Longer ORF PDB search and results

Mirte Ciz Marieke Kuijpers

02/03/2022

The code in this document is made to be useful with either the long or the short ORF, but in the set-up below the sequence to use is set to the long ORF.

# Set-up  
library("bio3d")  
library("ggplot2")  
library("ggrepel")  
library("msa")  
library("bio3d.view")  
  
# Load sequence of POI  
seqL <- read.fasta("long.ORF.fa")  
seqS <- read.fasta("short.ORF.fa")  
  
#Choose which sequence to use  
seq <- seqS

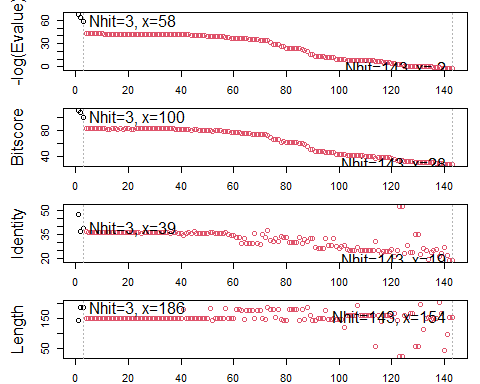
After set up the blast search can be completed and the summary statistics of this search can be plotted.

# Blast search  
blast <- blast.pdb(seq, database = "pdb")

## Searching ... please wait (updates every 5 seconds) RID = 1ZR0NWEN013   
## .  
## Reporting 143 hits

# Plot summary statistics of results  
hits <- plot(blast)

## \* Possible cutoff values: 58 -3   
## Yielding Nhits: 3 143   
##   
## \* Chosen cutoff value of: 58   
## Yielding Nhits: 3



# Print the IDs of the hits above the threshold  
hit.IDs <- hits$pdb.id  
hit.IDs

## [1] "6GYH\_A" "7BMH\_A" "5AWZ\_A"

There are 3 hits that pass the statistical threshold, namely: 6GYH\_A, 7BMH\_A, 5AWZ\_A. More information can be found on these by interrogating the blast results.

# Show the hit table for the top hits which pass the threshold  
head(blast$hit.tbl, n = length(hit.IDs))

## queryid subjectids identity alignmentlength mismatches gapopens q.start  
## 1 Query\_40791 6GYH\_A 47.222 144 74 2 35  
## 2 Query\_40791 7BMH\_A 36.898 187 112 2 1  
## 3 Query\_40791 5AWZ\_A 38.710 186 85 5 2  
## q.end s.start s.end evalue bitscore positives mlog.evalue pdb.id acc  
## 1 177 77 219 7.74e-30 110 61.81 67.03115 6GYH\_A 6GYH\_A  
## 2 181 92 278 2.13e-28 108 52.94 63.71626 7BMH\_A 7BMH\_A  
## 3 176 61 228 4.29e-26 100 51.08 58.41093 5AWZ\_A 5AWZ\_A

We can also download these PDB files, annotate them for more information and align them with our sequence to get an overview of sequence alignment.

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

## | | | 0% | |======================= | 33% | |=============================================== | 67% | |======================================================================| 100%

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

## Reading PDB files:  
## pdbs/split\_chain/6GYH\_A.pdb  
## pdbs/split\_chain/7BMH\_A.pdb  
## pdbs/split\_chain/5AWZ\_A.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  
## .  
##   
## Extracting sequences  
##   
## pdb/seq: 1 name: pdbs/split\_chain/6GYH\_A.pdb   
## PDB has ALT records, taking A only, rm.alt=TRUE  
## pdb/seq: 2 name: pdbs/split\_chain/7BMH\_A.pdb   
## PDB has ALT records, taking A only, rm.alt=TRUE  
## pdb/seq: 3 name: pdbs/split\_chain/5AWZ\_A.pdb   
## PDB has ALT records, taking A only, rm.alt=TRUE

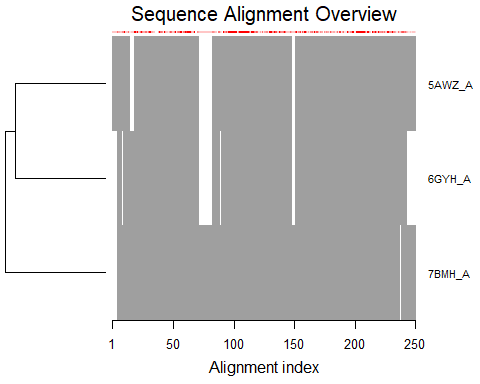
# Vector containing PDB codes for figure axis  
ids <- basename.pdb(pdbs$id)  
  
# Annotate hits for more information on the hits  
anno <- pdb.annotate(ids)  
  
# Find the organisms these PDB hits come from  
unique(anno$source)

## [1] "Coccomyxa subellipsoidea C-169" "Leptosphaeria maculans"   
## [3] "Acetabularia acetabulum"

# View more information on the hits  
anno

## structureId chainId macromoleculeType chainLength experimentalTechnique  
## 6GYH\_A 6GYH A Protein 236 X-ray  
## 7BMH\_A 7BMH A Protein 324 X-ray  
## 5AWZ\_A 5AWZ A Protein 244 X-ray  
## resolution scopDomain pfam  
## 6GYH\_A 2.00 <NA> Bacteriorhodopsin-like protein (Bac\_rhodopsin)  
## 7BMH\_A 2.20 <NA> Bacteriorhodopsin-like protein (Bac\_rhodopsin)  
## 5AWZ\_A 1.57 <NA> Bacteriorhodopsin-like protein (Bac\_rhodopsin)  
## ligandId  
## 6GYH\_A RET,CLR,OLB (4)  
## 7BMH\_A LFA (22),OLA (3)  
## 5AWZ\_A OCT (2),C14,RET,OLB,D12 (2),D10 (3)  
## ligandName  
## 6GYH\_A RETINAL,CHOLESTEROL,(2S)-2,3-dihydroxypropyl (9Z)-octadec-9-enoate (4)  
## 7BMH\_A EICOSANE (22),OLEIC ACID (3)  
## 5AWZ\_A N-OCTANE (2),TETRADECANE,RETINAL,(2S)-2,3-dihydroxypropyl (9Z)-octadec-9-enoate,DODECANE (2),DECANE (3)  
## source  
## 6GYH\_A Coccomyxa subellipsoidea C-169  
## 7BMH\_A Leptosphaeria maculans  
## 5AWZ\_A Acetabularia acetabulum  
## structureTitle  
## 6GYH\_A Crystal structure of the light-driven proton pump Coccomyxa subellipsoidea Rhodopsin CsR  
## 7BMH\_A Crystal structure of a light-driven proton pump LR (Mac) from Leptosphaeria maculans  
## 5AWZ\_A Crystal Structure of the Cell-Free Synthesized Membrane Protein, Acetabularia Rhodopsin I, at 1.57 angstrom  
## citation rObserved  
## 6GYH\_A Fudim, R., et al. Sci Signal (2019) 0.19398  
## 7BMH\_A Zabelskii, D., et al. Commun Biol (2021) 0.23840  
## 5AWZ\_A Furuse, M., et al. Acta Crystallogr D Biol Crystallogr (2015) 0.17760  
## rFree rWork spaceGroup  
## 6GYH\_A 0.22493 0.19231 H 3  
## 7BMH\_A 0.28470 0.23610 P 21 21 21  
## 5AWZ\_A 0.19410 0.17690 C 1 2 1

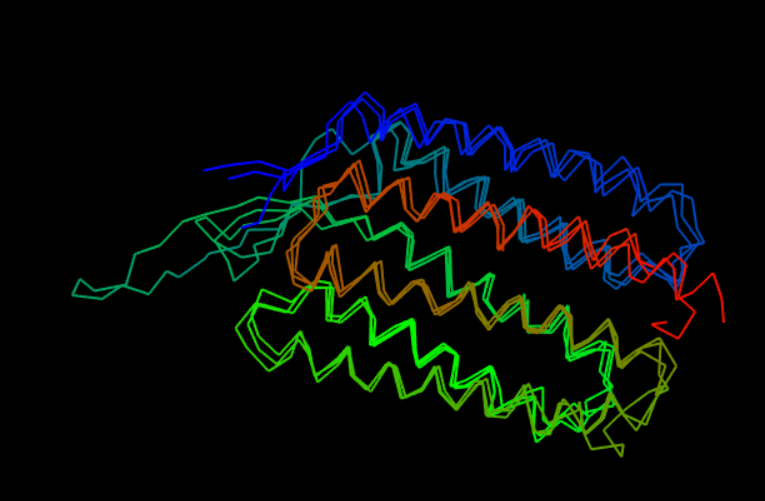
# Draw schematic alignment  
plot(pdbs, labels=ids)

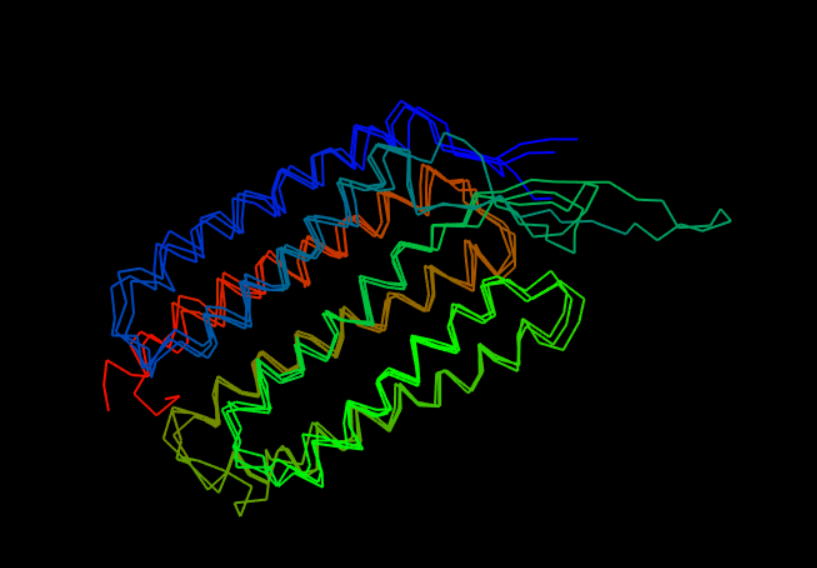


We can also plot the three structures we have found as follows:

# Set up  
library(bio3d.view)  
library(rgl)  
  
# Plot  
#view.pdbs(pdbs)

The View.pdbs() function brings up an interactive viewer, which cannot be directly viewed in the markdown document, so instead two screen-shots of this have been inserted.





With more proteins it could be interesting to plot variability, or even do PCA using the amino acid position data, but with only three proteins this is not useful.

# Session Information

sessionInfo()

## R version 4.1.2 (2021-11-01)  
## Platform: x86\_64-w64-mingw32/x64 (64-bit)  
## Running under: Windows 10 x64 (build 22000)  
##   
## Matrix products: default  
##   
## locale:  
## [1] LC\_COLLATE=English\_United Kingdom.1252   
## [2] LC\_CTYPE=English\_United Kingdom.1252   
## [3] LC\_MONETARY=English\_United Kingdom.1252  
## [4] LC\_NUMERIC=C   
## [5] LC\_TIME=English\_United Kingdom.1252   
##   
## attached base packages:  
## [1] stats4 stats graphics grDevices utils datasets methods   
## [8] base   
##   
## other attached packages:  
## [1] rgl\_0.108.3 bio3d.view\_0.1.0.9000 msa\_1.26.0   
## [4] Biostrings\_2.62.0 GenomeInfoDb\_1.30.1 XVector\_0.34.0   
## [7] IRanges\_2.28.0 S4Vectors\_0.32.3 BiocGenerics\_0.40.0   
## [10] ggrepel\_0.9.1 ggplot2\_3.3.5 bio3d\_2.4-3.9000   
##   
## loaded via a namespace (and not attached):  
## [1] tidyselect\_1.1.2 xfun\_0.29 purrr\_0.3.4   
## [4] colorspace\_2.0-2 vctrs\_0.3.8 generics\_0.1.2   
## [7] htmltools\_0.5.2 yaml\_2.2.2 utf8\_1.2.2   
## [10] rlang\_1.0.1 pillar\_1.7.0 glue\_1.6.1   
## [13] withr\_2.4.3 GenomeInfoDbData\_1.2.7 lifecycle\_1.0.1   
## [16] stringr\_1.4.0 zlibbioc\_1.40.0 munsell\_0.5.0   
## [19] gtable\_0.3.0 htmlwidgets\_1.5.4 evaluate\_0.15   
## [22] knitr\_1.37 extrafont\_0.17 fastmap\_1.1.0   
## [25] curl\_4.3.2 parallel\_4.1.2 fansi\_1.0.2   
## [28] Rttf2pt1\_1.3.10 highr\_0.9 Rcpp\_1.0.8   
## [31] scales\_1.1.1 jsonlite\_1.8.0 digest\_0.6.29   
## [34] stringi\_1.7.6 dplyr\_1.0.8 grid\_4.1.2   
## [37] cli\_3.2.0 tools\_4.1.2 bitops\_1.0-7   
## [40] magrittr\_2.0.2 RCurl\_1.98-1.6 tibble\_3.1.6   
## [43] extrafontdb\_1.0 crayon\_1.5.0 pkgconfig\_2.0.3   
## [46] ellipsis\_0.3.2 httr\_1.4.2 rmarkdown\_2.11   
## [49] rstudioapi\_0.13 R6\_2.5.1 compiler\_4.1.2