

# Multiple Sequence Alignment

# Multiple Sequence Alignment Problem

Given

- A set of sequences  $s_1, s_2, \dots, s_k$  (of length  $n$ )
- An objective function

Find:

- an  $\ell$  by  $k$  matrix ( $\ell \geq n$ )
- where row  $i$  contains the characters from sequence  $s_i$  in order with inserted gap characters
- that is optimal under the objective function.

Input

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| A | G | T | P | N | G | N | P |   |
| A | G | P | G | N | P |   |   |   |
| A | G | T | T | P | N | G | N | P |
| C | G | T | P | N | P |   |   |   |
| A | C | G | T | U | N | G | N | P |

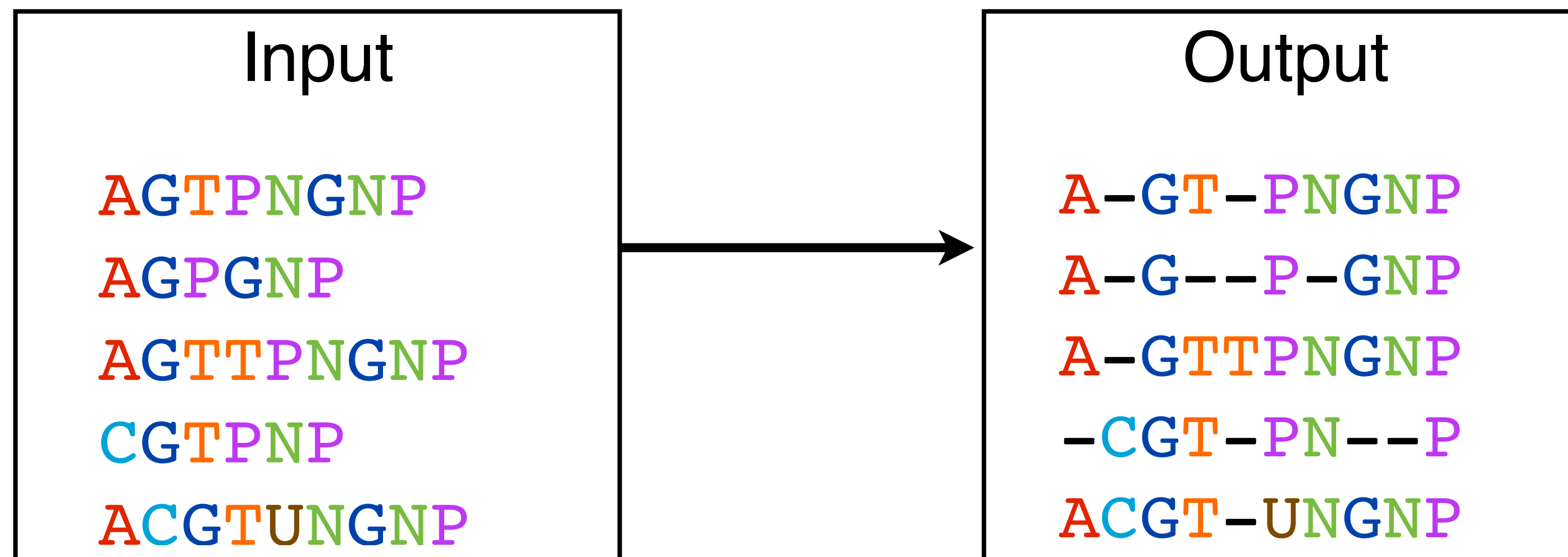
# Multiple Sequence Alignment Problem

Given

- A set of sequences  $s_1, s_2, \dots, s_k$  (of length  $n$ )
- An objective function

Find:

- an  $\ell$  by  $k$  matrix ( $\ell \geq n$ )
- where row  $i$  contains the characters from sequence  $s_i$  in order with inserted gap characters
- that is optimal under the objective function.



# Multiple Sequence Alignment

Whats the objective function:

- most popular -- **Sum-of-Pairs Objective:**
  - given some scoring function for a pairwise alignment  $PairScore(s_1', s_2')$  the score of the multiple alignment is:

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A-GT-PNGNP

$s'_2$  A-G--P-GNP

$s'_3$  A-GTTPNCNP

$s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A-GT-PNGNP

$s'_2$  A-G--P-GNP

$$PairScore(s'_1, s'_2) = 10 \times$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times 5$$



$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times 5 - 3 \times$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times 5 - 3 \times 0$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times 5 - 3 \times 0 - 1 \times$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times 5 - 3 \times 0 - 1 \times 2$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times 5 - 3 \times 0 - 1 \times 2 - 4 \times 2$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times 5 - 3 \times 0 - 1 \times 2 - 4 \times 2 = 40$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT PNGNP

$s'_2$  A G- P-GNP

$$PairScore(s'_1, s'_2) = 10 \times 5 - 3 \times 0 - 1 \times 2 - 4 \times 2 = 40$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A-GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times$$

$s'_3$  A-GTTPNCNP



$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times$$

$s'_3$  A GTTPNCNP

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times 7$$

$s'_3$  A GTTPNCNP

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times 7 - 3 \times$$

$s'_3$  A GTTPNCNP

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times 7 - 3 \times 1$$

$s'_3$  A GTTPNCNP

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times 7 - 3 \times 1 - 1 \times$$

$s'_3$  A GTTPNCNP

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times 7 - 3 \times 1 - 1 \times 1$$

$s'_3$  A GTTPNCNP

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times 7 - 3 \times 1 - 1 \times 1 - 4 \times 1 = 40$$

$s'_3$  A GTTPNCNP

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times 7 - 3 \times 1 - 1 \times 1 - 4 \times 1 = 62$$

$s'_3$  A GTTPNCNP



$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A GT-PNGNP

$$PairScore(s'_1, s'_3) = 10 \times 7 - 3 \times 1 - 1 \times 1 - 4 \times 1 = 62$$

$s'_3$  A GTTPNCNP

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A-GT-PNGNP

$$PairScore(s'_1, s'_4) = 10 \times$$

$s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A-GT PNGNP

$$PairScore(s'_1, s'_4) = 10 \times$$

$s'_4$  -CGT PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A-GT PNGNP

$$PairScore(s'_1, s'_4) = 10 \times 5$$

$s'_4$  -CGT PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40                      62

$s'_1$  **A-GT** **PNGNP**

$$PairScore(s'_1, s'_4) = 10 \times 5 - 3 \times$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A-GT PNGNP

$$PairScore(s'_1, s'_4) = 10 \times 5 - 3 \times 0$$

$s'_4$  -CGT PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A-GT PNGNP

$$PairScore(s'_1, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times$$

$s'_4$  -CGT PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A-GT PNGNP

$$PairScore(s'_1, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times 4$$

$s'_4$  -CGT PN--P



$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A-GT PNGNP

$$PairScore(s'_1, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times 4 - 4 \times 3$$

$s'_4$  -CGT PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

$s'_1$  A-GT PNGNP

$$PairScore(s'_1, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times 4 - 4 \times 3 = 34$$

$s'_4$  -CGT PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_1$  A-GT PNGNP

$$PairScore(s'_1, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times 4 - 4 \times 3 = 34$$

$s'_4$  -CGT PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  **A**-**G**---**P**-**G****N****P**  
 $s'_3$  **A**-**G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  **A** **G**---**P**-**G****N****P**  
 $s'_3$  **A** **G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  **A** **G**---**P**-**G****N****P**  
 $s'_3$  **A** **G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times 5$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  **A** **G**---**P**-**G****N****P**  
 $s'_3$  **A** **G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times 5 - 3 \times$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  **A** **G**---**P**-**G****N****P**  
 $s'_3$  **A** **G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times 5 - 3 \times 1$$



$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  **A** **G**---**P**-**G****N****P**  
 $s'_3$  **A** **G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times 5 - 3 \times 1 - 1 \times$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  **A** **G**---**P**-**G****N****P**  
 $s'_3$  **A** **G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times 5 - 3 \times 1 - 1 \times 3$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  A G---P-GNP

$s'_3$  A GTTPNCNP

$$PairScore(s'_2, s'_3) = 10 \times 5 - 3 \times 1 - 1 \times 3 - 4 \times 2 = 29$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34

$s'_2$  **A** **G**---**P**-**G****N****P**  
 $s'_3$  **A** **G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times 5 - 3 \times 1 - 1 \times 3 - 4 \times 2 = 36$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36

$s'_2$  **A** **G**---**P**-**G****N****P**  
 $s'_3$  **A** **G****T****T****P****N****C****N****P**

$$PairScore(s'_2, s'_3) = 10 \times 5 - 3 \times 1 - 1 \times 3 - 4 \times 2 = 36$$

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36

$s'_2$  **A-G--P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times$$

$s'_4$  **-CGT-PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times 3$$

$s'_4$  **-CGT** **PN--P**



$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

**40**
**62**
**34**
**36**

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times 3 - 3 \times$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

**40**
**62**
**34**
**36**

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times 3 - 3 \times 0$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times 3 - 3 \times 0 - 1 \times$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times 3 - 3 \times 0 - 1 \times 6$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times 3 - 3 \times 0 - 1 \times 6 - 4 \times 5$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times 3 - 3 \times 0 - 1 \times 6 - 4 \times 5 = 4$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4

$s'_2$  **A-G-** **P-GNP**

$$PairScore(s'_2, s'_4) = 10 \times 3 - 3 \times 0 - 1 \times 6 - 4 \times 5 = 4$$

$s'_4$  **-CGT** **PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4

$$PairScore(s'_3, s'_4) = 10 \times$$

$s'_3$  A-GTTPNCNP

$s'_4$  -CGT-PN--P



$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4

$$PairScore(s'_3, s'_4) = 10 \times 5$$

$s'_3$  A-GTTPNCNP  
 $s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4

$$PairScore(s'_3, s'_4) = 10 \times 5 - 3 \times$$

$s'_3$  A-GTTPNCNP

$s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

**40**
**62**
**34**
**36**
**4**

$$PairScore(s'_3, s'_4) = 10 \times 5 - 3 \times 0$$

$s'_3$  **A-GTTPNCNP**  
 $s'_4$  **-CGT-PN--P**

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4

$$PairScore(s'_3, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times$$

$s'_3$  A-GTTPNCNP  
 $s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4

$$PairScore(s'_3, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times 5$$

$s'_3$  A-GTTPNCNP

$s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4

$$PairScore(s'_3, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times 5 - 4 \times 4$$

$s'_3$  A-GTTPNCNP

$s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4

$$PairScore(s'_3, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times 5 - 4 \times 4 = 29$$

$s'_3$  A-GTTPNCNP  
 $s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40
62
34
36
4
29

$$PairScore(s'_3, s'_4) = 10 \times 5 - 3 \times 0 - 1 \times 5 - 4 \times 4 = 29$$

$s'_3$  A-GTTPNCNP  
 $s'_4$  -CGT-PN--P



$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4)$$

40

62

34

36

4

29

$s'_1$  A-GT-PNGNP

$s'_2$  A-G---P-GNP

$s'_3$  A-GTTPNCNP

$s'_4$  -CGT-PN--P

$$SPScore(\{s'_1, s'_2, \dots, s'_k\}) := \sum_{1 \leq i < j \leq k} PairScore(s'_i, s'_j)$$

$$PairScore(s'_i, s'_j) := 10\mathbf{mt} - 3\mathbf{ms} - 1\mathbf{id} - 4\mathbf{gp}$$

$$PairScore(s'_1, s'_2) + PairScore(s'_1, s'_3) + PairScore(s'_1, s'_4) + PairScore(s'_2, s'_3) + PairScore(s'_2, s'_4) + PairScore(s'_3, s'_4) = 205$$

40

62

34

36

4

29

$s'_1$  A-GT-PNGNP

$s'_2$  A-G---P-GNP

$s'_3$  A-GTTPNCNP

$s'_4$  -CGT-PN--P

# Finding an optimal MSA

Can we find an optimal multiple sequence alignment?

# Finding an optimal MSA

Can we find an optimal multiple sequence alignment?

- yes! we can use the same dynamic programming methods we had for pairwise alignment

# Finding an optimal MSA

Can we find an optimal multiple sequence alignment?

- yes! we can use the same dynamic programming methods we had for pairwise alignment
- assume there are only 3 sequences, then the recursion is the following:

# Finding an optimal MSA

Can we find an optimal multiple sequence alignment?

- yes! we can use the same dynamic programming methods we had for pairwise alignment
- assume there are only 3 sequences, then the recursion is the following:

$$V[i, j, k] = \max \left\{ \begin{array}{l} V[i-1, j-1, k-1] + \delta(s_1[i], s_2[j]) + \delta(s_2[j], s_3[k]) + \delta(s_1[i], s_3[k]) \\ V[i-1, j-1, k] + \delta(s_1[i], s_2[j]) + \delta(s_2[j], '-') + \delta(s_1[i], '-') \\ V[i-1, j, k-1] + \delta(s_1[i], '-') + \delta(s_2[j], s_3[k]) + \delta(s_1[i], s_3[k]) \\ V[i, j-1, k-1] + \delta('-', s_2[j]) + \delta(s_2[j], s_3[k]) + \delta('-', s_3[k]) \\ V[i-1, j, k] + 2\delta(s_1[i], '-') \\ V[i, j-1, k] + 2\delta(s_2[j], '-') \\ V[i, j, k-1] + 2\delta(s_3[k], '-') \end{array} \right.$$

# Finding an optimal MSA

Can we find an optimal multiple sequence alignment?

- yes! we can use the same dynamic programming methods we had for pairwise alignment
- assume there are only 3 sequences, then the recursion is the following:

$$V[i, j, k] = \max \left\{ \begin{array}{l} V[i-1, j-1, k-1] + \delta(s_1[i], s_2[j]) + \delta(s_2[j], s_3[k]) + \delta(s_1[i], s_3[k]) \\ V[i-1, j-1, k] + \delta(s_1[i], s_2[j]) + \delta(s_2[j], '-') + \delta(s_1[i], '-') \\ V[i-1, j, k-1] + \delta(s_1[i], '-') + \delta(s_2[j], s_3[k]) + \delta(s_1[i], s_3[k]) \\ V[i, j-1, k-1] + \delta('-', s_2[j]) + \delta(s_2[j], s_3[k]) + \delta('-', s_3[k]) \\ V[i-1, j, k] + 2\delta(s_1[i], '-') \\ V[i, j-1, k] + 2\delta(s_2[j], '-') \\ V[i, j, k-1] + 2\delta(s_3[k], '-') \end{array} \right.$$

What happens with 4 sequences? How many clauses are in the max? How big is  $V$ ?

# Finding an optimal MSA

Can we find an optimal multiple sequence alignment?

- yes! we can use the same dynamic programming methods we had for pairwise alignment
- assume there are only 3 sequences, then the recursion is the following:

$$V[i, j, k] = \max \begin{cases} V[i-1, j-1, k-1] & +\delta(s_1[i], s_2[j]) + \delta(s_2[j], s_3[k]) + \delta(s_1[i], s_3[k]) \\ V[i-1, j-1, k] & +\delta(s_1[i], s_2[j]) + \delta(s_2[j], '-' ) + \delta(s_1[i], '-' ) \\ V[i-1, j, k-1] & +\delta(s_1[i], '-' ) + \delta(s_2[j], s_3[k]) + \delta(s_1[i], s_3[k]) \\ V[i, j-1, k-1] & +\delta('-', s_2[j]) + \delta(s_2[j], s_3[k]) + \delta(s_1[i], s_3[k]) \\ V[i-1, j, k] & +2\delta(s_1[i], '-' ) \\ V[i, j-1, k] & +2\delta(s_2[j], '-' ) \\ V[i, j, k-1] & +2\delta(s_3[k], '-' ) \end{cases}$$

**$O(k^2 2^k n^k)$ -time!!**

What happens with 4 sequences? How many clauses are in the max? How big is  $V$ ?

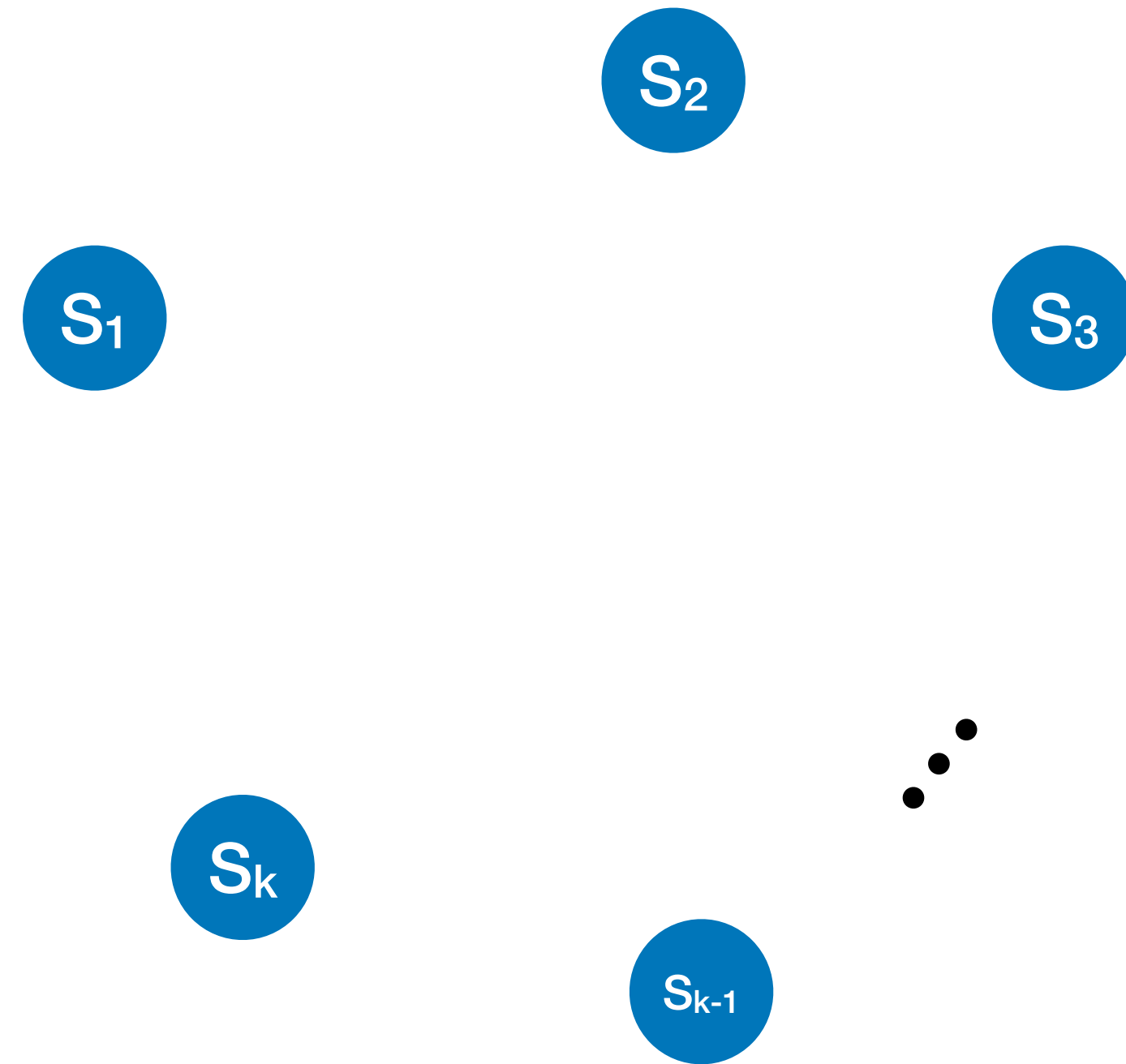


# Finding an optimal MSA

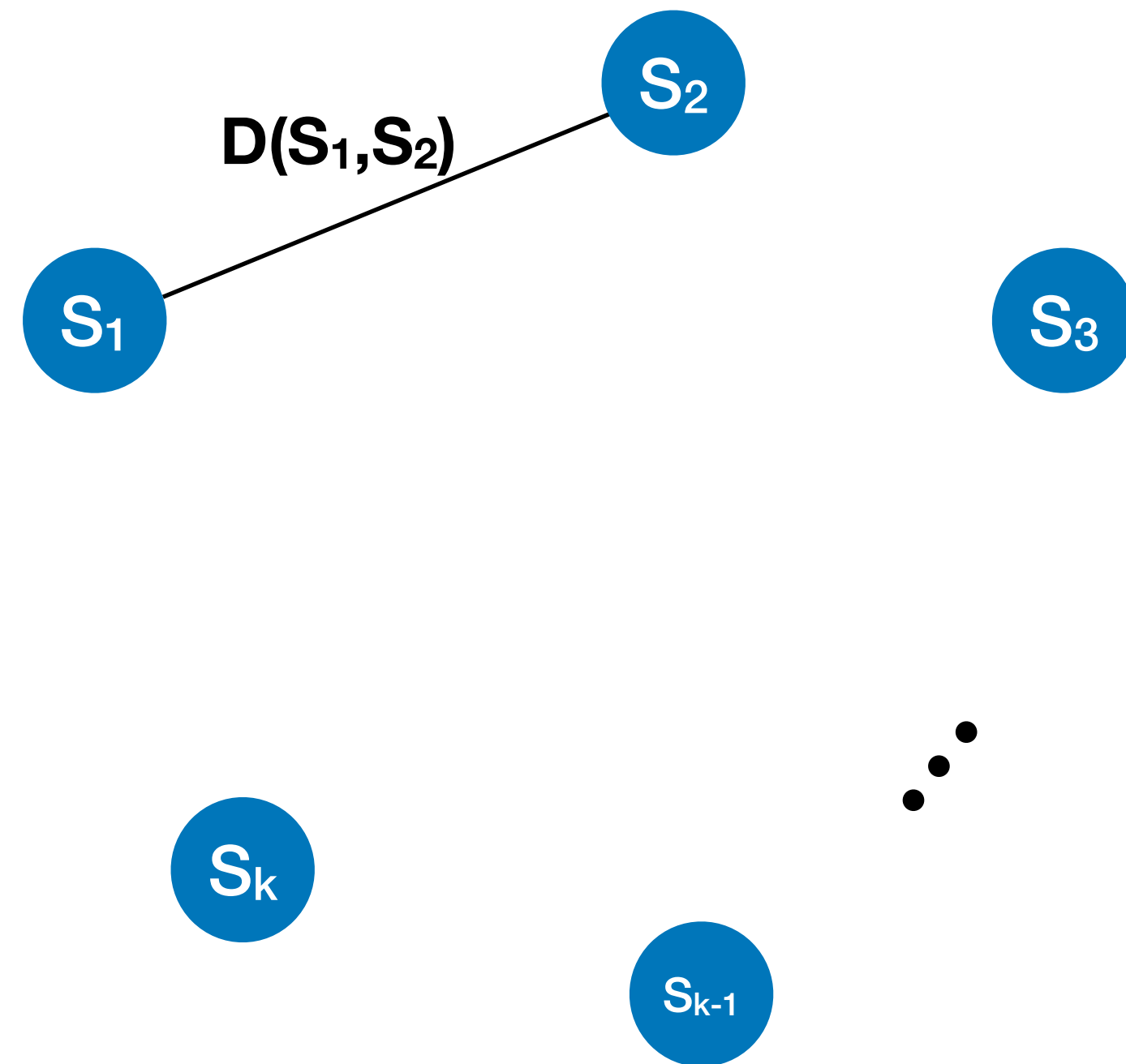
Kececioglu (1993)\* showed that the problem of finding multiple sequence alignments under any standard formulation is **NP-Hard!**

\*This was somewhat concurrent with Wang and Jiang (1994) both for constant gap costs, but later work by Kececioglu and Sarrett showed that this is also true for linear (affine) gap costs.

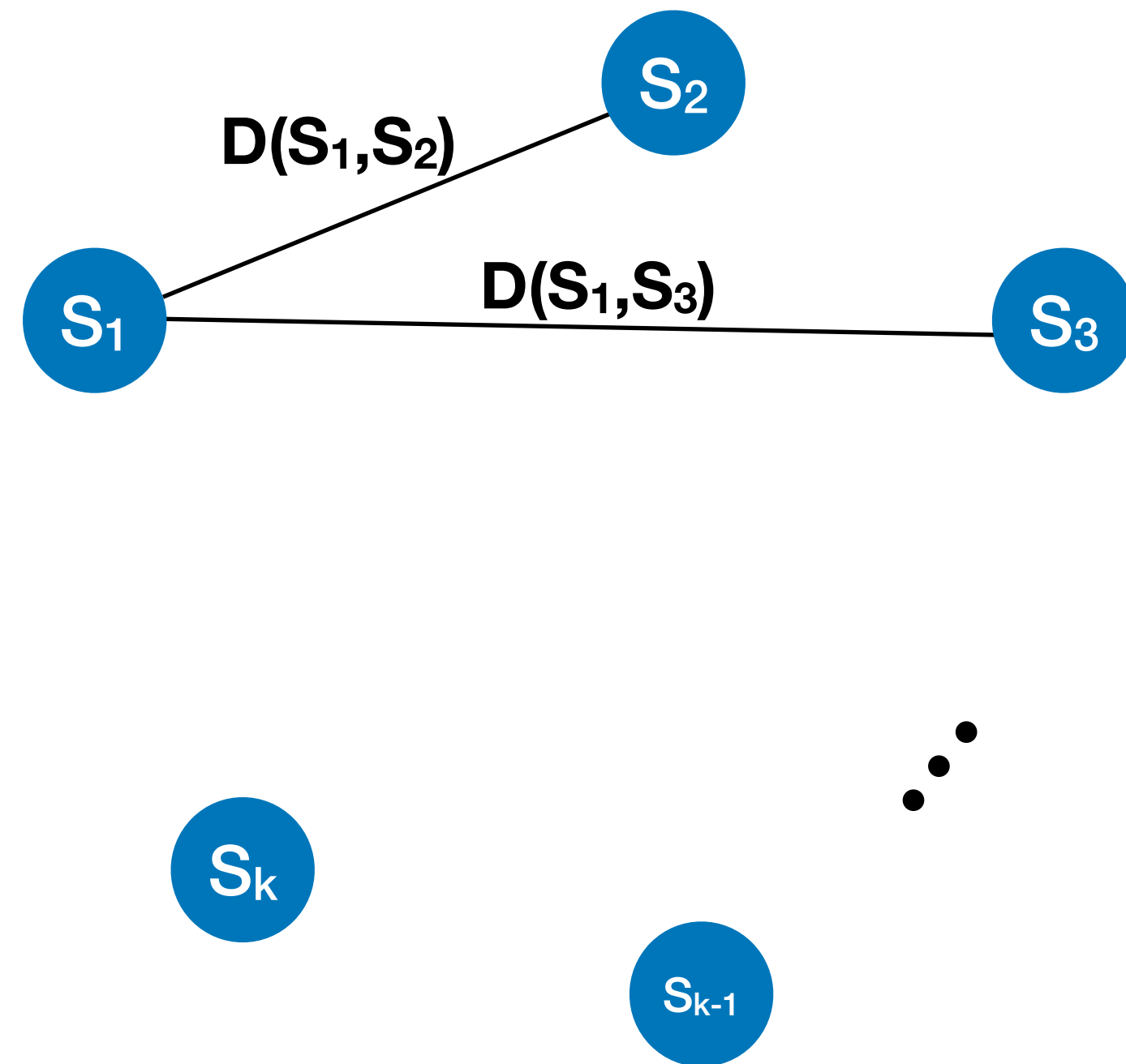
# The Center Star Method



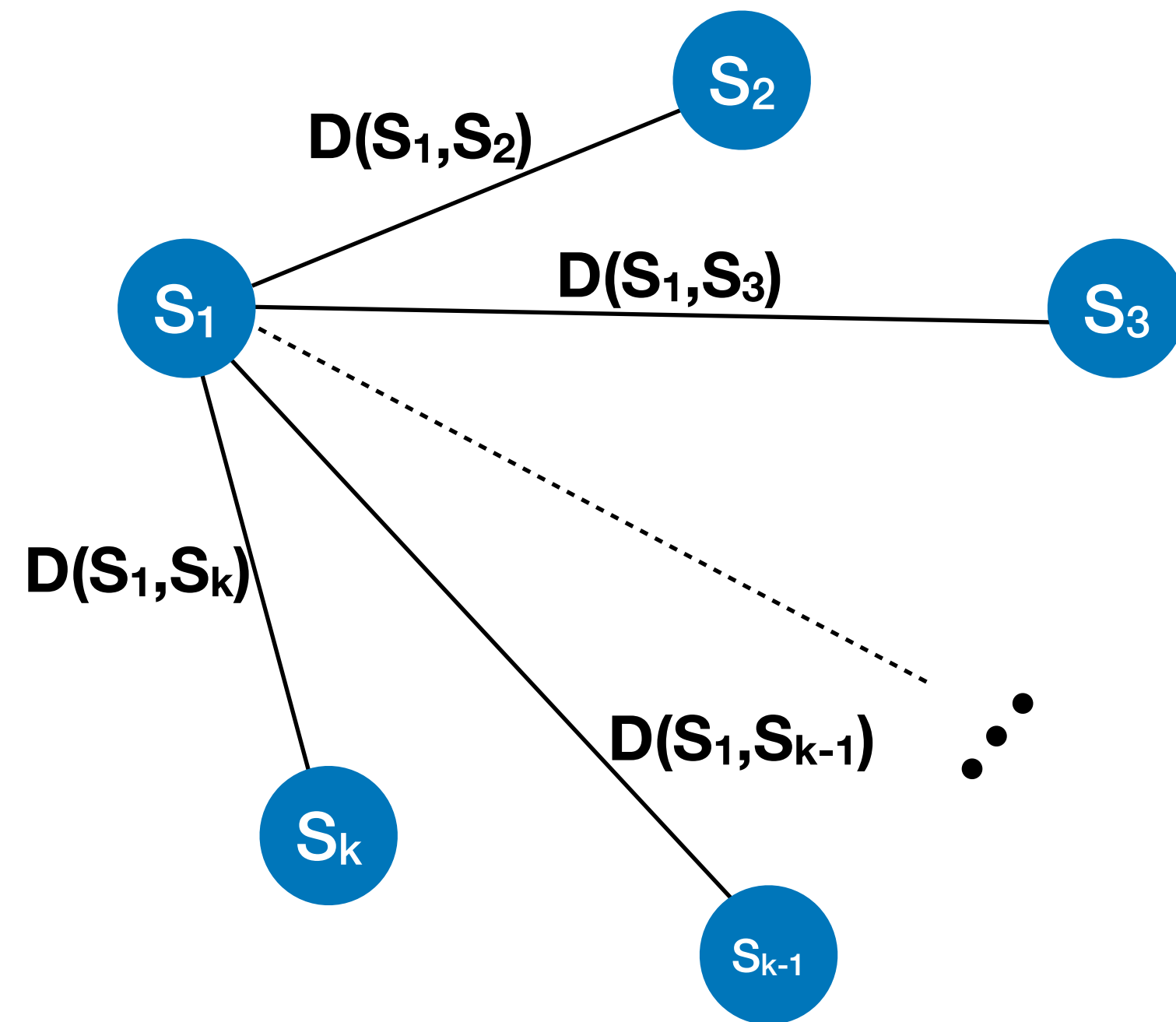
# The Center Star Method



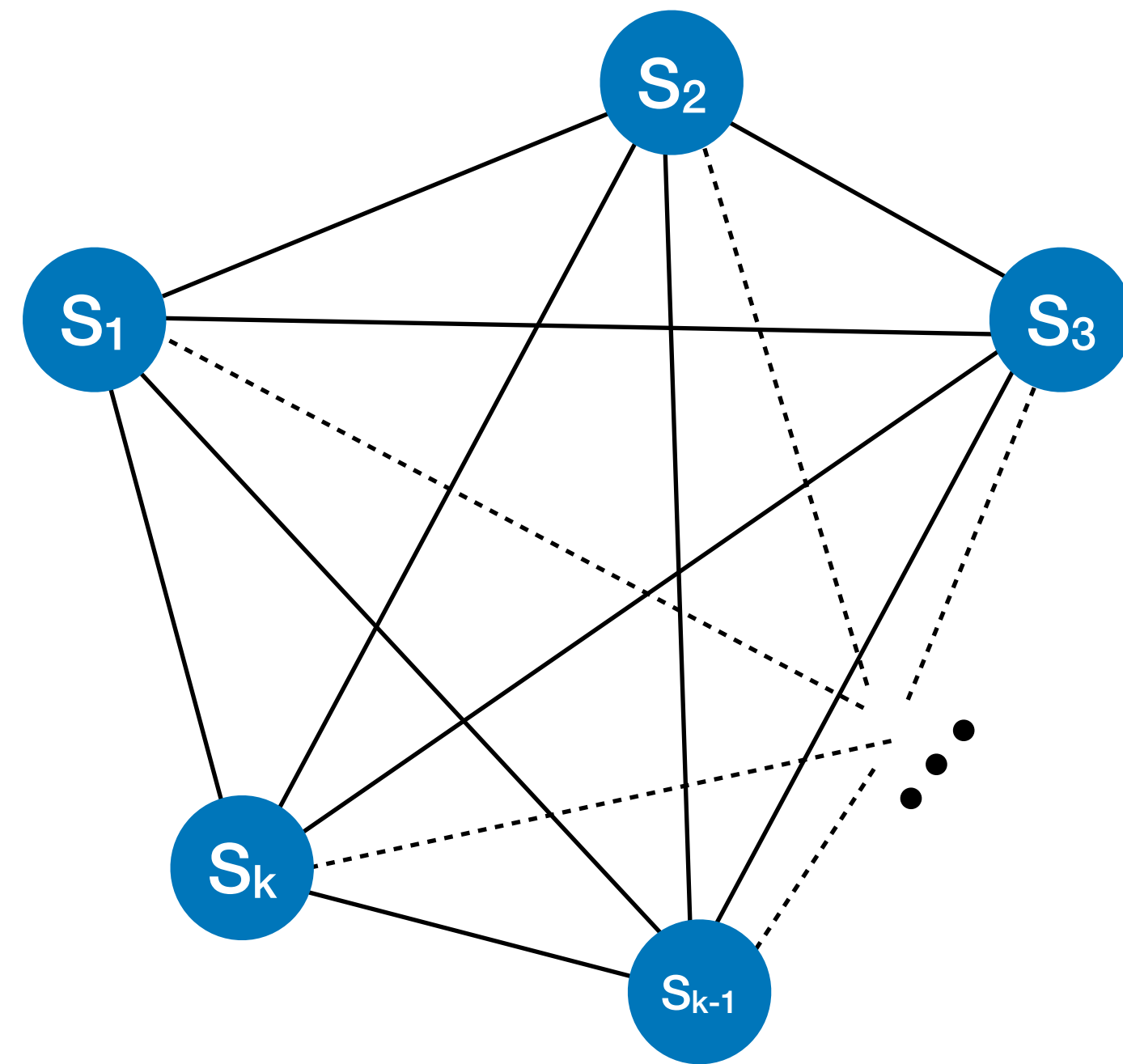
# The Center Star Method



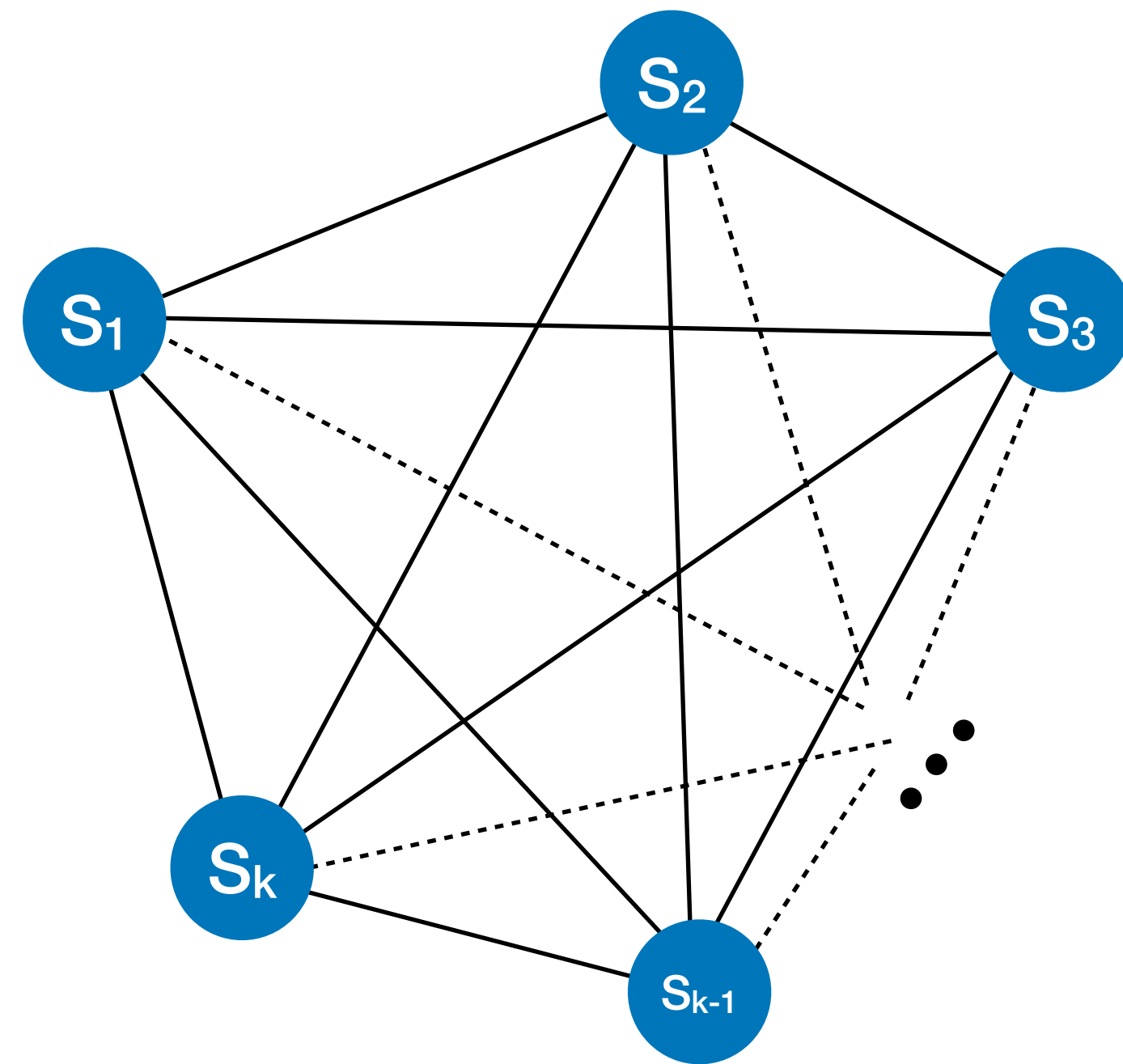
# The Center Star Method



# The Center Star Method

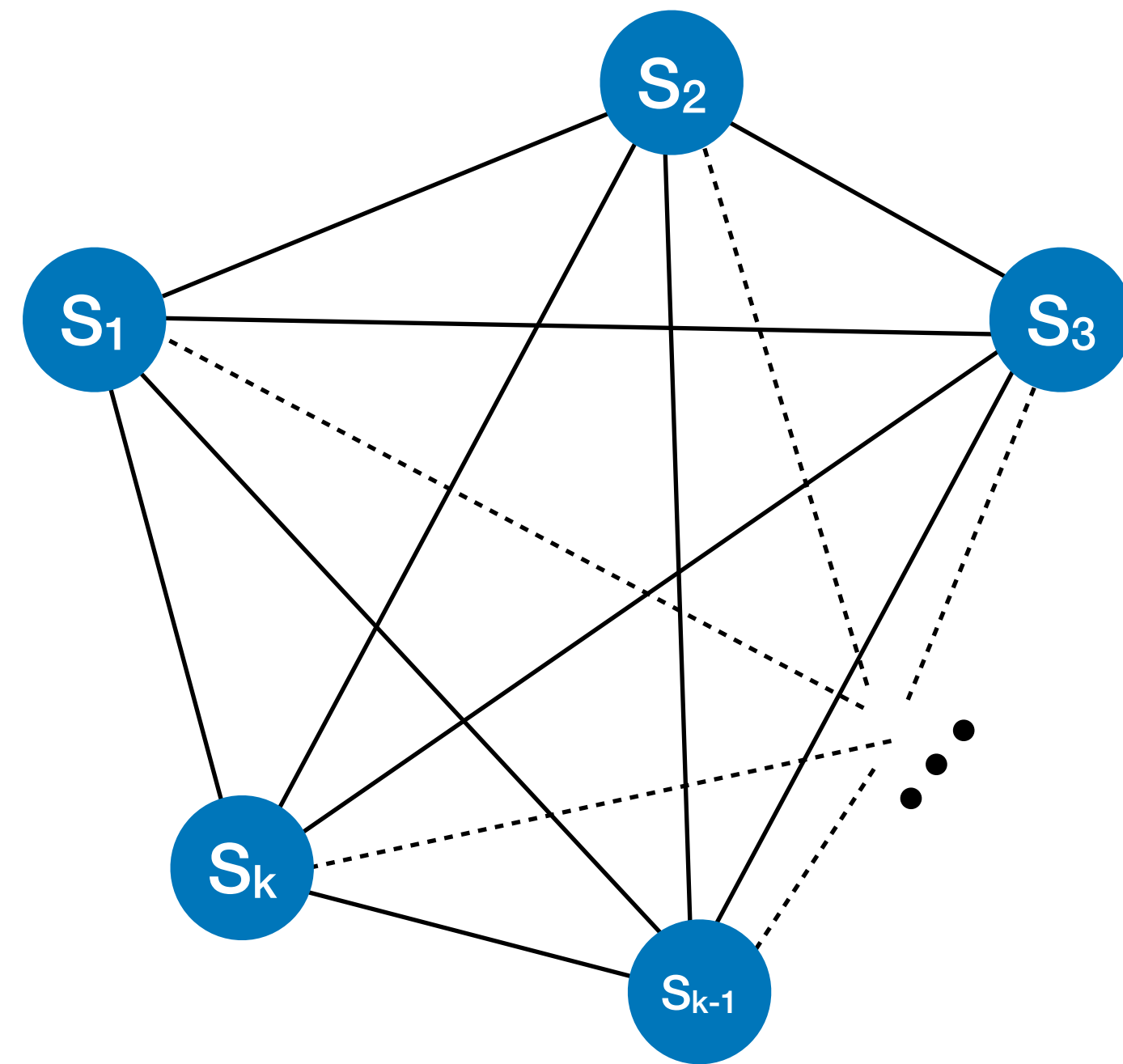


# The Center Star Method



$$S_c = \mathbf{arg} \min_{1 \leq i \leq k} \left\{ \sum_{1 \leq j \leq k} D(S_i, S_j) \right\}$$

# The Center Star Method



$$S_c = \mathbf{arg} \min_{1 \leq i \leq k} \left\{ \sum_{1 \leq j \leq k} D(S_i, S_j) \right\}$$

The final step is to build an alignment so that all of the alignments between  $S_c$  and  $S_i$  are satisfied.



# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG



# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_3$ : GATG-TGCAG

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_3$ : GATG-TGCAG



# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG  
 $S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG  
 $S_4$ : CC-GCTAGCAG

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG

$S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG  
 $S_2$ : GATG-T-GCCG  
 $S_3$ : GATG-T-GCAG  
 $S_4$ : CC-GCTAGCAG



# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG

$S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG

$S_5$ : CCTG-TAG--G

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG  
 $S_2$ : GATG-T-GCCG  
 $S_3$ : GATG-T-GCAG  
 $S_4$ : CC-GCTAGCAG

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG

$S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG

$S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG

$S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG

$S_5$ : CCTG-TAG--G

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG  
 $S_2$ : GATG-T-GCCG  
 $S_3$ : GATG-T-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG,  
 $S_5$ : CCTG-TAG--G

# The Center Star Method

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATGTGCCG  
 $S_3$ : GATGTGCAG  
 $S_4$ : CCGCTAGCAG  
 $S_5$ : CCTGTAGG

|       | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|-------|-------|-------|-------|-------|-------|
| $S_1$ | 0     | 4     | 3     | 2     | 4     |
| $S_2$ |       | 0     | 1     | 6     | 5     |
| $S_3$ |       |       | 0     | 5     | 5     |
| $S_4$ |       |       |       | 0     | 4     |
| $S_5$ |       |       |       |       | 0     |

$$\sum_{i=1}^k D(S_1, S_i) = 13$$

$$\sum_{i=1}^k D(S_2, S_i) = 16$$

$$\sum_{i=1}^k D(S_3, S_i) = 14$$

$$\sum_{i=1}^k D(S_4, S_i) = 17$$

$$\sum_{i=1}^k D(S_5, S_i) = 18$$

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG  
 $S_5$ : CCTG-TAG--G

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG

$S_1$ : CCTGCTGCAG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCTGCAG  
 $S_2$ : GATG-TGCCG  
 $S_3$ : GATG-TGCAG

$S_1$ : CCTGCT-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG  
 $S_2$ : GATG-T-GCCG  
 $S_3$ : GATG-T-GCAG  
 $S_4$ : CC-GCTAGCAG

$S_1$ : CCTGCT-GCAG  
 $S_5$ : CCTG-TAG--G

$S_1$ : CCTGCT-GCAG  
 $S_2$ : GATG-T-GCCG  
 $S_3$ : GATG-T-GCAG  
 $S_4$ : CC-GCTAGCAG  
 $S_5$ : CCTG-TAG--G

# The Center Star Method

Assume all sequences are length  $n$



# The Center Star Method

Assume all sequences are length  $n$

- computing all of the pairwise scores is  $O(k^2n^2)$ -time ( $O(n^2)$  each)



# The Center Star Method

Assume all sequences are length  $n$

- computing all of the pairwise scores is  $O(k^2n^2)$ -time ( $O(n^2)$  each)
- finding  $S_c$  takes  $O(k^2)$ -time

# The Center Star Method

Assume all sequences are length  $n$

- computing all of the pairwise scores is  $O(k^2n^2)$ -time ( $O(n^2)$  each)
- finding  $S_c$  takes  $O(k^2)$ -time
- finding the pairwise alignments takes  $O(kn^2)$ -time

# The Center Star Method

Assume all sequences are length  $n$

- computing all of the pairwise scores is  $O(k^2n^2)$ -time ( $O(n^2)$  each)
- finding  $S_c$  takes  $O(k^2)$ -time
- finding the pairwise alignments takes  $O(kn^2)$ -time
- inserting additional gaps for each sequence takes  $O(k^2n)$ -time.

# The Center Star Method

# The Center Star Method

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

# The Center Star Method

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

# The Center Star Method

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

# The Center Star Method

$$\sum_{1 \leq i < j \leq k} d_{M^*}(i, j) \geq \sum_{1 \leq i < j \leq k} D(S_i, S_j)$$

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.



# The Center Star Method

$$\begin{aligned}\sum_{1 \leq i < j \leq k} d_{M^*}(i, j) &\geq \sum_{1 \leq i < j \leq k} D(S_i, S_j) \\ &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_i, S_j)\end{aligned}$$

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

# The Center Star Method

$$\begin{aligned}\sum_{1 \leq i < j \leq k} d_{M^*}(i, j) &\geq \sum_{1 \leq i < j \leq k} D(S_i, S_j) \\ &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_i, S_j) \\ &\geq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_c, S_j)\end{aligned}$$

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

# The Center Star Method

$$\begin{aligned}\sum_{1 \leq i < j \leq k} d_{M^*}(i, j) &\geq \sum_{1 \leq i < j \leq k} D(S_i, S_j) \\ &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_i, S_j) \\ &\geq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_c, S_j) \\ &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j)\end{aligned}$$

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

# The Center Star Method

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

$$\begin{aligned} \sum_{1 \leq i < j \leq k} d_{M^*}(i, j) &\geq \sum_{1 \leq i < j \leq k} D(S_i, S_j) \\ &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_i, S_j) \\ &\geq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_c, S_j) \\ &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j) \end{aligned}$$

$$\sum_{1 \leq i < j \leq k} d_M(i, j) = \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} d_M(i, j)$$

# The Center Star Method

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

$$\begin{aligned}\sum_{1 \leq i < j \leq k} d_{M^*}(i, j) &\geq \sum_{1 \leq i < j \leq k} D(S_i, S_j) \\ &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_i, S_j) \\ &\geq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_c, S_j) \\ &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j)\end{aligned}$$

$$\begin{aligned}\sum_{1 \leq i < j \leq k} d_M(i, j) &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} d_M(i, j) \\ &\leq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} \left[ D(S_i, S_c) + D(S_c, S_j) \right]\end{aligned}$$

—

# The Center Star Method

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

$$\begin{aligned} \sum_{1 \leq i < j \leq k} d_{M^*}(i, j) &\geq \sum_{1 \leq i < j \leq k} D(S_i, S_j) \\ &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_i, S_j) \\ &\geq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_c, S_j) \\ &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j) \end{aligned}$$

$$\begin{aligned} \sum_{1 \leq i < j \leq k} d_M(i, j) &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} d_M(i, j) \\ &\leq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} \left[ D(S_i, S_c) + D(S_c, S_j) \right] \\ &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_i, S_c) + \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j) \end{aligned}$$

# The Center Star Method

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

$$\begin{aligned}\sum_{1 \leq i < j \leq k} d_{M^*}(i, j) &\geq \sum_{1 \leq i < j \leq k} D(S_i, S_j) \\ &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_i, S_j) \\ &\geq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_c, S_j) \\ &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j)\end{aligned}$$

$$\begin{aligned}\sum_{1 \leq i < j \leq k} d_M(i, j) &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} d_M(i, j) \\ &\leq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} \left[ D(S_i, S_c) + D(S_c, S_j) \right] \\ &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_i, S_c) + \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j) \\ &= k \sum_{1 \leq j \leq k} D(S_c, S_j)\end{aligned}$$

# The Center Star Method

How much worse than the optimal alignment is the multiple alignment produced using the center star method?

Assume the distances (alignment scores) satisfy the triangle inequality:

- given an MSA  $M$ , let  $d_M(X, Y)$  be the alignment score for the induced pairwise alignment between  $X$  and  $Y$  in  $M$ .
- the triangle inequality says:  $d_M(X, Y) \leq d_M(X, Z) + d_M(Z, Y)$

Let  $M$  be the alignment computed by the center star method and  $M^*$  be the optimal alignment.

$$\begin{aligned}
 \sum_{1 \leq i < j \leq k} d_{M^*}(i, j) &\geq \sum_{1 \leq i < j \leq k} D(S_i, S_j) \\
 &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_i, S_j) \\
 &\geq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} D(S_c, S_j) \\
 &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j) \\
 &\leq \frac{\sum_{1 \leq i < j \leq k} d_M(i, j)}{\sum_{1 \leq i < j \leq k} d_{M^*}(i, j)} \leq 2
 \end{aligned}$$

$$\begin{aligned}
 \sum_{1 \leq i < j \leq k} d_M(i, j) &= \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} d_M(i, j) \\
 &\leq \frac{1}{2} \sum_{1 \leq i \leq k} \sum_{1 \leq j \leq k} [D(S_i, S_c) + D(S_c, S_j)] \\
 &= \frac{k}{2} \sum_{1 \leq j \leq k} D(S_i, S_c) + \frac{k}{2} \sum_{1 \leq j \leq k} D(S_c, S_j) \\
 &= k \sum_{1 \leq j \leq k} D(S_c, S_j)
 \end{aligned}$$



# Progressive Alignment

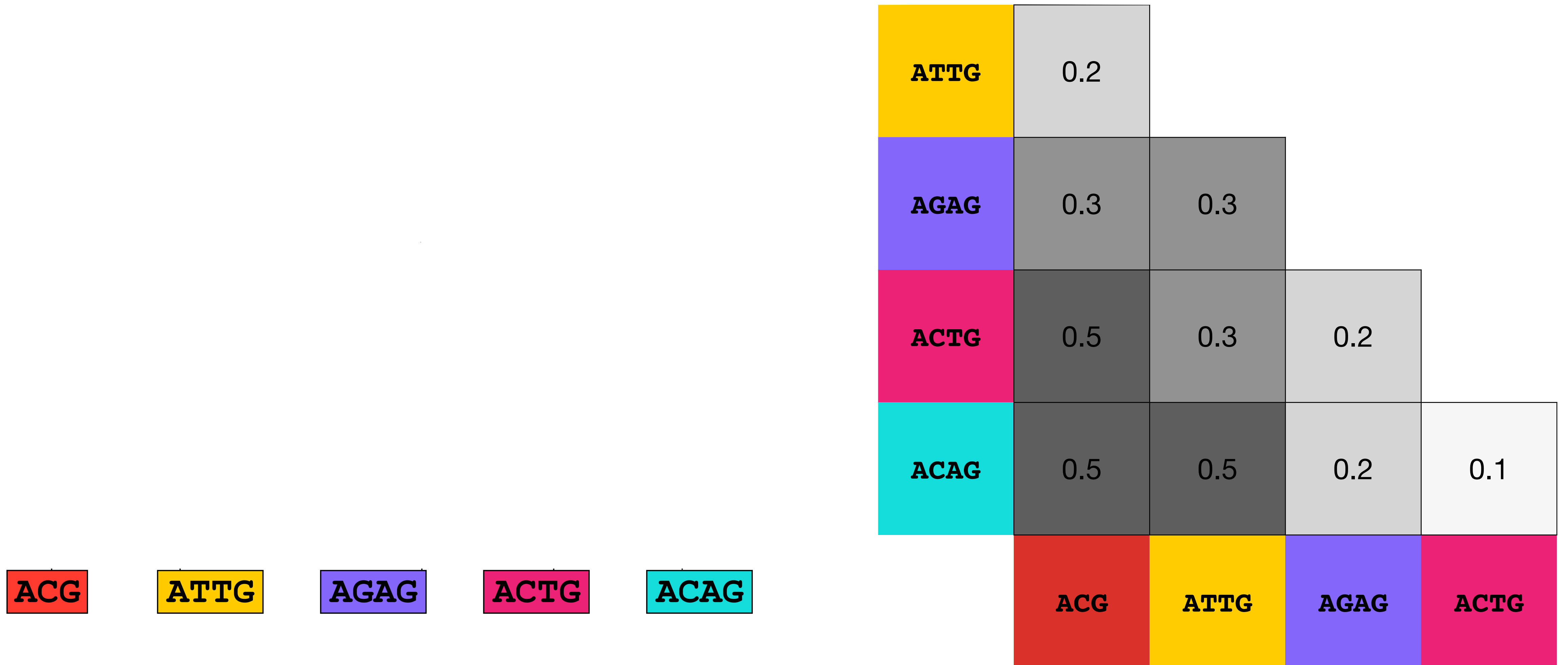
Similar to center star in that we use pairwise alignments to help build multiple alignments.

Introduced by Feng and Doolittle in 1987.

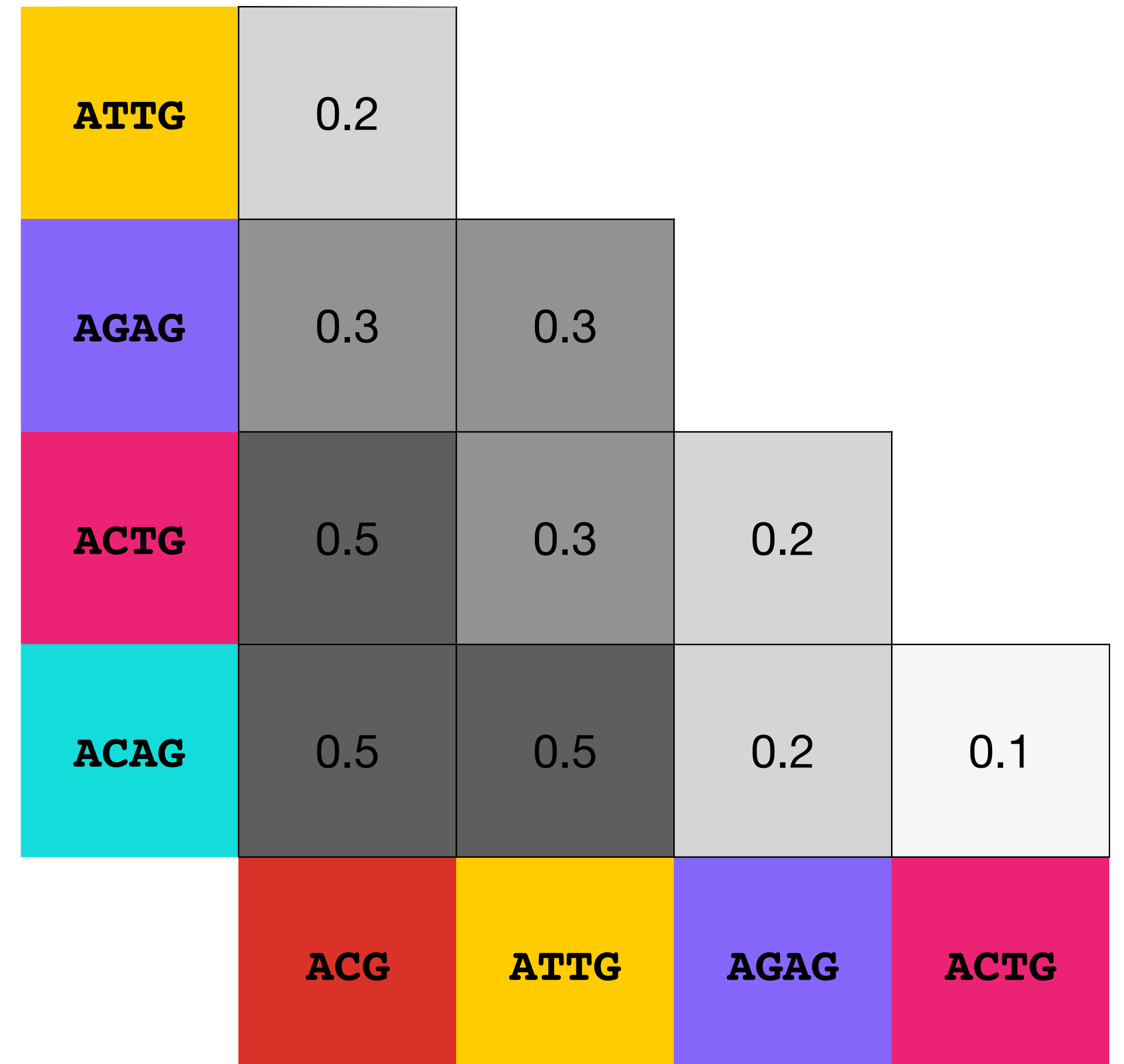
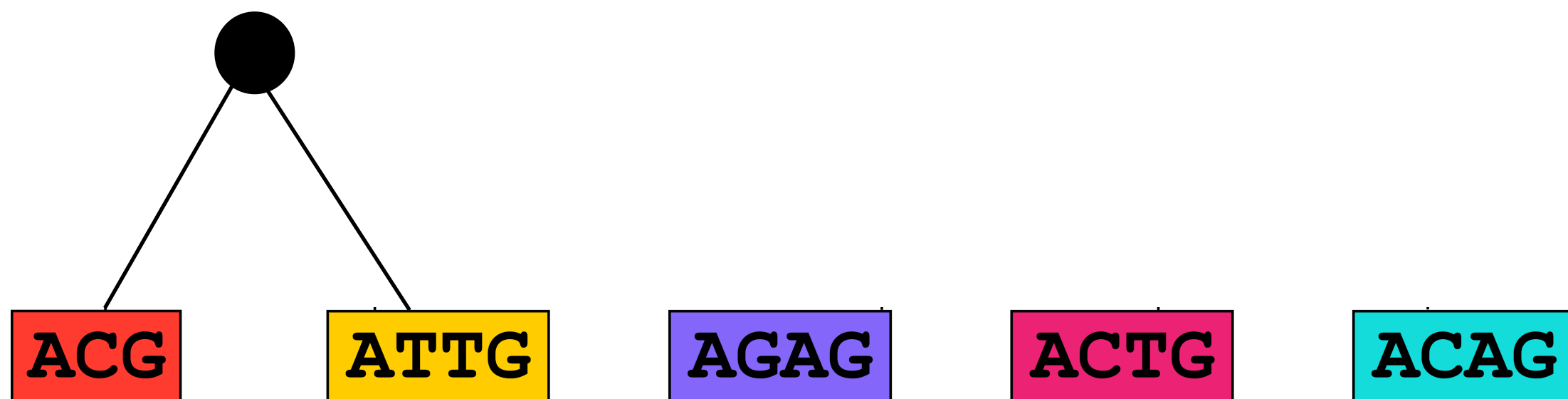
Basic idea:

- compute pairwise alignment scores for each pair of sequences
- generate a **guide tree** which ensures similar sequences are near to each other
- align sequences (or groups) one-by-one from the leaves of the tree

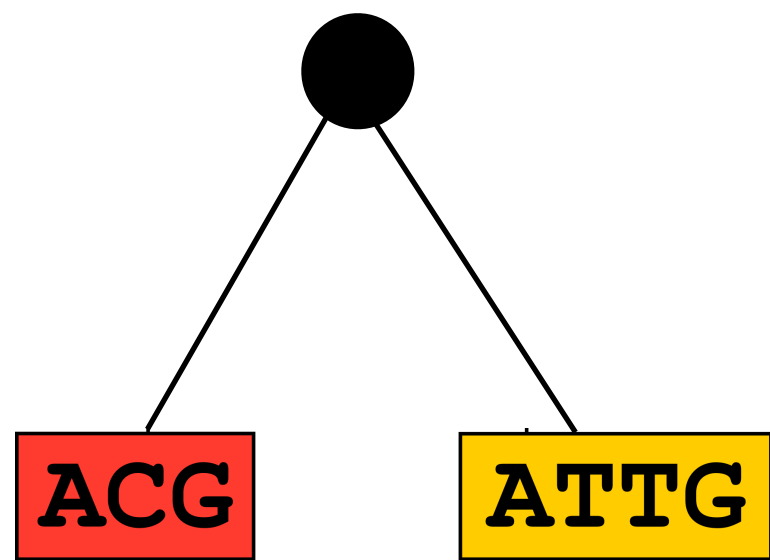
# Progressive Alignment



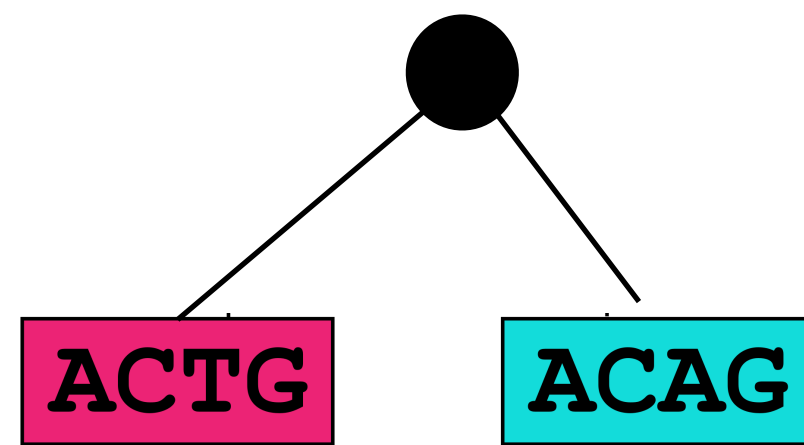
# Progressive Alignment



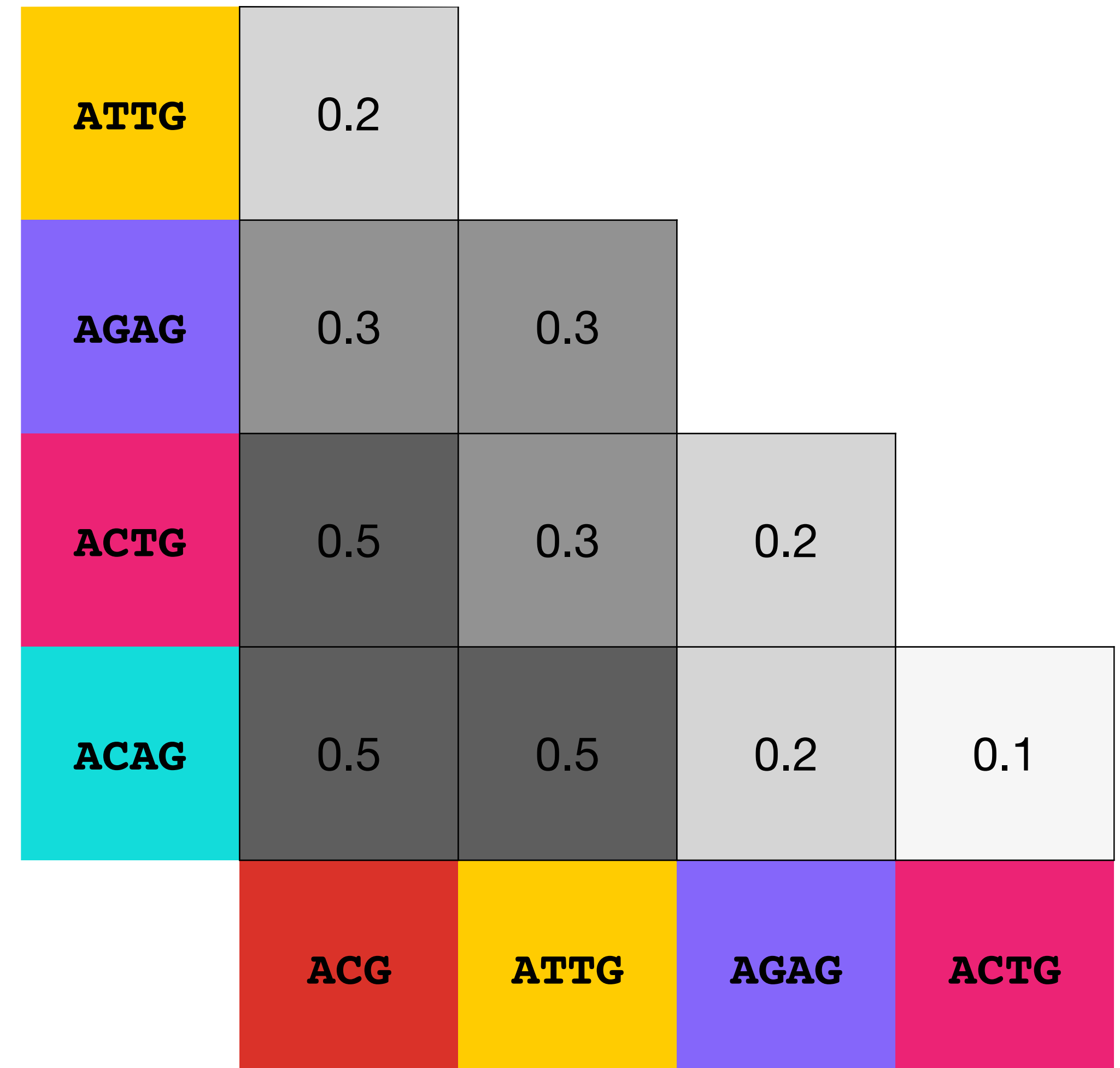
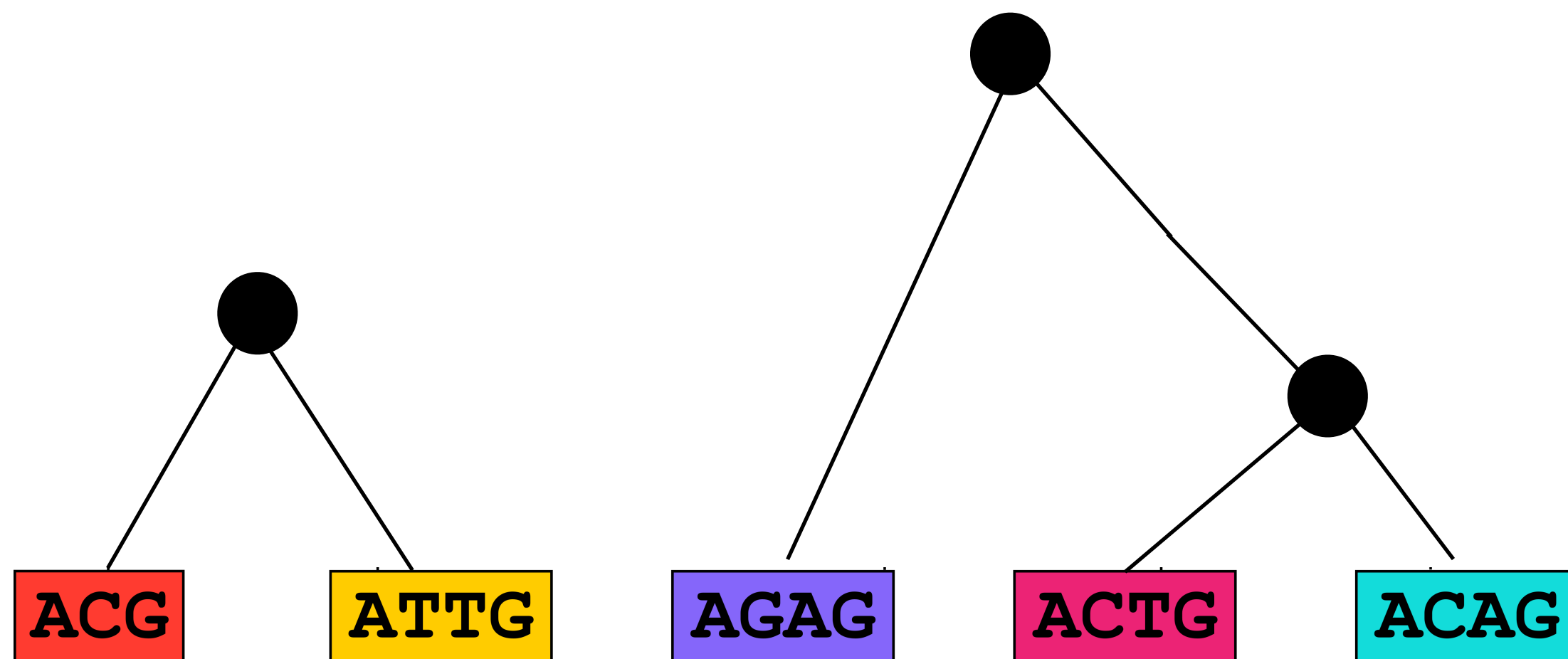
# Progressive Alignment



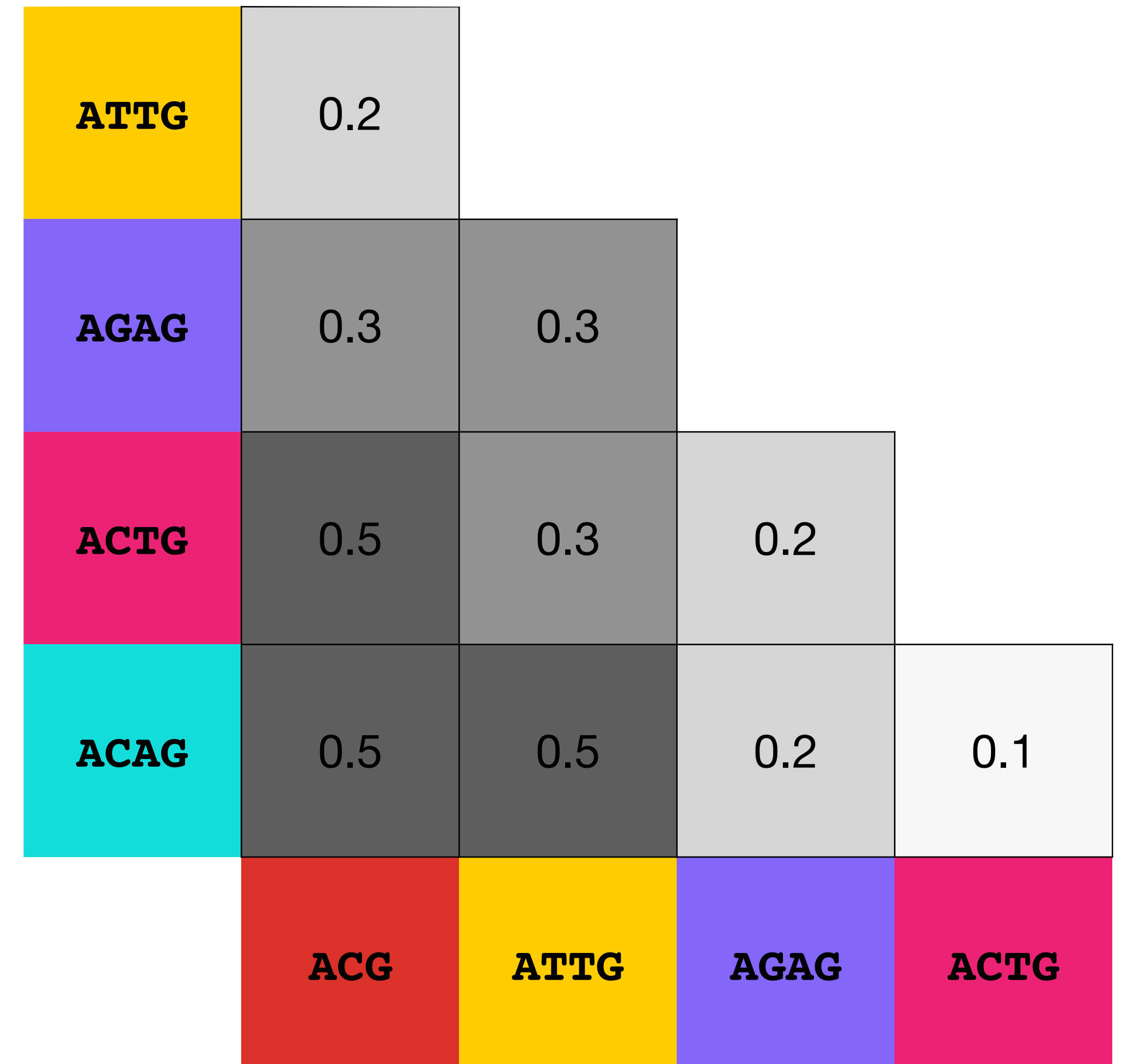
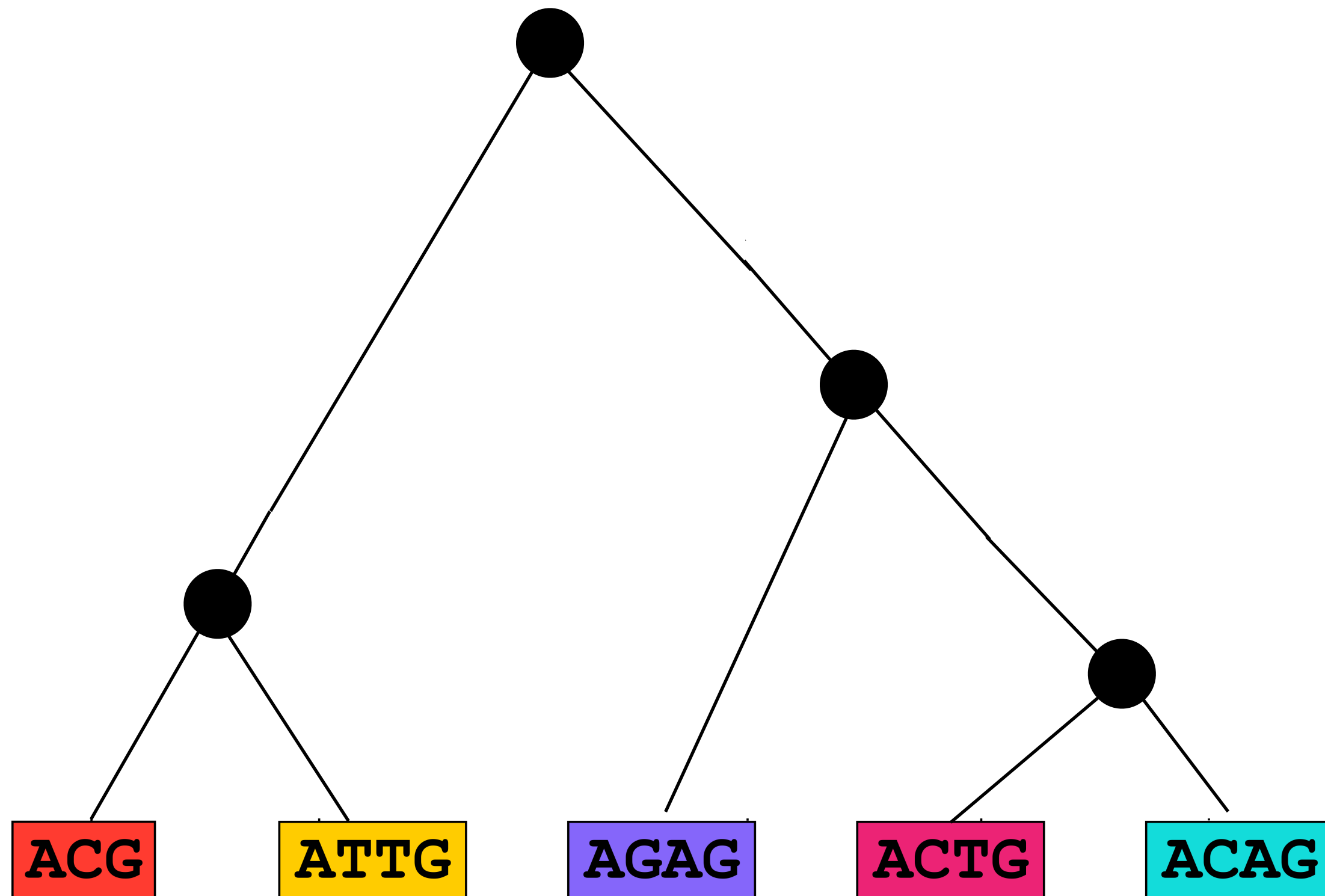
AGAG



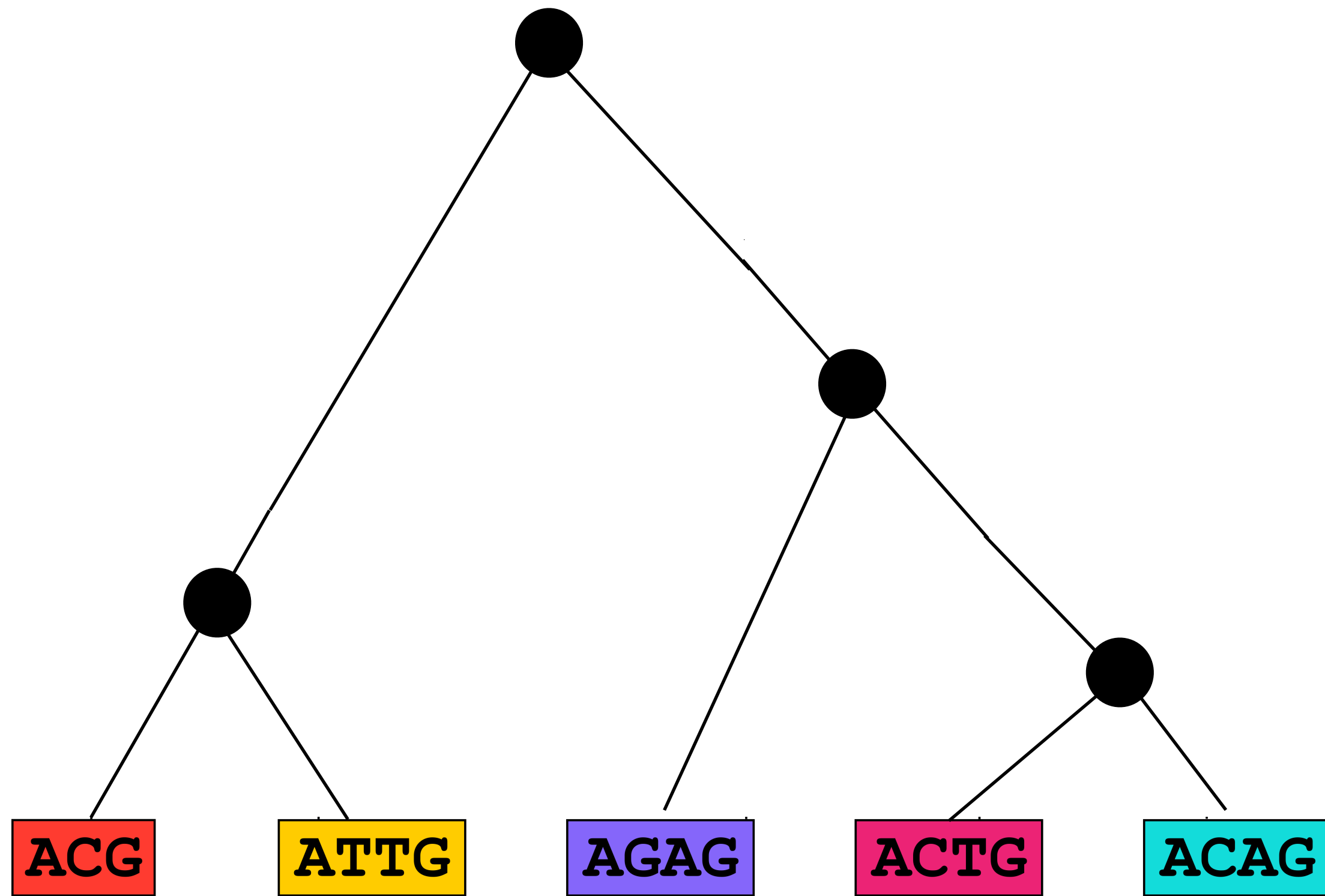
# Progressive Alignment



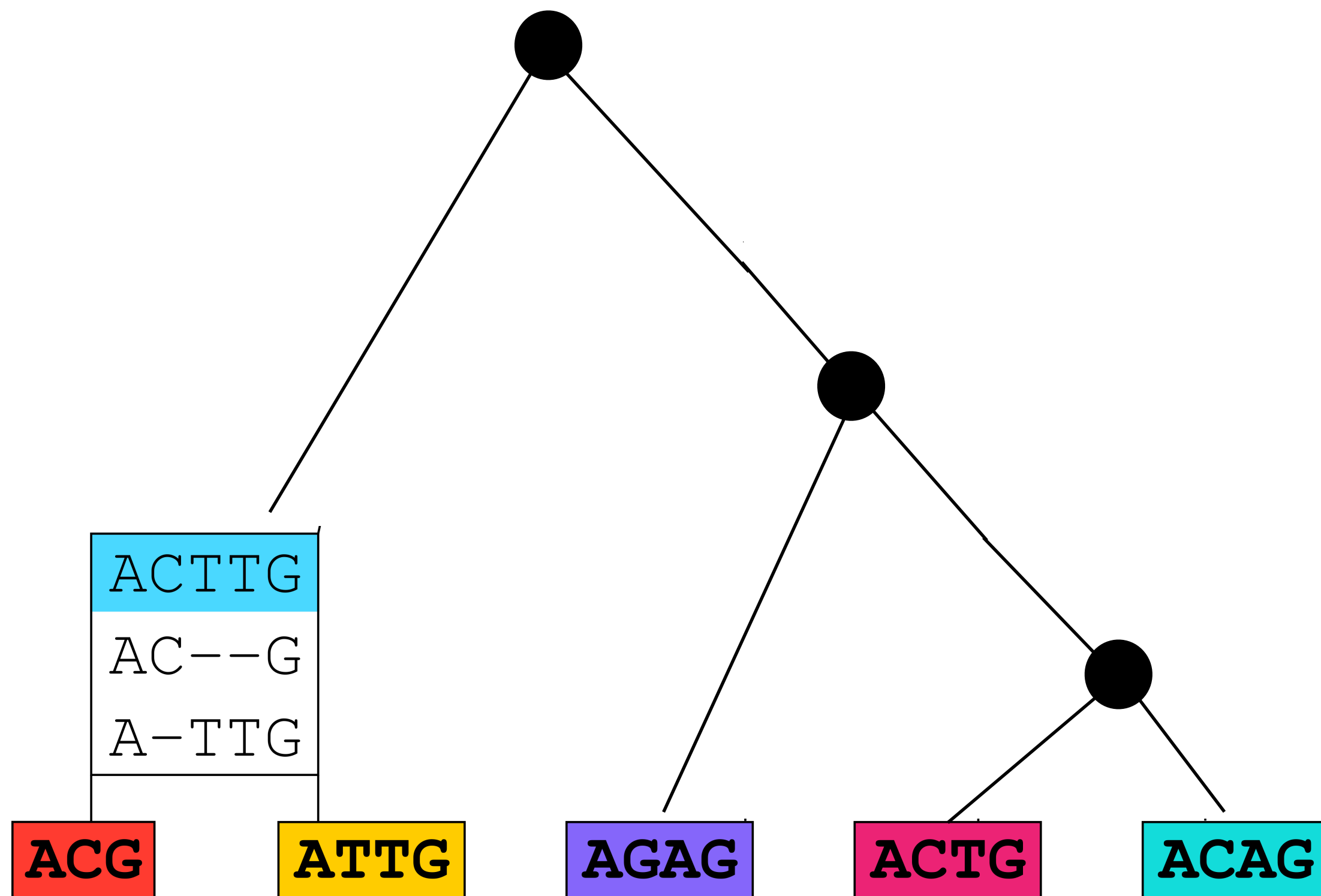
# Progressive Alignment



# Progressive Alignment

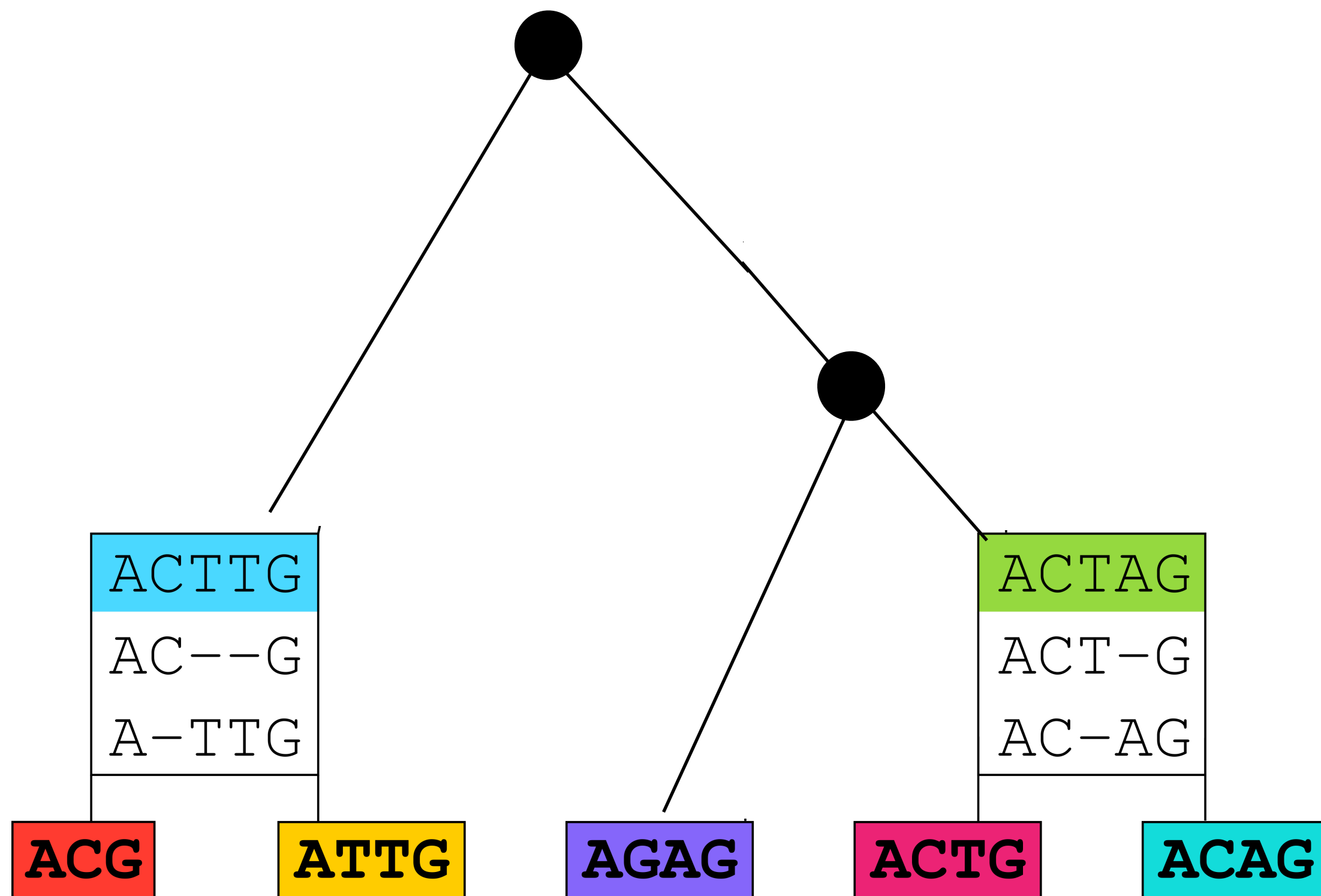


# Progressive Alignment

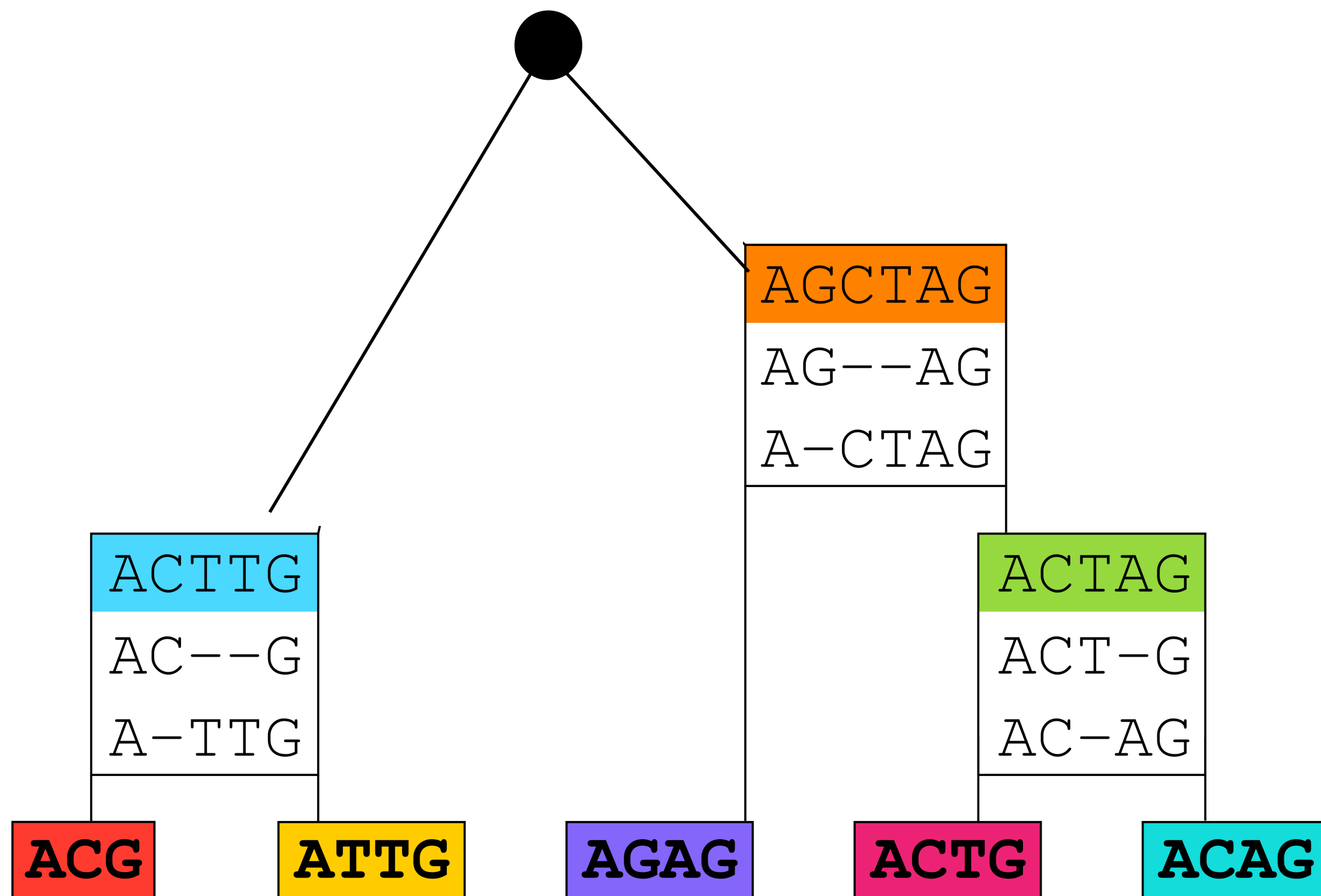




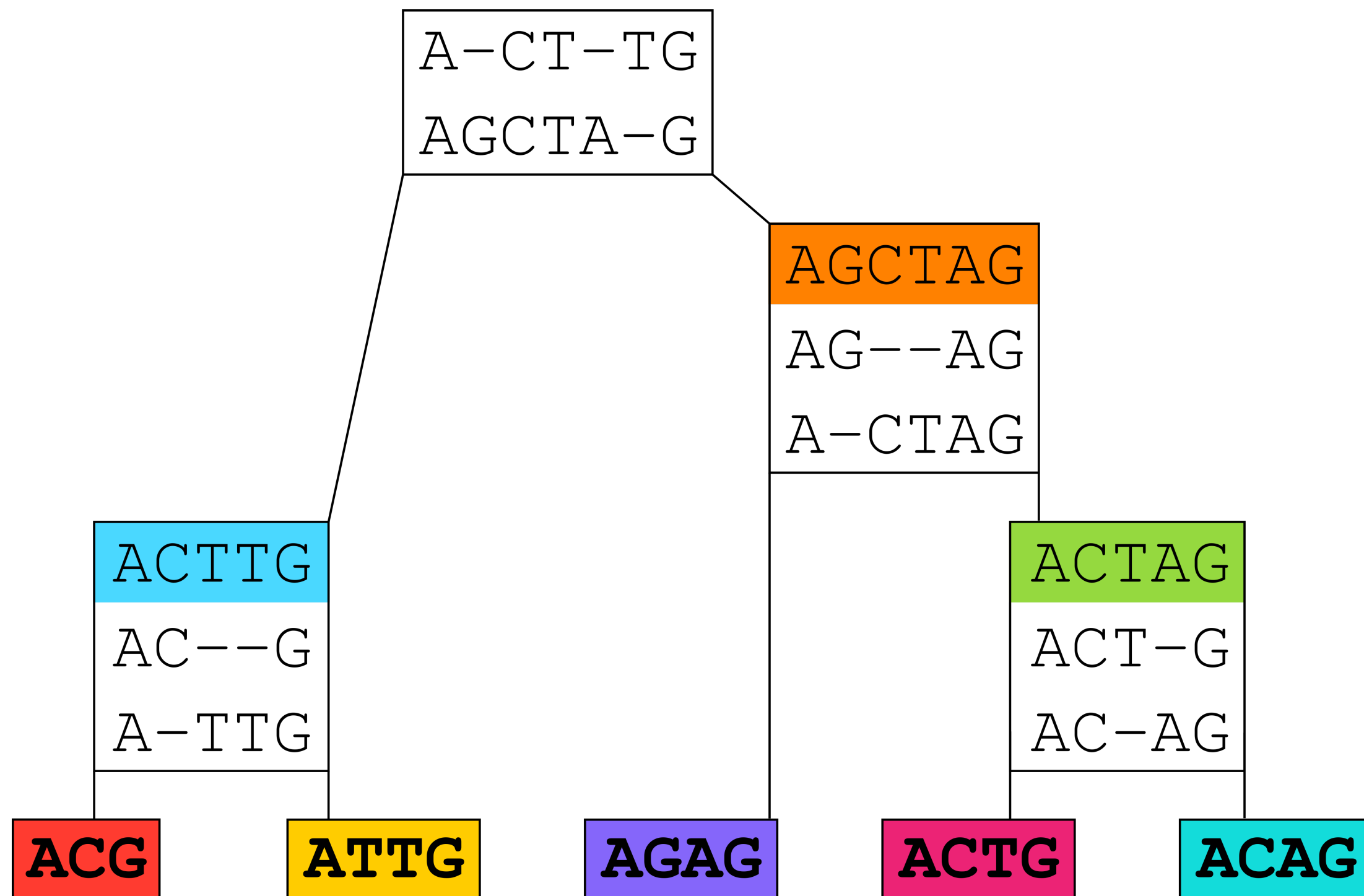
# Progressive Alignment



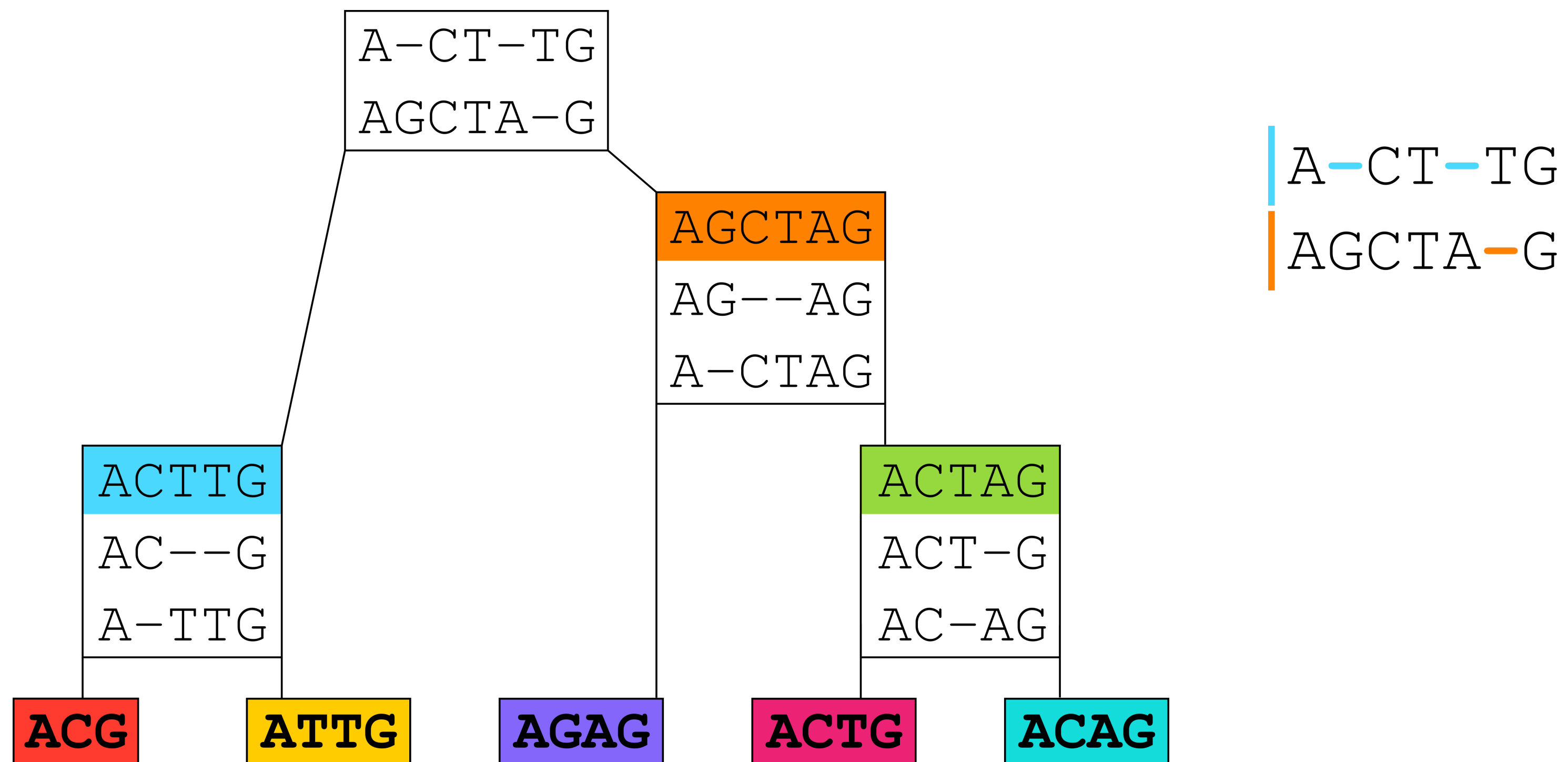
# Progressive Alignment



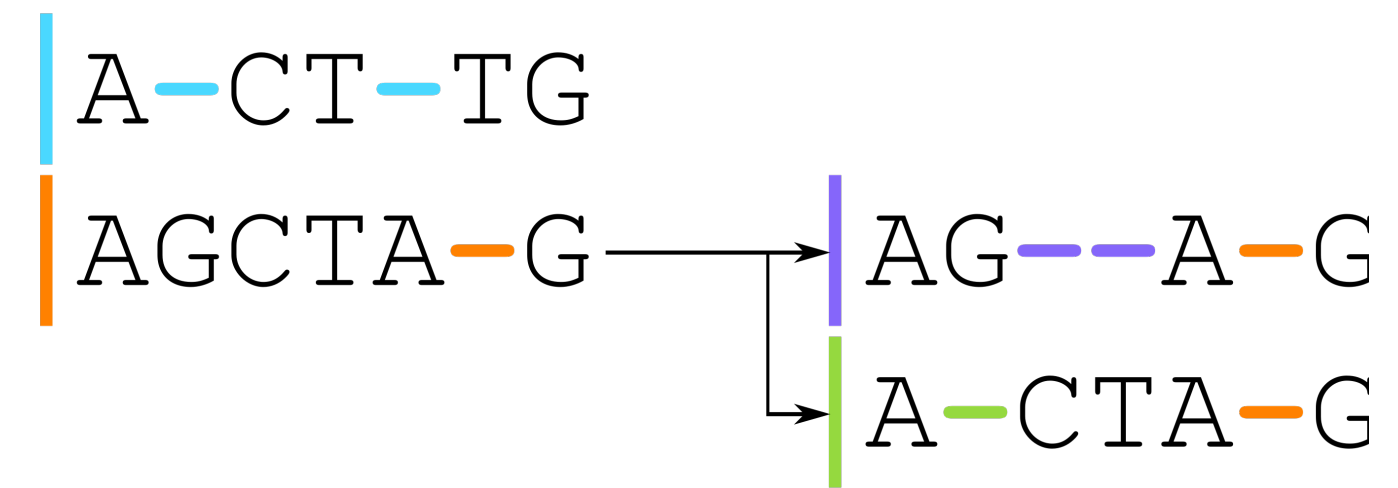
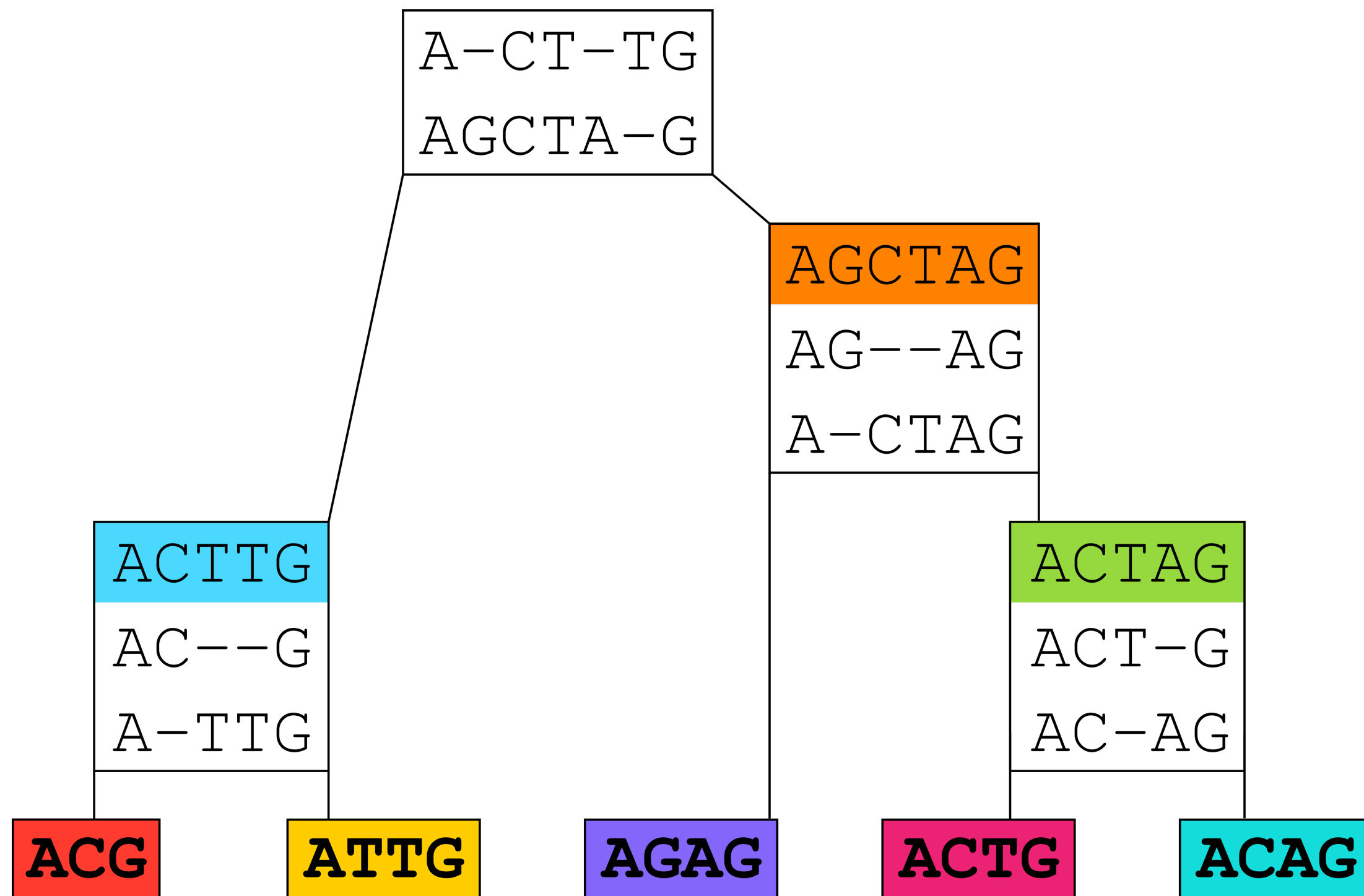
# Progressive Alignment



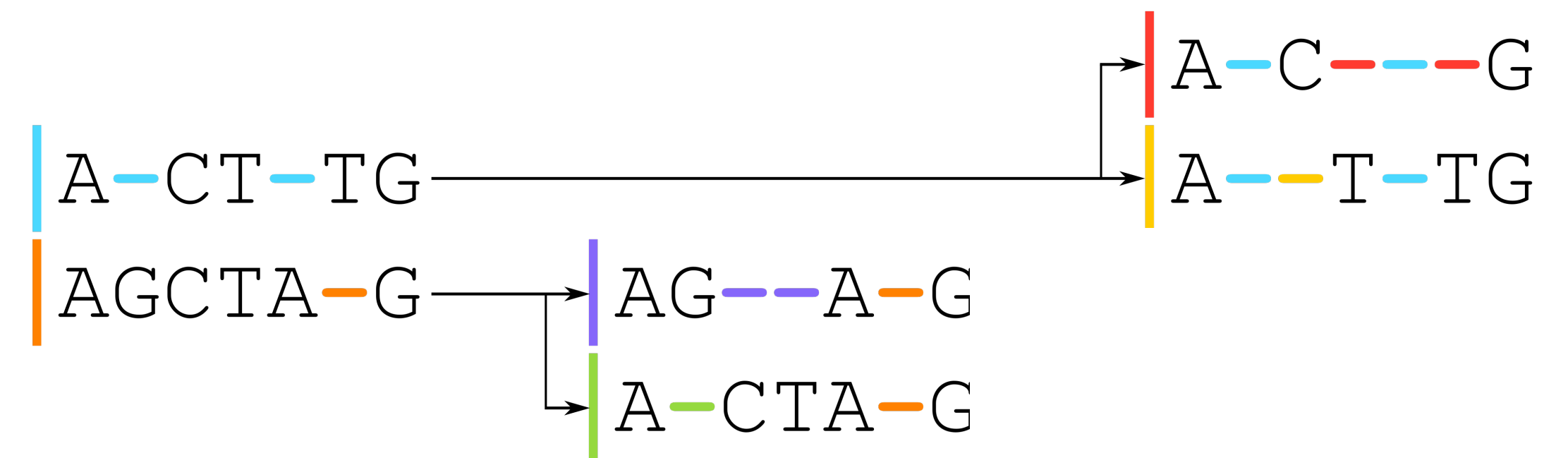
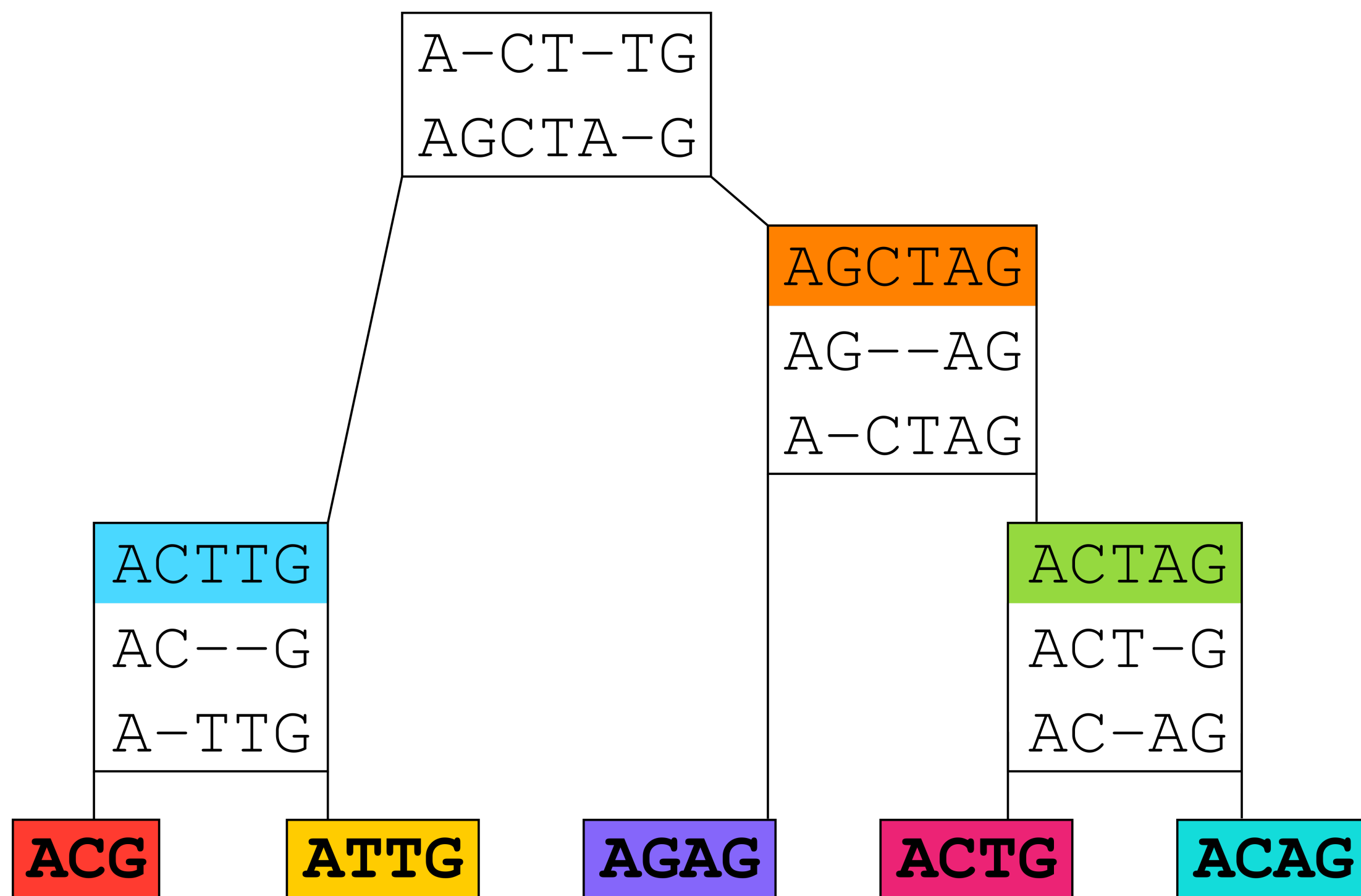
# Progressive Alignment



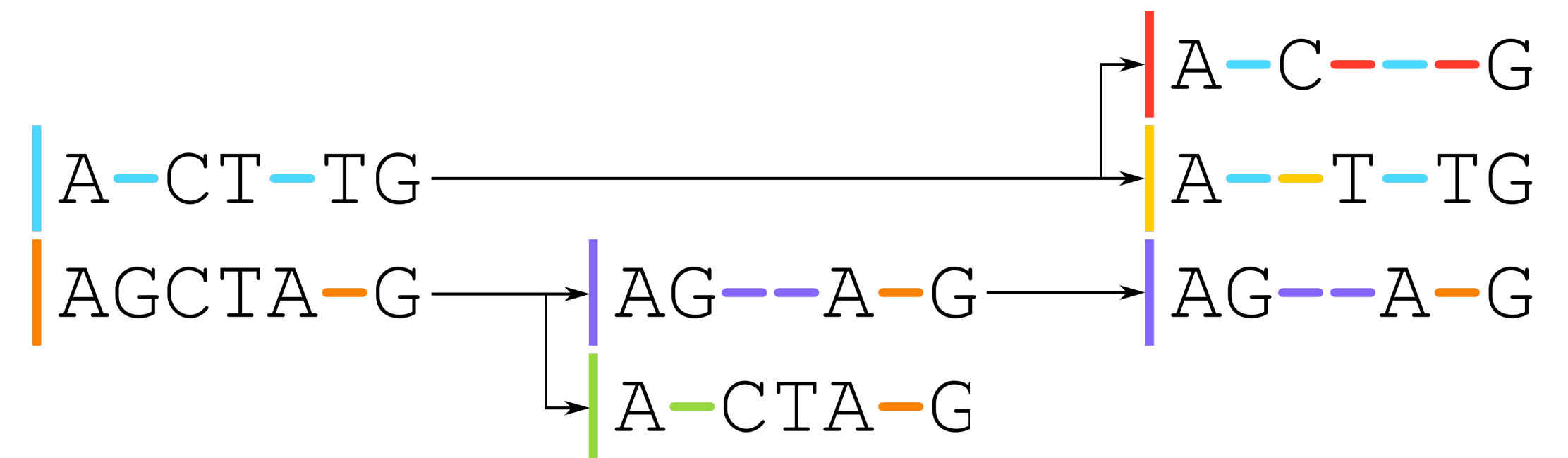
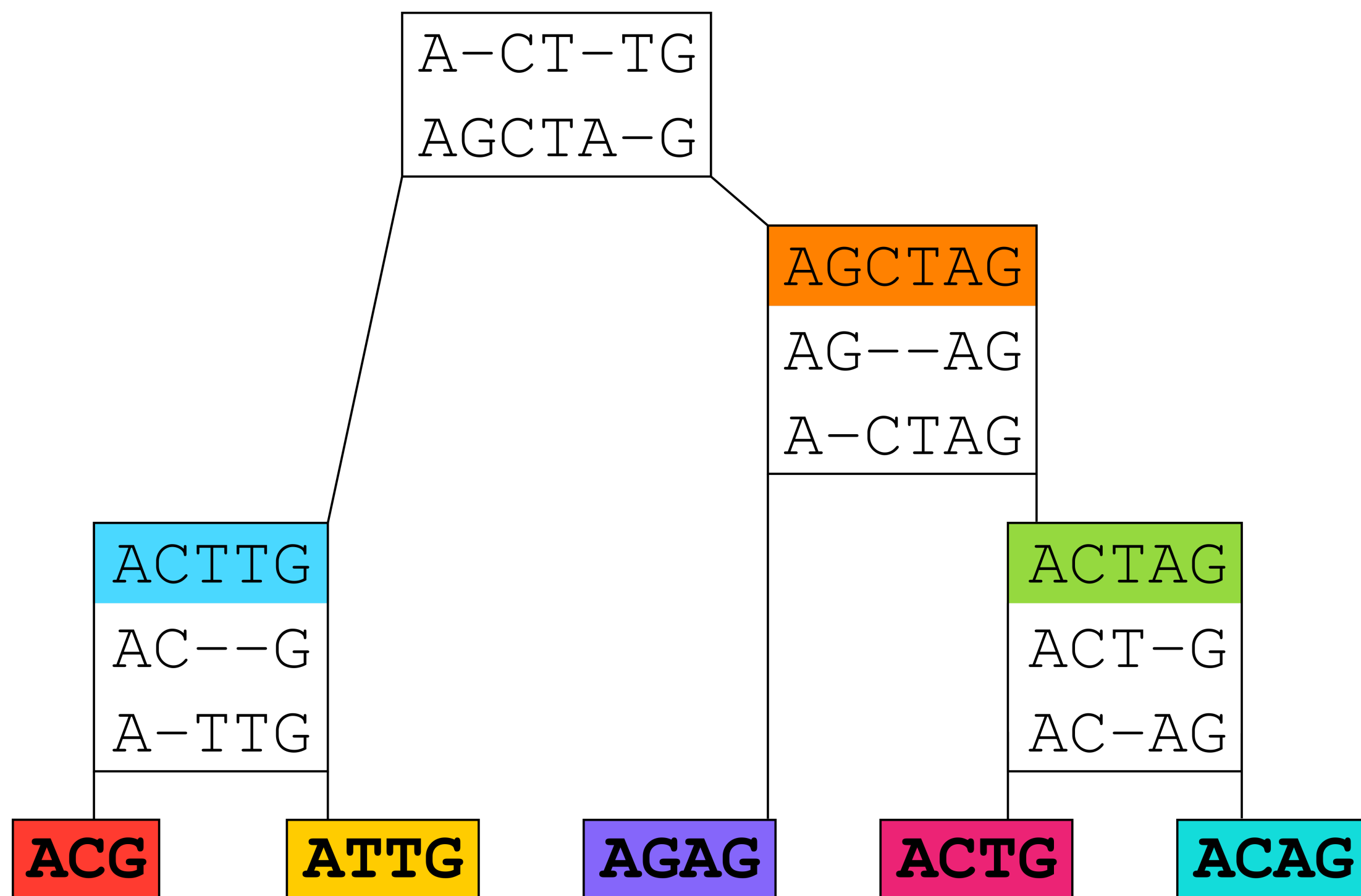
# Progressive Alignment



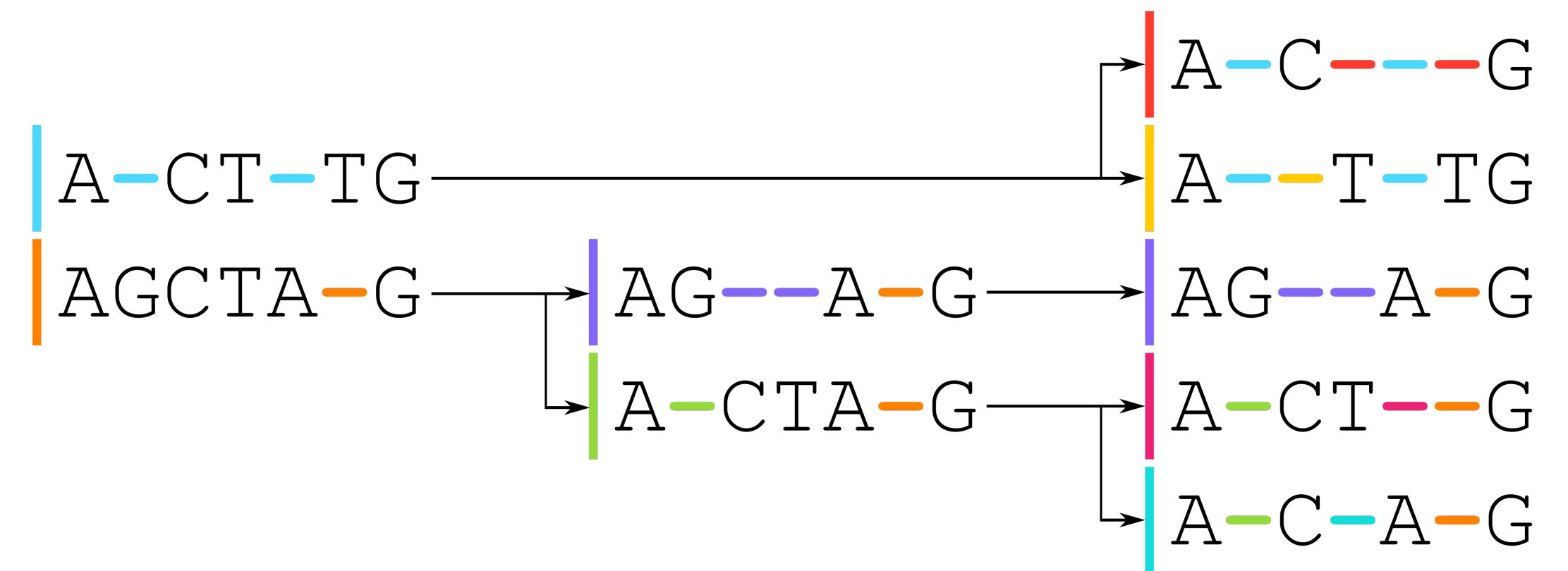
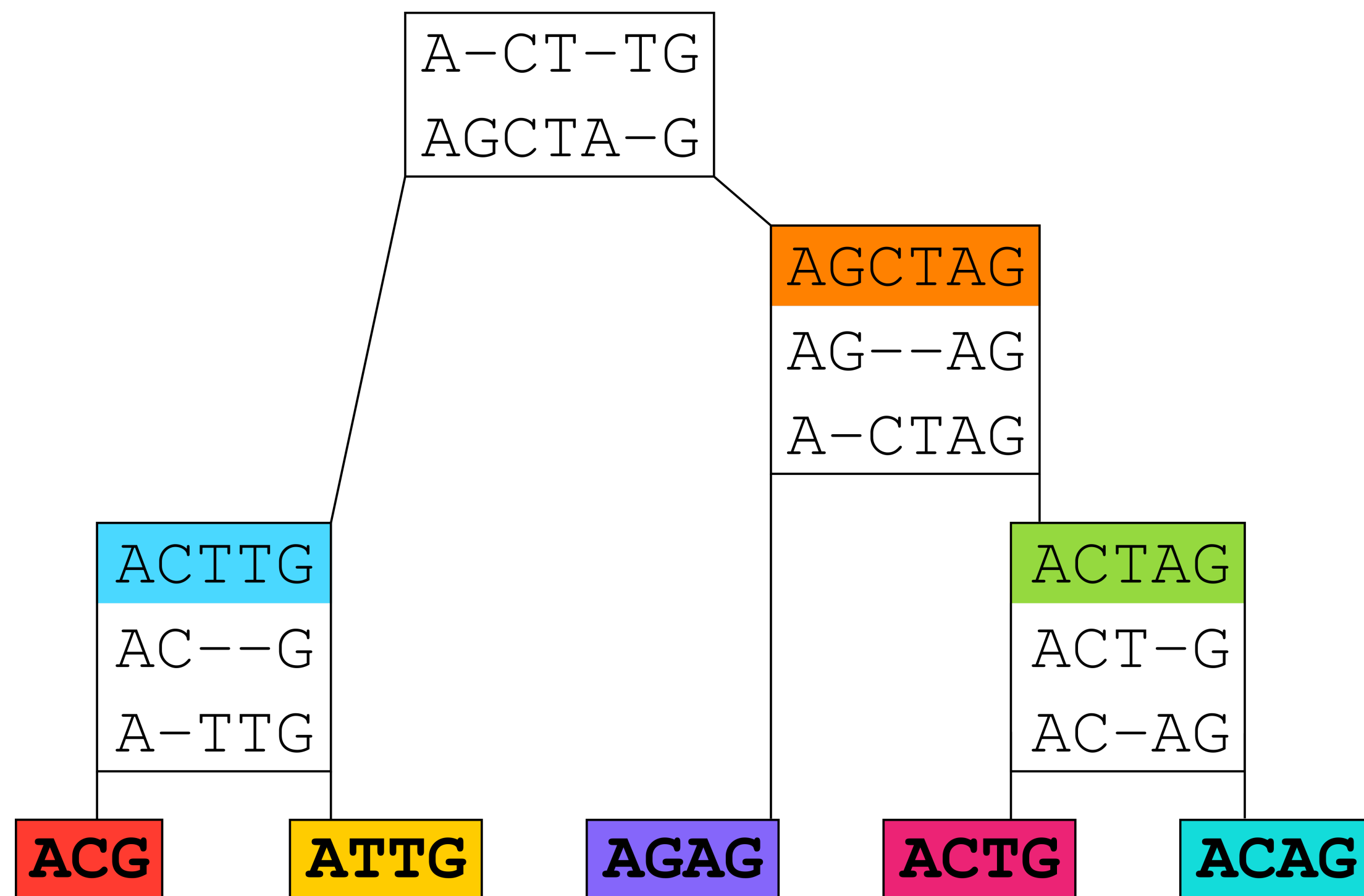
# Progressive Alignment



# Progressive Alignment

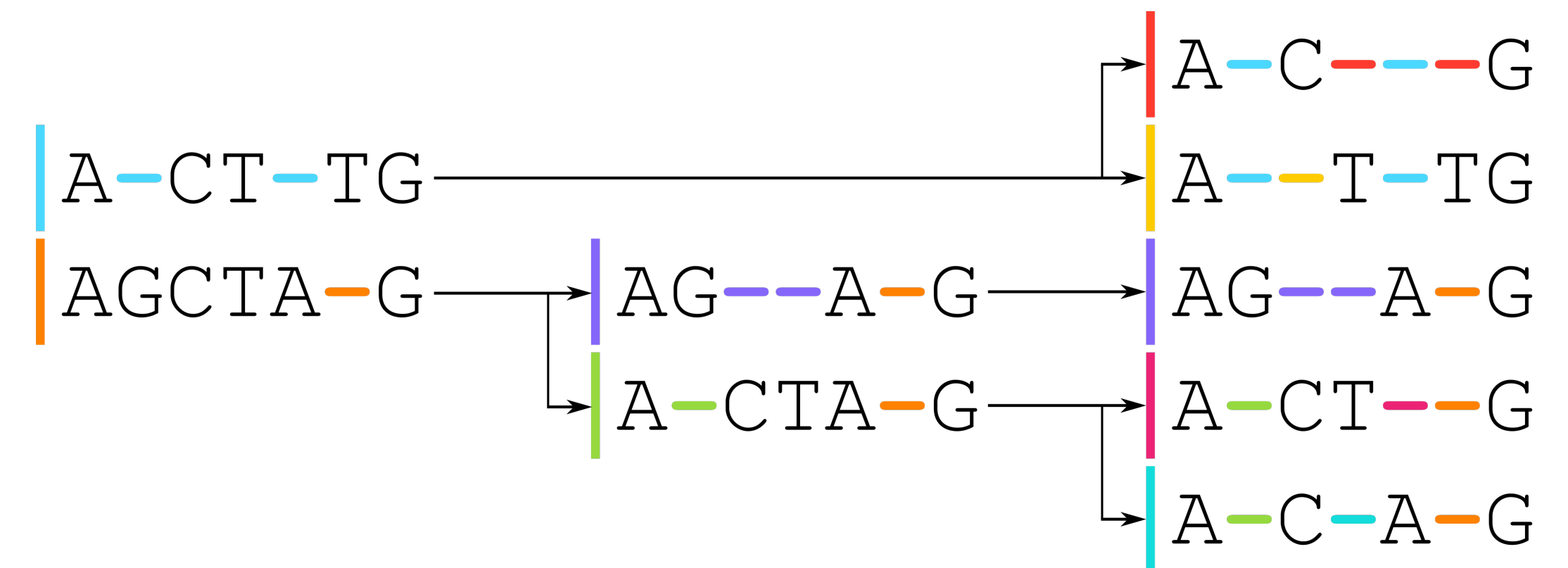
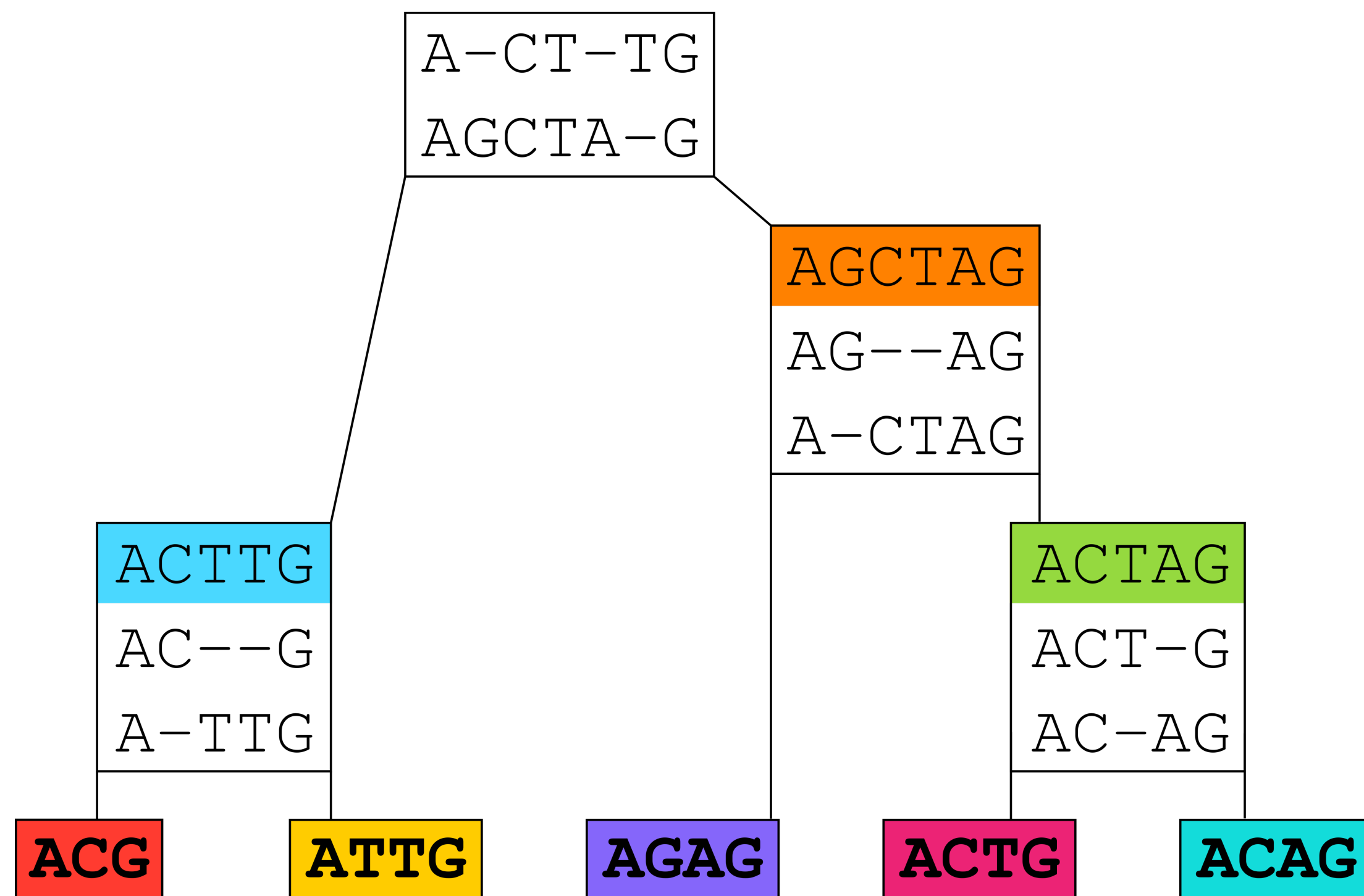


# Progressive Alignment





# Progressive Alignment



# ClustalW

Most used multiple sequence alignment tool, like BLAST its available in many web-based repositories.

Originally published by Higgins and Sharp (1987) and made widely available by Thompson, Higgins, and Gibson (1994).

# ClustalW

Algorithm

# ClustalW

## Algorithm

- Calculate the  $\binom{n}{2}$  pairwise alignments.

# ClustalW

## Algorithm

- Calculate the  $\binom{n}{2}$  pairwise alignments.
- Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.

# ClustalW

## Algorithm

- Calculate the  $\binom{n}{2}$  pairwise alignments.
- Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).

# ClustalW

## Algorithm

- Calculate the  $\binom{n}{2}$  pairwise alignments.
- Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).
- From the leaves compute the alignment at each internal node

# ClustalW

## Algorithm

- Calculate the  $\binom{n}{2}$  pairwise alignments.
- Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).
- From the leaves compute the alignment at each internal node
  - each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.



# ClustalW

## Algorithm

- Calculate the  $\binom{n}{2}$  pairwise alignments.
- Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).
- From the leaves compute the alignment at each internal node
  - each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.

**How do you align two alignments?**

# ClustalW

Profile-to-profile alignment:

# ClustalW

Profile-to-profile alignment:

- for an individual column define

$$PSP(A_1[i], A_2[j]) =: \sum_{x,y \in \Sigma} g_x^i g_y^j \delta(x, y)$$

- here  $g_x^i$  is the count of  $x$ s in column  $i$  of  $A_1$  (similarly for  $g_y^j$ )

# ClustalW

Profile-to-profile alignment:

- for an individual column define

$$PSP(A_1[i], A_2[j]) =: \sum_{x,y \in \Sigma} g_x^i g_y^j \delta(x, y)$$

- here  $g_x^i$  is the count of  $x$ s in column  $i$  of  $A_1$  (similarly for  $g_y^j$ )
- then we can find the best (non-affine gap) global alignment using the following recurrence:

$$V(i, j) = \max \begin{cases} V(i-1, j-1) + PSP(A_1[i], A_2[j]) \\ V(i-1, j) + PSP(A_1[i], -) \\ V(i, j-1) + PSP(-, A_2[j]) \end{cases}$$

# ClustalW

Profile-to-profile alignment:

- for an individual column define

$$PSP(A_1[i], A_2[j]) =: \sum_{x,y \in \Sigma} g_x^i g_y^j \delta(x, y)$$

- here  $g_x^i$  is the count of  $x$ s in column  $i$  of  $A_1$  (similarly for  $g_y^j$ )
- then we can find the best (non-affine gap) global alignment using the following recurrence:

$$V(i, j) = \max \begin{cases} V(i-1, j-1) + PSP(A_1[i], A_2[j]) \\ V(i-1, j) + PSP(A_1[i], -) \\ V(i, j-1) + PSP(-, A_2[j]) \end{cases}$$

Computing the  $g$  values can be done in a total of  $O(k_1 n_1)$  for  $A_1$ , and  $O(k_2 n_2)$  for  $A_2$ .

# ClustalW

Profile-to-profile alignment:

- for an individual column define

$$PSP(A_1[i], A_2[j]) =: \sum_{x,y \in \Sigma} g_x^i g_y^j \delta(x, y)$$

- here  $g_x^i$  is the count of  $x$ s in column  $i$  of  $A_1$  (similarly for  $g_y^j$ )
- then we can find the best (non-affine gap) global alignment using the following recurrence:

$$V(i, j) = \max \begin{cases} V(i-1, j-1) + PSP(A_1[i], A_2[j]) \\ V(i-1, j) + PSP(A_1[i], -) \\ V(i, j-1) + PSP(-, A_2[j]) \end{cases}$$

Computing the  $g$  values can be done in a total of  $O(k_1 n_1)$  for  $A_1$ , and  $O(k_2 n_2)$  for  $A_2$ .

- Then computing table  $V$  takes  $O(n_1 n_2)$ -time to compute.

# ClustalW

Profile-to-profile alignment:

- for an individual column define

$$PSP(A_1[i], A_2[j]) =: \sum_{x,y \in \Sigma} g_x^i g_y^j \delta(x, y)$$

- here  $g_x^i$  is the count of  $x$ s in column  $i$  of  $A_1$  (similarly for  $g_y^j$ )
- then we can find the best (non-affine gap) global alignment using the following recurrence:

$$V(i, j) = \max \begin{cases} V(i-1, j-1) + PSP(A_1[i], A_2[j]) \\ V(i-1, j) + PSP(A_1[i], -) \\ V(i, j-1) + PSP(-, A_2[j]) \end{cases}$$

Computing the  $g$  values can be done in a total of  $O(k_1 n_1)$  for  $A_1$ , and  $O(k_2 n_2)$  for  $A_2$ .

- Then computing table  $V$  takes  $O(n_1 n_2)$ -time to compute.
- The total running time is then  $O(k_1 n_1 + k_2 n_2 + n_1 n_2) \in O(kn + n^2)$ .

# ClustalW

**Align**

```
PPGVKSEDCAS  
PATGVKEDCAS  
PPDGKSED--S
```

```
GATGKDCCS  
GATGKDCAS
```



# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
| P | P | P |   |   |   |   |   |   |   |   |   |
| P | A | P |   |   |   |   |   |   |   |   |   |
| G | T | D |   |   |   |   |   |   |   |   |   |
| V | G | G |   |   |   |   |   |   |   |   |   |
| K | V | K |   |   |   |   |   |   |   |   |   |
| S | K | S |   |   |   |   |   |   |   |   |   |
| E | E | E |   |   |   |   |   |   |   |   |   |
| D | D | D |   |   |   |   |   |   |   |   |   |
| C | C | - |   |   |   |   |   |   |   |   |   |
| A | A | - |   |   |   |   |   |   |   |   |   |
| S | S | S |   |   |   |   |   |   |   |   |   |





# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

S  
S  
S  
S  
S

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

AS

AS

-S

CS

AS

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

CAS

CAS

--S

CCS

CAS

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

DCAS

DCAS

D--S

-CCS

-CAS

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

EDCAS

EDCAS

ED--S

--CCS

--CAS



# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

SEDCAS

KEDCAS

SED--S

D--CCS

D--CAS

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

KSEDCAS

VKEDCAS

KSED--S

KD--CCS

KD--CAS

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

VKSEDCAS

GVKEDCAS

GKSED--S

GKD--CCS

GKD--CAS

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

GVKSEDCAS

TGVKEDCAS

DGKSED--S

TGKD--CCS

TGKD--CAS

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

PGVKSEDCAS

ATGVKEDCAS

PDGKSED--S

ATGKD--CCS

ATGKD--CAS

# ClustalW

|   |   |   | G | A | T | G | K | D | C | C | S |
|---|---|---|---|---|---|---|---|---|---|---|---|
|   |   |   | G | A | T | G | K | D | C | A | S |
|   |   |   | # | # | # | # | # | # | # | # | # |
| P | P | P | # | # | # | # | # | # | # | # | # |
| P | A | P | # | # | # | # | # | # | # | # | # |
| G | T | D | # | # | # | # | # | # | # | # | # |
| V | G | G | # | # | # | # | # | # | # | # | # |
| K | V | K | # | # | # | # | # | # | # | # | # |
| S | K | S | # | # | # | # | # | # | # | # | # |
| E | E | E | # | # | # | # | # | # | # | # | # |
| D | D | D | # | # | # | # | # | # | # | # | # |
| C | C | - | # | # | # | # | # | # | # | # | # |
| A | A | - | # | # | # | # | # | # | # | # | # |
| S | S | S | # | # | # | # | # | # | # | # | # |

PPGVKSEDCAS  
 PATGVKEDCAS  
 PPDGKSED--S

GATGKD--CCS  
 GATGKD--CAS

# ClustalW

## Algorithm

- Calculate the  $\binom{n}{2}$  pairwise alignments.
- Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).
- From the leaves compute the alignment at each internal node
  - each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.

# ClustalW

## Algorithm

- $O(k^2n^2)$**  • Calculate the  $\binom{n}{2}$  pairwise alignments.
- Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).
- From the leaves compute the alignment at each internal node
  - each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.



# ClustalW

## Algorithm

- $O(k^2n^2)$**  • Calculate the  $\binom{n}{2}$  pairwise alignments.
- $O(k^2)$**  • Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).
- From the leaves compute the alignment at each internal node
  - each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.

# ClustalW

## Algorithm

- $O(k^2n^2)$**  • Calculate the  $\binom{n}{2}$  pairwise alignments.
- $O(k^2)$**  • Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- $O(k^3)$**  • Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).
  - From the leaves compute the alignment at each internal node
    - each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.

# ClustalW

## Algorithm

- $O(k^2n^2)$**  • Calculate the  $\binom{n}{2}$  pairwise alignments.
- $O(k^2)$**  • Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.
- $O(k^3)$**  • Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).
  - From the leaves compute the alignment at each internal node
    - each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.  **$O(kn+n^2)$**

# ClustalW

## Algorithm

**$O(k^2n^2)$**  • Calculate the  $\binom{n}{2}$  pairwise alignments.

**$O(k^2)$**  • Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.

**$O(k^3)$**  • Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).

**$O(k^2n+kn^2)$**  • From the leaves compute the alignment at each internal node  
• each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.  **$O(kn+n^2)$**

# ClustalW

## Algorithm

$O(k^2n^2)$  • Calculate the  $\binom{n}{2}$  pairwise alignments.

$O(k^2)$  • Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.

$O(k^3)$  • Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).

$O(k^2n+kn^2)$  • From the leaves compute the alignment at each internal node  
• each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.  $O(kn+n^2)$

$$O(k^2n^2 + k^2 + k^3 + k^2n + kn^2)$$

# ClustalW

## Algorithm

$O(k^2n^2)$  • Calculate the  $\binom{n}{2}$  pairwise alignments.

$O(k^2)$  • Compute the pairwise distance between sequences as  $1 - \frac{x}{y}$  where  $x$  is the number of gap characters, and  $y$  is the number of matches.

$O(k^3)$  • Use the neighbor-joining method to create the guide tree (we will talk about the details of this later).

$O(k^2n+kn^2)$  • From the leaves compute the alignment at each internal node  
• each alignment will be between either: (i) two sequences, (ii) two partial alignment, or (iii) a sequence and a partial alignment.  $O(kn+n^2)$

$$O(k^2n^2 + k^2 + k^3 + k^2n + kn^2) \in O(k^2n^2 + k^3)$$

# Issues with progressive alignment

"Once a gap, always a gap"

- Progressive alignment does not realign sequences, so once a gap is introduced at a certain position, it cannot be reconsidered.
- This means the order things are aligned can influence the alignment.

# Iterative Methods

To overcome the issues with progressive alignment, some methods first create an initial multiple alignment, then continually break apart then re-align subsets of sequences to improve the alignment.

Popular examples are MAFFT (Kato, 2002) and MUSCLE (Edgar, 2004).



# MUSCLE

**(Multiple Sequence Comparison by Log-Expectation)**

The Log-Expectation (LE) score replaces the PSP score in ClustalW.

# MUSCLE

**(Multiple Sequence Comparison by Log-Expectation)**

The Log-Expectation (LE) score replaces the PSP score in ClustalW.

Does not ignore spaces in the two profiles when aligning.

# MUSCLE

## (Multiple Sequence Comparison by Log-Expectation)

The Log-Expectation (LE) score replaces the PSP score in ClustalW.

Does not ignore spaces in the two profiles when aligning.

$$LE(A_1[i], A_2[j]) = (1 - f_G^i)(1 - f_G^j) \log \sum_{x,y \in \Sigma} f_x^i f_y^j$$

# MUSCLE

## (Multiple Sequence Comparison by Log-Expectation)

The Log-Expectation (LE) score replaces the PSP score in ClustalW.

Does not ignore spaces in the two profiles when aligning.

$$LE(A_1[i], A_2[j]) = (1 - f_G^i)(1 - f_G^j) \log \sum_{x,y \in \Sigma} f_x^i f_y^j$$

- here  $f_G^i$  is the frequency of gaps in column  $i$  of  $A_1$  (similarly for  $f_G^j$ ).

# MUSCLE

## (Multiple Sequence Comparison by Log-Expectation)

The Log-Expectation (LE) score replaces the PSP score in ClustalW.

Does not ignore spaces in the two profiles when aligning.

$$LE(A_1[i], A_2[j]) = (1 - f_G^i)(1 - f_G^j) \log \sum_{x,y \in \Sigma} f_x^i f_y^j$$

- here  $f_G^i$  is the frequency of gaps in column  $i$  of  $A_1$  (similarly for  $f_G^j$ ).
- $f_x^i$  and  $f_y^j$  are the *normalized* frequency of other characters:

# MUSCLE

## (Multiple Sequence Comparison by Log-Expectation)

The Log-Expectation (LE) score replaces the PSP score in ClustalW.

Does not ignore spaces in the two profiles when aligning.

$$LE(A_1[i], A_2[j]) = (1 - f_G^i)(1 - f_G^j) \log \sum_{x,y \in \Sigma} f_x^i f_y^j$$

- here  $f_G^i$  is the frequency of gaps in column  $i$  of  $A_1$  (similarly for  $f_G^j$ ).
- $f_x^i$  and  $f_y^j$  are the *normalized* frequency of other characters:

$$f_x^i =: \frac{g_x^i}{\left( \sum_{x' \in \Sigma} g_{x'}^j \right)}$$

# MUSCLE

(Multiple Sequence Comparison by Log-Expectation)

Algorithm:

# MUSCLE

## (Multiple Sequence Comparison by Log-Expectation)

Algorithm:

1. **draft progressive alignment** -- similar to ClustalW but with
  - LE score for aligning profiles,
  - a more efficient tree building algorithm, and
  - a more efficient pairwise comparison (using *k*-mer counting).



# MUSCLE

## (Multiple Sequence Comparison by Log-Expectation)

Algorithm:

1. **draft progressive alignment** -- similar to ClustalW but with
  - LE score for aligning profiles,
  - a more efficient tree building algorithm, and
  - a more efficient pairwise comparison (using  $k$ -mer counting).
2. **improved progressive alignment** -- using the alignment from (1)
  - redefine the pairwise distances using the Kimura distance  $-\ln\left(1 - D - \frac{D^2}{5}\right)$
  - $D$  is the fraction of matches.
  - re-align.

# MUSCLE

## (Multiple Sequence Comparison by Log-Expectation)

Algorithm:

1. **draft progressive alignment** -- similar to ClustalW but with
  - LE score for aligning profiles,
  - a more efficient tree building algorithm, and
  - a more efficient pairwise comparison (using  $k$ -mer counting).
2. **improved progressive alignment** -- using the alignment from (1)
  - redefine the pairwise distances using the Kimura distance  $-\ln\left(1 - D - \frac{D^2}{5}\right)$
  - $D$  is the fraction of matches.
  - re-align.
3. **refinement** -- deleting an edge in the guide tree creates two sub-groups of sequences with induced sub-alignments.
  - Extract those two sub-alignments and realign them.
  - Only keep the new alignment if the  $SP$  score is increased.
  - Stop when  $SP$  has not improved: in a predefined number of iterations or when all edges are visited.

# So many aligners!!

- ClustalW
- ClustalΩ
- MAFFT
- MUSCLE
- PREFAB
- Opal
- Kalign
- T-Coffee
- MSA
- Dialign
- MACAW
- MUMMALS
- Prank
- Probalign
- Probcons
- POA
- SATé
- PASTA
- MSAProbs
- M-Coffee (meta-alignment)
- FastR (RNA)
- PMFastR (RNA)
- ...