

BIOINFORMÁTICA
Práctica 2.1
Los algoritmos de similitud de secuencias.
Implementación de los algoritmos
Needleman-Wunsch y Smith-Waterman

Juan Miguel García Gómez
(juanmig@upvnet.upv.es)
IBIME-ITACA, Dpto. de Física Aplicada
Facultad de Informática-UPV

23 de abril de 2010

Índice

1. Objetivo de la práctica	1
2. Material de la práctica	2
3. Memoria de la práctica	2
4. Ejercicio. Implementación del algoritmo Needleman-Wunsch	2
4.1. Objetivo	2
4.2. Notas	2
4.3. Desarrollo	3
5. Ejercicio. Implementación del algoritmo Smith-Waterman	4
5.1. Objetivo	4
5.2. Notas	4
5.3. Desarrollo	4
6. Ejercicio. Comparación de algoritmos	6
6.1. Objetivo	6
6.2. Notas	6
6.3. Desarrollo	6

1. Objetivo de la práctica

La programación dinámica es el marco ideal para entender los algoritmos de alineamiento de secuencias. El Algoritmo de Needleman-Wunsch obtiene el alineamiento

to global óptimo entre dos secuencias, mientras que la modificación introducida por Smith-Waterman obtiene los alineamientos óptimos locales .

En la práctica implementaremos los algoritmos Needleman-Wunsch y Smith-Waterman en el lenguaje de programación R. Estas implementaciones nos permitirán estudiar las trazas de cada algoritmo al alinear un par de secuencias.

Debemos recordar que, si bien estos algoritmos son la base de la comparación de secuencias en biología molecular, su aplicación directa en grandes búsquedas en base de datos de secuencias es computacionalmente inabordable y su uso queda limitado a ciertas aplicaciones específicas. Esto ha llevado al diseño de algoritmos como BLAST, que no aseguran el alineamiento óptimo, pero dan resultados interesantes biológicamente y computacionalmente viables.

2. Material de la práctica

- R > 2,2,0 (<http://www.r-project.org>).
- Plantilla 21AlineamientoProgramacionDinamicaCOMPLETAR.R con el código base de las implementaciones de los algoritmos.

3. Memoria de la práctica

La memoria de la práctica será el script `21AlineamientoProgramacionDinamica.R` con las funciones `Needleman-Wunsch` y `SmithWaterman` completadas correctamente.

4. Ejercicio. Implementación del algoritmo Needleman-Wunsch

4.1. Objetivo

Completa los bucles 1,2 y 3 de la función `NeedlemanWunsch` para calcular, mediante programación dinámica, el valor del alineamiento de las subcadenas $S_{1,i-1}$ y $R_{1,j-1}$.

4.2. Notas

- La variable `trellis` es una matriz de m filas y n columnas, donde m es la longitud(S)+1 y n es la longitud(R)+1.
- La matriz `delta` indica la puntuación de la sustitución del símbolo A por B , incluido el borrado (sustitución A por $-$) y la inserción (sustitución de $-$ por B).
- La variable `decisionf` almacena la función `max` (cálculo del máximo) o `min` (cálculo de mínimo). Esto es debido a que según el caso, puede quererse utilizar una matriz `delta` con penalizaciones (e.g. distancias) o con puntuaciones. En caso de utilizar distancias utilizaremos `min`; en caso de puntuaciones utilizaremos `max`.

- Las matrices `backi` y `backj` almacenan las coordenadas del nodo desde el que se obtuvo la mejor puntuación.
- Para contemplar la cadena vacía en el trellis, insertamos el símbolo `—` (vacío) delante de las cadenas `S` y `R`. La cadena `u` será `—S` y la cadena `v` será `—R`. Por lo tanto, los símbolos u_i son S_{i-1} y los símbolos v_j son R_{j-1} .
- `costes["B"]` almacena temporalmente la puntuación de borrar el símbolo `u[i]`.
- `costes["I"]` almacena temporalmente la puntuación de insertar el símbolo `v[j]`.
- `costes["MoS"]` almacena temporalmente la puntuación de un match o una sustitución el símbolo `u[i]` por `v[j]`.

4.3. Desarrollo

1. Analiza el código escrito en la plantilla:
 - a) ¿Cuál es el coste del nodo inicial?
 - b) ¿Qué instrucciones eligen el camino (borrado, inserción o sustitución) utilizado en el caso general?
 - c) Cuando realizamos un borrado en el nodo i, j , ¿cuál es el nodo de procedencia?
2. Completa el código necesario para calcular la puntuación del borrado de $u[i]$ producido en la primera columna del trellis:

```
for (i in 2:m)
{
  trellis[i,1] <- #COMPLETAR
  backi[i,1] <- i-1
  backj[i,1] <- 1
}
```

3. Completa el código necesario para calcular la puntuación de la inserción de $v[j]$ producida en la primera fila del trellis:

```
for (j in 2:n)
{
  trellis[1,j] <- #COMPLETAR
  backi[1,j] <- 1
  backj[1,j] <- j-1
}
```

4. Completa el código necesario para calcular la puntuación del borrado, la inserción y la sustitución producida en el caso general i, j .

```
for (i in 2:m)
{
  for (j in 2:n)
  {
```

```

#borrado de u[i]
costes["B"] <- #COMPLETAR

#insercion de v[j]
costes["I"] <- #COMPLETAR

#sustitucion o match u[i]==v[j]
costes["MoS"] <- #COMPLETAR

#asignacion de coste
trellis[i,j] <- decisionf(costes)

#preparacion del backtracking
operacion <- names(which(costes==trellis[i,j])[1])
if (operacion=="B")
  {backi[i,j] <- i-1; backj[i,j] <- j;}
else if (operacion=="I")
  {backi[i,j] <- i; backj[i,j] <- j-1;}
else if (operacion=="MoS")
  {backi[i,j] <- i-1; backj[i,j] <- j-1;}
else warning("Operacion no conocida en trellis")

} #for j
} #for i

```

5. Ejercicio. Implementación del algoritmo Smith-Waterman

5.1. Objetivo

Modifica el código del algoritmo Needleman-Wunsch para implementar el algoritmo Smith-Waterman

5.2. Notas

- Revisa las notas del ejercicio anterior

5.3. Desarrollo

1. Añade una posición $\text{costes}[\text{"Ini"}]=0$ al vector costes para implementar la puntuación de llegar desde el nodo superior-izquierdo del trellis hasta cada nodo (i, j) con coste 0.
2. Modifica el bucle de borrado para calcular las puntuaciones de la primera columna. Las aristas que llegan al nodo llegan del nodo inicial y del nodo de la fila anterior.

```

costes <- c(Peor, Peor, Peor, #COMPLETAR)
names(costes) <- c("MoS", "I", "B", "Ini")

#borrado de u[i]

```

```

for (i in 2:m)
{
  costes["B"] <- #COMPLETAR

  #asignacion de coste
  trellis[i,1] <- decisionf(costes)

  #preparacion del backtracking
  operacion <- names(which(costes==trellis[i,1])[1])
  if (operacion=="B")
    {backi[i,1] <- i-1; backj[i,1] <- 1;}
  else if (operacion=="Ini")
    {backi[i,1] <- 1; backj[i,1] <- 1;}
  else warning("Operacion no conocida en trellis")
}

```

3. Modifica el bucle de inserción para calcular las puntuaciones de la primera fila. Las aristas que llegan al nodo llegan del nodo inicial y del nodo de la columna anterior.

```

costes <- c(Peor,Peor,Peor,#COMPLETAR)
names(costes) <- c("MoS", "I", "B","Ini")

#insercion de v[j]
for (j in 2:n)
{
  costes["I"] <- #COMPLETAR

  #asignacion de coste
  trellis[1,j] <- decisionf(costes)

  #preparacion del backtracking
  operacion <- names(which(costes==trellis[1,j])[1])
  if (operacion=="B")
    {backi[1,j] <- 1; backj[1,j] <- j-1;}
  else if (operacion=="Ini")
    {backi[1,j] <- 1; backj[1,j] <- 1;}
  else warning("Operacion no conocida en trellis")
}

```

4. Completa el código necesario para calcular la puntuación del borrado ("B"), la inserción (I), la sustitución ("MoS") y el comienzo de una nueva subsecuencia (Ini) producida en el caso general i, j .

```

costes <- c(Peor,Peor,Peor, #COMPLETAR )
names(costes) <- c("MoS", "I", "B","Ini")

#caso general

```

```

for (i in 2:m)
{
  for (j in 2:n)
  {
    #borrado de u[i]
    costes["B"] <- #COMPLETAR

    #insercion de v[j]
    costes["I"] <- #COMPLETAR

    #sustitucion o match u[i]==v[j]
    costes["MoS"] <- #COMPLETAR

    #asignacion de coste
    trellis[i,j] <- decisionf(costes)

    #preparacion del backtracking
    operacion <- names(which(costes==trellis[i,j])[1])
    if (operacion=="B")
      {backi[i,j] <- i-1; backj[i,j] <- j;}
    else if (operacion=="I")
      {backi[i,j] <- i; backj[i,j] <- j-1;}
    else if (operacion=="MoS")
      {backi[i,j] <- i-1; backj[i,j] <- j-1;}
    else if (operacion=="Ini")
      {backi[i,j] <- 1; backj[i,j] <- 1;}
    else warning("Operacion no conocida en trellis")

  } #for j
} #for i

```

6. Ejercicio. Comparación de algoritmos

6.1. Objetivo

Analizar los algoritmos implementados en los ejercicios anteriores para el alineamiento pares de secuencias y comparar las trazas producidas.

6.2. Notas

- Utiliza la función BackTracking para recuperar el mejor camino obtenido por el algoritmo Needleman-Wunsch
- Utiliza la función BackTrackingLocal para recuperar el mejor camino obtenido por el algoritmo SmithWaterman

6.3. Desarrollo

Ejecuta y analiza las trazas que se muestran a continuación.

```

> m <- 1
> s <- -1
> b <- -1
> i <- -1
> delta <- matrix(c(m, s, s, s, b, s, m, s, s, b, s, s, m, s, b,
+ s, s, s, m, b, i, i, i, i, 0), dimnames = list(c("A", "T",
+ "C", "G", "-"), c("A", "T", "C", "G", "-")), nrow = 5, ncol = 5,
+ byrow = TRUE)
> seq1 <- "ATATTTATCG"
> seq2 <- "ATTTATCG"
> seq3 <- "ATAAAATCTATCG"
> seq4 <- "AAAAAAAAAATATTTATCG"
> seq5 <- "CCATATTTATCGAAAAAAAA"
> seq6 <- "CCATATTTTTATCGAAAAAAAA"
> seq7 <- "AAAAAAAAAATATTTCTATCG"
> print(seq1)

[1] "ATATTTATCG"

> print(seq1)

[1] "ATATTTATCG"

> t11nw <- NeedlemanWunsch(seq1, seq1, delta)

  - A T A T T T A T C G
-  0 -1 -2 -3 -4 -5 -6 -7 -8 -9 -10
A -1  1  0 -1 -2 -3 -4 -5 -6 -7 -8
T -2  0  2  1  0 -1 -2 -3 -4 -5 -6
A -3 -1  1  3  2  1  0 -1 -2 -3 -4
T -4 -2  0  2  4  3  2  1  0 -1 -2
T -5 -3 -1  1  3  5  4  3  2  1  0
T -6 -4 -2  0  2  4  6  5  4  3  2
A -7 -5 -3 -1  1  3  5  7  6  5  4
T -8 -6 -4 -2  0  2  4  6  8  7  6
C -9 -7 -5 -3 -1  1  3  5  7  9  8
G -10 -8 -6 -4 -2  0  2  4  6  8  10
10,10 <-          9,9:          10 MoS          G G
9,9 <-          8,8:          9 MoS          C C
8,8 <-          7,7:          8 MoS          T T
7,7 <-          6,6:          7 MoS          A A
6,6 <-          5,5:          6 MoS          T T
5,5 <-          4,4:          5 MoS          T T
4,4 <-          3,3:          4 MoS          T T
3,3 <-          2,2:          3 MoS          A A
2,2 <-          1,1:          2 MoS          T T
1,1 <-          0,0:          1 MoS          A A

> t11sw <- SmithWaterman(seq1, seq1, delta)

  - A T A T T T A T C G
-  0 0 0 0 0 0 0 0 0 0 0

```

```

A 0 1 0 1 0 0 0 1 0 0 0
T 0 0 2 1 2 1 1 0 2 1 0
A 0 1 1 3 2 1 0 2 1 1 0
T 0 0 2 2 4 3 2 1 3 2 1
T 0 0 1 1 3 5 4 3 2 2 1
T 0 0 1 0 2 4 6 5 4 3 2
A 0 1 0 2 1 3 5 7 6 5 4
T 0 0 2 1 3 2 4 6 8 7 6
C 0 0 1 1 2 2 3 5 7 9 8
G 0 0 0 0 1 1 2 4 6 8 10
10,10 <-          9,9:          10 MoS          G G
9,9 <-          8,8:          9 MoS          C C
8,8 <-          7,7:          8 MoS          T T
7,7 <-          6,6:          7 MoS          A A
6,6 <-          5,5:          6 MoS          T T
5,5 <-          4,4:          5 MoS          T T
4,4 <-          3,3:          4 MoS          T T
3,3 <-          2,2:          3 MoS          A A
2,2 <-          1,1:          2 MoS          T T
1,1 <-          0,0:          1 MoS          A A

> print(seq1)

[1] "ATATTTATCG"

> print(seq2)

[1] "ATTTATCG"

> t12nw <- NeedlemanWunsch(seq1, seq2, delta)

  -  A  T  T  T  A  T  C  G
-   0 -1 -2 -3 -4 -5 -6 -7 -8
A  -1  1  0 -1 -2 -3 -4 -5 -6
T  -2  0  2  1  0 -1 -2 -3 -4
A  -3 -1  1  1  0  1  0 -1 -2
T  -4 -2  0  2  2  1  2  1  0
T  -5 -3 -1  1  3  2  2  1  0
T  -6 -4 -2  0  2  2  3  2  1
A  -7 -5 -3 -1  1  3  2  2  1
T  -8 -6 -4 -2  0  2  4  3  2
C  -9 -7 -5 -3 -1  1  3  5  4
G -10 -8 -6 -4 -2  0  2  4  6
10,8 <-          9,7:          6 MoS          G G
9,7 <-          8,6:          5 MoS          C C
8,6 <-          7,5:          4 MoS          T T
7,5 <-          6,4:          3 MoS          A A
6,4 <-          5,3:          2 MoS          T T
5,3 <-          4,2:          1 MoS          T T
4,2 <-          3,1:          0 MoS          T T
3,1 <-          2,0:         -1 MoS          A A
2,0 <-          1,0:         -2 B          T -
1,0 <-          0,0:         -1 B          A -

```



```

> t12sw <- SmithWaterman(seq1, seq2, delta)

  - A T T T A T C G
- 0 0 0 0 0 0 0 0 0
A 0 1 0 0 0 1 0 0 0
T 0 0 2 1 1 0 2 1 0
A 0 1 1 1 0 2 1 1 0
T 0 0 2 2 2 1 3 2 1
T 0 0 1 3 3 2 2 2 1
T 0 0 1 2 4 3 3 2 1
A 0 1 0 1 3 5 4 3 2
T 0 0 2 1 2 4 6 5 4
C 0 0 1 1 1 3 5 7 6
G 0 0 0 0 0 2 4 6 8
10,8 <-          9,7:          8 MoS          G G
9,7 <-          8,6:          7 MoS          C C
8,6 <-          7,5:          6 MoS          T T
7,5 <-          6,4:          5 MoS          A A
6,4 <-          5,3:          4 MoS          T T
5,3 <-          4,2:          3 MoS          T T
4,2 <-          3,1:          2 MoS          T T
3,1 <-          2,0:          1 MoS          A A

> print(seq1)

[1] "ATATTTATCG"

> print(seq3)

[1] "ATAAAATCTATCG"

> t13nw <- NeedlemanWunsch(seq1, seq3, delta)

  - A T A A A A T C T A T C G
- 0 -1 -2 -3 -4 -5 -6 -7 -8 -9 -10 -11 -12 -13
A -1 1 0 -1 -2 -3 -4 -5 -6 -7 -8 -9 -10 -11
T -2 0 2 1 0 -1 -2 -3 -4 -5 -6 -7 -8 -9
A -3 -1 1 3 2 1 0 -1 -2 -3 -4 -5 -6 -7
T -4 -2 0 2 2 1 0 1 0 -1 -2 -3 -4 -5
T -5 -3 -1 1 1 1 0 1 0 1 0 -1 -2 -3
T -6 -4 -2 0 0 0 0 1 0 1 0 1 0 -1
A -7 -5 -3 -1 1 1 1 0 0 0 2 1 0 -1
T -8 -6 -4 -2 0 0 0 2 1 1 1 3 2 1
C -9 -7 -5 -3 -1 -1 -1 1 3 2 1 2 4 3
G -10 -8 -6 -4 -2 -2 -2 0 2 2 1 1 3 5
10,13 <-          9,12:          5 MoS          G G
9,12 <-          8,11:          4 MoS          C C
8,11 <-          7,10:          3 MoS          T T
7,10 <-          6,9:          2 MoS          A A
6,9 <-          5,8:          1 MoS          T T
5,8 <-          4,7:          0 MoS          T C

```

```

4,7 <-          3,6:          1 MoS          T T
3,6 <-          2,5:          0 MoS          A A
2,5 <-          2,4:          -1 I           - A
2,4 <-          2,3:          0 I           - A
2,3 <-          2,2:          1 I           - A
2,2 <-          1,1:          2 MoS          T T
1,1 <-          0,0:          1 MoS          A A

```

```
> t13sw <- SmithWaterman(seq1, seq3, delta)
```

```

- A T A A A A T C T A T C G
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0
A 0 1 0 1 1 1 1 0 0 0 1 0 0 0
T 0 0 2 1 0 0 0 2 1 1 0 2 1 0
A 0 1 1 3 2 1 1 1 1 0 2 1 1 0
T 0 0 2 2 2 1 0 2 1 2 1 3 2 1
T 0 0 1 1 1 1 0 1 1 2 1 2 2 1
T 0 0 1 0 0 0 0 1 0 2 1 2 1 1
A 0 1 0 2 1 1 1 0 0 1 3 2 1 0
T 0 0 2 1 1 0 0 2 1 1 2 4 3 2
C 0 0 1 1 0 0 0 1 3 2 1 3 5 4
G 0 0 0 0 0 0 0 0 2 2 1 2 4 6
10,13 <-          9,12:          6 MoS          G G
9,12 <-          8,11:          5 MoS          C C
8,11 <-          7,10:          4 MoS          T T
7,10 <-          6,9:          3 MoS          A A
6,9 <-          5,8:          2 MoS          T T
5,8 <-          4,7:          1 MoS          T C
4,7 <-          3,6:          2 MoS          T T
3,6 <-          2,5:          1 MoS          A A
2,5 <-          1,4:          0 MoS          T A
1,4 <-          0,3:          1 MoS          A A

```

```
> print(seq4)
```

```
[1] "AAAAAAAAAATATTTATCG"
```

```
> print(seq5)
```

```
[1] "CCATATTTATCGAAAAAAA"
```

```
> t45nw <- NeedlemanWunsch(seq4, seq5, delta)
```

```

-   C   C   A   T   A   T   T   T   A   T   C   G   A   A   A   A   A   A   A
-   0  -1  -2  -3  -4  -5  -6  -7  -8  -9 -10 -11 -12 -13 -14 -15 -16 -17 -18 -19
A  -1  -1  -2  -1  -2  -3  -4  -5  -6  -7  -8  -9 -10 -11 -12 -13 -14 -15 -16 -17
A  -2  -2  -2  -1  -2  -1  -2  -3  -4  -5  -6  -7  -8  -9 -10 -11 -12 -13 -14 -15
A  -3  -3  -3  -1  -2  -1  -2  -3  -4  -3  -4  -5  -6  -7  -8  -9 -10 -11 -12 -13
A  -4  -4  -4  -2  -2  -1  -2  -3  -4  -3  -4  -5  -6  -5  -6  -7  -8  -9 -10 -11
A  -5  -5  -5  -3  -3  -1  -2  -3  -4  -3  -4  -5  -6  -5  -4  -5  -6  -7  -8  -9
A  -6  -6  -6  -4  -4  -2  -2  -3  -4  -3  -4  -5  -6  -5  -4  -3  -4  -5  -6  -7

```

A	-7	-7	-7	-5	-5	-3	-3	-3	-4	-3	-4	-5	-6	-5	-4	-3	-2	-3	-4	-5
A	-8	-8	-8	-6	-6	-4	-4	-4	-4	-3	-4	-5	-6	-5	-4	-3	-2	-1	-2	-3
A	-9	-9	-9	-7	-7	-5	-5	-5	-5	-3	-4	-5	-6	-5	-4	-3	-2	-1	0	-1
A	-10	-10	-10	-8	-8	-6	-6	-6	-6	-4	-4	-5	-6	-5	-4	-3	-2	-1	0	1
A	-11	-11	-11	-9	-9	-7	-7	-7	-7	-5	-5	-5	-6	-5	-4	-3	-2	-1	0	1
T	-12	-12	-12	-10	-8	-8	-6	-6	-6	-6	-4	-5	-6	-6	-5	-4	-3	-2	-1	0
A	-13	-13	-13	-11	-9	-7	-7	-7	-7	-5	-5	-5	-6	-5	-5	-4	-3	-2	-1	0
T	-14	-14	-14	-12	-10	-8	-6	-6	-6	-6	-4	-5	-6	-6	-6	-5	-4	-3	-2	-1
T	-15	-15	-15	-13	-11	-9	-7	-5	-5	-6	-5	-5	-6	-7	-7	-6	-5	-4	-3	-2
T	-16	-16	-16	-14	-12	-10	-8	-6	-4	-5	-5	-6	-6	-7	-8	-7	-6	-5	-4	-3
A	-17	-17	-17	-15	-13	-11	-9	-7	-5	-3	-4	-5	-6	-5	-6	-7	-6	-5	-4	-3
T	-18	-18	-18	-16	-14	-12	-10	-8	-6	-4	-2	-3	-4	-5	-6	-7	-7	-6	-5	-4
C	-19	-17	-17	-17	-15	-13	-11	-9	-7	-5	-3	-1	-2	-3	-4	-5	-6	-7	-6	-5
G	-20	-18	-18	-18	-16	-14	-12	-10	-8	-6	-4	-2	0	-1	-2	-3	-4	-5	-6	-6
A																				
-	-20																			
A	-18																			
A	-16																			
A	-14																			
A	-12																			
A	-10																			
A	-8																			
A	-6																			
A	-4																			
A	-2																			
A	0																			
A	2																			
T	1																			
A	1																			
T	0																			
T	-1																			
T	-2																			
A	-2																			
T	-3																			
C	-4																			
G	-5																			
20,20	<-	19,20:	-5	B	G	-														
19,20	<-	18,20:	-4	B	C	-														
18,20	<-	17,20:	-3	B	T	-														
17,20	<-	16,19:	-2	MoS	A	A														
16,19	<-	15,19:	-3	B	T	-														
15,19	<-	14,19:	-2	B	T	-														
14,19	<-	13,19:	-1	B	T	-														
13,19	<-	12,18:	0	MoS	A	A														
12,18	<-	11,18:	-1	B	T	-														
11,18	<-	10,17:	0	MoS	A	A														
10,17	<-	9,16:	-1	MoS	A	A														
9,16	<-	8,15:	-2	MoS	A	A														
8,15	<-	7,14:	-3	MoS	A	A														
7,14	<-	6,13:	-4	MoS	A	A														

```
> print(seq5)
```

```
[1] "CCATATTTATCGAAAAAAAA"
```

```
> print(seq5)
```

```
[1] "CCATATTTATCGAAAAAAAA"
```

```
> t55nw <- NeedlemanWunsch(seq5, seq5, delta)
```

	-	C	C	A	T	A	T	T	T	A	T	C	G	A	A	A	A	A	A	A
-	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19
C	-1	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17
C	-2	0	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15
A	-3	-1	1	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13
T	-4	-2	0	2	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11
A	-5	-3	-1	1	3	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
T	-6	-4	-2	0	2	4	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
T	-7	-5	-3	-1	1	3	5	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5
T	-8	-6	-4	-2	0	2	4	6	8	7	6	5	4	3	2	1	0	-1	-2	-3
A	-9	-7	-5	-3	-1	1	3	5	7	9	8	7	6	5	4	3	2	1	0	-1
T	-10	-8	-6	-4	-2	0	2	4	6	8	10	9	8	7	6	5	4	3	2	1
C	-11	-9	-7	-5	-3	-1	1	3	5	7	9	11	10	9	8	7	6	5	4	3
G	-12	-10	-8	-6	-4	-2	0	2	4	6	8	10	12	11	10	9	8	7	6	5
A	-13	-11	-9	-7	-5	-3	-1	1	3	5	7	9	11	13	12	11	10	9	8	7
A	-14	-12	-10	-8	-6	-4	-2	0	2	4	6	8	10	12	14	13	12	11	10	9
A	-15	-13	-11	-9	-7	-5	-3	-1	1	3	5	7	9	11	13	15	14	13	12	11
A	-16	-14	-12	-10	-8	-6	-4	-2	0	2	4	6	8	10	12	14	16	15	14	13
A	-17	-15	-13	-11	-9	-7	-5	-3	-1	1	3	5	7	9	11	13	15	17	16	15
A	-18	-16	-14	-12	-10	-8	-6	-4	-2	0	2	4	6	8	10	12	14	16	18	17
A	-19	-17	-15	-13	-11	-9	-7	-5	-3	-1	1	3	5	7	9	11	13	15	17	19
A	-20	-18	-16	-14	-12	-10	-8	-6	-4	-2	0	2	4	6	8	10	12	14	16	18
A																				
-	-20																			
C	-18																			
C	-16																			
A	-14																			
T	-12																			
A	-10																			
T	-8																			
T	-6																			
T	-4																			
A	-2																			
T	0																			
C	2																			
G	4																			
A	6																			
A	8																			
A	10																			
A	12																			
A	14																			
A	16																			
A	18																			

```

A 20
20,20 <-      19,19:      20 MoS      A A
19,19 <-      18,18:      19 MoS      A A
18,18 <-      17,17:      18 MoS      A A
17,17 <-      16,16:      17 MoS      A A
16,16 <-      15,15:      16 MoS      A A
15,15 <-      14,14:      15 MoS      A A
14,14 <-      13,13:      14 MoS      A A
13,13 <-      12,12:      13 MoS      A A
12,12 <-      11,11:      12 MoS      G G
11,11 <-      10,10:      11 MoS      C C
10,10 <-      9,9:      10 MoS      T T
9,9 <-      8,8:      9 MoS      A A
8,8 <-      7,7:      8 MoS      T T
7,7 <-      6,6:      7 MoS      T T
6,6 <-      5,5:      6 MoS      T T
5,5 <-      4,4:      5 MoS      A A
4,4 <-      3,3:      4 MoS      T T
3,3 <-      2,2:      3 MoS      A A
2,2 <-      1,1:      2 MoS      C C
1,1 <-      0,0:      1 MoS      C C

> t55sw <- SmithWaterman(seq5, seq5, delta)

- C C A T A T T T A T C G A A A A A A A A
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
C 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
C 0 1 2 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
A 0 0 1 3 2 1 0 0 0 1 0 0 0 1 1 1 1 1 1 1
T 0 0 0 2 4 3 2 1 1 0 2 1 0 0 0 0 0 0 0 0
A 0 0 0 1 3 5 4 3 2 2 1 1 0 1 1 1 1 1 1 1
T 0 0 0 0 2 4 6 5 4 3 3 2 1 0 0 0 0 0 0 0
T 0 0 0 0 1 3 5 7 6 5 4 3 2 1 0 0 0 0 0 0
T 0 0 0 0 1 2 4 6 8 7 6 5 4 3 2 1 0 0 0 0
A 0 0 0 1 0 2 3 5 7 9 8 7 6 5 4 3 2 1 1 1
T 0 0 0 0 2 1 3 4 6 8 10 9 8 7 6 5 4 3 2 1 0
C 0 1 1 0 1 1 2 3 5 7 9 11 10 9 8 7 6 5 4 3 2
G 0 0 0 0 0 0 1 2 4 6 8 10 12 11 10 9 8 7 6 5 4
A 0 0 0 1 0 1 0 1 3 5 7 9 11 13 12 11 10 9 8 7 6
A 0 0 0 1 0 1 0 0 2 4 6 8 10 12 14 13 12 11 10 9 8
A 0 0 0 1 0 1 0 0 1 3 5 7 9 11 13 15 14 13 12 11 10
A 0 0 0 1 0 1 0 0 0 2 4 6 8 10 12 14 16 15 14 13 12
A 0 0 0 1 0 1 0 0 0 1 3 5 7 9 11 13 15 17 16 15 14
A 0 0 0 1 0 1 0 0 0 1 2 4 6 8 10 12 14 16 18 17 16
A 0 0 0 1 0 1 0 0 0 1 1 3 5 7 9 11 13 15 17 19 18
A 0 0 0 1 0 1 0 0 0 1 0 2 4 6 8 10 12 14 16 18 20
20,20 <-      19,19:      20 MoS      A A
19,19 <-      18,18:      19 MoS      A A
18,18 <-      17,17:      18 MoS      A A
17,17 <-      16,16:      17 MoS      A A

```

```

16,16 <-          15,15:          16 MoS          A A
15,15 <-          14,14:          15 MoS          A A
14,14 <-          13,13:          14 MoS          A A
13,13 <-          12,12:          13 MoS          A A
12,12 <-          11,11:          12 MoS          G G
11,11 <-          10,10:          11 MoS          C C
10,10 <-          9,9:           10 MoS          T T
9,9 <-           8,8:           9 MoS          A A
8,8 <-           7,7:           8 MoS          T T
7,7 <-           6,6:           7 MoS          T T
6,6 <-           5,5:           6 MoS          T T
5,5 <-           4,4:           5 MoS          A A
4,4 <-           3,3:           4 MoS          T T
3,3 <-           2,2:           3 MoS          A A
2,2 <-           1,1:           2 MoS          C C
1,1 <-           0,0:           1 MoS          C C

```

```
> print(seq6)
```

```
[1] "CCATATTTTTATCGAAAAAAAA"
```

```
> print(seq7)
```

```
[1] "AAAAAAAAAATATTTCTATCG"
```

```
> t67nw <- NeedlemanWunsch(seq6, seq7, delta)
```

	-	A	A	A	A	A	A	A	A	A	A	A	T	A	T	T	T	C	T	A
-	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19
C	-1	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-15	-16	-17
C	-2	-2	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-15	-16	-17
A	-3	-1	-1	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-15
T	-4	-2	-2	-2	-2	-3	-4	-5	-6	-7	-8	-9	-8	-9	-10	-11	-12	-13	-14	-15
A	-5	-3	-1	-1	-1	-1	-2	-3	-4	-5	-6	-7	-8	-7	-8	-9	-10	-11	-12	-13
T	-6	-4	-2	-2	-2	-2	-2	-3	-4	-5	-6	-7	-6	-7	-6	-7	-8	-9	-10	-11
T	-7	-5	-3	-3	-3	-3	-3	-3	-4	-5	-6	-7	-6	-7	-6	-5	-6	-7	-8	-9
T	-8	-6	-4	-4	-4	-4	-4	-4	-4	-5	-6	-7	-6	-7	-6	-5	-4	-5	-6	-7
T	-9	-7	-5	-5	-5	-5	-5	-5	-5	-5	-6	-7	-6	-7	-6	-5	-4	-5	-4	-5
T	-10	-8	-6	-6	-6	-6	-6	-6	-6	-6	-6	-7	-6	-7	-6	-5	-4	-5	-4	-5
A	-11	-9	-7	-5	-5	-5	-5	-5	-5	-5	-5	-5	-6	-5	-6	-6	-5	-5	-5	-3
T	-12	-10	-8	-6	-6	-6	-6	-6	-6	-6	-6	-6	-4	-5	-4	-5	-5	-6	-4	-4
C	-13	-11	-9	-7	-7	-7	-7	-7	-7	-7	-7	-7	-5	-5	-5	-5	-6	-4	-5	-5
G	-14	-12	-10	-8	-8	-8	-8	-8	-8	-8	-8	-8	-6	-6	-6	-6	-6	-5	-5	-6
A	-15	-13	-11	-9	-7	-7	-7	-7	-7	-7	-7	-7	-7	-5	-6	-7	-7	-6	-6	-4
A	-16	-14	-12	-10	-8	-6	-6	-6	-6	-6	-6	-6	-7	-6	-6	-7	-8	-7	-7	-5
A	-17	-15	-13	-11	-9	-7	-5	-5	-5	-5	-5	-5	-6	-6	-7	-7	-8	-8	-8	-6
A	-18	-16	-14	-12	-10	-8	-6	-4	-4	-4	-4	-4	-5	-5	-6	-7	-8	-9	-9	-7
A	-19	-17	-15	-13	-11	-9	-7	-5	-3	-3	-3	-3	-4	-4	-5	-6	-7	-8	-9	-8
A	-20	-18	-16	-14	-12	-10	-8	-6	-4	-2	-2	-2	-3	-3	-4	-5	-6	-7	-8	-8
A	-21	-19	-17	-15	-13	-11	-9	-7	-5	-3	-1	-1	-2	-2	-3	-4	-5	-6	-7	-7
A	-22	-20	-18	-16	-14	-12	-10	-8	-6	-4	-2	0	-1	-1	-2	-3	-4	-5	-6	-6

	T	C	G				
-	-20	-21	-22				
C	-18	-19	-20				
C	-18	-17	-18				
A	-16	-17	-18				
T	-14	-15	-16				
A	-14	-15	-16				
T	-12	-13	-14				
T	-10	-11	-12				
T	-8	-9	-10				
T	-6	-7	-8				
T	-4	-5	-6				
A	-4	-5	-6				
T	-2	-3	-4				
C	-3	-1	-2				
G	-4	-2	0				
A	-5	-3	-1				
A	-5	-4	-2				
A	-6	-5	-3				
A	-7	-6	-4				
A	-8	-7	-5				
A	-9	-8	-6				
A	-8	-9	-7				
A	-7	-8	-8				
22,22	<-			21,22:	-8 B	A	-
21,22	<-			20,22:	-7 B	A	-
20,22	<-			19,22:	-6 B	A	-
19,22	<-			18,22:	-5 B	A	-
18,22	<-			17,22:	-4 B	A	-
17,22	<-			16,22:	-3 B	A	-
16,22	<-			15,22:	-2 B	A	-
15,22	<-			14,22:	-1 B	A	-
14,22	<-			13,21:	0 MoS	G	G
13,21	<-			12,20:	-1 MoS	C	C
12,20	<-			11,19:	-2 MoS	T	T
11,19	<-			10,18:	-3 MoS	A	A
10,18	<-			9,17:	-4 MoS	T	T
9,17	<-			8,16:	-5 MoS	T	C
8,16	<-			7,15:	-4 MoS	T	T
7,15	<-			6,14:	-5 MoS	T	T
6,14	<-			5,13:	-6 MoS	T	T
5,13	<-			4,12:	-7 MoS	A	A
4,12	<-			3,11:	-8 MoS	T	T
3,11	<-			2,10:	-9 MoS	A	A
2,10	<-			1,9:	-10 MoS	C	A
1,9	<-			0,8:	-9 MoS	C	A
0,8	<-			0,7:	-8 I	-	A
0,7	<-			0,6:	-7 I	-	A
0,6	<-			0,5:	-6 I	-	A
0,5	<-			0,4:	-5 I	-	A


```

0,4 <-          0,3:          -4 I          - A
0,3 <-          0,2:          -3 I          - A
0,2 <-          0,1:          -2 I          - A
0,1 <-          0,0:          -1 I          - A

```

```
> t67sw <- SmithWaterman(seq6, seq7, delta)
```

```

- A A A A A A A A A A A T A T T T C T A T C G
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0
C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0
A 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 0 0 0 1 0 0 0
T 0 0 0 0 0 0 0 0 0 0 0 0 0 2 1 2 1 1 0 1 0 2 1 0
A 0 1 1 1 1 1 1 1 1 1 1 1 1 3 2 1 0 0 0 2 1 1 0
T 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 4 3 2 1 1 1 3 2 1
T 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 3 5 4 3 2 1 2 2 1
T 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 2 4 6 5 4 3 2 1 1
T 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 3 5 5 6 5 4 3 2
T 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 2 4 4 6 5 6 5 4
A 0 1 1 1 1 1 1 1 1 1 1 1 1 0 2 1 1 3 3 5 7 6 5 4
T 0 0 0 0 0 0 0 0 0 0 0 0 0 2 1 3 2 2 2 4 6 8 7 6
C 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 2 2 1 3 3 5 7 9 8
G 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 2 2 4 6 8 10
A 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 0 1 1 3 5 7 9
A 0 1 2 2 2 2 2 2 2 2 2 2 1 1 0 0 0 0 0 2 4 6 8
A 0 1 2 3 3 3 3 3 3 3 3 3 2 2 1 0 0 0 0 1 3 5 7
A 0 1 2 3 4 4 4 4 4 4 4 4 3 3 2 1 0 0 0 1 2 4 6
A 0 1 2 3 4 5 5 5 5 5 5 5 4 4 3 2 1 0 0 1 1 3 5
A 0 1 2 3 4 5 6 6 6 6 6 6 5 5 4 3 2 1 0 1 0 2 4
A 0 1 2 3 4 5 6 7 7 7 7 7 6 6 5 4 3 2 1 1 0 1 3
A 0 1 2 3 4 5 6 7 8 8 8 8 7 7 6 5 4 3 2 2 1 0 2
14,22 <-          13,21:          10 MoS          G G
13,21 <-          12,20:          9 MoS          C C
12,20 <-          11,19:          8 MoS          T T
11,19 <-          10,18:          7 MoS          A A
10,18 <-          9,17:          6 MoS          T T
9,17 <-          8,16:          5 MoS          T C
8,16 <-          7,15:          6 MoS          T T
7,15 <-          6,14:          5 MoS          T T
6,14 <-          5,13:          4 MoS          T T
5,13 <-          4,12:          3 MoS          A A
4,12 <-          3,11:          2 MoS          T T
3,11 <-          2,10:          1 MoS          A A
2,10 <-          0,0:          0 Ini

```