12. Übung

Einführung in die Bioinformatik I, 1. Teil
Wintersemester 2019/2020

Aufgabe 1 (6 Punkte): Berechnen Sie mit Hilfe einer DP-Matrix alle optimalen globalen Alignments (mit Einheitskosten) von `GTGCACA` und `TCACTTA`. Wie viele sind es?

	ε	G	T	G	C	A	C	A
ε								
T								
C								
A								
C								
T								
T								
A								

Aufgabe 1 (6 Punkte): Berechnen Sie mit Hilfe einer DP-Matrix alle optimalen globalen Alignments (mit Einheitskosten) von `GTGCACA` und `TCACTTA`. Wie viele sind es?

	ε	G	T	G	C	A	C	A
ε	0	1	2	3	4	5	6	7
T	1	1	1	2	3	4	5	6
C	2	2	2	2	2	3	4	5
A	3	3	3	3	3	2	3	4
C	4	4	4	4	3	3	2	3
T	5	5	4	5	4	4	3	3
T	6	6	5	5	5	5	4	4
A	7	7	6	6	6	5	5	**4**

```
GTGCAC--A
-T-CACTTA
```

Aufgabe 2 (6 Punkte): Die vier organischen Basen Adenin, Cytosin, Guanin und Thymin der DNA lassen sich in sogenannte *Purine* (Adenin und Guanin) und *Pyrimidine* (Cytosin und Thymin) unterteilen. Bei Mutationen im genetischen Code ist es wahrscheinlicher, dass eine Purinbase duch eine andere oder eine Pyrimidinbase durch eine andere substituiert wird (*Transition*) als dass eine Purinbase duch eine Pyrimidinbase ersetzt wird oder umgekehrt (*Transversion*).

Diese Tatsache berücksichtigen wir in der folgenden Kostenfunktion δ:

$$\delta(a,b) = \begin{cases} 0 & a = b \\ 1 & a,b \in \{\texttt{A},\texttt{G}\} \text{ und } a \neq b \\ & a,b \in \{\texttt{C},\texttt{T}\} \text{ und } a \neq b \\ 2 & \textit{sonst} \end{cases}$$

Berechnen Sie die DP-Matrix von `GTGCACA` und `TCACTTA` unter Verwendung dieser Kostenfunktion, und geben Sie wie oben alle optimalen Alignments an.

Hinweis: Wenn keine Einheitskosten verwendet werden, gilt für die DP-Matrix:

$$D[i,0] = \sum_{k=1}^{i} \delta(u_k, \texttt{-}) \qquad \text{und} \qquad D[0,j] = \sum_{k=1}^{j} \delta(\texttt{-}, v_k)$$

	ε	G	T	G	C	A	C	A
ε								
T								
C								
A								
C								
T								
T								
A								

Aufgabe 2 (6 Punkte): Die vier organischen Basen Adenin, Cytosin, Guanin und Thymin der DNA lassen sich in sogenannte *Purine* (Adenin und Guanin) und *Pyrimidine* (Cytosin und Thymin) unterteilen. Bei Mutationen im genetischen Code ist es wahrscheinlicher, dass eine Purinbase duch eine andere oder eine Pyrimidinbase durch eine andere substituiert wird (*Transition*) als dass eine Purinbase duch eine Pyrimidinbase ersetzt wird oder umgekehrt (*Transversion*).

Diese Tatsache berücksichtigen wir in der folgenden Kostenfunktion δ:

$$\delta(a,b) = \begin{cases} 0 & a = b \\ 1 & a,b \in \{\texttt{A},\texttt{G}\} \text{ und } a \neq b \\ & a,b \in \{\texttt{C},\texttt{T}\} \text{ und } a \neq b \\ 2 & \textit{sonst} \end{cases}$$

Berechnen Sie die DP-Matrix von `GTGCACA` und `TCACTTA` unter Verwendung dieser Kostenfunktion, und geben Sie wie oben alle optimalen Alignments an.

Hinweis: Wenn keine Einheitskosten verwendet werden, gilt für die DP-Matrix:

$$D[i,0] = \sum_{k=1}^{i} \delta(u_k, \texttt{-}) \qquad \text{und} \qquad D[0,j] = \sum_{k=1}^{j} \delta(\texttt{-}, v_k)$$

	Ɛ	G	T	G	C	A	C	A
Ɛ	0	2	4	6	8	10	12	14
T	2	2	2	4	6	8	10	12
C	4	4	3	4	4	6	8	10
A	6	5	5	4	6	4	6	8
C	8	7	6	6	4	6	4	6
T	10	9	7	8	6	6	6	6
T	12	11	9	9	8	8	7	8
A	14	13	11	10	10	8	9	**7**

GTGCACA
TCACTTA

Aufgabe 3 (2 Punkte): Im Verlaufe der Evolution haben sich Mensch und Maus aus einem gemeinsamen Vorfahren entwickelt. Daher wäre es praktischer, Gene von Mensch und Maus nicht miteinander, sondern mit denen des gemeinsamen Vorfahren zu alignieren. Warum vergleichen wir nicht gegen den Vorfahren? Warum ergibt es trotzdem Sinn, Gene von Mensch und Maus miteinander zu alignieren?

Aufgabe 3 (2 Punkte): Im Verlaufe der Evolution haben sich Mensch und Maus aus einem gemeinsamen Vorfahren entwickelt. Daher wäre es praktischer, Gene von Mensch und Maus nicht miteinander, sondern mit denen des gemeinsamen Vorfahren zu alignieren. Warum vergleichen wir nicht gegen den Vorfahren? Warum ergibt es trotzdem Sinn, Gene von Mensch und Maus miteinander zu alignieren?

Der gemeinsame Vorfahre von Mensch und Maus (und dessen Genom) ist unbekannt.
Wenn wir davon ausgehen, dass die Evolution möglichst „sparsam“ abläuft (Maximum-Parsimony-Ansatz), sich also in der Evolution jede Funktionalität nur einmal entwickelt hat, können wir durch den Vergleich genetischen Codes von Mensch und Maus auf den gemeinsamen Vorfahren schließen: Gleicher oder ähnlicher genetischer Code weist auf gleiche oder ähnliche Funktionalität hin, die bereits der gemeinsame Vorfahr besessen haben muss.

Bonus (6 Punkte): Berechnen Sie mit der folgenden Rekurrenz die Anzahl aller globalen Alignments zweier Strings u, v für alle Längen von u und v bis einschließlich vier:

$$N_{\mathcal{A}}[i,0] = N_{\mathcal{A}}[0,j] = 1 \qquad N_{\mathcal{A}}[i,j] = N_{\mathcal{A}}[i-1,j] + N_{\mathcal{A}}[i,j-1] + N_{\mathcal{A}}[i-1,j-1]$$

Wieso funktioniert diese Rekurrenz?

N_A	0	1	2	3	4
0					
1					
2					
3					
4					

Bonus (6 Punkte): Berechnen Sie mit der folgenden Rekurrenz die Anzahl aller globalen Alignments zweier Strings u, v für alle Längen von u und v bis einschließlich vier:

$$N_A[i,0] = N_A[0,j] = 1 \qquad N_A[i,j] = N_A[i-1,j] + N_A[i,j-1] + N_A[i-1,j-1]$$

Wieso funktioniert diese Rekurrenz?

N_A	0	1	2	3	4
0	1	1	1	1	1
1	1	3	5	7	9
2	1	5	13	25	41
3	1	7	25	63	129
4	1	9	41	129	321

Dieses Verfahren funktioniert, indem es immer auf die Anzahl von Alignments kürzerer Strings zurückgreift.
Die Ränder sind klar, denn wenn der eine String Länge 0 hat, kann man den anderen nur mit Gaps alignieren.

Für zwei Strings u, v der Länge $i, j > 0$ gibt es drei Möglichkeiten:

- Man aligniert $u_1 \dots u_{i-1}$ mit $v_1 \dots v_{j-1}$ und u_i mit v_j, dafür gibt es N_A[i-1, j-1] Möglichkeiten.
- Man aligniert $u_1 \dots u_{i-1}$ mit $v_1 \dots v_j$ und u_i mit einem Gap, dafür gibt es N_A[i-1, j] Möglichkeiten.
- Man aligniert $u_1 \dots u_i$ mit $v_1 \dots v_{j-1}$ und v_j mit einem Gap, dafür gibt es $N_A[i, j-1]$ Möglichkeiten.

Insgesamt gibt es also N_A[i-1, j-1] + N_A[i-1, j] + N_A[i, j-1] Alignments für diese Strings.