Class 11

Ashley Martinez (A17891957)

AlphaFold Data Base

The EBI maintians the largest database of AlphaFold structure prediction models at: https://alphafold.ebi.ac.uk/

From last class before Halloween, we saw that the PDB had 244,290 (Oct 2025)

The total number of protein sequences in UniProtKB is 199,579,901

> Key Point: This is a tiny fraction of sequence space that has structual coverage(0.12%)

```
244290/199579901*100
```

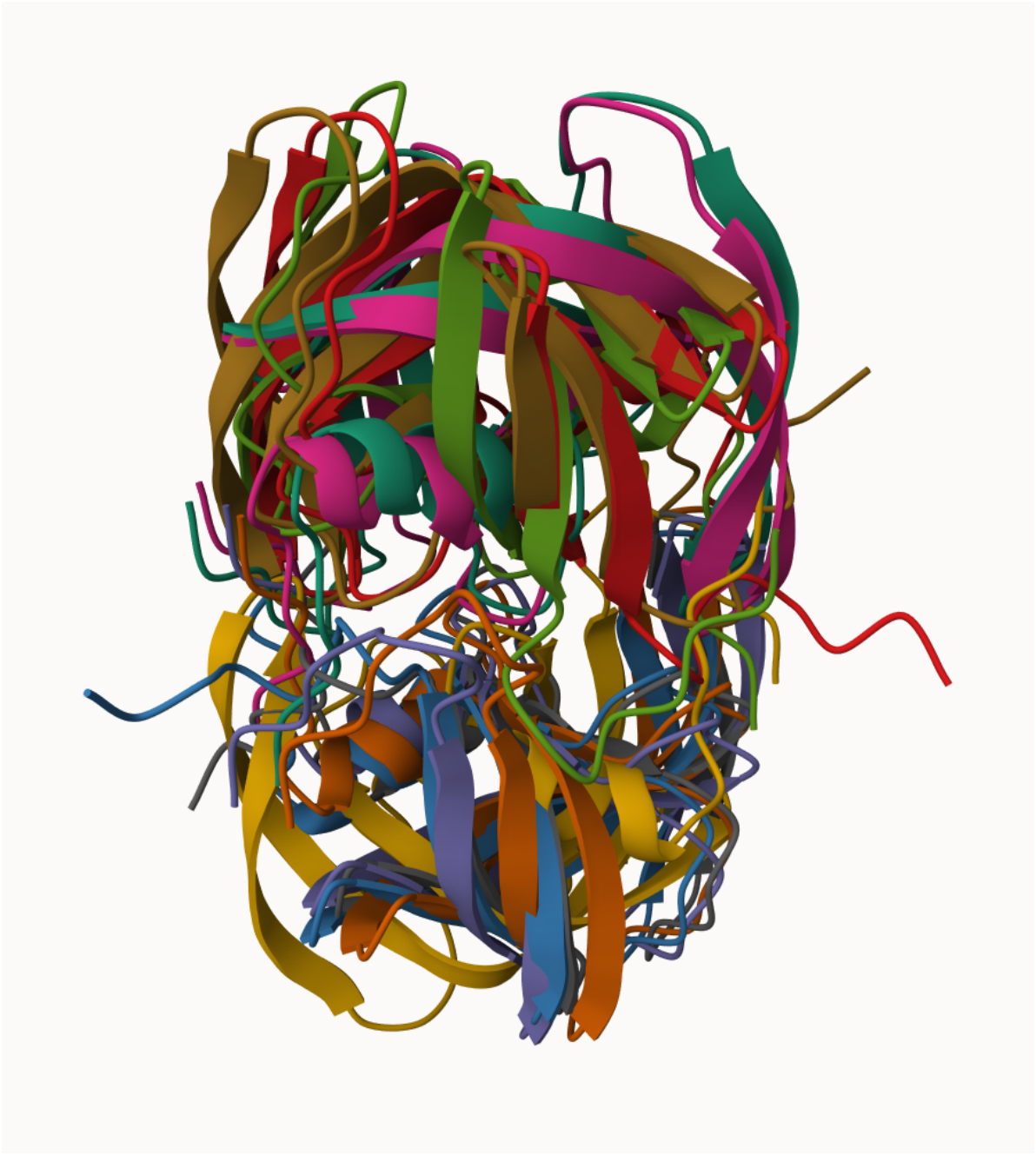
```
[1] 0.1224021
```

AFDB is attempting to address this gap...

There are two "Quality Scores" from AlphaFold one for residues (i.e. each amino acid) this is called **pLDDT** score the second is the **PAE** score. PAE measures the confidence in the relative position of two residues (i.e. a score for every pair of residues).

Generating your own structure predictions

Figure of 5 generated HIV-PR models



And the top ranked model colored by chain and model 5

Custom analysis of resulting models in R

Read key result files into R. The first thing I need to know is what my results directory is called (i.e. if name is different for every AlphaFold run/job)

```
results_dir<-"HIVPR_dimer_23119"

#File
pdb_files <- list.files(path=results_dir,
                        pattern="*.pdb",
```

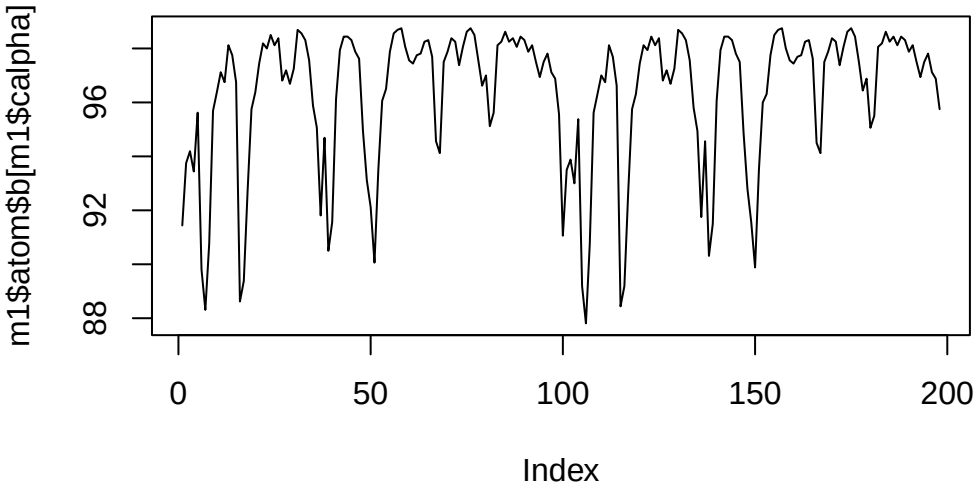
```
                          full.names = TRUE)
#Print our PDB File names
basename(pdb_files)
```

```
[1] "HIVPR_dimer_23119_unrelaxed_rank_001_alphafold2_multimer_v3_model_4_seed_000.pdb"
[2] "HIVPR_dimer_23119_unrelaxed_rank_002_alphafold2_multimer_v3_model_1_seed_000.pdb"
[3] "HIVPR_dimer_23119_unrelaxed_rank_003_alphafold2_multimer_v3_model_5_seed_000.pdb"
[4] "HIVPR_dimer_23119_unrelaxed_rank_004_alphafold2_multimer_v3_model_2_seed_000.pdb"
[5] "HIVPR_dimer_23119_unrelaxed_rank_005_alphafold2_multimer_v3_model_3_seed_000.pdb"
```

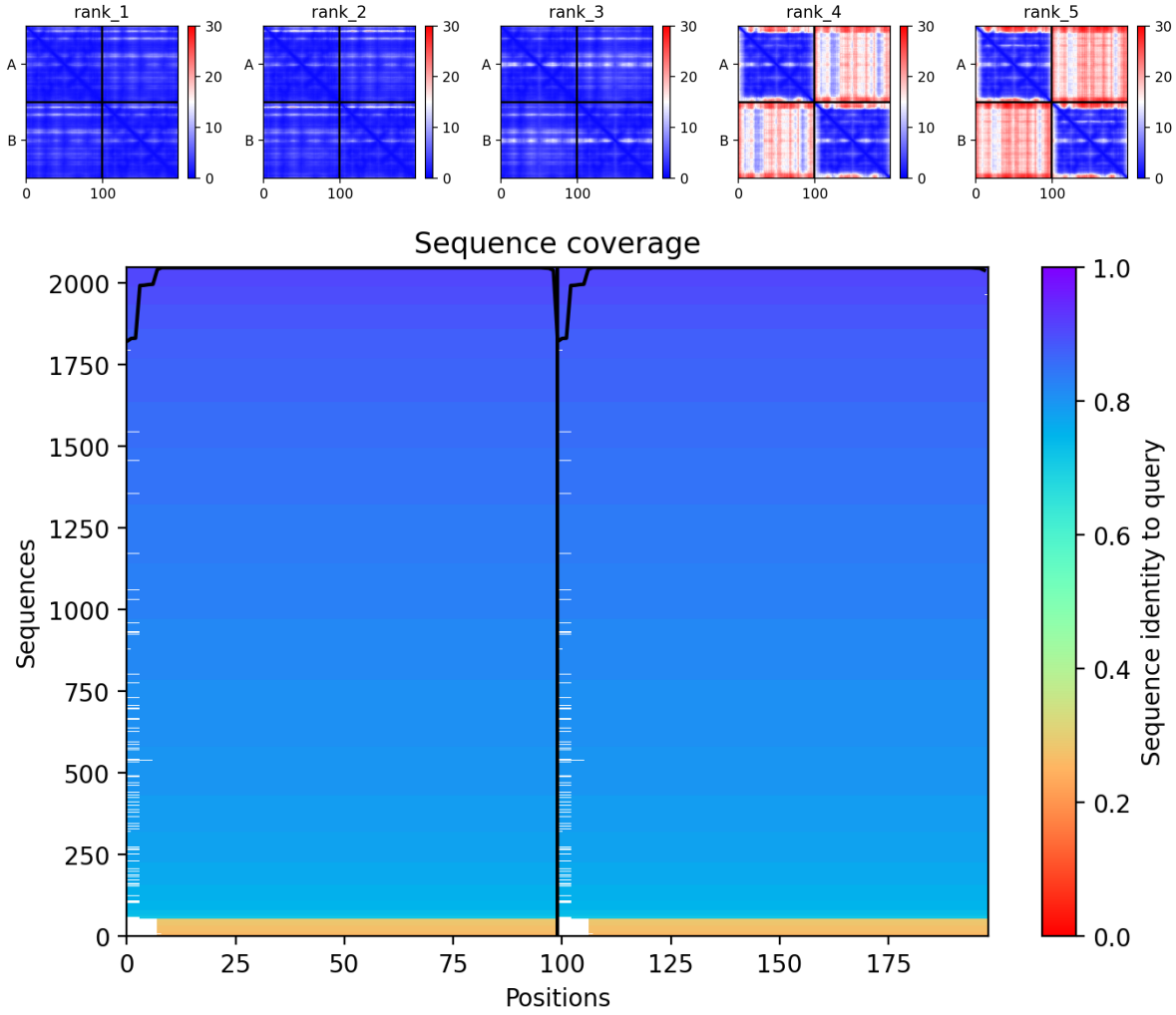
```
library(bio3d)
m1<-read.pdb(pdb_files[1])
pdbs <- pdbaln(pdb_files, fit=TRUE, exefile="msa")
```

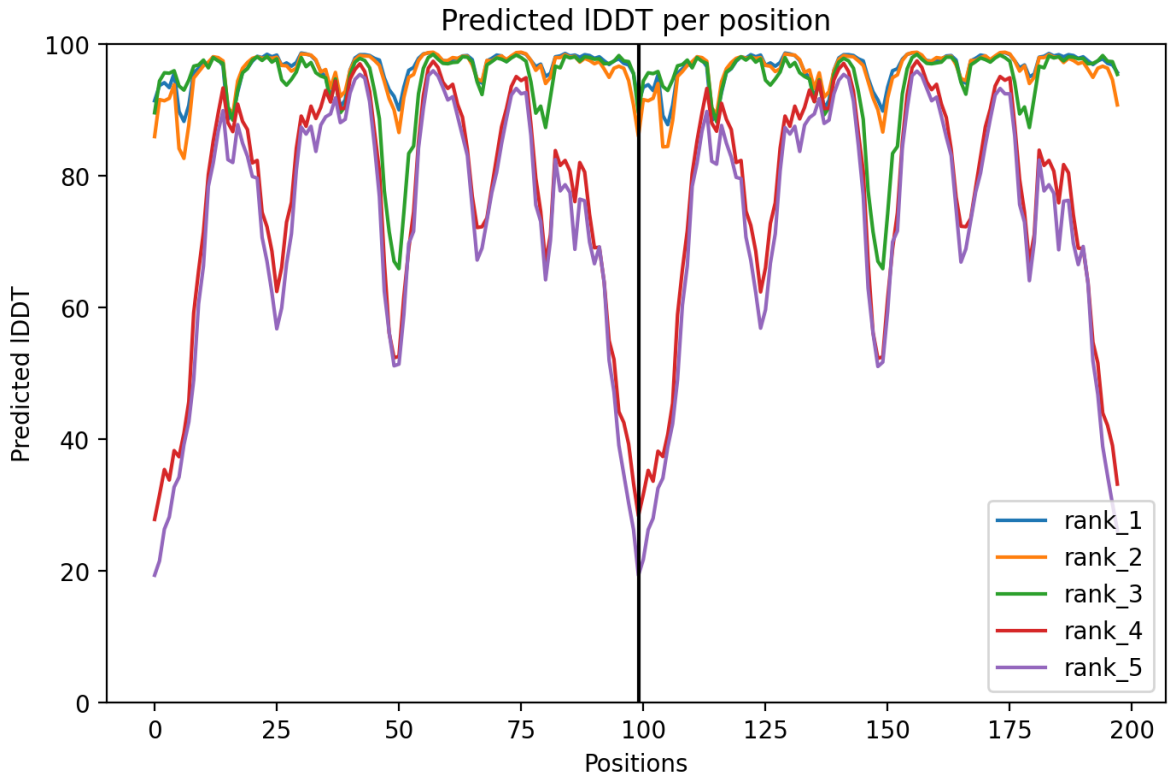
```
Reading PDB files:
HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_001_alphafold2_multimer_v3_model_4_seed_00
HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_002_alphafold2_multimer_v3_model_1_seed_00
HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_003_alphafold2_multimer_v3_model_5_seed_00
HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_004_alphafold2_multimer_v3_model_2_seed_00
HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_005_alphafold2_multimer_v3_model_3_seed_00
.....

Extracting sequences

pdb/seq: 1   name: HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_001_alphafold2_multimer
pdb/seq: 2   name: HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_002_alphafold2_multimer
pdb/seq: 3   name: HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_003_alphafold2_multimer
pdb/seq: 4   name: HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_004_alphafold2_multimer
pdb/seq: 5   name: HIVPR_dimer_23119/HIVPR_dimer_23119_unrelaxed_rank_005_alphafold2_multimer
```

```
plot(m1$atom$b[m1$calpha],typ="l")
```


m1$atom$b[m1$calpha]
88
92
96
0
50
100
150
200
Index

Predicted Alignment Error for Domains





Residue conservation from alignment file

Find the large AlphaFold alignment file

```
aln_file <- list.files(path=results_dir,
                       pattern=".a3m$",
                        full.names = TRUE)
aln_file
```

```
[1] "HIVPR_dimer_23119/HIVPR_dimer_23119.a3m"
```

How many sequences are in this alignment

```
aln <- read.fasta(aln_file[1], to.upper = TRUE)
```

```
[1] " ** Duplicated sequence id's: 101 **"
[2] " ** Duplicated sequence id's: 101 **"
```

```
dim(aln$ali)
```

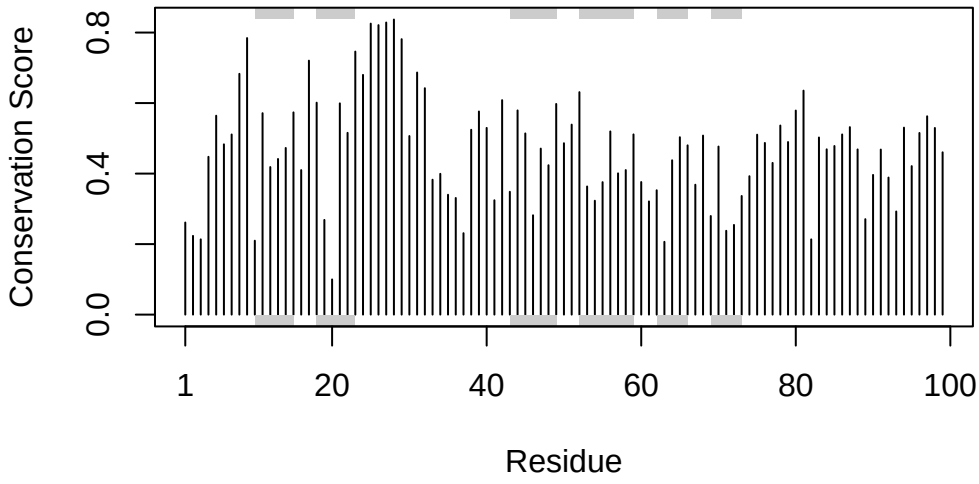
```
[1] 5397  132
```

We can score residue conservation in the alignment with the conserv() function.

```
sim <- conserv(aln)
pdb <- read.pdb("1hsg")
```

```
  Note: Accessing on-line PDB file
```

```
plotb3(sim[1:99], sse=trim.pdb(pdb, chain="A"),
       ylab="Conservation Score")
```




```
con <- consensus(aln, cutoff = 0.9)
con$seq
```

```
  [1] "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-"
 [19] "-" "-" "-" "-" "-" "-" "D" "T" "G" "A" "-" "-" "-" "-" "-" "-" "-" "-"
 [37] "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-"
```

```
 [55] "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-"
 [73] "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-"
 [91] "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-"
[109] "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-" "-"
[127] "-" "-" "-" "-" "-" "-"
```

```
m1.pdb <- read.pdb(pdb_files[1])
occ <- vec2resno(c(sim[1:99], sim[1:99]), m1.pdb$atom$resno)
write.pdb(m1.pdb, o=occ, file="m1_conserv.pdb")
```