

class 11 alphafold

Anqi Feng

Today, before delving into structural prediction with AlphaFold, I will finish off our previous lab10 “comparative structural analysis”

1. sequence identifier

```
library(bio3d)
id <-"1ake_A"
aa<- get.seq(id)
```

Warning in get.seq(id): Removing existing file: seqs.fasta

Fetching... Please wait. Done.

aa

```

      1      .      .      .      .      .      .      60
pdb|1AKE|A  MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMRLAAVKSGSELGKQAKDIMDAGKLV
      1      .      .      .      .      .      .      60

      61      .      .      .      .      .      .      120
pdb|1AKE|A  DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDRI
      61      .      .      .      .      .      .      120

      121      .      .      .      .      .      .      180
pdb|1AKE|A  VGRRVHAPSGRVYHVKNPPKVEGKDDVTGEELTRKDDQEETVRKRLVEYHQMTAPLIG
      121      .      .      .      .      .      .      180

      181      .      .      .      214
pdb|1AKE|A  YYSKEAEAGNTKYAKVDGTPVAEVRADLEKILG
      181      .      .      .      214
```

Call:

```
read.fasta(file = outfile)
```

Class:

```
fasta
```

Alignment dimensions:

```
1 sequence rows; 214 position columns (214 non-gap, 0 gap)
```

```
+ attr: id, ali, call
```

```
b <- blast.pdb(aa)
```

```
Searching ... please wait (updates every 5 seconds) RID = JS115HDG016
```

```
.....
```

```
Reporting 85 hits
```

```
attributes(b)
```

```
$names
```

```
[1] "hit.tbl" "raw"      "url"
```

```
$class
```

```
[1] "blast"
```

```
head(b$hit.tbl)
```

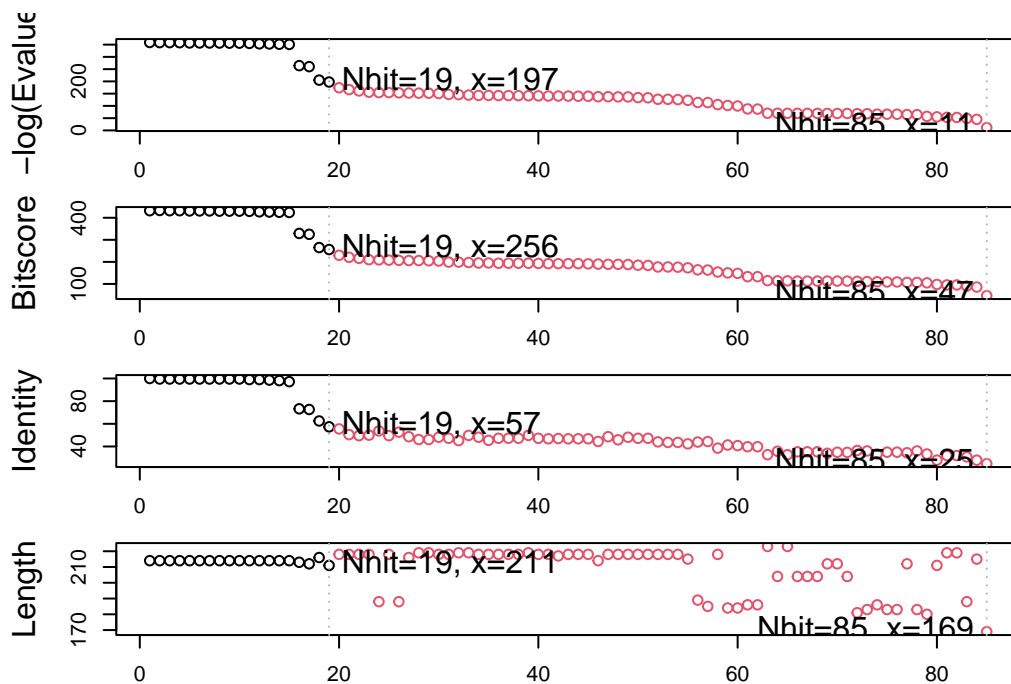
	queryid	subjectids	identity	alignmentlength	mismatches	gapopens	q.start
1	Query_1479401	1AKE_A	100.000	214	0	0	1
2	Query_1479401	8BQF_A	99.533	214	1	0	1
3	Query_1479401	4X8M_A	99.533	214	1	0	1
4	Query_1479401	6S36_A	99.533	214	1	0	1
5	Query_1479401	8Q2B_A	99.533	214	1	0	1
6	Query_1479401	8RJ9_A	99.533	214	1	0	1

	q.end	s.start	s.end	evaluate	bitscore	positives	mlog.evaluate	pdb.id	acc
1	214	1	214	1.58e-156	432	100.00	358.7458	1AKE_A	1AKE_A
2	214	21	234	2.58e-156	433	100.00	358.2555	8BQF_A	8BQF_A
3	214	1	214	2.82e-156	432	100.00	358.1665	4X8M_A	4X8M_A
4	214	1	214	4.14e-156	432	100.00	357.7826	6S36_A	6S36_A
5	214	1	214	1.10e-155	431	99.53	356.8054	8Q2B_A	8Q2B_A
6	214	1	214	1.10e-155	431	99.53	356.8054	8RJ9_A	8RJ9_A

```
hits <- plot(b)
```

```
* Possible cutoff values: 197 11
    Yielding Nhits:      19 85
```

```
* Chosen cutoff value of: 197
    Yielding Nhits:      19
```



```
attributes(hits)
```

```
$names
[1] "hits" "pdb.id" "acc" "inds"
```

```
$class
[1] "blast"
```

Top hits we like from our blast results would be

```
hits$pdb.id
```

```
[1] "1AKE_A" "8BQF_A" "4X8M_A" "6S36_A" "8Q2B_A" "8RJ9_A" "6RZE_A" "4X8H_A"
[9] "3HPR_A" "1E4V_A" "5EJE_A" "1E4Y_A" "3X2S_A" "6HAP_A" "6HAM_A" "4K46_A"
[17] "4NP6_A" "3GMT_A" "4PZL_A"
```

```
#Download related pdb files
files<- get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip=TRUE)
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/1AKE.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/8BQF.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/4X8M.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/6S36.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/8Q2B.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/8RJ9.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/6RZE.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/4X8H.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/3HPR.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/1E4V.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):
pdbs/5EJE.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):  
pdbs/1E4Y.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):  
pdbs/3X2S.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):  
pdbs/6HAP.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):  
pdbs/6HAM.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):  
pdbs/4K46.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):  
pdbs/4NP6.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):  
pdbs/3GMT.pdb.gz exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", aplit = TRUE, gzip = TRUE):  
pdbs/4PZL.pdb.gz exists. Skipping download
```

I have now found and downloaded all adk structures in the new database but viewing them is difficult as they need to be aligned and superposed (bioconductor software)

I am going to install BiocManager package from CRAN, then I can use `BiocManager::install()` to install any bioconductor package

```
#install.packages("BiocManager")
```

```
# Align releated PDBs  
pdbs <- pdbaln(files, fit = TRUE, exefile="msa")
```

Reading PDB files:

pdb/1AKE.pdb
pdb/8BQF.pdb
pdb/4X8M.pdb
pdb/6S36.pdb
pdb/8Q2B.pdb
pdb/8RJ9.pdb
pdb/6RZE.pdb
pdb/4X8H.pdb
pdb/3HPR.pdb
pdb/1E4V.pdb
pdb/5EJE.pdb
pdb/1E4Y.pdb
pdb/3X2S.pdb
pdb/6HAP.pdb
pdb/6HAM.pdb
pdb/4K46.pdb
pdb/4NP6.pdb
pdb/3GMT.pdb
pdb/4PZL.pdb

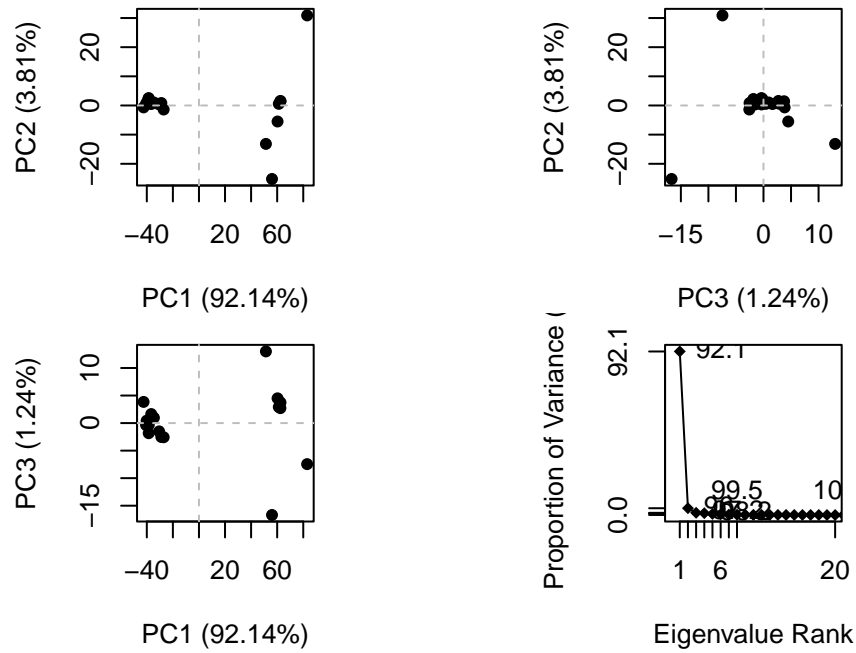
 PDB has ALT records, taking A only, rm.alt=TRUE
. PDB has ALT records, taking A only, rm.alt=TRUE
.. PDB has ALT records, taking A only, rm.alt=TRUE
. PDB has ALT records, taking A only, rm.alt=TRUE
. PDB has ALT records, taking A only, rm.alt=TRUE
. PDB has ALT records, taking A only, rm.alt=TRUE
.. PDB has ALT records, taking A only, rm.alt=TRUE
.. PDB has ALT records, taking A only, rm.alt=TRUE
.... PDB has ALT records, taking A only, rm.alt=TRUE
. PDB has ALT records, taking A only, rm.alt=TRUE
. PDB has ALT records, taking A only, rm.alt=TRUE
.. PDB has ALT records, taking A only, rm.alt=TRUE
.

Extracting sequences

pdb/seq: 1 name: pdb/1AKE.pdb
 PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 2 name: pdb/8BQF.pdb
 PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 3 name: pdb/4X8M.pdb
pdb/seq: 4 name: pdb/6S36.pdb
 PDB has ALT records, taking A only, rm.alt=TRUE

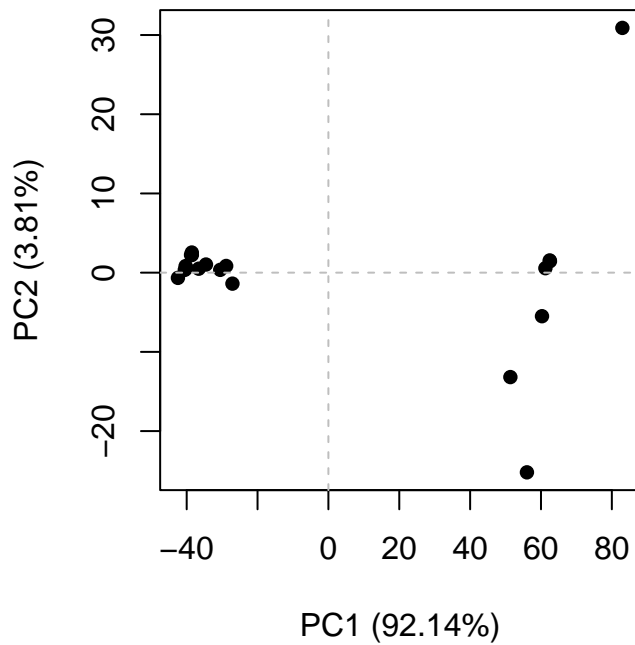
```
pdb/seq: 5    name: pdbc/8Q2B.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 6    name: pdbc/8RJ9.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 7    name: pdbc/6RZE.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 8    name: pdbc/4X8H.pdb
pdb/seq: 9    name: pdbc/3HPR.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 10   name: pdbc/1E4V.pdb
pdb/seq: 11   name: pdbc/5EJE.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 12   name: pdbc/1E4Y.pdb
pdb/seq: 13   name: pdbc/3X2S.pdb
pdb/seq: 14   name: pdbc/6HAP.pdb
pdb/seq: 15   name: pdbc/6HAM.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 16   name: pdbc/4K46.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 17   name: pdbc/4NP6.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 18   name: pdbc/3GMT.pdb
pdb/seq: 19   name: pdbc/4PZL.pdb
  PDB has ALT records, taking A only, rm.alt=TRUE
```

```
pc<- pca(pdbc)
plot(pc)
```



To examine in more detail what PC1 (or any PC) is capturing here we can plot the loadings or make a wee movie (trajectory) of moving along PC1

```
plot(pc, pc.axes=c(1,2))
```




```
mktrj(pc,pc=1, file = "pc1.pdb")
```