.524	6.678	1
1599	HETATM	1601	0 <NA>	HOH	A	553	<NA>	5.943	34.223	6.748	1
1600	HETATM	1602	0 <NA>	HOH	A	561	<NA>	0.934	40.259	19.405	1
1601	HETATM	1603	0 <NA>	HOH	A	567	<NA>	29.539	25.486	13.281	1
1602	HETATM	1604	0 <NA>	HOH	A	572	<NA>	24.552	17.352	10.295	1
1603	HETATM	1605	0 <NA>	HOH	A	575	<NA>	23.112	15.510	8.776	1
1604	HETATM	1606	0 <NA>	HOH	B	301	<NA>	20.445	8.036	-12.631	1
1605	HETATM	1607	0 <NA>	HOH	B	303	<NA>	20.044	14.822	4.638	1
1606	HETATM	1608	0 <NA>	HOH	B	304	<NA>	21.538	6.875	-10.099	1
1607	HETATM	1609	0 <NA>	HOH	B	306	<NA>	22.449	23.958	5.252	1
1608	HETATM	1610	0 <NA>	HOH	B	308	<NA>	11.720	21.289	7.190	1
1609	HETATM	1611	0 <NA>	HOH	B	312	<NA>	14.097	5.111	-11.638	1
1610	HETATM	1612	0 <NA>	HOH	B	313	<NA>	20.998	21.834	6.561	1
1611	HETATM	1613	0 <NA>	HOH	B	316	<NA>	22.659	14.583	-2.196	1
1612	HETATM	1614	0 <NA>	HOH	B	317	<NA>	28.724	15.629	-11.660	1
1613	HETATM	1615	0 <NA>	HOH	B	318	<NA>	16.539	45.207	0.079	1
1614	HETATM	1616	0 <NA>	HOH	B	319	<NA>	23.678	14.931	2.680	1
1615	HETATM	1617	0 <NA>	HOH	B	321	<NA>	20.718	15.976	-3.657	1
1616	HETATM	1618	0 <NA>	HOH	B	323	<NA>	31.249	26.796	-9.595	1
1617	HETATM	1619	0 <NA>	HOH	B	326	<NA>	28.813	28.445	-2.106	1

1618	HETATM	1620	0 <NA>	HOH	B	333	<NA>	12.251	39.551	-2.672	1
1619	HETATM	1621	0 <NA>	HOH	B	334	<NA>	25.465	12.592	-8.670	1
1620	HETATM	1622	0 <NA>	HOH	B	338	<NA>	12.998	36.205	-3.972	1
1621	HETATM	1623	0 <NA>	HOH	B	339	<NA>	17.541	17.060	-17.194	1
1622	HETATM	1624	0 <NA>	HOH	B	340	<NA>	5.321	14.325	-4.866	1
1623	HETATM	1625	0 <NA>	HOH	B	346	<NA>	9.314	17.330	-9.801	1
1624	HETATM	1626	0 <NA>	HOH	B	347	<NA>	7.435	26.652	-14.854	1
1625	HETATM	1627	0 <NA>	HOH	B	348	<NA>	4.405	16.704	-3.635	1
1626	HETATM	1628	0 <NA>	HOH	B	349	<NA>	19.414	7.026	4.428	1
1627	HETATM	1629	0 <NA>	HOH	B	350	<NA>	6.718	34.538	-1.322	1
1628	HETATM	1630	0 <NA>	HOH	B	354	<NA>	15.041	31.743	-13.235	1
1629	HETATM	1631	0 <NA>	HOH	B	355	<NA>	27.404	32.078	-10.860	1
1630	HETATM	1632	0 <NA>	HOH	B	356	<NA>	27.673	18.789	-6.155	1
1631	HETATM	1633	0 <NA>	HOH	B	358	<NA>	21.289	-1.161	-5.102	1
1632	HETATM	1634	0 <NA>	HOH	B	359	<NA>	6.973	36.523	1.489	1
1633	HETATM	1635	0 <NA>	HOH	B	360	<NA>	27.602	21.234	-0.635	1
1634	HETATM	1636	0 <NA>	HOH	B	362	<NA>	3.902	9.376	-0.027	1
1635	HETATM	1637	0 <NA>	HOH	B	364	<NA>	28.498	36.632	-7.529	1
1636	HETATM	1638	0 <NA>	HOH	B	366	<NA>	18.572	40.567	-10.042	1
1637	HETATM	1639	0 <NA>	HOH	B	367	<NA>	25.658	18.970	0.428	1
1638	HETATM	1640	0 <NA>	HOH	B	369	<NA>	20.843	1.263	-7.014	1
1639	HETATM	1641	0 <NA>	HOH	B	370	<NA>	13.975	15.741	12.070	1
1640	HETATM	1642	0 <NA>	HOH	B	374	<NA>	7.661	23.876	-6.324	1
1641	HETATM	1643	0 <NA>	HOH	B	375	<NA>	10.125	5.706	-1.458	1
1642	HETATM	1644	0 <NA>	HOH	B	376	<NA>	18.450	20.497	-18.728	1
1643	HETATM	1645	0 <NA>	HOH	B	377	<NA>	29.267	20.487	-3.497	1
1644	HETATM	1646	0 <NA>	HOH	B	379	<NA>	6.685	26.541	-5.608	1
1645	HETATM	1647	0 <NA>	HOH	B	381	<NA>	25.810	26.789	-19.106	1
1646	HETATM	1648	0 <NA>	HOH	B	383	<NA>	21.144	-4.428	-11.331	1
1647	HETATM	1649	0 <NA>	HOH	B	387	<NA>	16.904	27.594	-15.938	1
1648	HETATM	1650	0 <NA>	HOH	B	388	<NA>	23.926	45.612	-4.998	1
1649	HETATM	1651	0 <NA>	HOH	B	390	<NA>	25.300	17.493	3.076	1
1650	HETATM	1652	0 <NA>	HOH	B	392	<NA>	6.618	28.079	-3.427	1
1651	HETATM	1653	0 <NA>	HOH	B	393	<NA>	19.795	13.651	-16.606	1
1652	HETATM	1654	0 <NA>	HOH	B	395	<NA>	7.202	9.982	-4.103	1
1653	HETATM	1655	0 <NA>	HOH	B	400	<NA>	8.474	34.203	-4.893	1
1654	HETATM	1656	0 <NA>	HOH	B	405	<NA>	16.659	15.866	11.446	1
1655	HETATM	1657	0 <NA>	HOH	B	410	<NA>	26.400	10.057	-3.287	1
1656	HETATM	1658	0 <NA>	HOH	B	414	<NA>	9.503	3.489	-4.419	1
1657	HETATM	1659	0 <NA>	HOH	B	419	<NA>	15.438	12.973	-18.484	1
1658	HETATM	1660	0 <NA>	HOH	B	425	<NA>	11.428	19.956	-24.551	1
1659	HETATM	1661	0 <NA>	HOH	B	430	<NA>	18.725	43.171	-5.575	1
1660	HETATM	1662	0 <NA>	HOH	B	436	<NA>	32.141	29.620	-8.580	1
1661	HETATM	1663	0 <NA>	HOH	B	443	<NA>	8.811	13.667	-20.256	1
1662	HETATM	1664	0 <NA>	HOH	B	444	<NA>	4.071	26.169	-0.230	1
1663	HETATM	1665	0 <NA>	HOH	B	461	<NA>	11.425	44.636	-3.033	1
1664	HETATM	1666	0 <NA>	HOH	B	469	<NA>	6.902	23.686	-10.066	1
1665	HETATM	1667	0 <NA>	HOH	B	471	<NA>	5.749	25.785	-19.792	1
1666	HETATM	1668	0 <NA>	HOH	B	500	<NA>	25.592	16.404	-5.805	1
1667	HETATM	1669	0 <NA>	HOH	B	502	<NA>	4.040	15.516	-7.200	1
1668	HETATM	1670	0 <NA>	HOH	B	505	<NA>	28.640	34.232	-5.637	1

1669	HETATM	1671	0	<NA>	HOH	B	506	<NA>	8.979	11.173	11.112	1
1670	HETATM	1672	0	<NA>	HOH	B	509	<NA>	19.882	3.986	-18.136	1
1671	HETATM	1673	0	<NA>	HOH	B	514	<NA>	27.409	15.355	2.200	1
1672	HETATM	1674	0	<NA>	HOH	B	515	<NA>	17.222	39.766	-23.774	1
1673	HETATM	1675	0	<NA>	HOH	B	517	<NA>	28.742	24.158	-16.641	1
1674	HETATM	1676	0	<NA>	HOH	B	525	<NA>	22.694	-2.192	-12.589	1
1675	HETATM	1677	0	<NA>	HOH	B	526	<NA>	17.901	43.157	-14.082	1
1676	HETATM	1678	0	<NA>	HOH	B	531	<NA>	18.192	8.914	11.344	1
1677	HETATM	1679	0	<NA>	HOH	B	532	<NA>	19.507	45.215	1.709	1
1678	HETATM	1680	0	<NA>	HOH	B	548	<NA>	1.442	14.700	-6.128	1
1679	HETATM	1681	0	<NA>	HOH	B	549	<NA>	19.908	8.718	-19.215	1
1680	HETATM	1682	0	<NA>	HOH	B	556	<NA>	21.499	44.884	-1.280	1
1681	HETATM	1683	0	<NA>	HOH	B	564	<NA>	10.031	8.593	-22.052	1
1682	HETATM	1684	0	<NA>	HOH	B	568	<NA>	2.817	28.133	2.191	1
1683	HETATM	1685	0	<NA>	HOH	B	591	<NA>	15.835	40.105	-5.971	1
1684	HETATM	1686	0	<NA>	HOH	B	595	<NA>	4.515	36.451	-4.499	1
1685	HETATM	1687	0	<NA>	HOH	B	613	<NA>	24.127	-10.994	-0.982	1
1686	HETATM	1688	0	<NA>	HOH	B	617	<NA>	30.112	17.912	-4.791	1

b segid elesy charge

1560	63.07	<NA>	0	<NA>
1561	63.34	<NA>	0	<NA>
1562	66.96	<NA>	0	<NA>
1563	36.09	<NA>	0	<NA>
1564	64.67	<NA>	0	<NA>
1565	21.55	<NA>	0	<NA>
1566	26.65	<NA>	0	<NA>
1567	60.45	<NA>	0	<NA>
1568	25.82	<NA>	0	<NA>
1569	32.52	<NA>	0	<NA>
1570	41.02	<NA>	0	<NA>
1571	41.93	<NA>	0	<NA>
1572	27.94	<NA>	0	<NA>
1573	51.87	<NA>	0	<NA>
1574	66.74	<NA>	0	<NA>
1575	65.58	<NA>	0	<NA>
1576	67.74	<NA>	0	<NA>
1577	43.98	<NA>	0	<NA>
1578	37.23	<NA>	0	<NA>
1579	69.15	<NA>	0	<NA>
1580	70.78	<NA>	0	<NA>
1581	21.93	<NA>	0	<NA>
1582	46.57	<NA>	0	<NA>
1583	63.81	<NA>	0	<NA>
1584	47.08	<NA>	0	<NA>
1585	63.52	<NA>	0	<NA>
1586	31.73	<NA>	0	<NA>
1587	49.24	<NA>	0	<NA>
1588	65.44	<NA>	0	<NA>
1589	75.86	<NA>	0	<NA>
1590	67.42	<NA>	0	<NA>
1591	57.13	<NA>	0	<NA>

1592	60.42	<NA>	0	<NA>
1593	75.52	<NA>	0	<NA>
1594	38.21	<NA>	0	<NA>
1595	50.02	<NA>	0	<NA>
1596	53.78	<NA>	0	<NA>
1597	61.00	<NA>	0	<NA>
1598	73.78	<NA>	0	<NA>
1599	61.39	<NA>	0	<NA>
1600	49.60	<NA>	0	<NA>
1601	71.88	<NA>	0	<NA>
1602	66.74	<NA>	0	<NA>
1603	70.97	<NA>	0	<NA>
1604	63.94	<NA>	0	<NA>
1605	73.81	<NA>	0	<NA>
1606	42.37	<NA>	0	<NA>
1607	51.24	<NA>	0	<NA>
1608	18.18	<NA>	0	<NA>
1609	53.13	<NA>	0	<NA>
1610	47.68	<NA>	0	<NA>
1611	65.44	<NA>	0	<NA>
1612	38.53	<NA>	0	<NA>
1613	32.25	<NA>	0	<NA>
1614	61.86	<NA>	0	<NA>
1615	22.69	<NA>	0	<NA>
1616	59.93	<NA>	0	<NA>
1617	33.99	<NA>	0	<NA>
1618	79.22	<NA>	0	<NA>
1619	31.58	<NA>	0	<NA>
1620	47.41	<NA>	0	<NA>
1621	46.59	<NA>	0	<NA>
1622	48.25	<NA>	0	<NA>
1623	48.73	<NA>	0	<NA>
1624	54.68	<NA>	0	<NA>
1625	37.86	<NA>	0	<NA>
1626	68.44	<NA>	0	<NA>
1627	42.81	<NA>	0	<NA>
1628	60.62	<NA>	0	<NA>
1629	61.36	<NA>	0	<NA>
1630	35.03	<NA>	0	<NA>
1631	62.75	<NA>	0	<NA>
1632	71.64	<NA>	0	<NA>
1633	57.53	<NA>	0	<NA>
1634	50.97	<NA>	0	<NA>
1635	73.30	<NA>	0	<NA>
1636	62.30	<NA>	0	<NA>
1637	65.69	<NA>	0	<NA>
1638	61.76	<NA>	0	<NA>
1639	67.21	<NA>	0	<NA>
1640	61.89	<NA>	0	<NA>
1641	74.72	<NA>	0	<NA>
1642	48.75	<NA>	0	<NA>

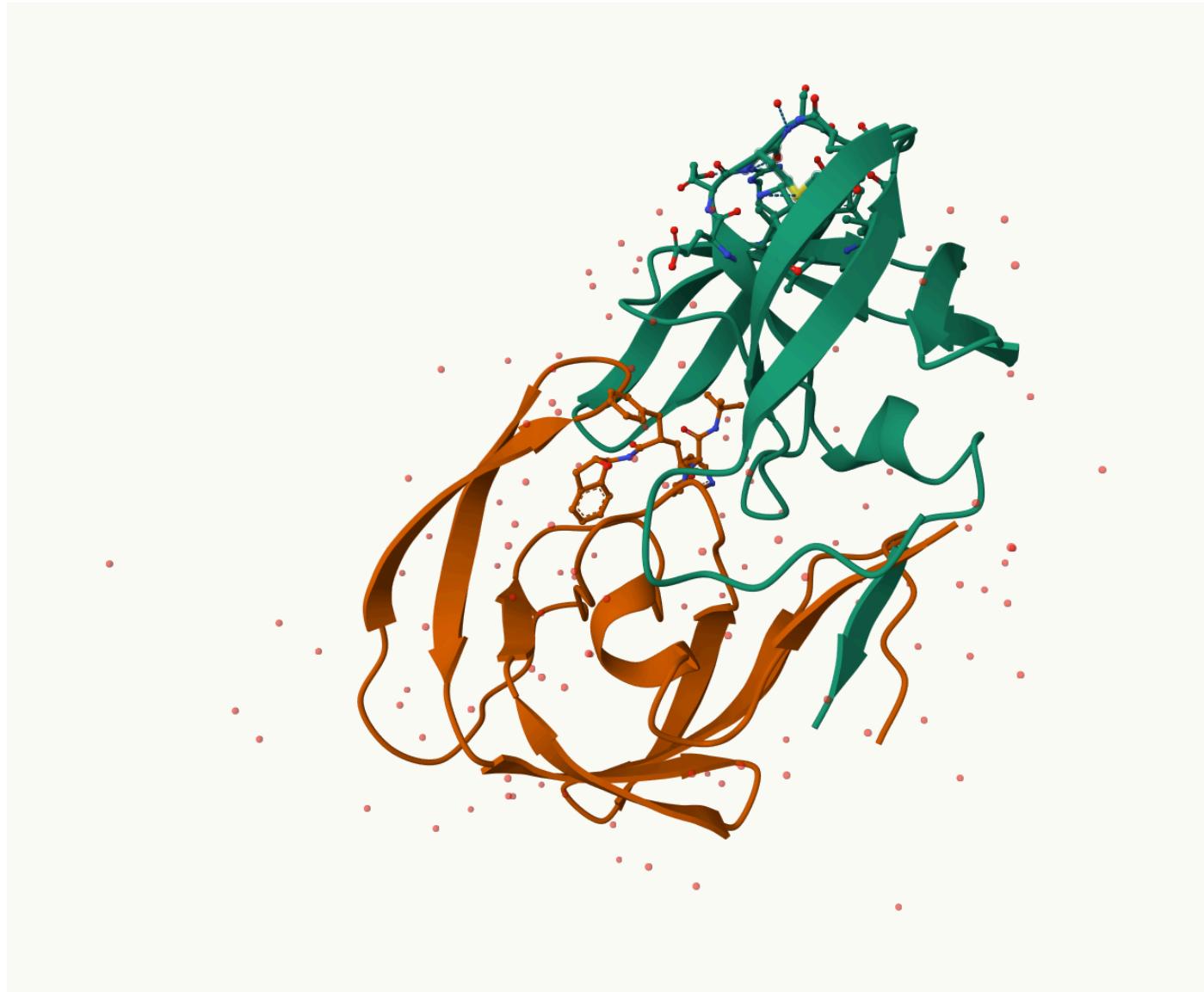
1643	60.17	<NA>	0	<NA>
1644	43.92	<NA>	0	<NA>
1645	70.16	<NA>	0	<NA>
1646	22.10	<NA>	0	<NA>
1647	27.84	<NA>	0	<NA>
1648	65.78	<NA>	0	<NA>
1649	67.04	<NA>	0	<NA>
1650	53.99	<NA>	0	<NA>
1651	54.21	<NA>	0	<NA>
1652	62.03	<NA>	0	<NA>
1653	63.64	<NA>	0	<NA>
1654	42.47	<NA>	0	<NA>
1655	65.50	<NA>	0	<NA>
1656	65.50	<NA>	0	<NA>
1657	73.55	<NA>	0	<NA>
1658	63.48	<NA>	0	<NA>
1659	52.97	<NA>	0	<NA>
1660	72.75	<NA>	0	<NA>
1661	75.75	<NA>	0	<NA>
1662	38.25	<NA>	0	<NA>
1663	68.43	<NA>	0	<NA>
1664	54.20	<NA>	0	<NA>
1665	63.96	<NA>	0	<NA>
1666	23.98	<NA>	0	<NA>
1667	52.93	<NA>	0	<NA>
1668	58.06	<NA>	0	<NA>
1669	64.79	<NA>	0	<NA>
1670	55.54	<NA>	0	<NA>
1671	61.69	<NA>	0	<NA>
1672	69.12	<NA>	0	<NA>
1673	78.93	<NA>	0	<NA>
1674	71.37	<NA>	0	<NA>
1675	78.14	<NA>	0	<NA>
1676	54.05	<NA>	0	<NA>
1677	72.78	<NA>	0	<NA>
1678	58.40	<NA>	0	<NA>
1679	58.78	<NA>	0	<NA>
1680	68.40	<NA>	0	<NA>
1681	64.90	<NA>	0	<NA>
1682	67.95	<NA>	0	<NA>
1683	53.68	<NA>	0	<NA>
1684	49.41	<NA>	0	<NA>
1685	64.49	<NA>	0	<NA>
1686	54.09	<NA>	0	<NA>

```
# Print the number of water molecules
cat("Total number of water molecules:", nrow(water_molecules), "\n")
```

Total number of water molecules: 127

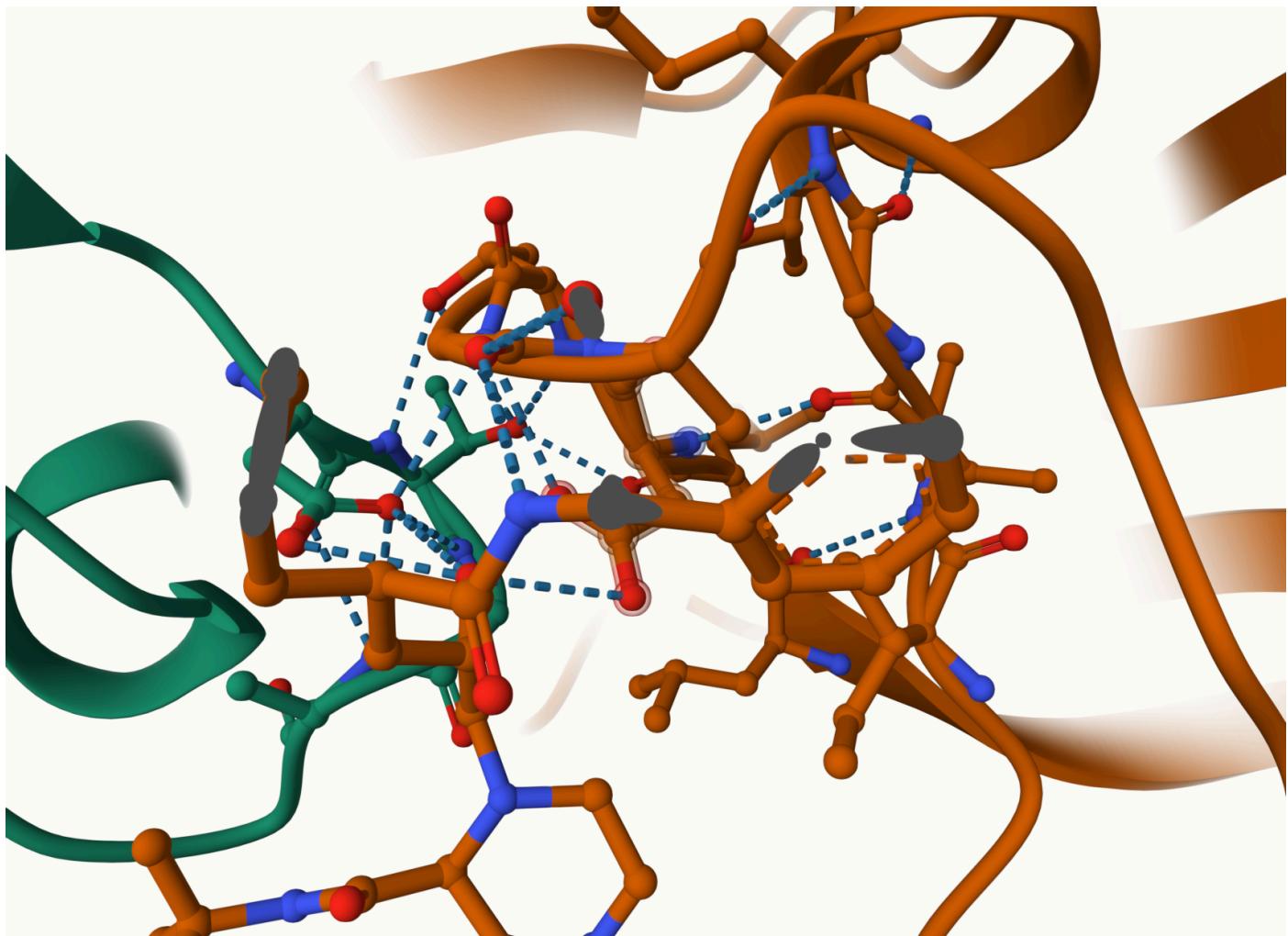
#127 water molecules

Using Mol*



My first image from Mol-star

Another Cool image



Another Cool image



The dot above is the water molecule.

Another Cool image

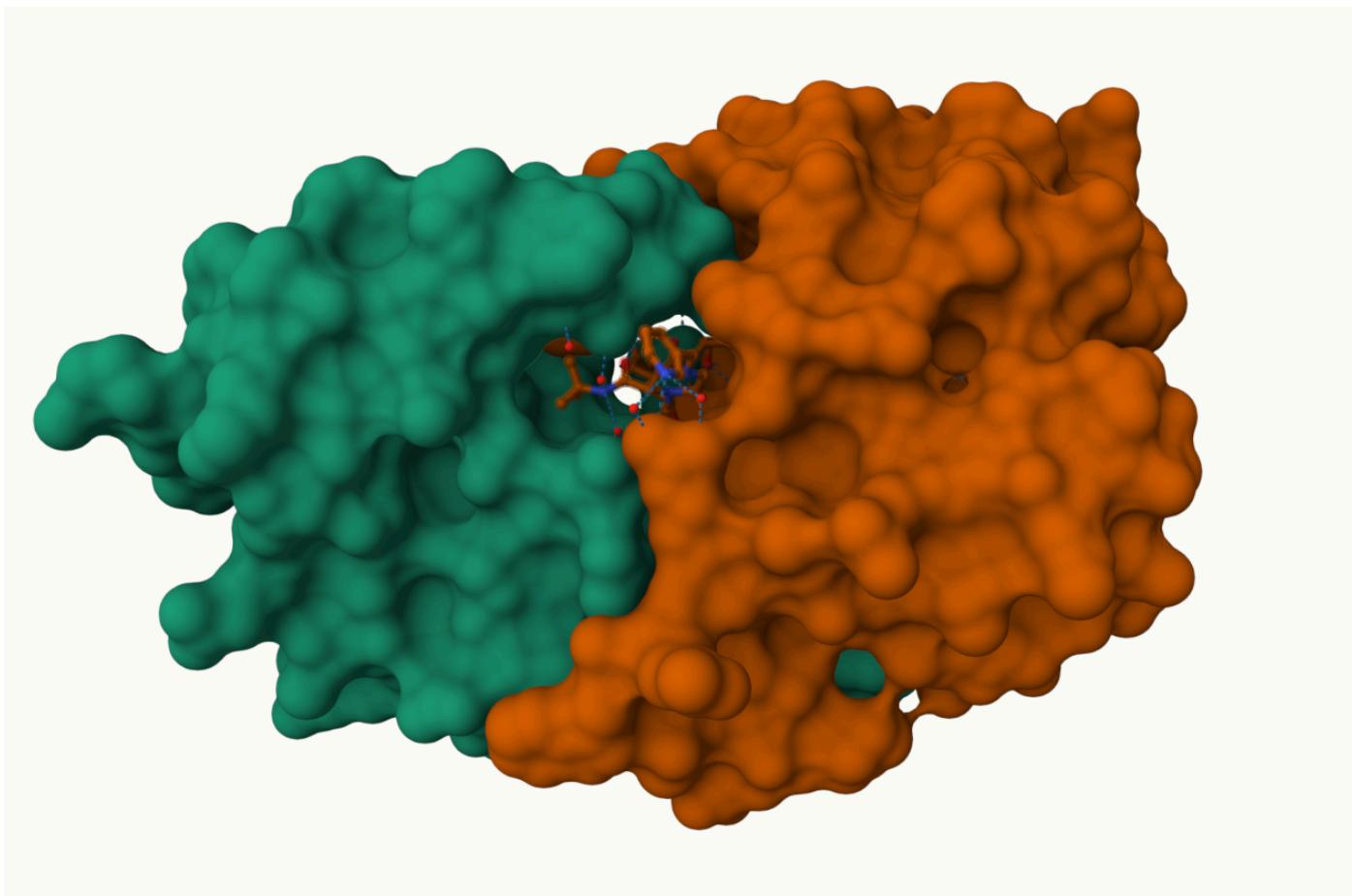


image 3

Another Cool image

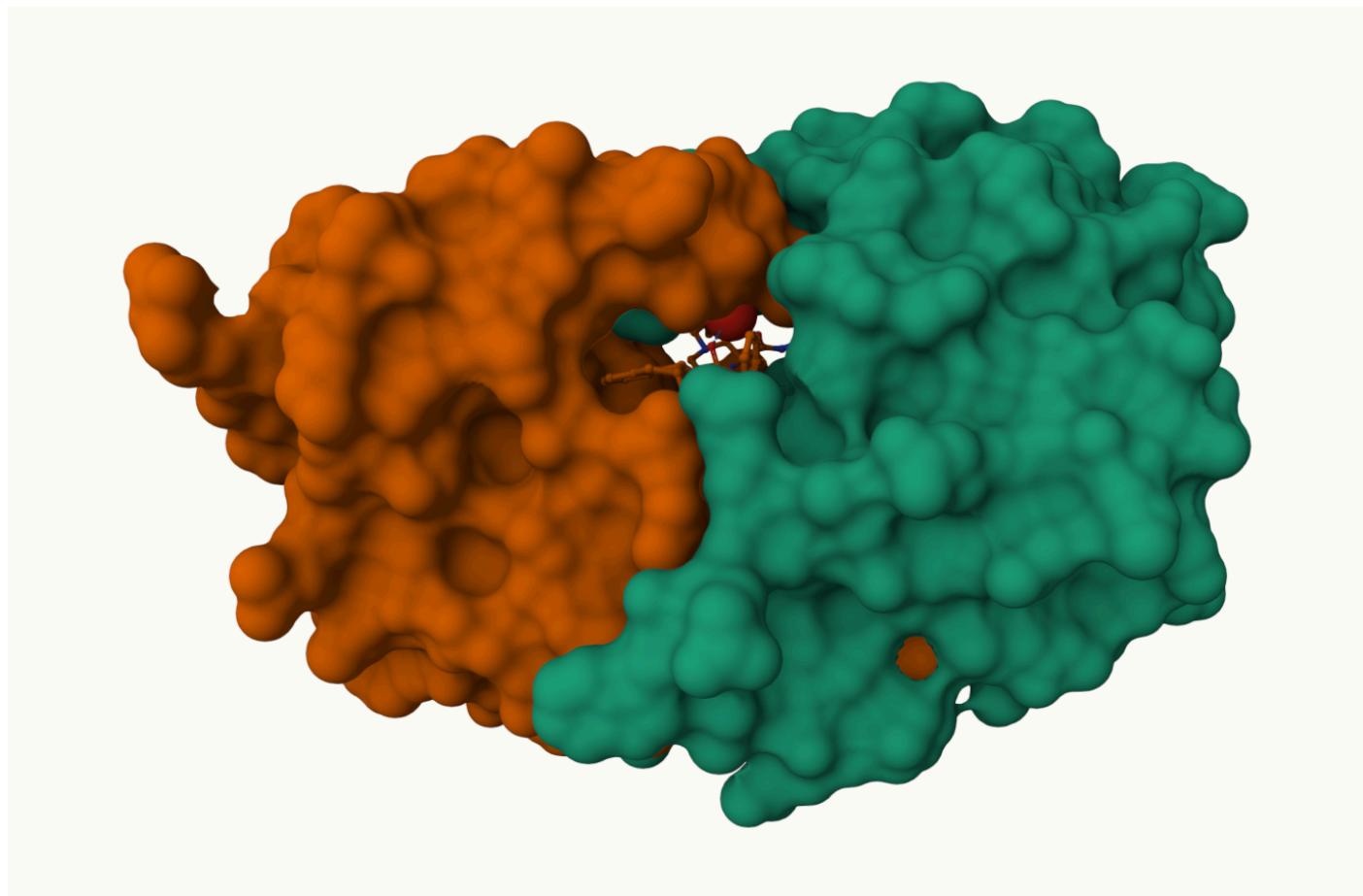


image 3

Another Cool image

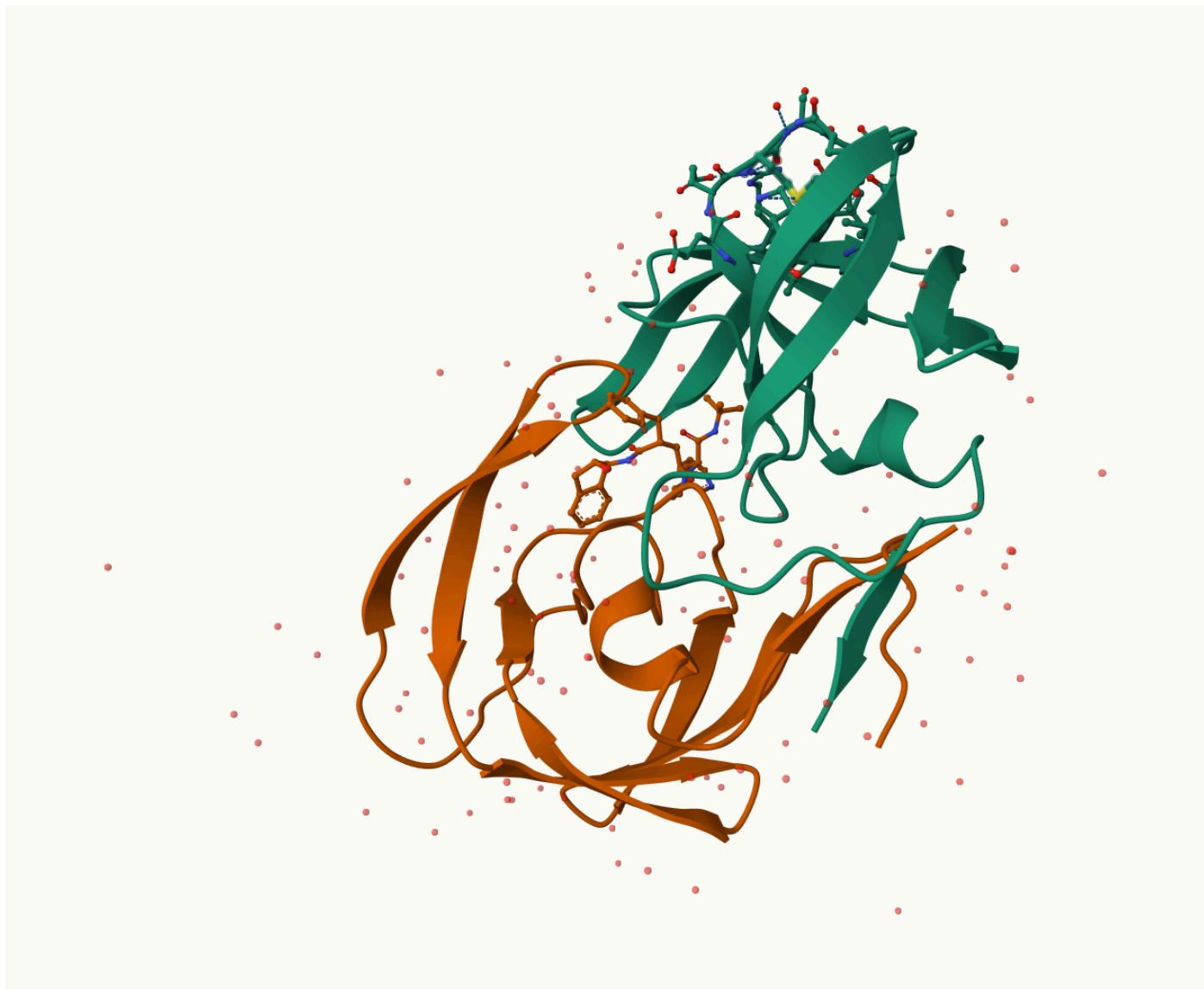


image 3

Q4 - Water molecules normally have 3 atoms. Why do we see just one atom per water molecule in this structure?

#H atom is on the beginning and it is very tiny. It is too small to see at this r

Q5 - There is a critical “conserved” water molecule in the binding site. Can you identify this water molecule? What residue number does this water molecule have

The watermolecule sits in the binding site.
Turn the water off, so that it is not visible.

Q6 - Generate and save a figure clearly showing the two distinct chains of HIV-protease along with the ligand. You might also consider showing the catalytic residues ASP 25 in each chain and the critical water (we recommend "Ball & Stick" for these side-chains). Add this figure to your Quarto document. Discussion Topic: Can you think of a way in which indinavir, or even larger ligands and substrates, could enter the binding site?

See previous uploaded

Q7 - [Optional] As you have hopefully observed HIV protease is a homodimer (i.e. it is composed of two identical chains). With the aid of the graphic display can you identify secondary structure elements that are likely to only form in the dimer rather than the monomer?

Optional

Interim

```
#Bio3D package for structural bioinformatics  
library(bio3d)  
pdb <- read.pdb("1hsg")
```

Note: Accessing on-line PDB file

pdb

Call: read.pdb(file = "1hsg")

Total Models#: 1
Total Atoms#: 1686, XYZs#: 5058 Chains#: 2 (values: A B)

Protein Atoms#: 1514 (residues/Calpha atoms#: 198)
Nucleic acid Atoms#: 0 (residues/phosphate atoms#: 0)

Non-protein/nucleic Atoms#: 172 (residues: 128)

Non-protein/nucleic resid values: [HOH (127), MK1 (1)]

Protein sequence:

```
PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWPKPMIGGIGGFVKVRQYD
QILIEICGHKAIGTVLVGPTPVNIIGRNLLTQIGCTLNFPQITLWQRPLVTIKIGGQLKE
ALLDTGADDTVLEEMSLPGRWPKPMIGGIGGFVKVRQYDQILIEICGHKAIGTVLGPTP
VNIIGRNLLTQIGCTLNF
```

+ attr: atom, xyz, seqres, helix, sheet,
calpha, remark, call

Interim 2

`attributes(pdb)`

```
$names
[1] "atom"   "xyz"    "seqres" "helix"  "sheet"  "calpha" "remark" "call"
$class
[1] "pdb"   "sse"
```

`head(pdb$atom)`

	type	eleno	elety	alt	resid	chain	resno	insert	x	y	z	o	b
1	ATOM	1	N	<NA>	PRO	A	1	<NA>	29.361	39.686	5.862	1	38.10
2	ATOM	2	CA	<NA>	PRO	A	1	<NA>	30.307	38.663	5.319	1	40.62
3	ATOM	3	C	<NA>	PRO	A	1	<NA>	29.760	38.071	4.022	1	42.64
4	ATOM	4	O	<NA>	PRO	A	1	<NA>	28.600	38.302	3.676	1	43.40
5	ATOM	5	CB	<NA>	PRO	A	1	<NA>	30.508	37.541	6.342	1	37.87
6	ATOM	6	CG	<NA>	PRO	A	1	<NA>	29.296	37.591	7.162	1	38.40
	segid	elesy	charge										
1	<NA>	N	<NA>										
2	<NA>	C	<NA>										
3	<NA>	C	<NA>										
4	<NA>	O	<NA>										
5	<NA>	C	<NA>										
6	<NA>	C	<NA>										

`pdbseq(pdb)`

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
"P"	"Q"	"I"	"T"	"L"	"W"	"Q"	"R"	"P"	"L"	"V"	"T"	"I"	"K"	"I"	"G"	"G"	"Q"	"L"	"K"
21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
"E"	"A"	"L"	"L"	"D"	"T"	"G"	"A"	"D"	"D"	"T"	"V"	"L"	"E"	"E"	"M"	"S"	"L"	"P"	"G"
41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
"R"	"W"	"K"	"P"	"K"	"M"	"I"	"G"	"G"	"I"	"G"	"G"	"F"	"I"	"K"	"V"	"R"	"Q"	"Y"	"D"
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
"Q"	"I"	"L"	"I"	"E"	"I"	"C"	"G"	"H"	"K"	"A"	"I"	"G"	"T"	"V"	"L"	"V"	"G"	"P"	"T"

```

  81   82   83   84   85   86   87   88   89   90   91   92   93   94   95   96   97   98   99   1
"P" "V" "N" "I" "I" "G" "R" "N" "L" "L" "T" "Q" "I" "G" "C" "T" "L" "N" "F" "P"
  2   3   4   5   6   7   8   9   10  11  12  13  14  15  16  17  18  19  20  21
"Q" "I" "T" "L" "W" "Q" "R" "P" "L" "V" "T" "I" "K" "I" "G" "G" "Q" "L" "K" "E"
 22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41
"A" "L" "L" "D" "T" "G" "A" "D" "D" "T" "V" "L" "E" "E" "M" "S" "L" "P" "G" "R"
 42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61
"W" "K" "P" "K" "M" "I" "G" "G" "I" "G" "G" "F" "I" "K" "V" "R" "Q" "Y" "D" "Q"
 62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81
"I" "L" "I" "E" "I" "C" "G" "H" "K" "A" "I" "G" "T" "V" "L" "V" "G" "P" "T" "P"
 82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99
"V" "N" "I" "I" "G" "R" "N" "L" "L" "T" "Q" "I" "G" "C" "T" "L" "N" "F"

```

`pdbseq(pdb) [25]`

25
"D"

Interim 3

Functional dynamics prediction

Predicting functional motions of a single

```

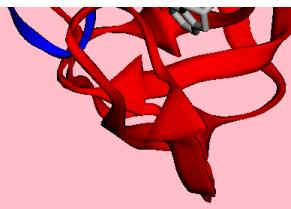
source("https://tinyurl.com/viewpdb")
library(r3dmol)
library(shiny)

```

Warning: package 'shiny' was built under R version 4.3.3

`view.pdb(pdb, backgroundColor = "pink")`





Interim 4

```
adk <- read.pdb("6s36")
```

Note: Accessing on-line PDB file
PDB has ALT records, taking A only, rm.alt=TRUE

```
adk
```

Call: read.pdb(file = "6s36")

Total Models#: 1

Total Atoms#: 1898, XYZs#: 5694 Chains#: 1 (values: A)

Protein Atoms#: 1654 (residues/Calpha atoms#: 214)

Nucleic acid Atoms#: 0 (residues/phosphate atoms#: 0)

Non-protein/nucleic Atoms#: 244 (residues: 244)

Non-protein/nucleic resid values: [CL (3), HOH (238), MG (2), NA (1)]

Protein sequence:

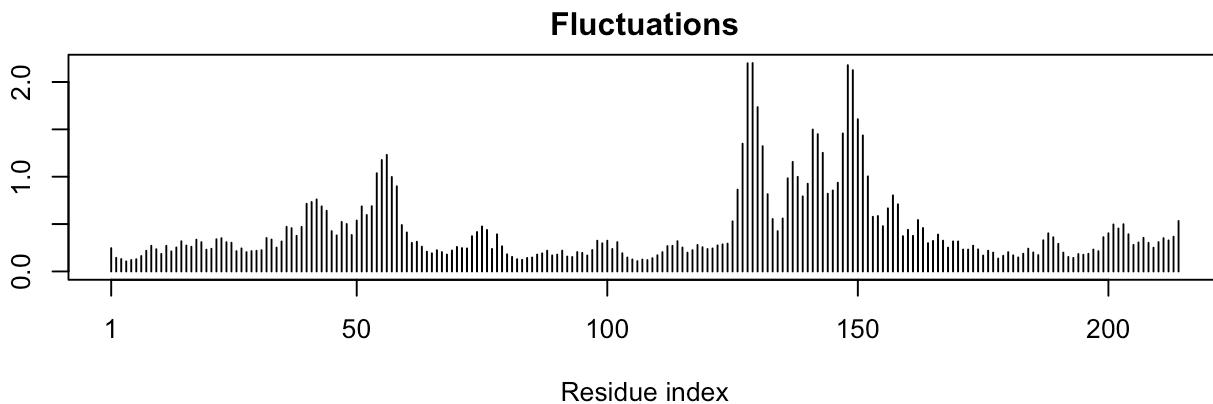
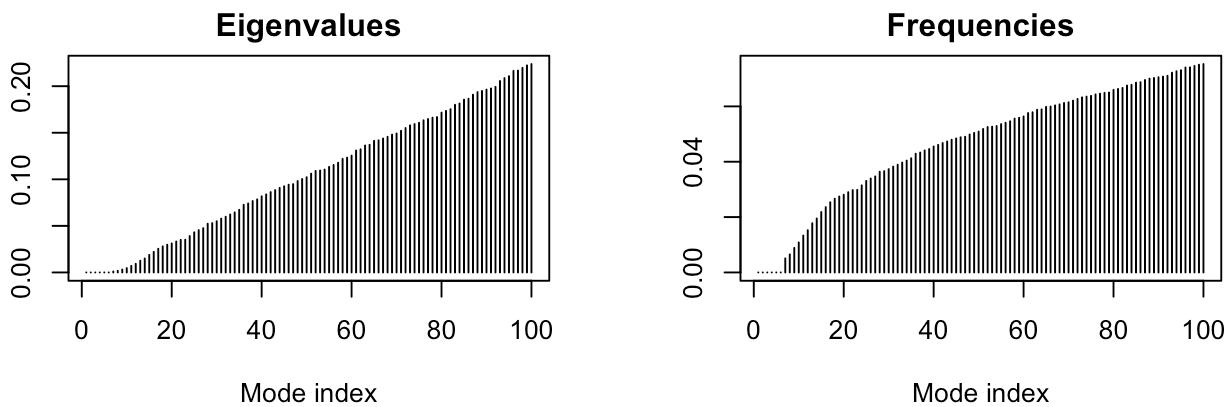
```
MRIILLGAPGAGKGTAQFIMEKYGIPQISTGMLRAAVKSGSELGKQAKDIDAGKLVT  
DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELVDKI  
VGRRVHAPSGRVYHVFKNPPKVEGKDDVTGEELTRKDDQEETVRKRLVEYHQMTAPLIG  
YYSKAEAGNTKYAKVDTGPVAEVRADLEKILG
```

```
+ attr: atom, xyz, seqres, helix, sheet,  
       calpha, remark, call
```

```
modes <- nma(adk)
```

Building Hessian... Done in 0.018 seconds.
Diagonalizing Hessian... Done in 0.384 seconds.

```
plot(modes)
```



```
mktrj(modes, file='adk.pdb') #this is only A amino acids
mktrj(modes, pdb=adk, file='adk.pdb')
```

Q7 - How many amino acid residues are there in this pdb object?

```
length(pdbseq(pdb)) #answer = 198
```

[1] 198

Q8 - Name one of the two non-protein residues?

```
#In the summary output of the pdb object, it shows that there are non-protein res
```

Q9 - How many protein chains are in this structure?

```
# Number of unique protein chains  
num_chains <- length(unique(pdb$atom$chain))  
print(num_chains) #answer is 2
```

[1] 2

Q10 - Which of the packages above is found only on BioConductor and not CRAN?

```
#msa  
# not needed, no time in class
```

Q11 - Which of the above packages is not found on BioConductor or CRAN?

```
#bitbucket  
# not needed, no time in class
```

Q12 - True or False? Functions from the devtools package can be used to install packages from GitHub and BitBucket?

```
#True  
# not needed, no time in class
```

Q13 - How many amino acids are in this sequence, i.e. how long is this sequence?

```
# not needed, no time in class
```

Q14 - What do you note about this plot? Are the black and colored lines similar or different? Where do you think they differ most and why?

```
# not needed, no time in class
```

