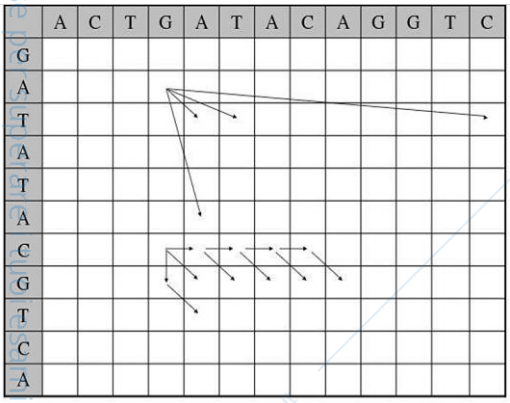
www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari

>Rivisitazione che velocizza il metodo di Needleman e Wunsch L6

Ad esempio, confrontando due sequenze della stessa lunghezza (caso più semplice: SXS), il calcolo dei percorsi possibile avviene casella-casella, effettuando un numero di calcoli pari ad n^2, in un tempo relativamente lungo. Dato che ogni casella ha un valore diverso, anche i calcoli per ciascuna di essi saranno differenti. Il costo sarà di un minimo di 0-1 n, fino ad un massimo di 2n (in alto a sinistra). Al salire di n (ovvero del numero di lunghezza delle sequenze che vanno a formare la matrice), il costo sale secondo una proporzionalità di n^3.

Un metodo più efficiente potrebbe essere quello di considerare non più una proporzionalità di n^3, ma di n^2 dimostrando che è possibile utilizzare due algoritmi diversi non per fare in modo differente una stessa cosa, ottenendo uno stesso risultato, ma per affrontare uno stesso problema in maniera differente.
Il problema nasce perché il percorso è rappresentato da nodi e da salti, quindi bisogna effettuare vari query per considerare tutte le possibilità. In altre parole presa una casella con un determinato valore si può procedere dritto (diagonale) in basso (verticale) o in orizzontale, è ovvio che la scelta cadrà sulla casella che possiederà il valore maggiore.
1. Gli angoli di partenza sono equivalenti (matrice polarizzata, ovvero con diverse polarità)
Normalmente si parte dall'angolo in basso a destra, ma essendo un quadrato, non esiste una discriminazione particolare sull'angolo di partenza.
Se esiste quindi un percorso ottimale che parte da un certo angolo, partendo da un altro angolo, deve esserci sempre un percorso ottimale con lo stesso valore e deve essere uguale.

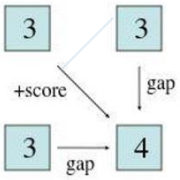


2. Migliore rappresentazione del percorso
Pensando a come viene rappresentato il percorso (attraverso frecce), potrebbero esserci problemi nel momento in cui si abbiano più alternative. Esistono quindi delle differenze quando si ha una corrispondenza tra coppie e quando si ha l'introduzione di uno o più gap.
Ammettere la presenza di un gap significa saltare la casella di una matrice, il che comporta allungare ulteriormente l'allineamento di uno: in termini pratici significa un'ulteriore complicazione del calcolo (operazione composta), il quale neanche con Needleman-Wunsch verrebbe semplificato.
Questa complessità è necessaria o si ha a disposizione un'alternativa che ci permette lo stesso di lavorare in modo efficiente? Un metodo grafico più efficiente sarebbe quello di differenziare in due step diversi l'aggiunta dell'appaiamento (diagonale → match) e l'introduzione dei gap (orizzontale → gap nella sequenza in basso). Camminare in verticale indica l'introduzione di gap nella sequenza in alto.
L'operazione viene scomposta in più operazioni semplici o elementari.
Esempio: saltare di una riga e due colonne potrebbe essere visto come un iniziale movimento in orizzontale e poi una discesa in diagonale di 45° (due operazioni semplici).
Il vantaggio è che il percorso effettuato con questo meccanismo si recupera lavoro, dato che seguendo le frecce arrivo direttamente a notare tutte le caselle possibili per i vari paths (circoscrivo con le frecce il rettangolo). L'intero processo non solo si semplifica, ma viene addirittura reso più breve in termini di tempo perché per ogni punto è sufficiente valutare solo tre possibilità.

Camminare in orizzontale significa che sono introdotti gap.
- Sottrarre i punti per il cammino orizzontale o verticale → GAP
- Aggiungere i punti per il cammino diagonale → MATCH

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari



	A	D	C	N	Y	R	Q	C	L	C	R	P	M
A	1	0	0	0	0	0	0	0	0	0	0	0	0
Y	0	1	1	1	1	1	1	1	1	1	1	1	1
C	0	1	2	2	2	2	2	2	2	2	2	2	2
Y	0	1	2	2	3	3	3	3	3	3	3	3	3
N	0	1	2	3	3	3	3	3	3	3	3	3	3
R	0	1	2	3	3	4	4	4	4	4	4	4	4
C	0	1	2	3	3	4	4	5	5	5	5	5	5
K	0	1	2	3	3	4	4	5	5	5	5	5	5
C	0	1	2	3	3	4	4	5	5	6	6	6	6
R	0	1	2	3	3	4	4	5	5	6	7	7	7
D	0	1	2	3	3	4	4	5	5	6	7	7	7
P	0	1	2	3	3	4	4	5	5	6	7	8	8

gap=0

```
ADC-NYRQCLCR-PM
* * * * * ** *
AYCYN-R-CKCRDP-
```

All'interno di una casella selezionata le operazioni non sono più tutte uguali, ma piuttosto differenziate, più facilmente prevedibili e in ultima analisi anche più semplici. Infatti si presentano 3 opzioni:
1) Andare lungo la diagonale, sommando i valori delle caselle in cui si arriva;
2) Camminare in verticale significa aggiungere un gap, ciò vuol dire che si deve sottrarre oppure dare un valore al gap pari a 0;
3) Procedere in orizzontale significa sempre aggiungere un gap.
Di queste 3 possibilità, si considera quella che fa ottenere il valore maggiore e di conseguenza, l'allineamento migliore. Questa è una tecnica più veloce e semplificata dell'algoritmo di N&W, ottenendo lo stesso risultato. Questo approccio tiene conto di [(n*n)*3] caselle a differenza di quello precedente che ne considera n^3, riducendo il numero di gap, in quanto ad essa verrà attribuito un valore fortemente negativo, preferendo il percorso lungo la diagonale.
Ma lo scoring system influenza i risultati? Sì, ogni singola variazione alle regole di calcolo dei punti equivale a contributi diversi a uno stesso allineamento, portando a considerare un percorso in modi differenti a seconda dei valori attribuiti; ne deriva che i parametri impiegati per il calcolo negli allineamenti sono determinanti. In sostanza, vengono scoraggiati i percorsi verticali ed orizzontali, introducendo valori di gap molto negativi. Un qualunque cambio di diagonale, si riduce.

Even that can be made better

	A	D	C	N	Y	R	Q	C	L	C	R	P	M
A	1												
Y					1								
C			1					1		1			
Y					1								
N				1									
R						1	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
D	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

n*n repeated n times = n^3

	A	D	C	N	Y	R	Q	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
Y	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
Y	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
D	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

n*n repeated 3 times = 3*n^2

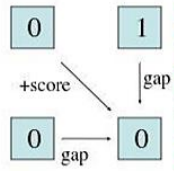
Waterman e Smith

L'algoritmo di Waterman e Smith è un algoritmo di bioinformatica pensato per l'allineamento di sequenze, cioè la determinazione del grado di similarità (detta anche omologia) di sequenze nucleotidiche o proteiche. L'algoritmo si basa sulla programmazione dinamica ed è una variazione dell'algoritmo di Needleman-Wunsch (proposto qualche anno prima). Effettuando un cambio nell'algoritmo di N&W, introducendo un ulteriore parametro di confronto, si ottiene W&S: dopo aver fatto il calcolo del valore della casella, se il valore è negativo, bisogna sostituirlo con 0. Inoltre, nel punteggio, bisogna valutare i mismatch e gap con valori negativi. > *Basta seguire il percorso che non fa perdere punti.*

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari

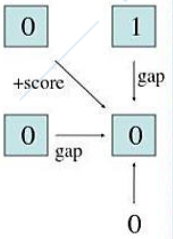
Obiettivo: "Date due sequenze relativamente lunghe, determinare la sottosequenza della prima sequenza che realizza il maggior grado di similarità con una sottosequenza della seconda sequenza".



	A	C	T	G	A	C	T	G	T	C	G	A	C	T	G	C	C	T	G
A	1	-1	-1	-1	1	-1	-1	-1	-1	-1	-1	1	-1	-1	-1	-1	-1	-1	-1
A	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
T	-1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
G	-1	1	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
C	-1	1	2	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4
T	-1	1	2	3	3	4	5	5	5	5	5	5	5	5	5	5	5	5	5
G	-1	1	2	3	3	4	5	6	6	6	6	6	6	6	6	6	6	6	6
C	-1	1	2	3	3	4	5	6	6	7	7	7	7	7	7	7	7	7	7
T	-1	1	2	3	3	4	5	6	7	7	7	7	7	8	8	8	8	8	8
G	-1	1	2	3	3	4	5	6	7	7	8	8	8	8	9	9	9	9	9
C	-1	1	2	3	3	4	5	6	7	8	8	8	9	9	9	10	10	10	10
T	-1	1	2	3	3	4	5	6	7	8	8	8	9	10	10	10	10	11	11
C	-1	1	2	3	3	4	5	6	7	8	8	8	9	10	10	11	11	11	11
C	-1	1	2	3	3	4	5	6	7	8	8	8	9	10	10	11	12	12	12
A	1	1	2	3	4	4	5	6	7	8	8	9	9	10	10	11	12	12	12
C	-1	2	2	3	4	5	5	6	7	8	8	9	10	10	10	11	12	12	12
T	-1	2	3	3	4	5	6	6	7	8	8	9	10	11	11	11	12	13	13
G	-1	2	3	4	4	5	6	7	7	8	9	9	10	11	12	12	12	13	14

match=1
mismatch=0
gap=0

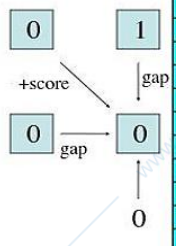
es. match=1; mismatch=-1; gap=0
NON NEGATIVE GAP >>>> N&W
Non è cambiato nulla, dato che con tali parametri non avviene il confronto con lo 0.I numeri continueranno a salire da sinistra verso il basso.



	A	C	T	G	A	C	T	G	T	C	G	A	C	T	G	C	C	T	G
A	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
A	1	1	0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0
T	0	1	2	0	0	1	2	0	1	0	0	0	1	2	0	0	0	1	0
G	0	0	1	3	0	0	1	3	0	1	1	0	0	1	3	0	0	0	2
C	0	1	0	1	3	1	0	1	3	1	1	1	1	0	1	4	1	0	0
T	0	0	2	0	1	3	2	0	2	3	1	1	1	2	0	1	4	2	0
G	0	0	0	3	0	1	3	3	0	2	4	1	1	1	3	0	1	4	3
C	0	1	0	0	3	1	1	3	3	1	2	4	2	1	1	4	1	1	4
T	0	0	2	0	0	3	2	1	4	3	1	2	4	3	1	1	4	2	1
G	0	0	0	3	0	0	3	3	1	4	4	1	2	4	4	1	1	4	3
C	0	1	0	0	3	1	0	3	3	2	4	4	2	2	4	5	2	1	4
T	0	0	2	0	0	3	2	0	4	3	2	4	4	3	2	4	5	3	1
C	0	1	0	2	0	1	3	2	1	5	3	2	5	4	3	3	5	5	3
C	0	1	1	0	2	1	1	3	2	2	5	3	3	5	4	4	4	5	5
A	1	0	1	1	1	2	1	1	3	2	2	6	3	3	5	4	4	4	5
C	0	2	0	1	1	2	2	1	1	4	2	3	7	4	3	6	5	4	4
T	0	0	3	0	1	1	3	2	2	1	4	2	4	8	5	3	6	6	4
G	0	0	0	4	1	1	1	4	2	2	2	4	2	5	9	6	3	6	7

match=1
mismatch=0
gap=-3

es. match=1; mismatch=0; gap=-3
NON NEGATIVE
MISMATCH>>>>>N&W



	A	C	T	G	A	C	T	G	T	C	G	A	C	T	G	C	C	T	G
A	1	-1	-1	-1	1	-1	-1	-1	-1	-1	-1	1	-1	-1	-1	-1	-1	-1	-1
A	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T	-1	0	1	0	0	0	1	0	1	0	0	0	0	1	0	0	0	1	0
G	-1	0	0	2	0	0	0	2	0	0	1	0	0	0	2	0	0	0	2
C	-1	0	0	0	1	1	0	0	1	1	0	0	1	0	0	3	1	0	0
T	-1	0	1	0	0	0	2	0	1	0	0	0	0	2	0	0	2	2	0
G	-1	0	0	2	0	0	0	3	0	0	1	0	0	0	3	0	0	1	3
C	-1	0	0	0	1	1	0	0	2	1	0	0	1	0	0	4	1	0	0
T	-1	0	1	0	0	0	2	0	1	1	0	0	0	2	0	1	3	2	0
G	-1	0	0	2	0	0	0	3	0	0	2	0	0	0	3	0	0	2	3
C	-1	0	0	0	1	1	0	0	2	1	0	1	1	0	0	4	1	0	1
T	-1	0	1	0	0	0	2	0	1	1	0	0	0	2	0	1	3	2	0
C	-1	0	0	0	0	1	0	1	0	2	0	0	1	0	1	1	2	2	1
C	-1	0	0	0	0	1	0	0	0	1	1	0	1	0	0	2	2	1	1
A	1	0	0	0	1	0	0	0	0	0	0	2	0	0	0	0	1	1	0
C	-1	2	0	0	0	2	0	0	0	1	0	0	3	0	0	1	1	0	0
T	-1	0	3	0	0	0	3	0	1	0	0	0	0	4	1	0	0	2	0
G	-1	0	0	4	1	0	0	4	1	0	1	0	0	1	5	2	0	0	3

match=1
mismatch=-1
gap=-3

es. match=1; mismatch=-1; gap=-2
SMITH and WATERMAN

>La parte dissimile scende di punteggio anche di molto sotto lo zero, a questo punto interviene il **confronto con lo 0**, azzerando il valore della casella. La parte simile è ovviamente favorita ed è quella che avrà i punteggi più alti.
>La regola n.2 è necessaria perché corrisponde all'unica differenza reale tra S&W e N&W. Inoltre, impostando la condizione che solo i gap devono essere negativi, si potrebbe avere una situazione di ridondanza, perché in presenza di un gap si avrebbe sempre un percorso alternativo con valori positivi (match) o al più altri gap pari a 0. In virtù di ciò, il **doppio negativo** (gap e mismatch) è fondamentale.

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari

>Nella matrice di S&W, a differenza di N&W, si avranno i valori massimi ovunque e non localizzati in una regione precisa. Conseguenza di ciò non si hanno dei pattern tutti in salita, ma il percorso prima sale arrivando ad un picco massimo e poi scende nuovamente fino ad arrivare sotto lo 0, azzerandosi per la regola 1. Si ottengono così allineamenti locali interessanti, cioè dispersi ovunque nella sequenza.
(es. esone umano confrontato con la sequenza genomica umana)

Algoritmi esaustivi
Local alignment: Smith e Waterman (1981) è utile per individuare pattern di similarità locali.
>> Water Matcher (programma)
Global alignment: Needleman e Wunsch (1970) è utile per gli allineamenti globali.
>> Needle Stretcher (programma)

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari