01a-Performing-MSA

March 26, 2025

1 Test 01: Performing Multiple Sequence Alignment

1.0.1 Overview

This notebook demonstrates alignment of a set of 6 sequences from a BAliBASE testcase using MAli v1.31.

The resulting alignment is then scored against a structural reference as this is a helpful way of showing that the output is valid.

Expected runtime: ~30 seconds or less

1.0.2 Context

This notebook is intended to test the following requirements of MAli:

Requirement 1.1 - Given sequences to align, produces a valid solution - independent of quality. - An alignment is performed, with the output shown below and scored against a reference using QScore. Invalid alignments would receive a score of 0.0 or yield no score at all.

Requirement 1.2 - Employs a heuristic to estimate a number of iterations needed to align each set of sequences. - MAli is invoked without specifying how many seconds or iterations to spend on alignment. MAli generally spends ~5 seconds on alignment by default.

Requirement 1.3 - Aligns sets of 6 typical protein sequences within 10 seconds on a university machine. - Every testcase in /data contains 6 biological sequences. As such, this notebook demonstrates alignment of a 6-sequence testcase within 10 seconds.

Requirement 2.1 - Employs a metaheuristic algorithm (such as Genetic Algorithm) to guide the alignment process. - MAli v1.31 uses a mew + lambda evolutionary algorithm to perform multiple sequence alignment, as demonstrated below.

Requirement 3.1 - Can load a set of biological sequences from an appropriate bioinformatics file format. - The input testcases for this demo are in the FASTA file format.

Requirement 3.2 - Can output aligned sets of sequences using an appropriate bioinformatics file format. - The output alignments for this demo are in the FASTA file format.

1.0.3 Installing Prerequisites

[1]:
```
!pip install biopython
```

```
Requirement already satisfied: biopython in
c:\users\pdmoo\appdata\local\programs\python\python310\lib\site-packages (1.85)
Requirement already satisfied: numpy in
c:\users\pdmoo\appdata\local\programs\python\python310\lib\site-packages (from
biopython) (1.26.2)
```

Imports

[2]:
```python
import os
import shutil
import subprocess
import time
from presentation_helper import PresentationHelper
from wrapped_scorer import WrappedScorer
```

MAli v1.31

[3]:
```python
ALIGNER_NAME = "MAli-v1.31"
ALIGNER_PATH = "MAli-v1.31/MAli.exe"
OUTPUT_FOLDER = "data/output"
```

[4]:
```python
# creating empty output folder
if os.path.exists(OUTPUT_FOLDER):
    shutil.rmtree(OUTPUT_FOLDER)
os.makedirs(OUTPUT_FOLDER)
```

Testcase The BB20016 testcase from BAliBASE has been chosen as it contains 6 biological sequences and has a structural reference available.

All testcases from BALIS-2 (subset of BAliBASE used for development) containing 6 sequences have been included in /data

[5]:
```python
TESTCASE_NAME = "BB20016"
INPUT_FILEPATH = f"data/input/{TESTCASE_NAME}"
OUTPUT_FILEPATH = f"data/output/{TESTCASE_NAME}"
```

Viewing Testcase

[6]:
```python
presenter = PresentationHelper()
```

[7]:
```python
presenter.present_unaligned_fasta(INPUT_FILEPATH)
```

```
Displaying Sequences from data/input/BB20016:

>1a7x_A
GVQVETISPGDGRTFPKRGQTCVVHYTGMLEDGKKFDSSRDRNKPFKFMLGKQEVIRGWEEGVAQMSVGQRAKLTISPDY
AYGATGHPGIIPPHATLVFDVELLKLE

>1jvw_A
AASHEERMNNYRKRVGRLFMEQKAAQPDAVKLPSGLVFQRIARGSGKRAPAIDDKCEVHYTGRLRDGTVFDSSRERGKPT
TFRPNEVIKGWTEALQLMREGDRWRLFIPYDLAYGVTGGGGMIPPYSPLEFDVELISIKDGGKGRTAEEVDEILRKAEED
```

```
>1kt0_A
VLKIVTPMIGDKVYVHYKGKLFDSPFVFSLGKGQVIKAWDIGVATMKRGEICHLLCKPEYAYGSAGSLPKIPSNATLFFE
IELLDFKGEDLFEDGGIIRRTKRKGEGYSNPNEGATVEIHLEGRCGGRMFDCRDVAFTVGEGEDHDIPIGIDKALEKMQR
EEQCILYLGPRYGFGEAGKPKFGIEPNAELIYEVTLKSFEKAKESWEMDTKEKLEQAAIVKEKGTVYFKGGKYMQAVIQY
GKIVSWLEMEYGLSEKESKASESFLLAAFLNLAMCYLKLREYTKAVECCDKALGLDSANEKGLYRRGEAQLLMNEFESAK
GDFEKVLEVNAARLQISMCQKKAKEHNERDRRIYANM

>1pbk_
PKYTKSVLKKGDKTNFPKKGDVVHCWYTGTLQDGTVFDTNIQTSAKKKKNAKPLSFKVGVGKVIRGWDEALLTMSKGEKA
RLEIEPEWAYGKKGQPDAKIPPNAKLTFEVELVDID

>1r9h_A
KIDITPKKDGGVLKLIKKEGQGVVKPTTGTTVKVHYVGTLENGTKFDSSRDRGDQFSFNLGRGNVIKGWDLGVATMTKGE
VAEFTIRSDYGYGDAGSPPKIPGGATLIFEVELFEWSA

>1l1p_A
GSHMQATWKEKDGAVEAEDRVTIDFTGSVDGEEFEGGKASDFVLAMGQGRMIPGFEDGIKGHKAGEEFTIDVTFPEEYHA
ENLKGKAAKFAINLKKVEERELPELT
```

Performing Alignment

[8]:
```
ALIGNMENT_COMMAND = f"{ALIGNER_PATH} -input {INPUT_FILEPATH} -output␣
↪{OUTPUT_FILEPATH}"
print(f"CLI command to be run: '{ALIGNMENT_COMMAND}'")
```

```
CLI command to be run: 'MAli-v1.31/MAli.exe -input data/input/BB20016 -output
data/output/BB20016'
```

[9]:
```
start_time = time.perf_counter()
subprocess.run(ALIGNMENT_COMMAND)
end_time = time.perf_counter()

time_in_milliseconds = (end_time - start_time) * 1000
time_in_milliseconds_rounded = round(time_in_milliseconds, 0)
time_in_seconds = time_in_milliseconds_rounded / 1000

print(f"Performed alignment of {TESTCASE_NAME} in {time_in_seconds} seconds")
```

```
Performed alignment of BB20016 in 5.158 seconds
```

Viewing Alignment Produced by MAli

[10]:
```
ALIGNMENT_FILEPATH = OUTPUT_FILEPATH + ".faa"
presenter.present_interleaved_aligned_fasta(ALIGNMENT_FILEPATH)
```

```
Displaying interleaved alignment from 'data/output/BB20016.faa:

1a7x_A          ------------------------------------------------------------
```

```
1jvw_A  ------------------------------------------------------------
1kt0_A  VLKIVTPMIGDKVYVHYKGKLFDSPFVFSLGKGQVIKAWDIGVATMKRGEICHLLCKPEY
1pbk_   ------------------------------------------------------------
1r9h_A  ------------------------------------------------------------
1l1p_A  ------------------------------------------------------------

1a7x_A  ------------------------------------------------------------
1jvw_A  ------------------------------------------------------------
1kt0_A  AYGSAGSLPKIPSNATLFFEIELLDFKGEDLFEDGGIIRRTKRKGEGYSNPNEGATVEIH
1pbk_   ------------------------------------------------------------
1r9h_A  ------------------------------------------------------------
1l1p_A  ------------------------------------------------------------

1a7x_A  ------------------------------------------------------GV---
1jvw_A  ------------------AAS-HEERMNNYRKRVGRLFMEQKAAQPDAVKLPSGLVFQR
1kt0_A  LEGRCGGRMFDCRDVAFTVGEGEDHDIPIGIDKALEKMQREEQCILYLGPRYGFGEA---
1pbk_   ----------------------------------------------------PKYTKSVL-
1r9h_A  ------------------------------------------------------KIDI
1l1p_A  ------------------------------------------------------GS---

1a7x_A  ---Q--VE-T-ISP-GDGRTFPKRGQTCVVH--YTGMLEDGKKFD----SS-R-DRN-KP
1jvw_A  IARGSGKRAPAIDDKCE------------VH--YTGRLRDGTVFD---S-S-RE-RG-KP
1kt0_A  ---G--KP-K-FGI-EPNAELIYEV--TLKSFEK---------------AKESWE----
1pbk_   ---K--K--------GDKTNFPKKG-DV-VHCWYTGTLQDGTVFDTNIQTSAKKKKNAKP
1r9h_A  TPKKDGGVLKLIKKEGQGVVKPTTGTTVKVH--YVGTLENGTKFD----SS-R-DRG-DQ
1l1p_A  ---H--MQAT-WKE-KDGAVEAEDRVTID----FTGSV-DGEEFE----GG-K-ASD---

1a7x_A  FKFMLGKQE-VIRGWEEGVAQMSVGQRAKLTISPDYAYGATGHP-GIIPPHA-TL--VFD
1jvw_A  TTF---RPNEVIKGWTEALQLMREGDRWRLFIPYDLAYGVTGGG-GMIPPYS-PL--EFD
1kt0_A  ----------------------MDTKEKLE-------------QAAIVKEKGTVYFKGG
1pbk_   LSFKVGVGK-VIRGWDEALLTMSKGEKARLEIEPEWAYGKKGQPDAKIPPNA-KL--TFE
1r9h_A  FSFNLGRGN-VIKGWDLGVATMTKGEVAEFTIRSDYGYGDAGSP-PKIPGGA-TL--IFE
1l1p_A  FVLAMGQGR-MIPGFEDGIKGHKAGEEFTIDVTFPEEYHAENLK-GKAAKFAINLKKVEE

1a7x_A  VELLKLE-----------------------------------------------------
1jvw_A  VELISIKDGGKGRTAEEVDEILRKAEED--------------------------------
1kt0_A  KYMQAVIQYGKIVSWLEMEYGLSEKESKASESFLLAAFLNLAMCYLKLREYTKAVECCDK
1pbk_   VELVDID-----------------------------------------------------
1r9h_A  VELFEWSA-----------------------------------------------------
1l1p_A  RELPELT------------------------------------------------------

1a7x_A  ------------------------------------------------------------
1jvw_A  ------------------------------------------------------------
1kt0_A  ALGLDSANEKGLYRRGEAQLLMNEFESAKGDFEKVLEVNAARLQISMCQKKAKEHNERDR
1pbk_   ------------------------------------------------------------
1r9h_A  ------------------------------------------------------------
1l1p_A  ------------------------------------------------------------
```

```
1a7x_A          ------
1jvw_A          ------
1kt0_A          RIYANM
1pbk_           ------
1r9h_A          ------
1l1p_A          ------
```

Viewing Structural Reference Alignment

[11]:
```
REFERENCE_FILEPATH = f"data/ref/{TESTCASE_NAME}"
presenter.present_interleaved_aligned_fasta(REFERENCE_FILEPATH)
```

```
Displaying interleaved alignment from 'data/ref/BB20016:

1a7x_A          …
1jvw_A          …
1kt0_A          vlkivtpmigdkvyvhykgklfdspfvfslgkgqvikawdigvatmkrgeichllckpey
1pbk_           …
1r9h_A          …
1l1p_A          …

1a7x_A          …gvQVETISPGdgrtFPKRGQTCVVH
1jvw_A          .aasheermnnyrkrvgrlfmeqkaaqpdavklpsglVFQRIARGsgkrAPAIDDKCEVH
1kt0_A          aygsagslpkipsnatlffeielldfkgedlfedggiIRRTKRKGegysNPNEGATVEIH
1pbk_           …pkyTKSVLKKGdktnFPKKGDVVHCW
1r9h_A          …kiditpkkdggvLKLIKKEGqgvvKPTTGTTVKVH
1l1p_A          …gsHMQATWKEkd.gAVEAEDRVTID

1a7x_A          YTGMLEdGKKFDSSrd…rnkPFKFMLGK..qevir.GWEEGvAQMSVGQRAKLT
1jvw_A          YTGRLRdGTVFDSSre…rgkPTTFRPNE…vik.GWTEAlQLMREGDRWRLF
1kt0_A          LEGRCG.GRMFDCR…DVAFTVGEgedhdipiGIDKAlEKMQREEQCILY
1pbk_           YTGTLQdGTVFDTNiqtsakkknakPLSFKVGV..gkvir.GWDEAlLTMSKGEKARLE
1r9h_A          YVGTLEnGTKFDSSrd…rgdQFSFNLGR..gnvik.GWDLGvATMTKGEVAEFT
1l1p_A          FTGSVD.GEEFEGG…kasDFVLAMGQ..grmip.GFEDGiKGHKAGEEFTID

1a7x_A          ISPDYAYGAT.ghpgiIPPHATLVFDVELLKLE…
1jvw_A          IPYDLAYGVT.ggggmIPPYSPLEFDVELISIKdggkgrtaeevdeilrkaeed…
1kt0_A          LGPRYGFGEAgkpkfgIEPNAELIYEVTLKSFEkakeswemdtkekleqaaivkekgtvy
1pbk_           IEPEWAYGKKgqpdakIPPNAKLTFEVELVDID…
1r9h_A          IRSDYGYGDA.gsppkIPGGATLIFEVELFEWSa…
1l1p_A          VTFPEEYHAE…NLKGKAAKFAINLKKVEerelpelt…

1a7x_A          …
1jvw_A          …
1kt0_A          fkggkymqaviqygkivswlemeyglsekeskasesfllaaflnlamcylklreytkave
1pbk_           …
1r9h_A          …
1l1p_A          …
```

```
1a7x_A          …
1jvw_A          …
1kt0_A          ccdkalgldsanekglyrrgeaqllmnefesakgdfekvlevnaarlqismcqkkakehn
1pbk_           …
1r9h_A          …
1l1p_A          …

1a7x_A          …
1jvw_A          …
1kt0_A          erdrriyanm
1pbk_           …
1r9h_A          …
1l1p_A          …
```

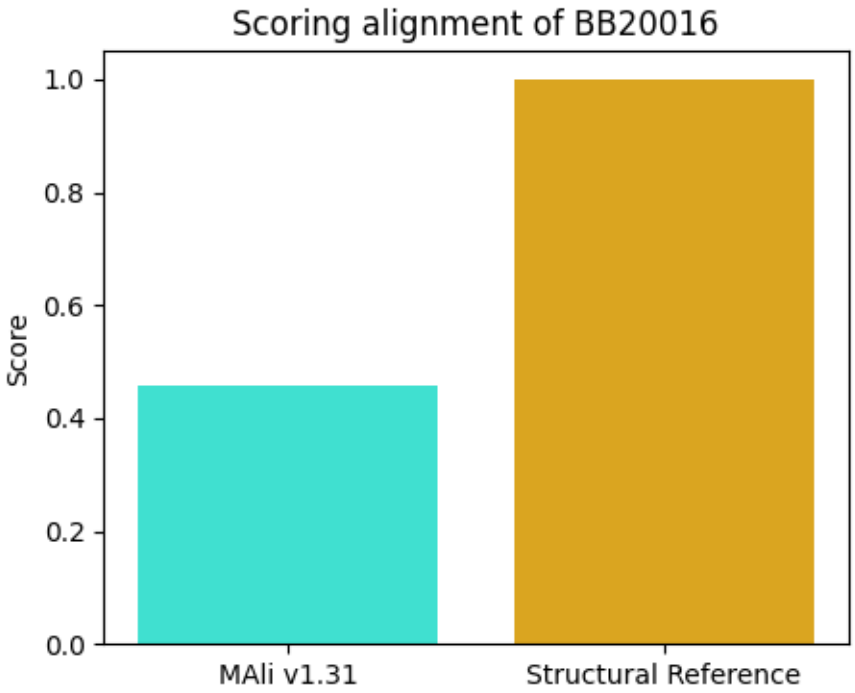
Scoring Alignment Against Reference

[12]:
```
SCORER_PATH = "QScore/qscore.exe"
scorer = WrappedScorer(SCORER_PATH)

score = scorer.score_testcase(ALIGNMENT_FILEPATH, REFERENCE_FILEPATH)
print(score)
```

```
0.459
```

[13]:
```
presenter.present_score(TESTCASE_NAME, score)
```



[ ]: