

Allineamenti multipli

Finora ci siamo occupati di allineamenti a coppie (pairwise), ma il modo migliore per conoscere le caratteristiche di una determinata famiglia è allineare molte proteine a funzione analoga.

I siti funzionalmente o strutturalmente più rilevanti tendono a mantenersi invariati nelle proteine omologhe, mentre i siti meno importanti possono cambiare anche molto.

Osservare e studiare le conservazioni significa capire come le famiglie di proteine funzionano, cosa la rende diverse tra loro, se esistono o meno relazioni filogenetiche inter e intrafamiglia.

In questo modo è possibile individuare la funzione di una proteina ignota solo **osservando la sequenza dei suoi residui.**

Similitudine e omologia

Omologia: carattere QUALITATIVO che posseggono quelle sequenze che derivano da un antenato comune in seguito al processo evolutivo. O due geni sono omologhi o non lo sono. Non esiste una percentuale di omologia.

Similitudine: carattere QUANTITATIVO che origina da un allineamento. Il grado di identità che si determina tra i residui allineati o il fatto che residui simili possano corrispondere in un allineamento, può essere quantificato disponendo di metri di valutazione oggettivi, come le matrici di sostituzione.

=> un'alta similitudine tra proteine può essere indice di omologia, ma non si può escludere il contrario. Esistono infatti proteine molto simili in organismi filogeneticamente non correlati tra loro e proteine molto diverse che possono essere ricondotte a omologhe mediante altri studi

Geni ortologhi e geni paraloghi

Geni ortologhi: geni simili riscontrabili in organismi correlati tra loro. Il fenomeno della speciazione porta alla divergenza dei geni e quindi delle proteine che essi codificano.

es. l' α -globina di uomo e di topo hanno iniziato a divergere circa 80 milioni di anni fa, quando avvenne la divisione che dette vita ai primati e ai roditori. I due geni sono da considerarsi ortologhi.

Geni paraloghi: geni originati dalla duplicazione di un unico gene nello stesso organismo.

es. α -globina e β -globina umana hanno iniziato a divergere in seguito alla duplicazione di un gene globinico ancestrale. I due geni sono da considerarsi paraloghi.

Le sequenze da multiallineare in genere si ottengono dalla ricerca in banca dati mediante i sistemi di ricerca per similarità come BLAST e FASTA (ma ce ne sono anche molti altri...).

Visto che derivano già da un allineamento (anche se prodotto con metodi euristici) e visto che si prendono in considerazione solo sequenze che hanno un alto score (o un basso E, expect value), l'allineamento mutiplo su questi DATASET darà risultati soddisfacenti.

In un allineamento multiplo si prendono in considerazione le colonne di residui, più che le proteine a cui appartengono. Ogni residuo incolonnato è da considerarsi in modo implicito come evolutivamente correlato, in qualche modo.

Un allineamento esatto come quello dei pairwise esaustivi richiederebbe un algoritmo di ordine $O(L^N)$, cioè un numero di operazioni che cresce con la lunghezza delle sequenze elevato al numero di sequenze stesse

⇒ 3 proteine da 200 residui = ordine 8×10^6

⇒ 5 proteine da 100 residui = ordine 10^{10}

Ovviamente i tempi di elaborazione sarebbero interminabili.

E' stata quindi proposta una soluzione semplice ed elegante:

L'ALLINEAMENTO PROGRESSIVO DI COPPIE DI SEQUENZE

basandosi sull'assunto che se una proteina può essere allineata con una seconda e una seconda con una terza, allora deve esistere un allineamento che le comprenda tutte e tre.

N sequenze (dataset)
disposte a caso, non
allineate

Allineare tutte le proteine
con tutte le proteine, a coppie
($N(N-1)/2$ allineamenti)

Determinare un
albero guida basato
sui punteggi di
similarità di tutte le
coppie

A partire dalla coppia più simile,
determinare le colonne
conservate, e allineare la coppia
successiva mantenendo queste
colonne e ricalcolando lo score
complessivo

N sequenze (dataset)
allineate

Clustal W

E' il programma per gli allineamenti multipli più utilizzato.

E' implementato sul server EBI, richiamato dalle pagine dei risultati quando si fanno ricerche di similitudine (FASTA o BLAST), ma ne esistono versioni gratuite che girano sotto Linux e DOS.

Inoltre ne esiste una versione con interfaccia grafica per Win32, **ClustalX**.

Di fatto, ClustalW è il migliore se nell'allineamento non ci sono troppi gaps e se la similarità media è superiore al 50%.

Se si è sicuri di aver prodotto un buon allineamento è possibile con ClustalW aggiungere sequenze a quelle già allineate. I criteri di allineamento non saranno ricalcolati, ma verranno utilizzati quelli preesistenti

Se gli allineamenti generano uno score basso, è possibile utilizzare **CRITERI STRUTTURALI** per migliorare l'allineamento: se si conosce la struttura di una delle proteine, questa verrà utilizzata per la costruzione dei punteggi di penalità nell'allineamento a coppie preliminare: infatti in una famiglia di proteine, la struttura terziaria e quella secondaria tendono a conservarsi più della struttura primaria.

Inoltre ClustalW può importare delle tavole di penalties predeterminate per aggiustare gli allineamenti secondo criteri arbitrari, allo scopo di migliorare lo score generale.

Lancia ClustalW

Valutare la bontà di un multi-allineamento

In genere: si sommano tutti gli score di tutte le possibili coppie di proteine allineate, pesando i valori in base alla similitudine nello stesso cluster per evitare che alcuni cluster prevalgano su altri nel conteggio finale. Ottengo un WSP (Weighted Sum of Pairs):

$$\text{WSP}_{\text{score}} = \sum_{i=1}^{N-1} \sum_{j=1}^N W_{ij} \text{QUAL}(A_{ij})$$

N: numero di sequenze i,j: coppia di sequenze

QUAL: punteggio di similarità della coppia W: peso per la coppia

Il valore complessivo del WSP dipende dai criteri di punteggio utilizzati nell'allineamento più che da considerazioni biologiche, ma è comunque un criterio valido per tutti gli allineamenti con gli stessi parametri

Uno score così è chiamato **Objective Function (OF)**

Strumenti per la visualizzazione dei multi-allineamenti

Visualizzare bene un multi-allineamento è importantissimo per apprezzare le informazioni che esso può fornire.

L'output dei programmi di multiallineamento è una stringa contenente le sequenze allineate formattata in modi diversi secondo vari standard:

MSF, NEXUS, PHYLIP, CLUSTAL, FASTA

In questi cambia l'intestazione, la porzione di proteina che si trova sulla stessa linea (per leggere tutto l'allineamento sulla stessa schermata), le informazioni collaterali che riguardano l'allineamento stesso.

Esistono programmi come ReadSeq in grado di convertire un formato in un altro agevolmente (anche ClustalW può farlo...)

Utilizzo dei colori

I file raw-text possono essere utilizzati per visualizzare le colonne, ma è possibile associare colori diversi per residui con caratteristiche chimico fisiche diverse. Questo facilita molto la visualizzazione dei multiallineamenti

```
GRVQGVWYRGWTVETAKGLG-LAGWVRNRADGT-VEALFHGPE-AAVEAMLIACRG-G-PPSARVDDLRVTP-VAAP-
GRVQGVWCYRNWTVENAEQLG-IRGWVRNRDGS-VEALFSGPP-EAVDEMHRQR-PPAAMVTGLEAFP-STEE-
GRVQGVGCRYAACADMAHALG-LRGWVRNRDGA-VEAFLAGPE-PNVLRMQAWMEE-G-PDLALVTQLRRTTPGDIEP-
GTVQGVGFRHATVVRQAHALG-IRGWVANLDDGS-VEAMLQGSA-NQVDRMLSWLRH-G-PPAARVTEWSGEEERSTER-
GRVQGVFFRQSMKEVAMRNG-VKGWVRNRSDGKTVEAVLEGP-DAVMKWLEWARI-G-PPGARVEDIEVQWEEYK-
GRVQGVSFRAVTRDRAREAQ-VKGWVRNLSDC-VEAVFEGTR-PAVQKLISWCYS-G-PSQAQVERVEVHWEEPTG-
GRVQGVGFRWSMQREARKLG-VNGWVRNLPDGS-VEAVLECEE-ERVEALIGWAHQ-G-PPFARVTRVEVHWEEPK-
GRVQGVGYRYSTVDTARQLG-LTGWVRNLPDNE-VEAVFEGAR-EVWDDMVVRWCHS-G-PPAAVVKDWWVEYEVPEG-
GRVQGVGFRYSTVDTASQLG-LTGWVRNLPDGE-VEAVFEGVR-DIVEDMVVRWCHA-G-PPAAVVDVAVEYEEPEG-
GVVQGVFFRASMREEALRLG-LSCWVRNLPDGESVEAVVEGRG-DAVERIICWCLR-G-PPAARVRELEVELEPYK-
GRVQGINFRSNTLSKALELN-VKGWVNRIDGS-VEAYFSGEL-CDVMSLINYCVS-D-MPYAVVKRYDV-YDIPYM-
GRVQGINFRSNTLVKALELG-VKGWIKNLPDGS-VEALFSGES-EQIEKLISYCVS-N-MPYAEVKRYDV-YIEPYT-
GRVQGVWCYRQGTALQAERLA-LAGWVRNLADGE-VEAVVECEE-AAVRELAEWLWR-G-PEQARVEGVELEEVGLQG-
GRVQGVGFRQATREEARLE-LDCWVRNLDG-VEVWVECEE-DAKALERWLGR-G-PRHAEVSAVEVEQMPLOQ-
```

[ESPrnt](#) e [PrettyPlot](#) sono programmi dedicati a questo tipo di analisi qualitativa disponibili in rete

Le sequenze consenso

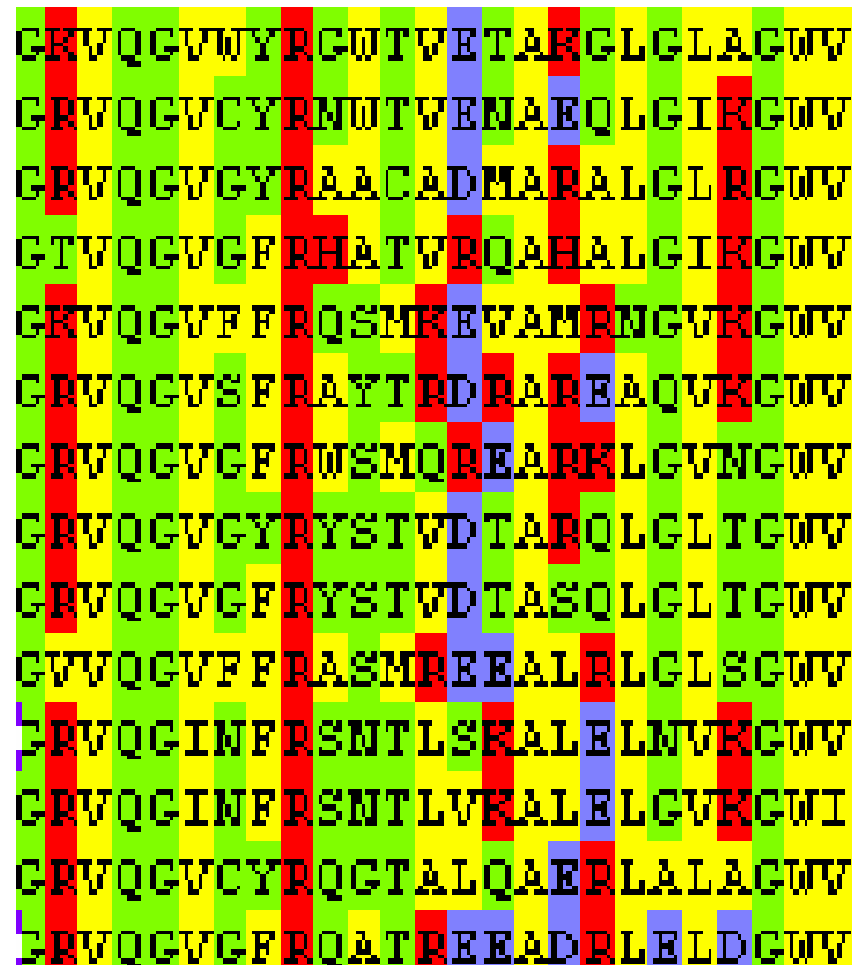
Si definisce sequenza consenso una sequenza derivata da un multiallineamento che presenta solo i residui più conservati per ogni posizione

⇒ riassume un multiallineamento.

⇒ non è identica a nessuna delle proteine del dataset.

⇒ si possono definire dei simboli che la definiscano e che indichino anche conservazioni non perfette in una posizione.

⇒ è possibile utilizzare una formattazione precisa che permetta di capire anche le variazioni in una posizione, non solo le conservazioni.



GRVQGV--R-----A--LG--GWV
 GRVQGH-aRvvvvvvAvvLGivGWV

GRVQG [VI] - [FY] R-----A-L-----GWY
 GRVQGV--R-6A-LG--GWV

Alcuni modi di indicare le
 sequenze consenso

WebLogo è una risorsa in rete
 per generare sequenze
 consenso

Consenso esatto

Consenso a simboli

Consenso con variazioni

Consenso con ripetizioni

Profili dei multi-allineamenti

Un multi-allineamento genera molte più informazioni per l'individuazione dei residui importanti per una famiglia di proteine di tanti allineamenti a coppie.

Diventa quindi basilare poter riassumere le conservazioni osservate in un unico formato.

Inoltre multi-allineare proteine divergenti tra loro è molto più informativo rispetto alla stessa analisi fatta su proteine molto simili.

Un PROFILO è un metodo di SCORING in cui ad ognuno dei venti amino acidi viene assegnato un punteggio basato sulla frequenza e sul valore in una matrice di sostituzione. Ogni cella di un profilo esprime quindi il peso da attribuire ad ogni aminoacido in quella posizione

I programmi che generano i profili (come **PROFILEMAKER** del pacchetto GCG) riportano sulla prima colonna la sequenza **CONSENSO**, cioè una sequenza derivante da tutti gli allineamenti e contenente solo i residui più frequenti.

Ogni colonna successiva descrive la situazione di tutti gli aminoacidi in quella posizione.

Un profilo può essere utilizzato per una ricerca in banca dati mediante la variante di Blast [PSI-BLAST](#). Il programma effettua, data una sequenza query, una serie di iterazioni in cui ogni volta l'utente sceglie un certo numero di sequenze individuate e su queste viene ricostruito il profilo.

Ad ogni iterazione successiva verranno individuate nuove sequenze, in modo più o meno accurato a seconda delle scelte fatte in precedenza.

Algoritmi ad apprendimento automatico

Se si possiedono dei buoni modelli probabilistici che descrivano bene l'informazione contenuta in un contesto biologico, è possibile far “imparare” le relazioni che il modello propone ad un computer.

L'applicazione pratica di questo concetto si ritrova in:

1- Hidden Markov Models

2- Reti neurali

3- Algoritmi genetici

Proprio perchè oggi disponiamo di un gran numero di informazioni ma piuttosto disordinate, è possibile cercare di istruire un computer allo scopo di capire da solo le relazioni che intercorrono tra i vari elementi.

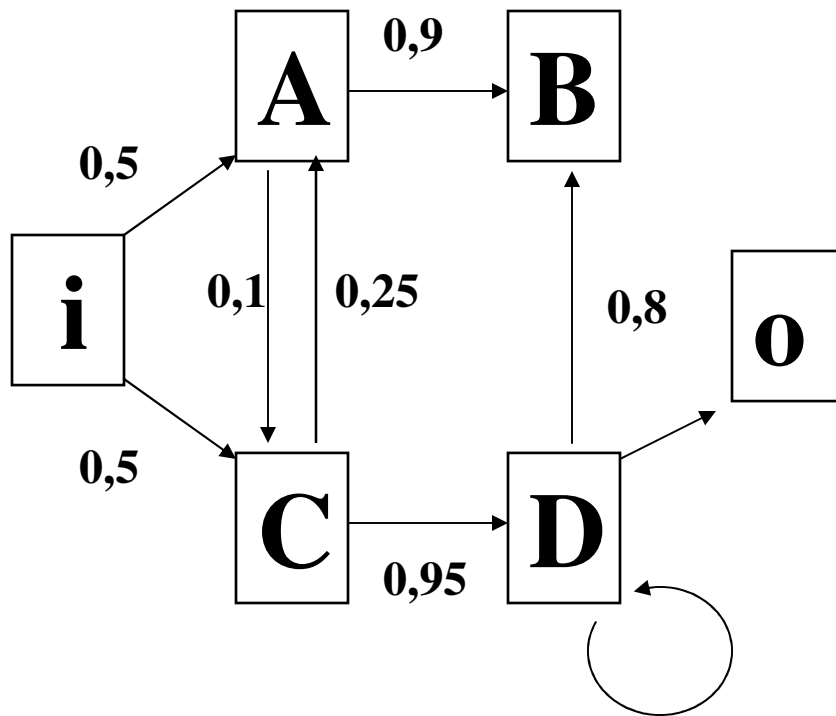
Hidden Markov Models (HMMs)

Sono modelli matematici che descrivono le probabilità di trovare una data sequenza in un database (che può essere anche un dataset di proteine multiallineamte) conoscendo il contenuto del database.

Una catena di Markov è un insieme di numeri in successione in cui ogni numero dipende solo dai k numeri che lo precedono (k si definisce come l'ordine della catena).

Questi numeri possono essere probabilità, e quindi una catena di Markov è come un modello che descrive le probabilità condizionate di avere un residuo data una serie di residui precedenti.

Il programma più utilizzato basato su questi modelli è [HMMER](#) che ha come input un multiallineamento precedente (o una ricerca i banca dati) ed è in grado di cercare in banca dati utilizzando non le sequenze ma solo i profili che da essi vengono generati.



Data la sequenza ABCD è possibile stabilire la sua probabilità moltiplicando le probabilità di ogni evento, se gli eventi sono indipendenti (Catena di Markov di ordine zero) , secondo

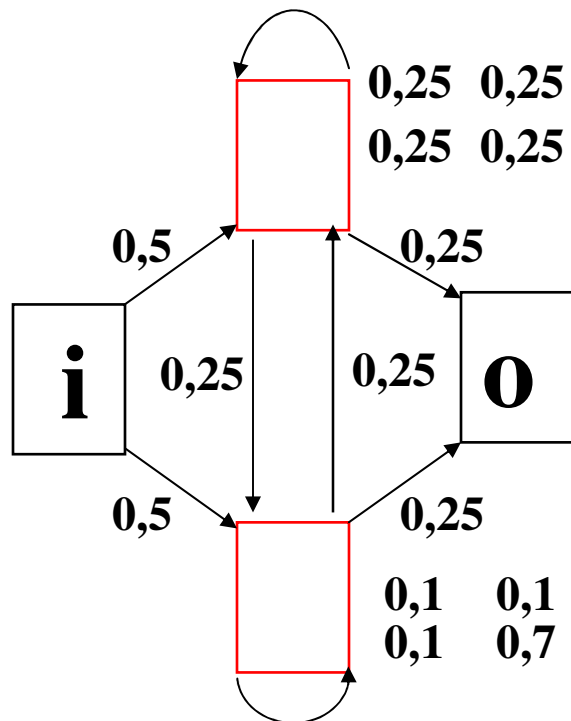
$$p(1234) = p(1)* p(2)* p(3)* p(4)$$

Il grafico sopra mostra le relazioni probabilistiche che stanno alla base di una sequenza di es. 4 residui (o basi) secondo una matrice precisa e secondo dei criteri osservati nel database, così da stabilire in modo completo tutti i percorsi possibili, posto che ogni evento dipenda solo dall'evento successivo.

E' poi possibile determinare le probabilità se gli eventi non sono indipendenti, ma dipendono dai k eventi successivi (catena di Markov di ordine k).

Gli eventi sono definiti stati della catena, ma sono nascosti e legati da relazioni di probabilità predeterminate, fin quando non si chiede al modello di generare (emettere) i simboli appropriati.

Ogni stato allora emetterà il suo simbolo a seconda degli eventi precedenti.



Input: mi serve una sequenza di 10 nucleotidi che abbia una probabilità p di presentarsi

Output: una sequenza di 10 residui con probabilità P di verificarsi

Input: che probabilità ho che questa sequenza si presenti per caso nel database?

Output: la probabilità è p

Reti neurali

Sono circuiti di informazioni con un fissato numero di nodi (**STATI**) in cui immagazzinare le informazioni risultanti dalle varie interconnessioni ed una precisa **ARCHITETTURA**, cioè una struttura di interconnessione dei nodi.

Se fornisco ad una rete neurale una informazione e il suo risultato (un **TRAINING SET**), gli stati memorizzano il modo di andare dall'informazione al risultato sfruttando le interconnessioni. Ripetendo molte volte il training con set diversi ma ugualmente veri, alla fine la rete è in grado di arrivare da sola al risultato.

Se fornisco alla rete una informazione con risultato incognito, essa risponderà con il risultato che per lei è appropriato, dato quello che ha imparato dai training set.

es. Se io fornisco un numero di multiallineamenti esatti, la rete impara a multiallineare, e alla fine, data una serie di sequenze, sarà in grado di multiallinearle.

Algoritmi genetici

Se consideriamo un problema che ha una soluzione dipendente da n parametri e da k valori, una esplorazione completa richiede k^n operazioni.

Ma se noi sappiamo come si può evolvere il sistema (perchè abbiamo un training set o sappiamo le regole) per ricavare il risultato, sappiamo che alcuni passaggi non sono possibili o non si sono mai verificati, e sappiamo che ci sono percorsi che sono preferiti ad altri.

Se l'algoritmo viene modellato per rispettare gli schemi osservati e viene calcolata per ogni passaggio una FITNESS, cioè un valore di attendibilità, posso arrivare entro un certo numero di cicli ad avere un risultato che ha una fitness ottimale per le mie aspettative

⇒ posso simulare un crossing over tra due sequenze visto che so come il crossing over avviene.

⇒ posso simulare la mutagenesi visto che conosco le frequenze di mutazioni e gli eventi mutageni che accadono

