rapport_ali

June 18, 2024

[ ]:

```
import MSA_entropy_plot as ali
import exon_changes as ex
```

1 ALGORITHM

1.0.1 Function `align_sequences_to_msas`

Input: - `input_file` (str): Path to the file containing the sequences in FASTA format to be analyzed. - `msa1_path` (str): Path to the file containing the first Multiple Sequence Alignment (MSA) in FASTA format. - `msa2_path` (str): Path to the file containing the second MSA in FASTA format. - `output_dir` (str): Path to the directory where the results will be saved. - `iterations` (int): Number of iterations for aligning the sequences.

Output: - No direct return; the modified MSAs and undecided sequences are written to files in `output_dir`.

Role: This function loads two MSAs and a set of sequences. It calculates important positions between the MSAs based on Kullback-Leibler divergence, constructs presence matrices for each MSA, assigns scores to sequences based on these matrices, and distributes the sequences between the two MSAs or marks them as undecided. These steps are repeated for the specified number of iterations, and the results are recorded in files.

1.0.2 Function `find_important_positions_with_weights`

Input: - `msa1` (MultipleSeqAlignment): First multiple sequence alignment. - `msa2` (MultipleSeqAlignment): Second multiple sequence alignment.

Output: - `important_positions` (dict): Dictionary of important positions with weights (average of two asymmetric entropies)

Role: Calculates and returns important positions between two MSAs using the Kullack-Leibler divergence for each position. Positions with high divergences are considered important and receive a weight, influencing the analysis of sequences when compared to these MSAs.

1.0.3 Function `calculate_presence_matrix`

Input: - `alignment` (MultipleSeqAlignment): An MSA from which to calculate the matrix. - `position_weights` (dict): Dictionary of weights for important positions in the MSA.

Output: - `presence_matrix` (np.array): Presence matrix of amino acids in the MSA, weighted by important positions.

Role: This function constructs a matrix describing the relative frequency of each amino acid at each position in the MSA, adjusted by the weights of the important positions. This matrix is essential for evaluating how much amino acids are conserved or vary across sequences.

1.0.4 Function `calculate_sequence_score`

Input: - `sequence` (Seq): Sequence to evaluate. - `presence_matrix` (np.array): Presence matrix from `calculate_presence_matrix`. - `amino_acids` (str): String of considered amino acids.

Output: - `score` (float): Score of the sequence based on its concordance with the presence matrix.

Role: Evaluates an individual sequence by calculating a score based on its correspondence with the presence matrix. This score is used to determine whether the sequence better matches the first or second MSA.

1.0.5 Function `calculate_kl_divergence`

Input: - `p` (list): Probability distribution of the first sequence. - `q` (a list): Probability distribution of the second sequence.

Output: - `kl_divergence` (float): Kullback-Leibler divergence value of `p` relative to `q`.

Role: Uses the function `rel_entr` to calculate the Kullback-Leibler divergence between two probability distributions. `rel_entr` (source: https://docs.scipy.org/doc/scipy/reference/generated/scipy.special.rel_entr.html): The function rel_entr comes from the scipy.special library and is used to calculate the relative entropy or Kullback-Leibler divergence between two probability distributions.

1.0.6 Function `calculate_aa_distribution`

Input: - `column` (list): List of amino acids at a certain position in the MSA.

Output: - `distribution` (list): Normalized distribution of the frequencies of each amino acid.

Role: Calculates the distribution of amino acid frequencies at a given position. This distribution is used to compute the Kullback-Leibler divergence between the columns of two MSAs by the function `calculate_kl_divergence`.

2 Understanding the Kullback-Leibler Divergence and Its Relevance in Weighting Factors

The Kullback-Leibler (KL) divergence is a key statistical tool used to quantify how one probability distribution diverges from another, typically a baseline or reference distribution. Commonly applied across fields such as information theory, statistics, and machine learning, the KL divergence is instrumental in analyzing how the distribution of a dataset, denoted (P), diverges from a comparative distribution, (Q).

Defined mathematically, the divergence from (P) to (Q) is expressed as:

$$D_{KL}(P \parallel Q) = \sum P(i) \cdot \log\left(\frac{Q(i)}{P(i)}\right)$$

Here, (P) and (Q) represent the respective probability distributions, and the summation extends over all conceivable events (i) within these distributions. The formula highlights several essential characteristics of the KL divergence:

- **Asymmetry in Measurement**: The divergence

  $$D_{KL}(P \parallel Q)$$

  is not the same as

  $$D_{KL}(Q \parallel P)$$

  . This reflects the asymmetrical "cost" of deviation, indicating that transitioning from (P) to (Q) incurs a different "cost" than moving from (Q) to (P).

- **Zero Divergence Interpretation**: When (P) perfectly matches (Q), the KL divergence reaches zero, indicating no discrepancy between the distributions. Conversely, larger values signify greater deviations between (P) and (Q), highlighting substantial differences.

- **Non-negative Values**: The value of

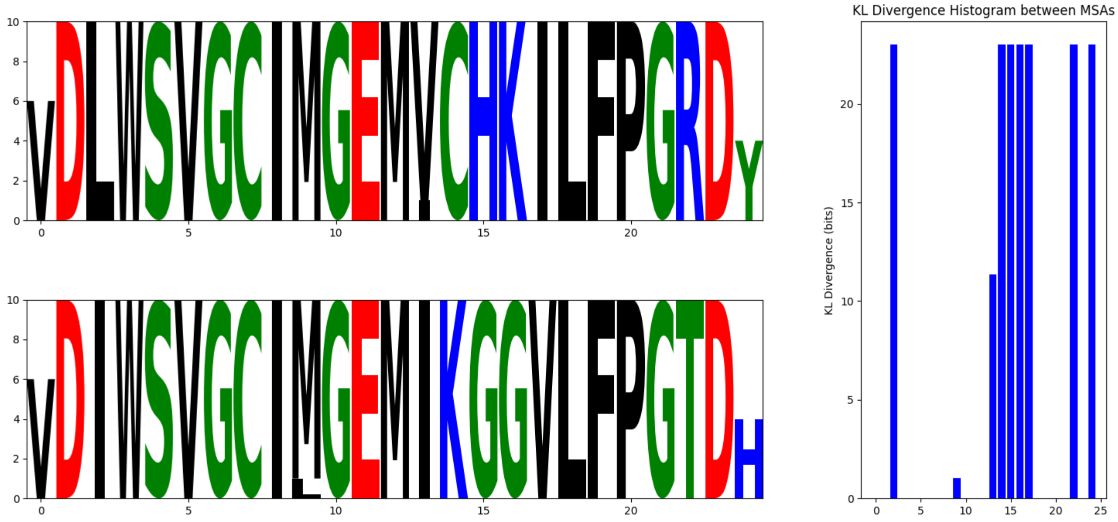
  $$D_{KL}(P \parallel Q)$$

  is always non-negative ((D_{KL}(P Q) 0)), achieving zero exclusively when (P) and (Q) are identical.

2.0.1 1/ 12 species for 12 species : x EM v Thoraxe

```
[ ]: can = "DATA/ENSG00000107643/thoraxe/msa/msa_s_exon_17_0.fasta"
     alt =  "DATA/ENSG00000107643/thoraxe/msa/combined_filtered_alt.fasta"

     ali.plot_combined_msa_analysis(can,alt)
```

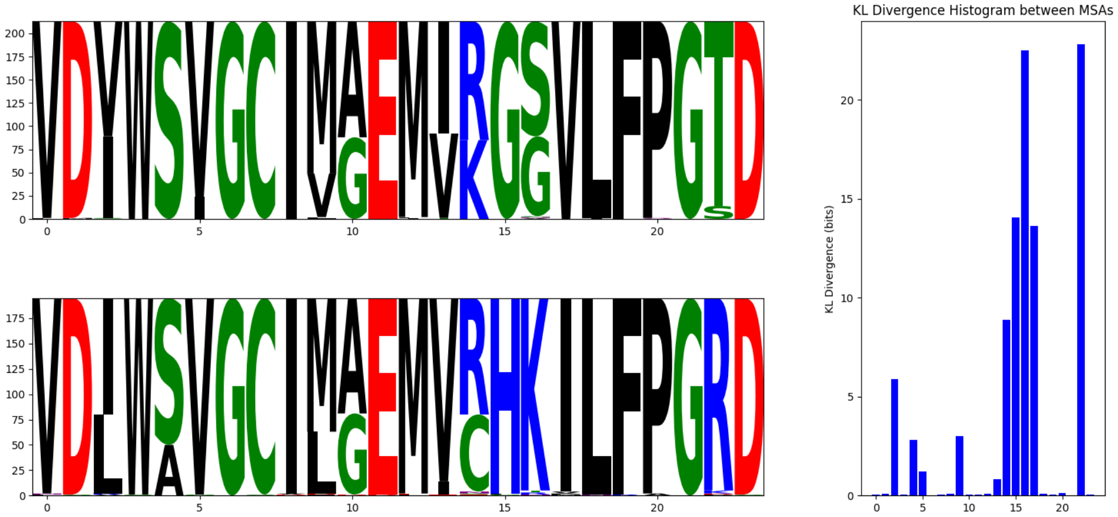
```
There are 10 sequences, all of length 25
There are 10 sequences, all of length 25
```



2.0.2 2/ 100 species for 100 species : x EM v Thoraxe

```
[ ]: msa1_path = "DATA/ENSG00000107643-100species//thoraxe_2/combined_filtered_can.
      ↪fasta"
     msa2_path = "DATA/ENSG00000107643-100species/thoraxe_2/
      ↪combined_filtered_minus_1_alt.fasta"

     print( msa1_path,msa2_path)
     ali.plot_combined_msa_analysis(msa1_path,msa2_path)
```

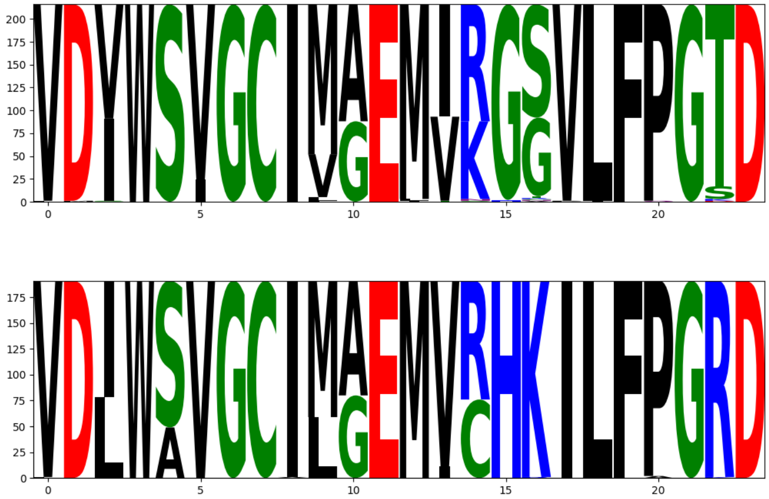
```
DATA/ENSG00000107643-100species//thoraxe_2/combined_filtered_can.fasta
DATA/ENSG00000107643-100species/thoraxe_2/combined_filtered_minus_1_alt.fasta
There are 195 sequences, all of length 24
There are 213 sequences, all of length 24
```

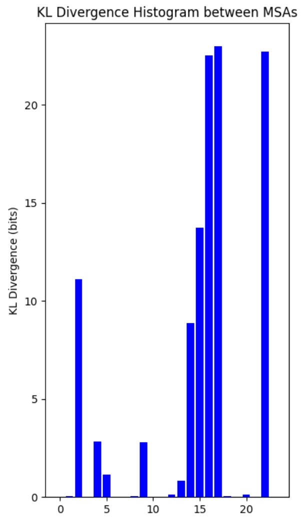


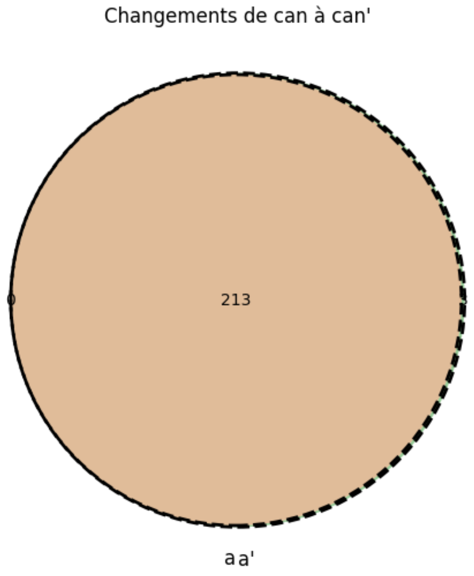
2.0.3 3/ 100 species for 12 species : v EM v Thoraxe

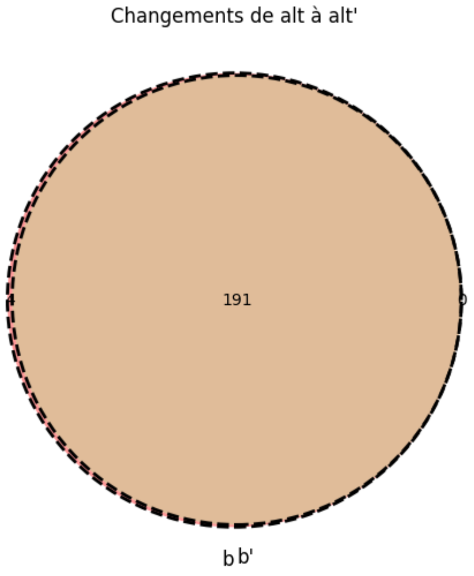
```
[ ]: msa3_path = "DATA/ENSG00000107643-100species/thoraxe_2/aligned_sequences/
      ↪can_ali.fasta"
     msa4_path = "DATA/ENSG00000107643-100species/thoraxe_2/aligned_sequences/
      ↪alt_ali.fasta"



     ali.plot_combined_msa_analysis(msa3_path,msa4_path)
     ex.analyze_changes(msa1_path,msa2_path,msa3_path,msa4_path)
```

```
There are 191 sequences, all of length 24
There are 216 sequences, all of length 24
```









```
[ ]:     Added to Can' Lost from Can Added to Alt'   Lost from Alt  \
    0  WBGene00002187           NaN           NaN  WBGene00002187
    1  WBGene00004056           NaN           NaN  WBGene00004056
    2  WBGene00002188           NaN           NaN  WBGene00004980
    3             NaN           NaN           NaN  WBGene00002188

     Exchanged to Can' Exchanged to Alt'
    0    WBGene00002187               NaN
    1    WBGene00002188               NaN
    2    WBGene00004056               NaN
```

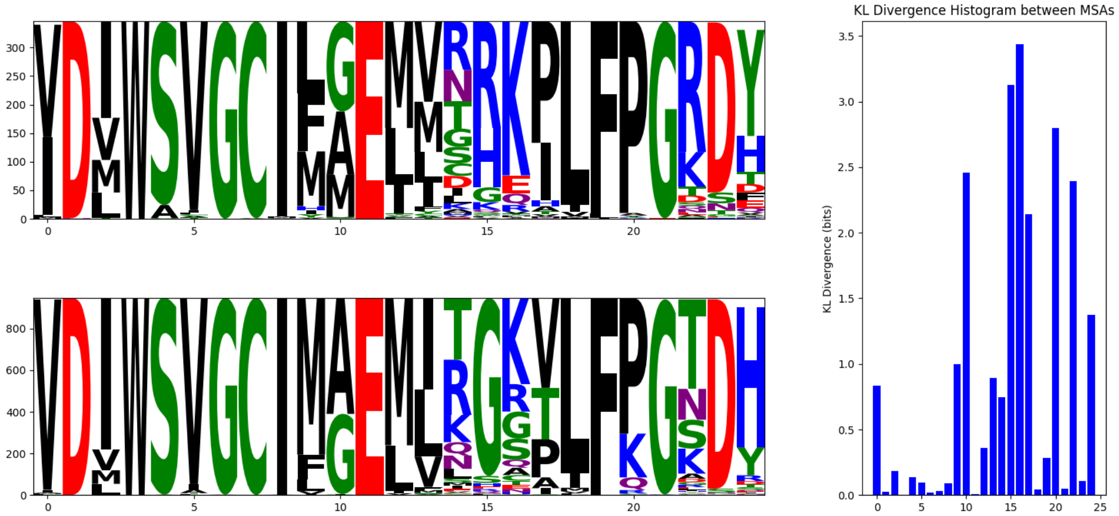
```
3                  NaN                  NaN
```

2.0.4 4/ a3m species for 12 species : v EM v Thoraxe

[ ]:

```
can = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences/can_ali.fasta"
alt = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences/alt_ali.fasta"

ali.plot_combined_msa_analysis(can,alt)
```

```
There are 948 sequences, all of length 25
There are 346 sequences, all of length 25
```

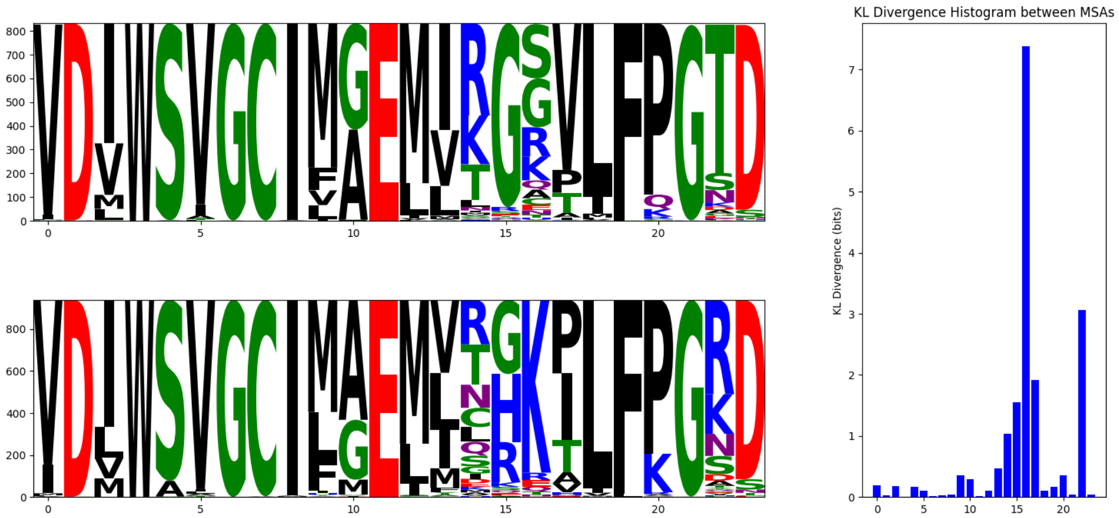


2.0.5 5/ a3m species for 100 species : x EM v Thoraxe

[ ]:

```
can = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences_100/can_ali_noEM.
  ↪fasta"
alt = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences_100/alt_ali_noEM.
  ↪fasta"

ali.plot_combined_msa_analysis(can,alt)
```

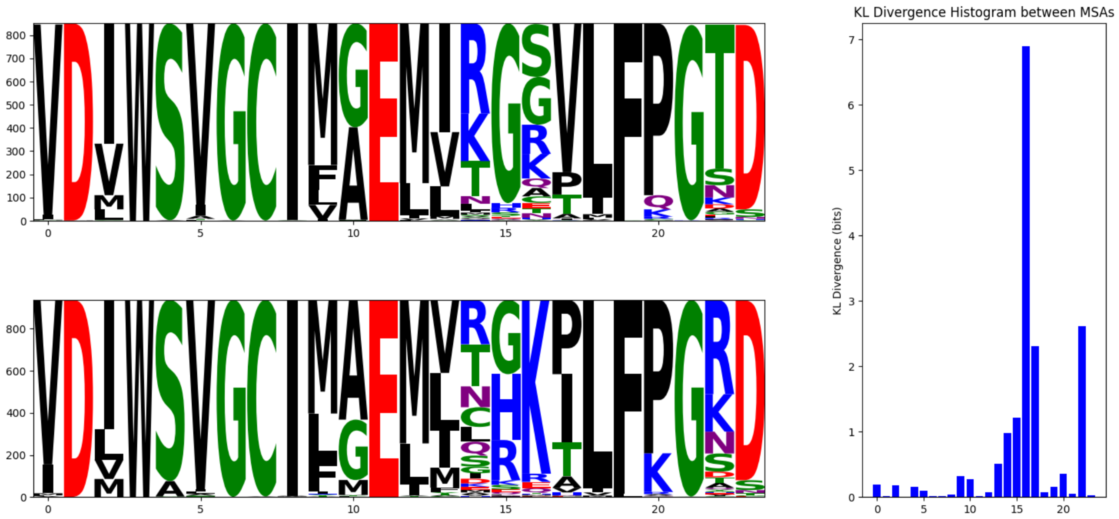
```
There are 940 sequences, all of length 24
There are 832 sequences, all of length 24
```



2.0.6 6/ a3m species for 100 species : x EM x Thoraxe

```
[ ]: can = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences_100_no_thoraxe/
      ↪can_ali_noEM.fasta"
     alt = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences_100_no_thoraxe/
      ↪alt_ali_noEM.fasta"

     ali.plot_combined_msa_analysis(can,alt)
```

```
There are 938 sequences, all of length 24
There are 851 sequences, all of length 24
```

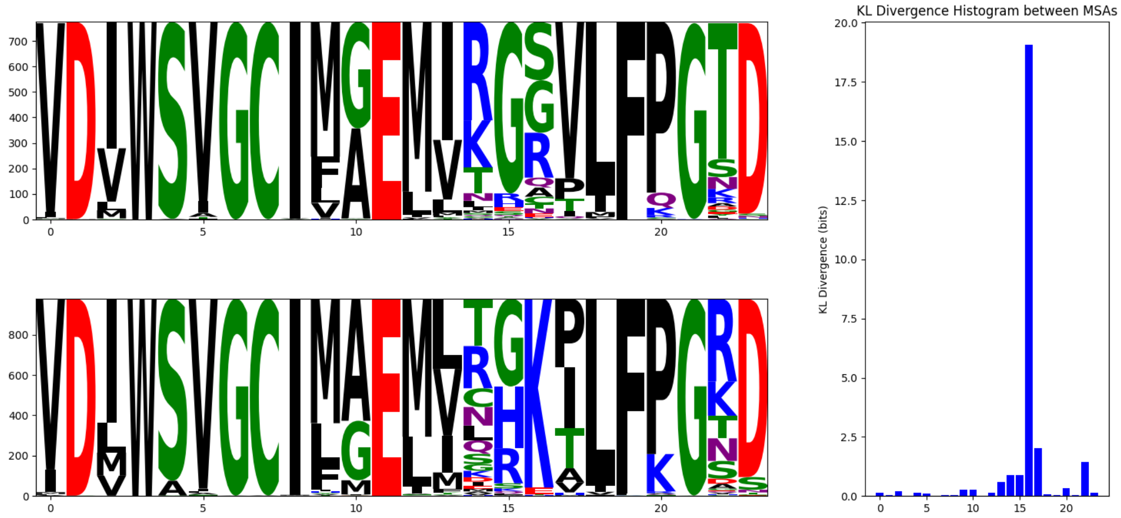


2.0.7 7/ a3m species for 100 species : v EM v Thoraxe

[ ]:

```
can = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences_100/can_ali.fasta"
alt = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences_100/alt_ali.fasta"

ali.plot_combined_msa_analysis(can,alt)
```

```
There are 979 sequences, all of length 24
There are 775 sequences, all of length 24
```

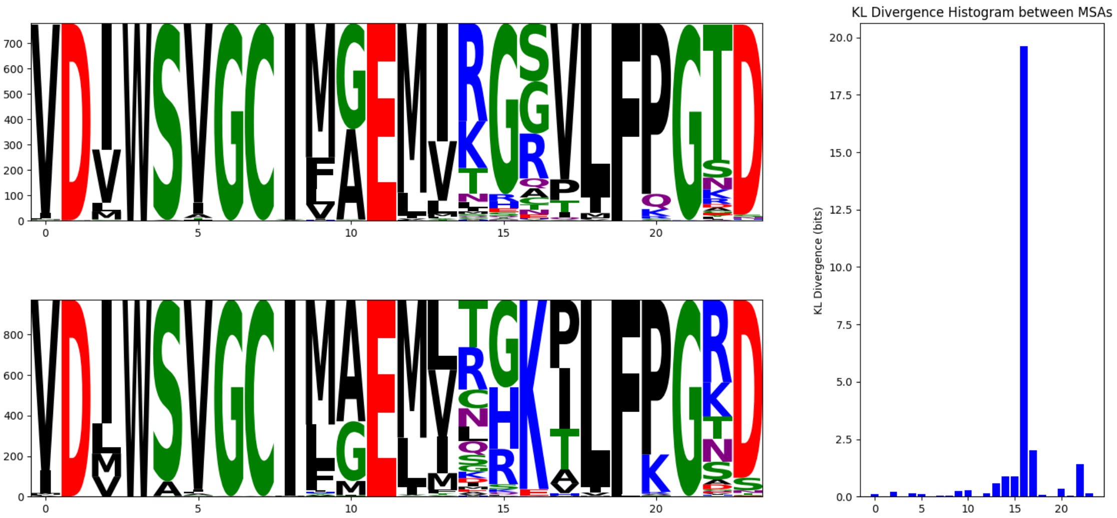


2.0.8 8/ a3m species for 100 species : v EM x Thoraxe

[ ]:

```
can = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences_100_no_thoraxe/
  ↪can_ali.fasta"
alt = "DATA/ENSG00000107643/Analyze_logo/aligned_sequences_100_no_thoraxe/
  ↪alt_ali.fasta"

ali.plot_combined_msa_analysis(can,alt)
```

```
There are 974 sequences, all of length 24
There are 778 sequences, all of length 24
```


VDIWSVGCIMGEMIRGSVLFPGTD
0
100
200
300
400
500
600
700
0
5
10
15
20
VDIWSVGCIMAEMLTGKPLFPGRD
0
200
400
600
800
0
5
10
15
20
KL Divergence Histogram between MSAs
KL Divergence (bits)
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0
5
10
15
20