

# **Laboratorio di Bioinformatica**

Dispense del corso - Modulo 1

**Chiara Solito**

Corso di Laurea in Bioinformatica  
Università degli studi di Verona  
A.A. 2021/22

La presente è una dispensa riguardante il corso di **Laboratorio di Bioinformatica** del CdS in Bioinformatica (Università degli Studi di Verona). Per la stesura di questa dispensa si è fatta fede al materiale didattico fornito direttamente dal professore nell'Anno Accademico 2021/2022. Eventuali variazioni al programma successive al suddetto anno non saranno quindi incluse.

Insieme a questo documento in formato PDF viene fornito anche il codice L<sup>A</sup>T<sub>E</sub>X con cui è stato generato.

## Contents

<b>1 Il corso</b>	<b>5</b>
<b>2 Cos'è la bioinformatica?</b>	<b>5</b>
2.1 Il flusso dell'informazione biologica . . . . .	5
2.2 Struttura degli acidi nucleici . . . . .	5
2.3 Le proteine . . . . .	6
<b>3 Il cosmo "omico"</b>	<b>6</b>
3.1 La genomica . . . . .	6
3.2 Trascrittomica . . . . .	7
3.3 Proteomica . . . . .	7
3.4 Genomica Strutturale . . . . .	7
3.5 Farmaco-genomica . . . . .	7
<b>4 L'evoluzione ed il confronto tra sequenze</b>	<b>7</b>
<b>5 Le Basi di Dati Biologiche</b>	<b>10</b>
5.1 Introduzione . . . . .	10
5.2 Dati di Sequenza . . . . .	11
<b>6 NCBI</b>	<b>12</b>
6.1 Com'è strutturato il database . . . . .	12
6.2 Operatori Booleani . . . . .	14
6.2.1 Operatore AND (&) . . . . .	14
6.2.2 Operatore OR ( ) . . . . .	15
6.2.3 Operatore NOT (!) . . . . .	15
6.2.4 Combinazione di Operatori Booleani . . . . .	15
6.3 Nel dettaglio . . . . .	15
<b>7 Proteine - Le banche dati proteiche più usate</b>	<b>16</b>
7.1 NCBI Protein - non molto ricco . . . . .	16
7.2 Uniprot . . . . .	16
7.2.1 Struttura del database . . . . .	17
7.3 ExPASy . . . . .	17
<b>8 Allineamenti di Sequenze</b>	<b>19</b>
8.1 Definizione - <u>Allineamento a coppie</u> . . . . .	19
8.2 Altre definizioni . . . . .	19
<b>9 Confrontare due sequenze</b>	<b>21</b>
9.1 Come identificare le zone di somiglianza locale tra due sequenze? . . . . .	22
9.1.1 Matrice a punti - dot plot . . . . .	22
9.1.2 In breve . . . . .	24

<b>10 Algoritmi dinamici di allineamento</b>	<b>25</b>
10.1 Il concetto . . . . .	25
10.2 Regole pratiche . . . . .	25
10.3 Step . . . . .	26
10.4 Needleman-Wunsch: programmazione dinamica . . . . .	28
<b>11 Waterman-Smith</b>	<b>30</b>
11.1 L'algoritmo . . . . .	30
<b>12 Allineamento: globale vs locale</b>	<b>30</b>
12.0.1 In conclusione . . . . .	32
12.1 Significatività statistica di un allineamento . . . . .	32
12.1.1 Significatività allineamento globale: lo Z score . . . . .	32
12.1.2 Significatività allineamento locale . . . . .	33
<b>13 Matrici di punteggio</b>	<b>34</b>
13.1 PAM: Point Accepted Mutation . . . . .	35
13.2 La mutabilità relativa degli amminoacidi . . . . .	37
13.2.1 Matrice PAM1 (probabilità) di Dayhoff . . . . .	38
<b>14 Matrici di sostituzione</b>	<b>38</b>
14.1 Moltiplicare le matrici . . . . .	38
14.2 Approccio Dayhoff . . . . .	40
<b>15 Matrici BLOSUM</b>	<b>42</b>
<b>16 BLOSUM vs. PAM</b>	<b>42</b>
16.1 La "Twilight zone" nell'allineamento di proteine . . . . .	43
<b>17 BLAST</b>	<b>45</b>
17.1 Problema con gli algoritmi dinamici . . . . .	45
17.2 Cos'è Blast? . . . . .	45
17.3 Ricerca su blast . . . . .	45
17.4 L'algoritmo . . . . .	47
17.4.1 Le 3 fasi . . . . .	48
17.5 Come interpretare una ricerca BLAST . . . . .	49
17.5.1 Il valore atteso E . . . . .	49
17.5.2 E-value e p-value . . . . .	50
17.5.3 Panoramica . . . . .	50
17.6 Strategia per la ricerca con BLAST . . . . .	51
17.6.1 Valutare se le proteine sono omologhe . . . . .	51
17.6.2 Due esempi di problemi che BLAST standard non può risolvere . . . . .	52
<b>18 PSI-BLAST (position specific iterated)</b>	<b>52</b>
18.1 Fasi di esecuzione . . . . .	52
18.1.1 Come costruire il profilo? . . . . .	53
18.1.2 Funzionamento di Psi-Blast . . . . .	53
18.1.3 Esempio di Cicli . . . . .	53
18.2 Psi-Blast: la corruzione . . . . .	56
<b>19 Allineamento multiplo di sequenze</b>	<b>59</b>
19.1 Visione Generale . . . . .	59
19.1.1 Una definizione . . . . .	59
19.1.2 Alcuni fatti . . . . .	59
19.1.3 Caratteristiche utili per realizzarla . . . . .	59
19.1.4 Utilizzi e Vantaggi . . . . .	59
19.2 Metodi Euristici . . . . .	60

<b>20 Clustal Omega</b>	<b>60</b>
20.1 Fasi di MSA . . . . .	60
20.1.1 Feng Doolittle fase 1: generare allineamenti . . . . .	60
20.1.2 Feng-Doolittle fase 2: albero guida . . . . .	60
20.1.3 Feng-Doolittle fase 3: allineamento progressivo 62	
20.1.4 Perché un GAP è per sempre? . . . . .	63
20.1.5 Esempio sulla variabilità dei MSA . . . . .	63
<b>21 Confronto</b>	<b>63</b>
21.1 ClustalW . . . . .	63
21.2 Praline . . . . .	63
21.3 MUSCLE . . . . .	64
21.4 Probcns . . . . .	64
21.5 TCoffee . . . . .	65
<b>22 Metodi Iterativi</b>	<b>65</b>
22.1 La consistenza . . . . .	65
<b>23 T-Coffee (2000)</b>	<b>66</b>
23.1 Vincoli . . . . .	66
<b>24 Metodi</b>	<b>66</b>
24.1 Strategia per la valutazione degli algoritmi per l'allineamento multiplo (benchmarking) . . . . .	67
<b>25 Homologene</b>	<b>67</b>
<b>26 Introduzione alle Banche dati di proteine</b>	<b>69</b>
26.1 Le proteine: complessi polimeri di amminoacidi . . . . .	69
26.2 Struttura secondaria . . . . .	69
26.3 Struttura terziaria e quaternaria . . . . .	70
<b>27 Protein Data Bank</b>	<b>70</b>
27.1 Il file .PDB . . . . .	71
27.2 Pfam e Prosite . . . . .	71
27.2.1 Pfam . . . . .	71
27.2.2 Prosite . . . . .	71
27.2.3 CATH . . . . .	71
<b>28 Reti Neurali</b>	<b>72</b>
28.1 Struttura di una rete neurale . . . . .	72
28.2 Architettura e apprendimento di reti neurali . . . . .	73
28.3 Collegamento con le proteine . . . . .	73
28.3.1 Storicamente: Profile Network from Heidelberg (PHD) . . . . .	75
28.3.2 Schema del funzionamento di PSIPRED . . . . .	75
28.3.3 Schema del funzionamento di JPRED . . . . .	76
28.4 Altre proprietà . . . . .	77
<b>29 Cos'è la Biologia dei Sistemi?</b>	<b>79</b>
29.1 Dai componenti ai sistemi . . . . .	79
29.2 Nascita della Biologia dei Sistemi . . . . .	79
29.3 È necessaria? . . . . .	79
29.4 La costruzione di modelli computazionali dev'essere sistemico . . . . .	80

<b>30 Scienze riduzionistiche</b>	<b>81</b>
30.0.1 Descrizione di sistemi/processi biochimici . . . . .	81
30.1 Qual è l'intersezione? . . . . .	81
30.2 Esempio pratico: Rete di Proteine . . . . .	81
<b>31 Modellare la struttura della rete</b>	<b>82</b>
31.1 Struttura . . . . .	82
31.2 Da dove partiamo? . . . . .	82
31.2.1 I Database di Biomodels . . . . .	83
31.3 Esempio di Struttura del Modello di Rete . . . . .	84
31.4 Model Tasks . . . . .	85
31.4.1 Espressioni matematiche nel modellamento dinamico . . . . .	85
31.4.2 Espressioni matematiche . . . . .	85
31.4.3 Determinazione dei parametri . . . . .	85
<b>32 Modelli standard nella Biologia dei sistemi</b>	<b>86</b>
32.1 Concetti generali e proprietà . . . . .	86
32.2 Linguaggio comune: SBML . . . . .	87
32.3 Framework deterministico: è realistico? . . . . .	89
32.4 Equazione chimica principale . . . . .	90
32.5 Quando stocastica e quando deterministica? . . . . .	90
32.5.1 Un altro sistema semplice . . . . .	90

## **1 Il corso**

Il corso si propone di presentare allo studente le basi teoriche e applicative di algoritmi e programmi utilizzati nella ricerca e nell'analisi dei dati contenuti nelle principali banche dati biologiche di uso corrente. Il corso si compone di due moduli di seguito specificati.

Modulo 1: In questo modulo verranno appresi gli strumenti volti all'utilizzo dell'informazione in prote-omica, genomica, biochimica, biologia molecolare e strutturale. Si fornisce inoltre un'introduzione all'analisi e la visualizzazione di dati strutturali relativi a macromolecole biologiche e loro complessi e la creazione di semplici modelli dinamici e statici di reti biomolecolari, che avvicinerà lo studente all'emergente disciplina della systems biology.

Modulo 2: In questo modulo lo studente acquisirà conoscenza pratica degli strumenti bioinformatici per l'analisi, l'interpretazione e la predizione di dati biologici in proteomica, genomica, biochimica, biologia molecolare e strutturale. In particolare, gli studenti avranno la possibilità di applicare strumenti della bioinformatica allo stato dell'arte a specifici problemi biologici.

## **Lezione 1: Introduzione**

Ripasso delle basi e introduzione dei concetti fondamentali

## 2 Cos'è la bioinformatica?

La bioinformatica è (oggi) una disciplina scientifica dedicata alla risoluzione di problemi biologici a livello molecolare con metodi informatici. Descrive fenomeni biologici in modo numerico/statistico.

La bioinformatica principalmente:

- Fornisce modelli per l'interpretazione di dati provenienti da esperimenti di biologia molecolare e biochimica al fine di identificare tendenze e leggi numeriche
- genera nuovi strumenti matematici per l'analisi di sequenze di DNA, RNA e proteine (frequenza di sequenze rilevanti, loro evoluzione e funzione).
- organizza le conoscenze acquisite in basi di dati al fine di rendere tali dati accessibili a tutti, ottimizzando gli algoritmi di ricerca dei dati

Condivide alcuni argomenti con:

- **Systems biology**

- Rappresenta i processi biologici come sistemi per comprenderne le funzioni e i principi in modo olistico per mezzo di modelli matematici

- **Computational biology**

- Integra i risultati sperimentali con quelli derivanti da esperimenti in silico, ottenuti quindi per mezzo di metodi informatici a partire da dati biologici.

### 2.1 Il flusso dell'informazione biologica

Ad ogni livello di organizzazione (da interazioni fra biomolecole fino a cellule, organismi, popolazioni) l'elemento unificante è l'**EVOLUZIONE**, unico vero fondamento teorico della disciplina.

- EVOLUZIONE: adattamento progressivo attraverso variabilità genetica casuale e selezione naturale (Darwin, 1859)
- Ad ogni livello biologico, il fenotipo (insieme di tratti e caratteri somatici) è codificato dal genotipo (il patrimonio genetico)
- Genotipo: sorgente primaria di variazione genetica; fenotipo: bersaglio della selezione naturale
- Il genotipo è conservato nel genoma (fatto di DNA, eccezione fatta per virus a RNA)

### 2.2 Struttura degli acidi nucleici

Sono poliesteri composti da nucleotidi (composti da una base azotata, uno zucchero 2'-deossi-ribosio (o ribosio in RNA) e un gruppo fosforico).

2 tipi di basi azotate: purine (adenina, guanina) e pirimidine (timina, citosina uracile).

L'RNA è meno stabile ma più versatile del DNA; è scarsamente reattivo (meglio per conservare l'informazione) e assume strutture 3D anche molto complesse, ne esistono diverse forme: mRNA, tRNA, rRNA e piccoli RNA; ciò è fondamentale per la trasmissione dell'informazione genetica.

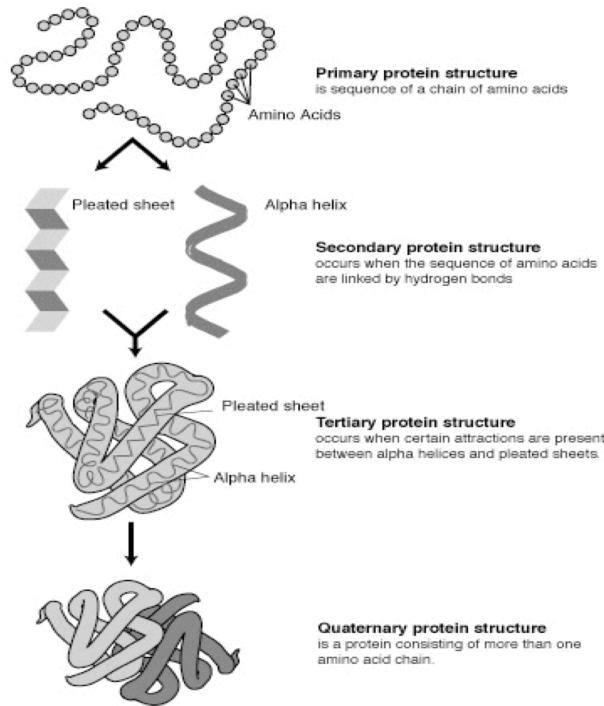
Un gene si trova in una precisa porzione fisica del genoma (**locus genico**). Es. Location: 6p21.1 significa cromosoma 6, braccio corto (p), regione 2 banda 1, sotto- banda 1.

In un gene le **Open Reading Frames** (parti di DNA/RNA codificanti) si trovano comprese fra la sequenza di inizio (codone d'inizio) e la sequenza di stop (codone di stop). Il genoma eucariotico contiene porzioni non codificanti importanti per la regolazione (**promotori**: vi si lega RNA polimerasi; **enhancers**: aumentano x200 la frequenza di trascrizione del gene) e per la costituzione (**introni**, sequenze ripetute). Lo splicing ("sal-datura") prepara il pre-mRNA per la traduzione. Nel genoma umano le porzioni non-codificanti sono in netta maggioranza. Diversa è la situazione nei genomi procariotici.

## 2.3 Le proteine

Sono il risultato del flusso dell'informazione genetica La presenza di 20 amminoacidi naturali con proprietà chimico- fisiche diverse conferisce una variabilità enorme. Il legame peptidico crea il backbone di qualunque proteina.

La struttura di una proteina si organizza in 4 livelli, visibili "srotolando" la matassa della luce di natale:



**La struttura 3D di una proteina è molto complessa** La determinazione della struttura 3D di proteine è un settore di ricerca molto attivo, come mostra la crescita esponenziale di strutture depositate nel Protein Data Bank.

## 3 Il cosmo "omico"

### 3.1 La genomica

- Genoma: Insieme dei geni di un organismo.
- Genomica: scienza che se ne occupa.
- Genoma Umano: Sequenziato completamente nel 2003.
- Occorre localizzare: Elementi Funzionali:
  - Regioni 'utili' → geni;
  - Sequenze codificanti, comprendere i meccanismi che regolano l'espressione, scoprire la funzione, e cercare d'intervenire specificamente su quest'ultima.

Il costo del sequenziamento del genoma oggi è alla portata di ciascun individuo.

### 3.2 Trascrittomica

- Trascrittoma: l'insieme di tutti i trascritti (RNA messaggeri, mRNA)
- Trascrittomica: scienza che se ne occupa.
- Occorre localizzare: Profili di espressione:
  - più dinamico del genoma
  - microarrays monitorano i livelli di espressione di migliaia di geni allo stesso tempo. Mirano ad individuare correlazioni e legami tra espressione genica, attivazione e inibizione. Esempi: studio nella differenziazione di cellule staminali o evoluzione di tumori.

### 3.3 Proteomica

- Proteoma: l'insieme di tutte le proteine in un sistema biologico o nel suo genoma
- Proteomica: scienza che se ne occupa.
- Occorre localizzare: sia le proteine codificate dai geni che le possibili modificazioni post- traduzionali (gruppi prostetici, multidomini, fosforilazione, ecc).
- Alcune tecniche
  - Gel:
    - 1° dimensione punto isoelettrico
    - 2° massa molecolare
  - Spettrometria di massa: identifica una proteina in base al suo rapporto massa/carica in seguito a ionizzazione

### 3.4 Genomica Strutturale

- Genomica strutturale: determinazione della struttura terziaria e quaternaria (3D e domini) delle proteine.
- Tecniche: cristallografia, NMR, homology modeling, cryoEM (microscopia crioelettronica) + AlphaFold (basato su AI)
- La struttura terziaria di una proteina è essenziale per determinarne la funzione

### 3.5 Farmaco-genomica

- Farmacogenomica: mira a prevedere la reazione di ciascun individuo verso un principio attivo in base al suo genotipo.
- Obiettivo: creare terapie farmacologiche personalizzate per ottimizzare il risultato minimizzando gli effetti collaterali.
- Esempio: previsione di gravi reazione avverse a Abacavir nella terapia dell'HIV

## 4 L'evoluzione ed il confronto tra sequenze

Un allele (variante di un gene presente contemporaneamente nella popolazione) può essere generato, fissato o mutare nel tempo.

**Uno degli obiettivi in senso lato della bioinformatica è stabilire se l'analisi dell'informazione riguardo a due oggetti biologici (e.g. geni o proteine) permette di stabilire una relazione di OMOLOGIA, cioè di discendenza da un antenato comune** Due sequenze che vengono separate fisicamente (per speciazione, duplicazione ecc.) non si scambiano più "informazione" ed evolvono indipendentemente, accumulando mutazioni. Spetta a noi trovare i tratti conservati dal comune antenato.

Un modo per muoversi in tal direzione è allineare le sequenze e determinare la percentuale di identità o sequence

identity (s.i.) (rapporto, in % tra il numero dei amminoacidi/basi identici rispetto al totale) o comunque il grado di similitudine. Di norma, sequenze nucleotidiche non correlate hanno una s.i. 50%; sequenze amminoacidiche non correlate hanno una s.i. 20%. Se tali valori aumentano, aumenta la probabilità che le sequenze siano omologhe. Ma tale indice dovrebbe tener conto anche della lunghezza delle sequenze. Una s.i. del 90% fra due sequenze di 100 a.a. ha un significato diverso rispetto alla stessa s.i. su sequenze di 30 a.a. **Allineare due sequenze significa stabilire se tra esse sussiste una relazione di omologia**

## **Lezione 2: Basi di Dati Biologiche**

## 5 Le Basi di Dati Biologiche

Il concetto di informazione è strettamente connesso a quello di dato e di struttura. Il dato è un osservabile (insieme di numeri, caratteri, simboli...) La struttura è l'organizzazione ordinata di dati che ne consente l'apprendimento.

Una banca dati è l'insieme di dati elementari, omogenei, ordinati e fruibili. In altre parole: è una collezione organizzata di dati. Esempio: elenco telefonico. L'informazione è strutturata in campi (nome, cognome ecc.). Ogni persona con i propri dati è un record. I dati biologici necessitano di un'organizzazione. Primo tentativo: Margaret Dayhoff (1925-1983): raccolse, nel 1965, le sequenze di 65 proteine (lavoro pionieristico per il tempo!). Le tecniche di sequenziamento rapido ed i progetti *-omici* hanno prodotto una quantità esplosiva di dati, anche di sequenze. L'avvento di Internet ha facilitato di gran lunga l'acquisizione e la distribuzione dell'informazione biologica in banche dati.

### 5.1 Introduzione

- Sono collezioni di dati:
  - strutturati
  - indicizzati
  - aggiornati
  - interconnessi
- I database biologici sono legati a strumenti per:
  - recuperare records al loro interno
  - aggiornare il database
  - combinare le informazioni
- Ci sono 6 principali categorie di basi di dati biologiche:
  - basi di dati di sequenze
    - DNA
    - RNA
    - Proteine
  - basi di dati per il mapping
    - geni
    - cromosomi
    - ...
  - Strutture3d (PDB)
  - Trascrittomico
  - Funzionali (KEGG)
  - Per la letteratura (PubMed), ontologies (GO), ...

A gennaio di ogni anno il Nucleic Acids Research pubblica un Database Issue, a gennaio:

- nel 2020 contiene 89 nuovi database e l'aggiornamento di 90 database
- classificati nelle seguenti categorie
  - Nucleotide Sequence Databases
  - RNA sequence databases
  - Protein sequence databases
  - Structure Databases

- Genomics Databases (non-vertebrate)
- Metabolic and Signaling Pathways
- Human and other Vertebrate Genomes
- Human Genes and Diseases
- Microarray Data and other Gene Expression Databases
- Proteomics Resources
- Other Molecular Biology Databases
- Organelle databases
- Plant databases
- Immunological databases
- Cell biology
- COVID-19 databases

Le banche dati si strutturano e si integrano per favorire lo studio del dogma centrale della biologia. Tre enti al mondo sono i principali.

- EMBL
- NCBI
- DDBJ

Integrando collegamenti esterni (Swiss-prot, ExPASy, UCSC, ecc, ecc...) sono un punto ideale di partenza.

## 5.2 Dati di Sequenza

Che dati si possono trovare?

- Principalmente sono presenti
  - sequenze di caratteri (nucleotidi, amminoacidi)
  - o strutture
- L'uso della rappresentazione dei dati biologici di varia natura come sequenze è la forma di gran lunga più diffusa.
- Sequenze di DNA: formate da 4 tipi di lettere (a,c,g,t), convenzionalmente minuscole
- Sequenze di RNA: formate da 4 tipi di lettere (A,C,G,U), convenzionalmente maiuscole
- Sequenze proteiche: formate da 20 lettere (A, C, D, E, F, G, H, I,K, L, M, N, P, Q, R, S, T, V, W, Y), convenzionalmente maiuscole

Il formato FASTA-Pearrson:

- Rappresentazione mediante testo di sequenze nucleotidiche o peptidiche (lettere MAIUSCOLE).
- La prima riga (di lunghezza arbitraria) è preceduta da ">" e rappresenta la descrizione della sequenza.
- Le linee precedute da ">" o ";" sono considerate di commento e non vengono interpretate come dato di sequenza
- Le linee successive (ciascuna di 80 caratteri) rappresentano la sequenza.
- Un file fasta può avere estensione (non c'è uno standard)

Il formato XML (eXtensible Markup Language).

- Replica la struttura logica del record nella banca dati
- I tag permettono di delimitare e definire campi e sottocampi

## 6 NCBI

NCBI (National Center for Biotechnology Information) presso il National Institute of Health. Offre accesso a tante risorse di vario tipo:

- Sequenze geniche e proteiche
- Strutture terziarie
- Genomi completi
- Pathways
- EST (expressed sequence tags)
- Profili trascrittomici
- Cataloghi tassonomici

Fornisce accesso a numerosi database attraverso il sistema Entrez:

- GenBank
- Swissprot
- PubMed
- GEO
- ...

Fornisce accesso anche a diversi software bioinformatici.

### 6.1 Com'è strutturato il database

Una ricerca qualunque dall'home page apre ENTREZ, interfaccia per l'accesso ai database presenti in NCBI.

- PubMed è l'interfaccia di accesso a MEDLINE. Con I suoi
  - 20 milioni di record fino agli anni '50
  - 4600 riviste da più di 70 paesiÈ la banca dati per la letteratura biomedica più completa. (Accessibile anche tramite EBI tramite 17 CiteXplore)
- Nucleotide è un database che raccoglie sequenze da diversi altri database di NCBI. Per sequenze nucleotidiche
  - EST ( expressed sequence tag )
  - GSS ( genome sequence surveys Gene è orientato ai geni, ai loci altre sequenze, B act A rtif C hromosome , Y east A rtif C hromosome ,...)

Inoltre:

- RefSeq ( sistema di identificazione )
- Unigene ( sequenze raggruppate )
- UniProt ( proteine )

- Gene è orientato ai geni, ai loci
- Proteins è la sezione focalizzata sulle proteine, alle quali possono corrispondere strutture
- PubChem dedicato ai composti chimici
- In Genome genomi completi con riferimenti alla ricerca effettuata, varianti genomiche, ecc

- Informazioni su profili di espressione genica in diverse condizioni, modifiche post-traduzionali GEO (Gene Expression Omnibus) repository

GenBank è la banca dati di tutte le sequenze in NCBI (sincronizzata con EMBL e DDBJ). Le sequenze derivano da diverse fonti e tipi:

- Geni (regioni di regolazione, esoni, introni: unità ereditarie)
- EST (Expressed Sequence Tags) brevi segmenti di DNA trascritti e sequenz. da cDNA (ottenuto da mRNA retrotrascritto)
- STS (sequence tagged site, dove l'informazione genetica è mappata fisicamente)
- GSS (Genome Survey Sequence, vettori sequenze solo parzialmente sequenziate)
- HTGS (High Throughput Genomic Sequence, sequenze prodotte da tecniche di seconda generazione per il sequenziamento veloce, messe qui in "preview")
- Sequenze di proteine (sezione nr, non redundant)

Così tanto materiale ha provocato l'esigenza di ordine: **RefSeq**.

RefSeq è stato ideato per far corrispondere a ciascun trascritto normalmente prodotto da un gene e a ciascuna proteina una sequenza di riferimento, un identificatore (accession number).

Altri esempi di identificatori NON RefSeq sono:

- X02775 GenBank/EMBL/DDBJ nucleotidic sequence
- Rs7079946 dbSNP (single nucleotide polymorphism)
- N91759.1 An expressed sequence tag
- AAC02945 GenBank protein
- Q28369 SwissProt protein
- 1KT7 Protein Data Bank structure record

Refseq fornisce un identificatore per la sequenza di riferimento, curato dal personale dell'NCBI. I formati principali degli id RefSeq sono:

- Complete genome/chromosome/plasmid **NC**\_#####
- Genomic contig (segmenti sovrapposti di DNA segments che rappresentano una sequenza consenso) **NT**\_#####
- mRNA (DNA format) **NM**\_#####
- Protein **NP**\_#####

**Un primo esempio di ricerca - L'Emoglobina** Una delle prime proteine ad essere studiata (anni '30 e '40, da Mulder, Liebing et al.).

È stata la prima proteina ad essere usata negli allineamenti multipli di sequenza: voglio fare dei confronti di sequenze (ad esempio per confrontare la stessa proteina prodotta da diverse specie). Con le prime tecniche di sequenziamento abbiamo scoperto che è stata localizzata in due loci, uno sul cromosoma 16 (subunità alfa) e 11 (subunità beta). I due geni sono regolati sia in base all'età che in base ai diversi tessuti.

È quindi un problema complesso che ha poi originato una serie di considerazioni. La mioglobina, una globina (struttura globulare a 8 eliche) che lega l'ossigeno nei tessuti muscolari, è stata la prima proteina la cui struttura tridimensionale è stata risolta tramite cristallografia.

L'emoglobina è un tetramero (due catene alfa e due beta negli adulti) è il principale trasportatore di ossigeno nei vertebrati. Assieme alla mioglobina è stata usata nei primi studi sugli allineamenti multipli.

Negli anni '80 con le prime tecniche di sequenziamento è stata localizzata in due loci, uno sul cromosoma 16 (subunità alfa) e 11 (subunità beta). I due geni sono regolati sia in base all'età che in base ai diversi tessuti.

## Ricerca dell'emoglobina

1. Inseriamo "beta globin" nella barra di ricerca
2. Seguiamo poi il link a "Gene"
3. Entrez Gene (ex LocusLink) è un portale curato che descrive loci genetici
  - nomenclatura
  - alias
  - accession numbers
  - fenotipi
  - OMIM (ereditarietà dei caratteri)
  - HomoloGene
  - mappatura sul genoma
  - collegamenti esterni
4. In generale ad oggi questa ricerca trova 126 entries
5. Intestazione: Entrez Gene, Noa: "Official Symbol", HBB per la beta globina
6. Limitiamoci alla ricerca per Homo Sapiens (selezionando sulla destra da Results by taxon)
7. Cliccando la specie si aggiorna automaticamente la stringa di ricerca: (beta globin) AND "Homo sapiens" [ porgn:\_txid9606 ]
8. Con il limite Homo Sapiens le entries sono solo 41
9. Apriamo la prima entry
10. Sulla dx in basso troviamo numerosi link a database esterni
11. Abbiamo una sezione sulle regioni genomiche, una sulla bibliografia
12. Sezione interessante: GeneRif (intended to facilitate access to publications documenting experiments that add to our understanding of a gene and its function)
13. E ancora Fenotipi, Variazione Genica, Pathways per Biosistemi e Interazioni note con altri geni.
14. Ontologia: (fondamentale per sistemi automatici di apprendimento). Classificazione e organizzazione dei dati in categorie predefinite così da agevolare l'individuazione di analogie e caratteristiche primarie. Può essere di diversi tipi, ma la principale distingue:
  - Funzione molecolare
  - Localizzazione cellulare
  - Processo biologico
15. Catalogazione RefSeq (a fine pagina)

## 6.2 Operatori Booleani

### 6.2.1 Operatore AND (&)

Restringe il campo di ricerca, inserendo ad es. la stringa: equus caballus AND hemoglobin alpha  
La banca dati ci mostrerà una lista di sequenze proteiche i cui campi di descrizione contengono entrambe le parole. Quindi le sequenze proteiche del cavallo che non contengono nella descrizione la parola hemoglobin non vengono selezionate.

### 6.2.2 Operatore OR (|)

Estende il campo di ricerca, digitando ad esempio: Restringe il campo di ricerca, inserendo: **homo sapiens OR mus musculus**

Otterremo una lista di sequenze i cui campi contengono la parola homo sapiens o la parola mus musculus. L'operatore allarga l'insieme delle sequenze che incontrano le nostre esigenze.

### 6.2.3 Operatore NOT (!)

Restringe il campo di ricerca, inserendo: **homo sapiens NOT hemoglobin** Richiederemo sequenze i cui campi contengono la parola homo sapiens ma non la parola hemoglobin.

### 6.2.4 Combinazione di Operatori Booleani

Gli operatori booleani si possono combinare, vengono letti da sinistra a destra. Per questo sono utili le parentesi. Ad esempio: globin AND promoter OR enhancer produce quasi 5000 hits. Ma se si scrive globin AND (promoter OR enhancer) se ne ottengono circa 70.

Altre possibilità sono:

- Specificare un organismo (human, nella query: **human[ORGN]**)
- Usare l'asterisco: **glob \*** restituisce tutte le entry che contengono una stringa che inizia per "glob"
- Usare le virgolette “”. La ricerca di “**toxin B1**” restituirà le entries che contengono esattamente la stringa intera.
- ...

## 6.3 Nel dettaglio

**Homologene** la risorsa ideale per individuare gruppi di geni omologhi negli eucarioti presenti in NCBI

**OMIM** Catalogo di geni umani e disordini genetici

**SNP** Single Nucleotide Polymorphism

## 7 Proteine - Le banche dati proteiche più usate

**Uniprot** (Universal Protein Resource) raccoglie le informazioni dei database:

1. Swiss-prot (SIB)
2. TrEMBL (EBI)
3. PIR

Offre la possibilità di effettuare Text Search o Blast Search. Viene curato anche un database NON RIDONDANTE (UniRef).

**Swissprot** Molto curato e dettagliato, con annotazioni circa funzione, struttura, modificazione e altre informazioni utili.

**TrEMBL** È la traduzione in silico di ogni entry codificante del database primario dell'EMBL, non è accurato, ma è ricchissimo.

**PIR** È il discendente diretto del database della Dayhoff, è curato a mano e le annotazioni sono molto ricche e precise.

### 7.1 NCBI Protein - non molto ricco

Entrez Protein: Contiene diverse informazioni su proteine

- 147 amminoacidi
- PRI: primates
- *NP\_000509* (protein accession number)
- *NM\_000518.4* (mRNA, RefSeq)
- Riferimenti bibliografici
- Sequenze FASTA (Opzione Display)
- Siti di modifica post-traduzionale (AA94, AA121)
- Riferimenti ad altri database
- Sequenza amminoacidica (1 lettera)

È un record non molto ricco dal punto di vista dei dati delle proteine.

### 7.2 Uniprot

Uniprot è il più completo database centralizzato per le sequenze proteiche.  
È organizzato su 3 livelli:

1. Uniprot Knowledge Base
  - Swiss-Prot (curato)
  - TrEMBL (automatico)
2. UniProt Reference clusters (UniRef)
  - Cluster di proteine che condividono il 50%, 90%, 100% di identità di sequenza
3. UniProt Archive (UniParc)
  - Archivio di sequenze proteiche stabile, non ridondante, da diverse 58 fonti

### 7.2.1 Struttura del database

Nella homepage abbiamo la classica barra di ricerca e subito sotto i link di accesso alle diverse informazioni contenute in Uniprot.

#### Un esempio di ricerca

1. Inseriamo "hbh" nella barra di ricerca.
2. Sulla sinistra possiamo selezionare gli organismi a cui restringere la ricerca. Selezioniamo Humans.
3. Questo aggiornerà automaticamente la stringa di ricerca: hbh AND organism: "Homo sapiens (Human) [9605]"
4. Selezioniamo la prima entry.
5. Sulla sinistra troviamo la tavola con tutti i contenuti disponibili.
6. Tra i più importanti abbiamo: "Function" (che specifica la funzione della proteina), "Pathology & Biotech", "Expression", "Interaction", "Family & Domains", ...
7. In "Structure" e altre sezioni troviamo i link a PDB (Protein Data Bank), database di strutture proteiche.
8. In "Sequence" troviamo tutta la sequenza proteica, scaricabile in formato FASTA.
9. Abbiamo inoltre vari link di collegamento ad altri database di sequenze (EMBL, GeneBank, DDBJ), vari-anti, ...

### 7.3 ExPASy

(Expert Protein Analysis System)

È una risorsa curata, espressione del SIB (Swiss Institute of Bioinformatics). Principalmente dedicata alle proteine.

La risorsa principale che ha prodotto è SwissProt (confluìta in Uniprot). Rimane un punto di riferimento per molti tools.

## **Lezione 3: Allineamenti di Sequenze - concetti e algoritmi**

## 8 Allineamenti di Sequenze

Un primo e precoce allineamento di sequenze si ha nel 1961: H.C. Watson and J.C. Kendrew, "Comparison Between the Amino-Acid Sequences of Sperm Whale Myoglobin and of Human Hæmoglobin." Nature 190:670-672, 1961.

**L'allineamento di sequenze a coppie è un'operazione fondamentale in bioinformatica** È utilizzato per decidere se due proteine (o geni) sono correlate strutturalmente e funzionalmente. Viene utilizzato per identificare i domini o motivi che sono condivisi tra le proteine. È alla base della ricerca con BLAST (prossime lezioni) e viene utilizzato anche per l'analisi dei genomi.

**Allineamento a coppie: sequenze di proteine possono essere più informative del DNA** Le proteine sono più informative del DNA (20 vs 4 caratteri); molti aminoacidi condividono proprietà biofisiche. Ricordiamo che i codoni sono degenerati: i cambiamenti in terza posizione spesso non alterano l'aminoacido che ne è specificato (mutazioni sinonime). Le sequenze di proteine offrono un più lungo tempo di "look-back" e le sequenze di DNA possono essere tradotte in proteine, e poi utilizzate negli allineamenti a coppie.

### 8.1 Definizione - Allineamento a coppie

Il processo che allinea due sequenze per raggiungere livelli massimi di identità (e conservazione, nel caso di sequenze di amminoacidi) al fine di valutare il grado di similitudine e la possibilità di omologia.

### 8.2 Altre definizioni

- **Identità**

La misura in cui due sequenze (di nucleotidi o aminoacidi) sono invarianti. (es. identità del 32% => 32 a.a. su 100 sono ordinatamente identici)

- **Conservazione**

In una sequenza, modifiche in una specifica posizione di un amminoacido (o meno comunemente, di un nucleotide) che preservano le proprietà fisico-chimiche del residuo originale.

- **Similitudine**

La misura in cui due sequenze (di nucleotidi o aminoacidi) sono correlate. Si basa su identità + conservazione.

- **Omologia**

Similitudine attribuita a discendenti da un antenato comune.

**! Nota bene:**

- OMOLOGIA indica che due entità (es. 2 sequenze) hanno una stessa origine filogenetica, cioè derivano da un antenato comune. È un carattere QUALITATIVO.

- SIMILITUDINE indica che due entità (es. 2 sequenze), in relazione ad un certo criterio comparativo, hanno un certo grado di similitudine. È un carattere QUANTITATIVO (vedremo tra breve come definirla).

**! Osservazioni:**

- La struttura di una proteina dipende della sua sequenza di a.a. (concetto alla base del Protein Folding).

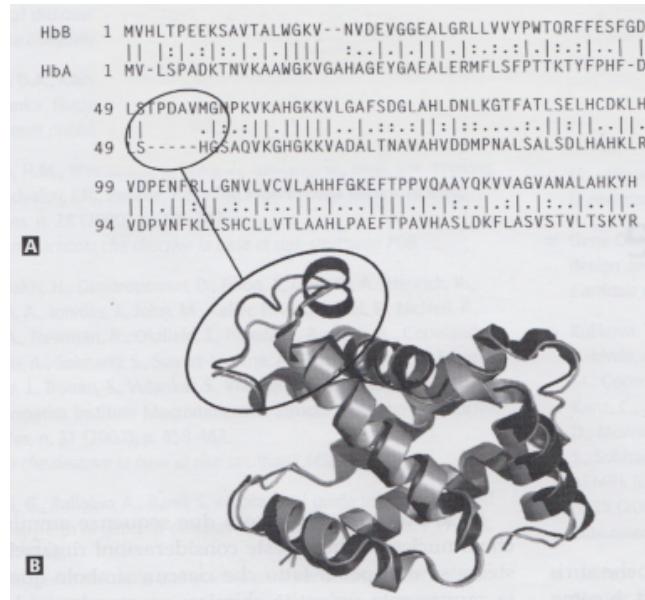
- La struttura determina la funzione molecolare della proteina.

- Se una sequenza proteica è conservata durante l'evoluzione ed è quindi presente in organismi diversi (famiglia di proteine) è ragionevole assumere che le funzioni che svolge siano simili o per lo meno correlate.

**! Passi per predizione di funzione:**

1. Identificazione delle proteine di una famiglia (evolute da un progenitore comune → sequenza di a.a. abbastanza simile.)
2. identificazione degli a.a. che svolgono un ruolo strutturale o funzionale analogo (allineamento).

**Esempio 1:** la catena  $\beta$  dell'emoglobina e mioglobina «si somigliano»



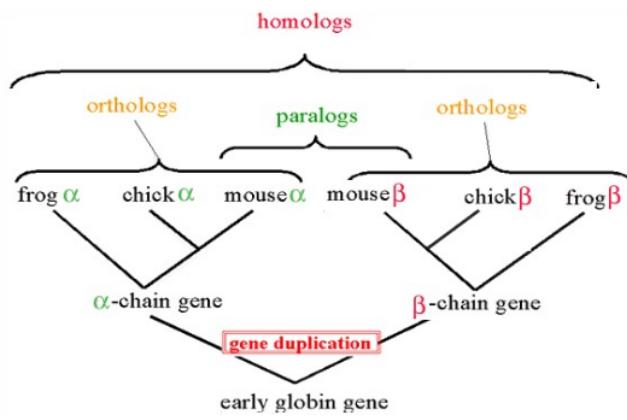
**Figure 1:** Le zone con indel nelle sequenze sono strutturalmente dissimili

### • Ortologi

Sequenze omologhe in diverse specie che derivano, tramite la speciazione, da un gene ancestrale comune. La funzione può essere o non essere simile.

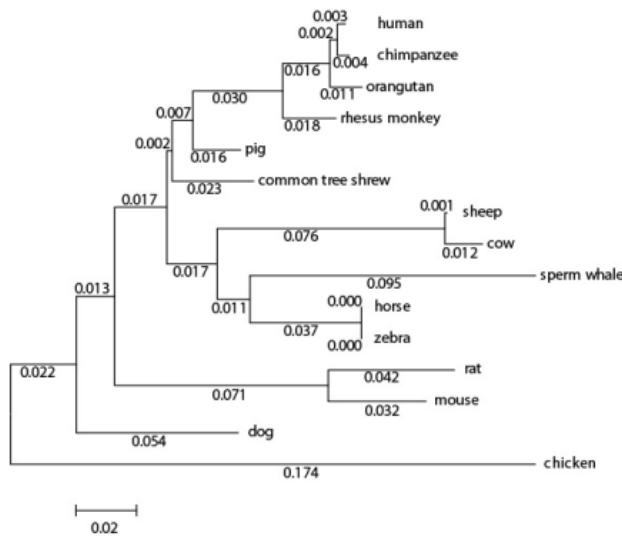
### • Paraloghi

Sequenze omologhe all'interno di una singola specie sorte dalla duplicazione genica.



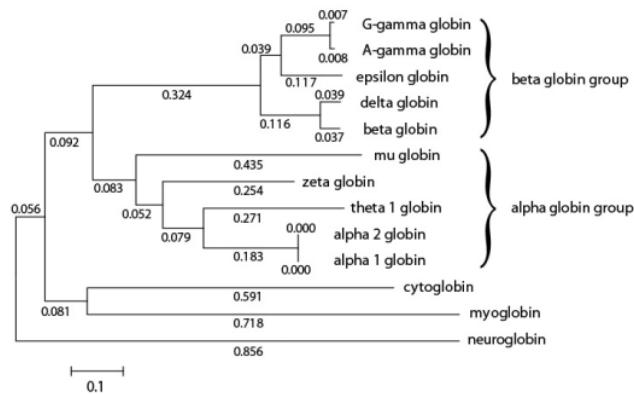
**Figure 2:** Ortologi e paraloghi sono spesso rappresentati in un albero singolo.

Ortologhi: membri di una famiglia di geni (proteine) in vari organismi.



**Figure 3:** Questo albero mostra gli ortologhi della globina.

Paraloghi: i membri di una famiglia di geni (proteine) all'interno di una specie.



**Figure 4:** Questo albero mostra i paraloghi della globina umana.

## 9 Confrontare due sequenze

**Come posso trasformare una stringa in un'altra?** Un modo semplice per capirlo è allineare le due stringhe:

ESEMPIO: 1 - LA CASA NUOVA; 2 - LA CASSA VUOTA

```

1 - L A C A - S A N U O V A
2 - L A C A S S A V U O T A
                    oppure
1 - L A C A - S A - N U O V A
2 - L A C A S S A V - U O T A

```

Nel secondo caso c'è un'operazione in più.

- Il numero minimo di operazioni necessarie per allineare due sequenze ne misura la distanza.

- La Natura dispone di varie operazioni per trasformare un oggetto nell'altro (mutazioni, indel...)
- L'evoluzione sceglie la via piu' breve (principio di massima parsimonia); cio' si manifesta tramite l'analisi dell'allineamento.

Dobbiamo avere chiari i concetti di match (residui appaiati), mismatch (sostituzioni) e gap (indel).

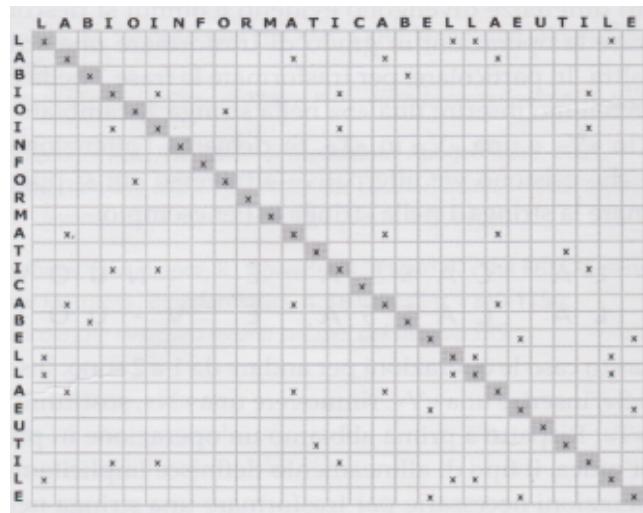
## 9.1 Come identificare le zone di somiglianza locale tra due sequenze?

### 9.1.1 Matrice a punti - dot plot

È un modo relativamente semplice.

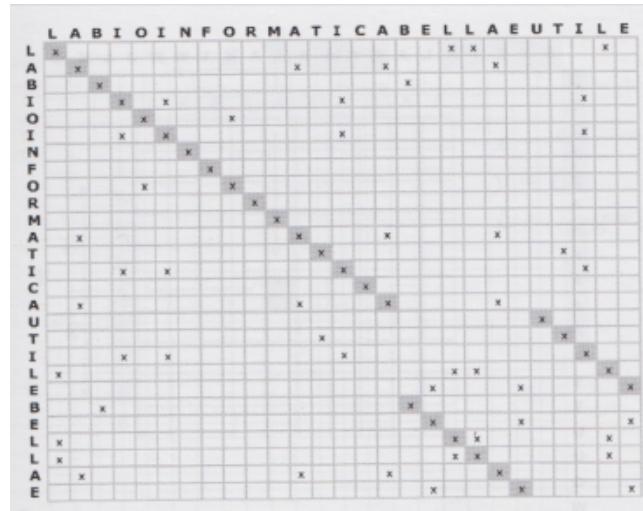
Confrontiamo la stringa con se stessa (autoconfronto):

1. Mettiamo una x per ogni identità



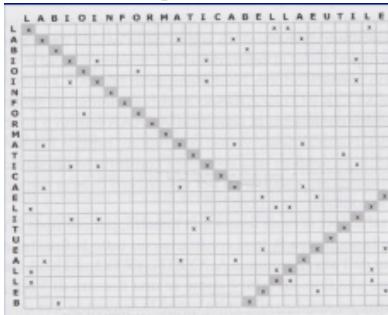
**Figure 5:** ... piuttosto banale. Cambiamo la seconda stringa.

Effettuiamo un'inversione. Il pattern delle diagonali lo mostra chiaramente: la diagonale principale si spezza ma porzioni delle stringhe sono identiche.

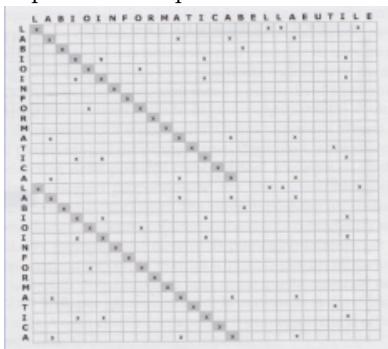


Possiamo individuare facilmente alcuni patterns mediante le matrici a punti (la prima stringa in alto resta immutata):

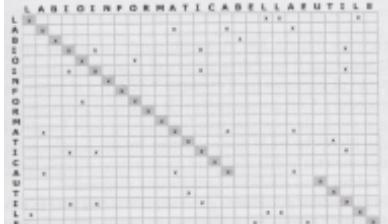
- Inversione di parole



- Ripetizione di parole



- Delezione di caratteri

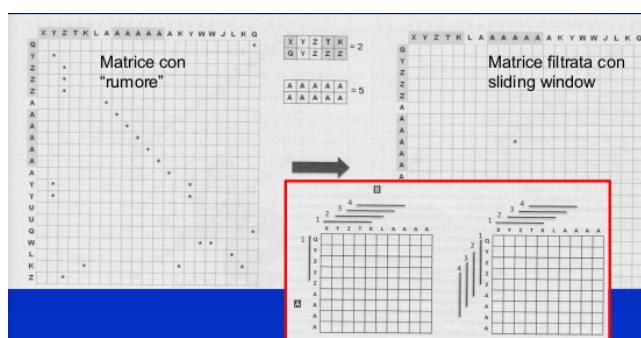


Se però allineamo sequenze di acidi nucleici (solo 4 lettere) il “segnaletico” di similitudine è mascherato dal grande rumore di fondo. Servono FILTRI per ridurre il rumore.

**Una semplice osservazione:** le zone delle sequenze più simili localmente si distribuiscono su diagonali; le altre somiglianze puntiformi si distribuiscono casualmente. Allora, è meglio confrontare le sequenze non per singole posizioni, ma per interi segmenti (FINESTRE).

Potremmo usare delle finestre scorrevoli.

ESEMPIO: confrontiamo una finestra di 5 residui in una seq con una finestra di 5 residui nell'altra. Confronto tutte le finestre (facendole scorrere), e metto una \* al centro della finestra solo se ho un match totale.



In generale: si fa scorrere una finestra alla volta.

## La procedura

1. Definiamo la posizione di una casella  $(x, y)$ .
2. Fissiamo il centro della finestra, di raggio  $g$ .
3. La lunghezza della finestra è dunque:  $L = 2g + 1$
4. Il numero  $N$  di residui identici in quella finestra è allora:  $N(x, y) = \sum_{h=-g}^{+g} S(x + h, y + h)$   
 $S = 1$  se il carattere in  $x + h$  è identico a quello in  $y + h$ ; altrimenti  $S = 0$

Questa regola è però molto restrittiva. A noi interessa anche la similitudine, non solo l'identità. Potremmo definire una soglia  $s$ , per cui:

- Se  $N(x, y) > s$  mette un simbolo nella casella in posizione  $x, y$ .

Dobbiamo quindi misurare la similitudine, e.g. tra aa o basi. Un esempio di matrice di punteggio (non di punti!!) per seq di nucleotidi può essere:

Esempio: secondo la matrice di punteggio a sx, l'allineamento ha punteggio 8

	A	T	C	G
A	2	1	0	0
T	1	2	0	0
C	0	0	2	1
G	0	0	1	2

A A A T C C G A A  
 A T A C A G A T T  
 2 +1 +2 +0 +0 +1 +0 +1 +1 = 8

Misurata la similitudine, possiamo allora attribuire un nuovo punteggio al confronto fra due finestre:

- Se  $N(x, y)$  è il numero dei residui simili, ed è la media dei punteggi delle singole coppie prelevati dalla matrice di punteggio scelta:
- $N(x, y) = \sum_{h=-g}^{+g} \frac{S(x+h, y+h)}{L}$

$S$  ora dipende da quale matrice di punteggio sceglio e dalla lunghezza della finestra. Possiamo quindi essere un po' più "elastici" pur mantenendo la regola:

- Se  $N(x, y) > s$  mette un simbolo nella casella in posizione  $x, y$ .
- lo decide la matrice di punteggio (cioè la similitudine).

### 9.1.2 In breve

In conclusione, la visualizzazione (ed il calcolo) di una matrice a punti dipende da:

1. La lunghezza  $L$  della finestra scorrevole scelta
2. Il metodo per misurare la similitudine  $S(x, y)$
3. La soglia  $s$  per "marcare" la casella rispettiva

In pratica conviene fissare 2 parametri e variare il terzo per rendere le zone di similitudine più evidenti. Molti programmi fanno questo.

Es. DOTTER/Dotlet - assegna un colore dipendentemente da  $S$ .

**Nota:** Analogamente all'identità di sequenza, che si può misurare in percentuale (%), anche la similitudine, una volta quantificata, si può misurare in %.

$S_1$  e  $S_2$  sono due sequenze lunghe, rispettivamente,  $L_1$  ed  $L_2$ .

Scelta  $L_1$  come riferimento si ha:

$$\text{SequenceIdentity}(s.i.) = \left( \frac{\#\text{matches}}{L_1} \right) * 100$$

$$\text{SequenceSimilarity}(s.s.) = \left( \frac{S_1 \text{ vs. } S_2 \text{ s.c.}}{S_1 \text{ vs. } S_2 \text{ i.c.}} \right) * 100$$

*s.c.* = similarity score, ed è ottenuto allineando  $S_1$  con  $S_2$ , e attribuendo il punteggio ottenuto dalla matrice discore.

*i.c.* = identity score ed è ottenuto allineando  $S_1$  con sè stessa ed attribuendo il punteggio ottenuto dalla matrice di score

Nota: se sono presenti indel, a denominatore metto la lunghezza dell'allineamento (compresi gli indel), e non la lunghezza originale della sequenza.

## 10 Algoritmi dinamici di allineamento

I dot plots non tengono in considerazione gli indel. Occorrono altri algoritmi che, passo a passo e seguendo una certa direzione, trovino l'allineamento con:

- Maggior numero di simboli identici
- Minor numero di indel (sfavorite evolutivamente)

Esempio: proviamo tutte le combinazioni da sx a dx, riempiendo una colonna alla volta (qui mostriamo solo i primi 3 residui!!)

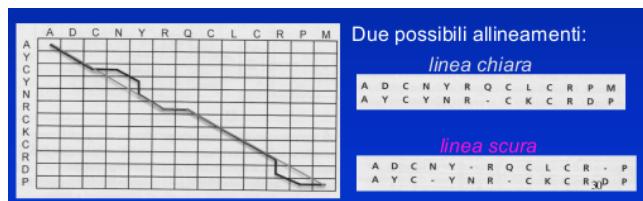
Stabiliamo inoltre i punteggi:

- -1 per indel
- 0 per residui diversi
- +1 per residui identici

Il numero delle combinazioni possibili è molto grande, specie per sequenze lunghe. Nel 1970 NEEDLEMAN e WUNSCH creano un algoritmo, poi migliorato ed esteso. Noi analizziamo l'originale.

### 10.1 Il concetto

distribuiamo le due sequenze in una matrice. Il possibile allineamento tra le due identifica un percorso che unisce le caselle dei residui appaiati.



### 10.2 Regole pratiche

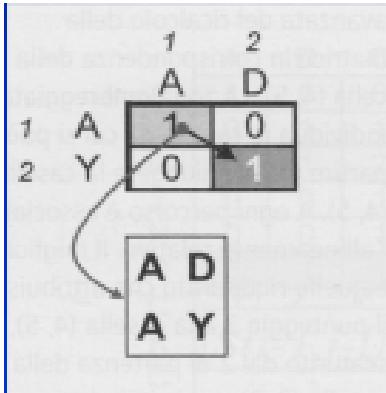
1. Il percorso ha una direzione e procede solo in avanti (NON si torna indietro!)
2. Occorre trovare il percorso con il **maggior** numero di aa identici e il **minor** numero di indel
3. Occorre anche tener conto della similitudine fra amminoacidi (significato evolutivo)
4. IMPORTANTE: un allineamento ottimale è sempre composto da suballineamenti ottimali (cioè: togliendo uno ad uno i residui dal fondo, l'allineamento deve restare ottimale, per poter ricostruire il percorso a ritroso)

### 10.3 Step

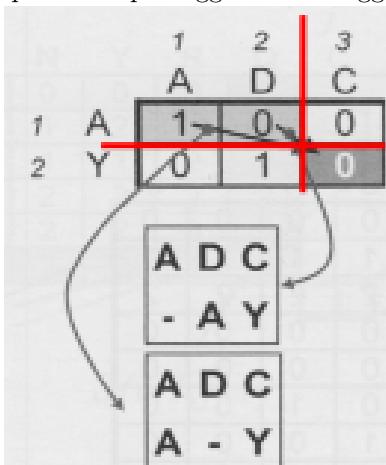
1. INIZIALIZZAZIONE della matrice: regola molto semplice; 1 se identici, 0 altrimenti. N.B. ora  $(x, y)$  identifica: (residuo in colonna, residuo in riga) (N.B. dopo aver trattato le matrici di score potremo normalizzare diversamente)

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	A	D	C	N	Y	R	Q	C	L	C	R	P	M
2	Y	0	0	0	1	0	0	0	0	0	0	0	0
3	G	0	0	1	0	0	0	0	1	0	1	0	0
4	Y	0	0	0	0	1	0	0	0	0	0	0	0
5	N	0	0	0	1	0	0	0	0	0	0	0	0
6	R	0	0	0	0	0	1	0	0	0	0	1	0
7	C	0	0	1	0	0	0	0	1	0	1	0	0
8	K	0	0	0	0	0	0	0	0	0	0	0	0
9	C	0	0	1	0	0	0	0	1	0	1	0	0
10	R	0	0	0	0	1	0	0	0	0	1	0	0
11	D	0	1	0	0	0	0	0	0	0	0	0	0
12	P	0	0	0	0	0	0	0	0	0	0	1	0

2. Partiamo da (1,1) (in alto a sx: la direzione è importante!!): (1,1) → (2,2) L'unico percorso possibile è questo: per "ricordarmelo" sommo il valore della cella (1,1) a quello della cella (2,2) da matrice inizializzata



3. Prossimo step: verso (2,3). Ci sono due possibili percorsi, corrispondenti a due diversi allineamenti: scelgo quello con punteggio finale maggiore, sempre sommando casella precedente a (2,3)



4. Procediamo analogamente: ad esempio, verso (4,4)

	C - - N				C - N				
	A	Y	C	Y	A	Y	C	Y	
1	A	1	0	0	0	0	0	0	0
2	Y	0	1	1	2	1	1	1	1
3	C	0	1	2	1	1	2	2	2
4	Y	0	0	1	0	0	0	0	0
5	N	0	0	1	0	0	0	0	0
6	R	0	0	0	0	1	0	0	0
7	C	0	0	1	0	0	0	1	0
8	K	0	0	0	0	0	0	0	0
9	C	0	0	1	0	0	0	1	0
10	R	0	0	0	0	1	0	0	0
11	D	0	1	0	0	0	0	0	0
12	P	0	0	0	0	0	0	0	1

Questo è l'allineamento  
migliore: aumentiamo il  
punteggio di (4,4) di 2

5. e oltre, ad es. verso (4,5): qui il nuovo punteggio sarà 3

	N - - Y				N - Y				
	A	Y	C	Y	A	Y	C	Y	
1	A	1	0	0	0	0	0	0	0
2	Y	0	1	1	2	1	1	1	1
3	C	0	1	2	1	1	2	2	2
4	Y	0	0	1	0	0	0	0	0
5	N	0	0	0	1	0	0	0	0
6	R	0	0	0	0	1	0	0	0
7	C	0	0	1	0	0	0	1	0
8	K	0	0	0	0	0	0	0	0
9	C	0	0	1	0	0	0	1	0
10	R	0	0	0	0	1	0	0	0
11	D	0	1	0	0	0	0	0	0
12	P	0	0	0	0	0	0	0	1

6. Alla fine arriveremo fino all'ultima casella in basso a dx:

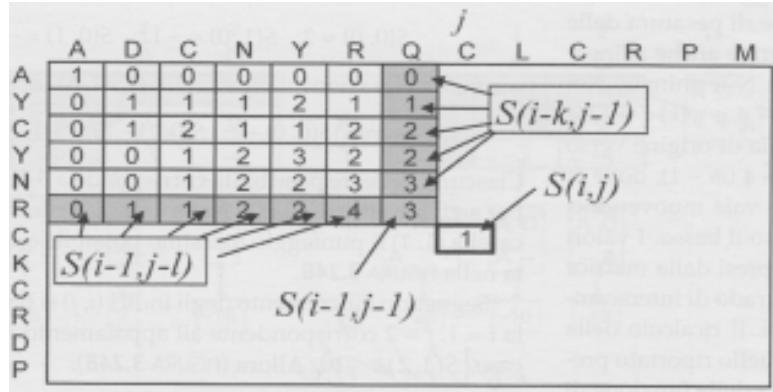
	1	2	3	4	5	6	7	8	9	10	11	12	13
	A	D	C	N	Y	R	Q	C	L	C	R	P	M
1	A	1	0	0	0	0	0	0	0	0	0	0	0
2	Y	0	1	1	2	1	1	1	1	1	1	1	1
3	C	0	1	2	1	1	2	2	3	2	3	2	2
4	Y	0	1	1	2	3	2	2	2	3	3	3	3
5	N	0	1	1	3	2	3	3	3	3	3	3	3
6	R	0	1	1	2	3	4	3	3	3	3	4	3
7	C	0	1	2	2	3	3	4	5	4	5	4	4
8	K	0	1	1	2	3	3	4	4	5	5	5	5
9	C	0	1	2	2	3	3	4	4	5	5	5	5
10	R	0	1	1	2	3	4	4	5	5	5	6	6
11	D	0	2	1	2	3	3	4	4	5	5	6	7
12	P	0	1	2	2	3	3	4	4	5	5	6	7

7. Da essa possiamo spostarci a ritroso poiché abbiamo memorizzato i migliori punteggi dalle caselle precedenti. Abbiamo così determinato il miglior allineamento:

A	D	C	N	Y	R	Q	C	L	C	R	P	M	Il numero di identità
A	Y	C		Y	N	R	C	K	C	R	D	P	
1	0	1	0	1	0	1	0	1	0	1	1	0	= 8

Qui il punteggio totale coincide con il numero di residui identici.  
 $\frac{8}{15} = 0.53 = 53\%$  identità di sequenza

**In termini formali . . .** La casella  $(i,j)$  avrà lo score  $S(i,j)$  ricalcolato a partire dalla matrice di inizializzazione in questo modo:



$$S(i, j) = s(a, b) + \max[S(i - 1, j - 1), S(i - k, j - 1), S(i - 1, j - l)]$$

Dovremmo però trovare un modo più efficace di inizializzare la matrice tenendo conto della similarità fra aa.

#### 10.4 Needleman-Wunsch: programmazione dinamica

NW garantisce l'ottimalità dell'allineamento, anche se l'algoritmo non calcola tutti i possibili allineamenti. È un esempio di un algoritmo di programmazione dinamica: un percorso ottimale (allineamento) è identificato dall'estensione graduale di sottopercorsi localmente ottimali. Dunque, una serie di decisioni è effettuata ad ogni passo dell'allineamento per trovare la coppia di residui con il miglior punteggio per quel passo.

L'algoritmo di Needleman-Wunsch è disponibile presso EBI, che ospita molti tools per allineamenti locali e globali di sequenze (Pairwise Sequence Alignment).

## **Lezione 4: Allineamenti di Sequenze 2**

### **Matrici di Sostituzione**

## 11 Waterman-Smith

Un problema dell'algoritmo di Needleman-Wunsch: non si tiene conto della penalizzazione delle indel.  
L'algoritmo di WATERMAN-SMITH (1976) introduce una funzione di penalizzazione delle indel, per migliorare l'algoritmo NW, serve un sistema di pesatura delle indel, ad esempio:

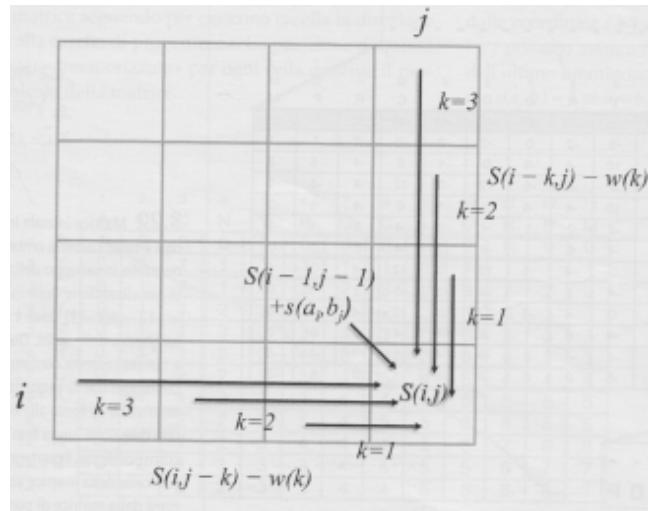
$$w(k) = g + e(k - 1)$$

Il peso  $w$  di una indel di lunghezza  $k$  dipende dalla penalizzazione per l'apertura di una singola indel ( $g$ ) e dalla penalizzazione per l'allungamento ( $e$ ).

### 11.1 L'algoritmo

Nella pratica l'algoritmo procede in questo modo:

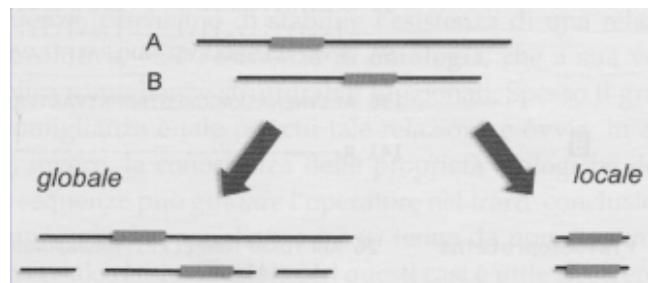
1. Inserisce una riga e una colonna 0-ime alla matrice di inizializzazione (calcolata ad esempio partire da BLOSUM o PAM che vedremo, ecco perché non ha solo 0 o 1).  
Nella riga e colonna ombreggiate è sviluppata la funzione di penalizzazione:  $w(k) = -12 - 4(k - 1)$   
La riga e la colonna 0-sime contengono il punteggio che la sequenza avrebbe se allineata a una delezione lunga fino alla cella corrispondente.
2. Tiene conto dei possibili modi per arrivare alla casella  $(i, j)$ . Il suo punteggio  $S(i, j)$  dipende da essi:
  - (a) Mi muovo in diagonale: no indel e punteggio dato da: punteggio della casella di partenza + punteggio della casella  $(i, j)$  secondo la matrice di inizializzazione (come in NW)
  - (b) Mi muovo in verticale o orizzontale: inserisco indel nella sequenza  $i$  e  $j$ . Il punteggio sarà dato da: punteggio della casella di partenza - funzione di penalizzazione  $w(k)$  ( $k$  è la lunghezza della indel).
  - (c) Scelgo alla fine il percorso che dà il punteggio migliore



## 12 Allineamento: globale vs locale

**Allineamento globale** (NW e SW visto finora) si estende da un capo all'altro di ogni sequenza.

**Allineamento locale** trova le regioni (sottosequenze) di due sequenze che si allineano in modo ottimale.

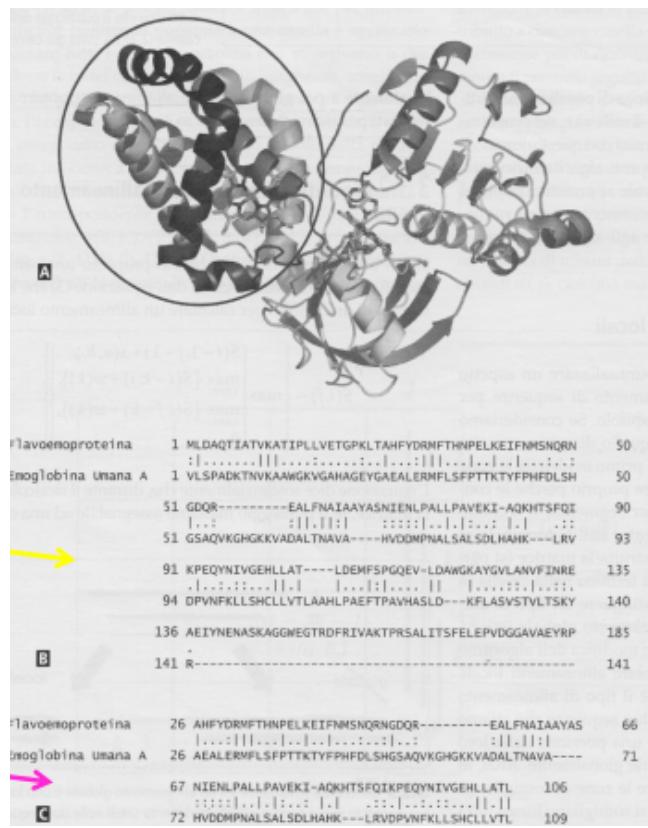


Qui l'allineamento globale maschera la corrispondenza tra zone somiglianti

SW si può modificare per renderlo capace di calcolare allineamenti locali: basta introdurre fra i casi possibili  $S(i, j) = 0$  nel caso in cui nell'allineamento globale fosse negativo

**Esempio:** L'allineamento fra una flavoemoproteina (con un dominio di tipo emoglobinico) e la catena A dell'emoglobina umana

- **globale:** più difficile notare quantitativamente la similitudine
- **locale:** più apparente



**Esempio:** SW locale:dopo aver ricalcolato la matrice cerco la cella con il valore massimo assoluto e parto da lì.

Gli stessi due peptidi di prima, allineati con Waterman-Smith globale e locale danno luogo a matrici ed allineamenti diversi. Partendo dalle caselle con score maggiore il percorso a ritroso individua allineamenti differenti (non sempre AL è sottoinsieme di AG)

	A	D	C	N	Y	R	S	C	L	C	R	P	M	
A	0	-12	15	-20	-24	-28	-32	-38	-40	-44	-48	-52	-56	-60
D	-12	2	-10	-14	-18	-22	-26	-30	-34	-38	-42	-46	-50	-54
C	-16	-10	-2	0	8	-12	-16	-20	-24	-28	-32	-36	-40	-44
N	-24	-18	-10	-2	0	8	-4	-8	-12	-16	-20	-24	-28	-32
Y	-32	-26	-23	-10	-6	-4	-12	-13	-20	-24	-28	-32	-36	-40
R	-32	-26	-23	-10	-6	-4	-12	-13	-20	-24	-28	-32	-36	-40
S	-28	-22	-16	-6	0	6	8	-3	-8	-12	-16	-20	-24	-28
C	-40	-34	-30	-18	-10	-14	-3	1	6	18	6	0	0	0
L	-44	-38	-38	-18	-20	-10	-6	8	13	6	30	18	14	10
C	-44	-38	-38	-18	-20	-10	-6	8	13	6	30	18	14	10
R	-48	-42	-39	-26	-18	-22	-4	-3	1	10	18	36	24	20
P	-46	-40	-39	-26	-24	-22	-15	-2	-3	-2	14	24	36	23
M	-56	-50	-47	-34	-31	-29	-20	-14	-9	-10	20	30	36	39

	A	D	C	N	Y	R	S	C	L	C	R	P	M	
A	0	-12	-16	-20	-24	-28	-32	-36	-40	-44	-48	-52	-56	-60
D	-12	2	0	0	0	0	0	0	0	0	0	0	0	0
C	-16	0	0	0	0	10	0	0	0	0	0	0	0	0
N	-24	0	0	0	0	0	10	0	2	0	11	0	8	0
Y	-32	0	0	0	0	0	0	10	0	2	0	11	1	7
R	-32	0	0	0	0	0	0	0	14	11	0	0	0	13
S	-36	0	0	0	0	0	0	0	12	1	13	0	0	0
C	-40	0	0	0	0	0	0	13	3	0	11	20	0	10
L	-44	0	0	0	0	0	0	12	1	13	1	0	15	8
C	-44	0	0	0	0	0	0	12	1	13	1	0	15	8
R	-48	0	0	0	0	0	0	12	1	19	7	3	12	26
P	-46	0	0	0	0	0	0	8	0	21	9	5	16	26
M	-56	0	0	0	0	0	0	0	0	0	0	0	0	0

	A	D	C	N	Y	R	S	C	L	C	R	P	M	
A	0	-12	-16	-20	-24	-28	-32	-36	-40	-44	-48	-52	-56	-60
D	-12	2	0	0	0	0	0	0	0	0	0	0	0	0
C	-16	0	0	0	0	0	0	0	0	0	0	0	0	0
N	-24	0	0	0	0	0	0	0	0	0	0	0	0	0
Y	-32	0	0	0	0	0	0	0	0	0	0	0	0	0
R	-32	0	0	0	0	0	0	0	0	0	0	0	0	0
S	-36	0	0	0	0	0	0	0	0	0	0	0	0	0
C	-40	0	0	0	0	0	0	0	0	0	0	0	0	0
L	-44	0	0	0	0	0	0	0	0	0	0	0	0	0
C	-44	0	0	0	0	0	0	0	0	0	0	0	0	0
R	-48	0	0	0	0	0	0	0	0	0	0	0	0	0
P	-46	0	0	0	0	0	0	0	0	0	0	0	0	0
M	-56	0	0	0	0	0	0	0	0	0	0	0	0	0

### 12.0.1 In conclusione

- L'allineamento locale è quasi sempre utilizzato per il ricerche su database (tramite BLAST). È 'utile per trovare domini (o regioni limitate di omologia) all'interno di sequenze.
- Smith e Waterman (1981) hanno risolto il problema dell'allineamento locale ottimale di sequenze.
- Altri metodi (BLAST, FASTA) sono più veloci ma meno accurati. Li vedremo in seguito.
- In ogni caso, qualunque metodo di allineamento si scelga esso fornirà un punteggio S all'allineamento. Ricordiamo sempre che lo score S dipende dal metodo di allineamento e non è assoluto!

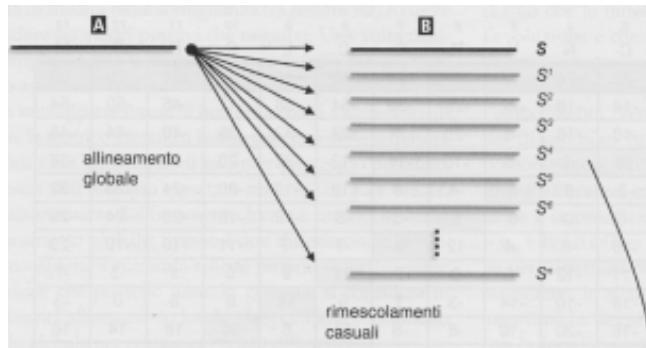
## 12.1 Significatività statistica di un allineamento

**Domanda:** Ho allineato due sequenze A e B, ottenuto il punteggio S. Come posso capire se sono omologhe? Che probabilità ho di trovare il punteggio S "per caso"?

Il problema è più facilmente risolvibile per gli allineamenti locali, e meno per quelli globali.

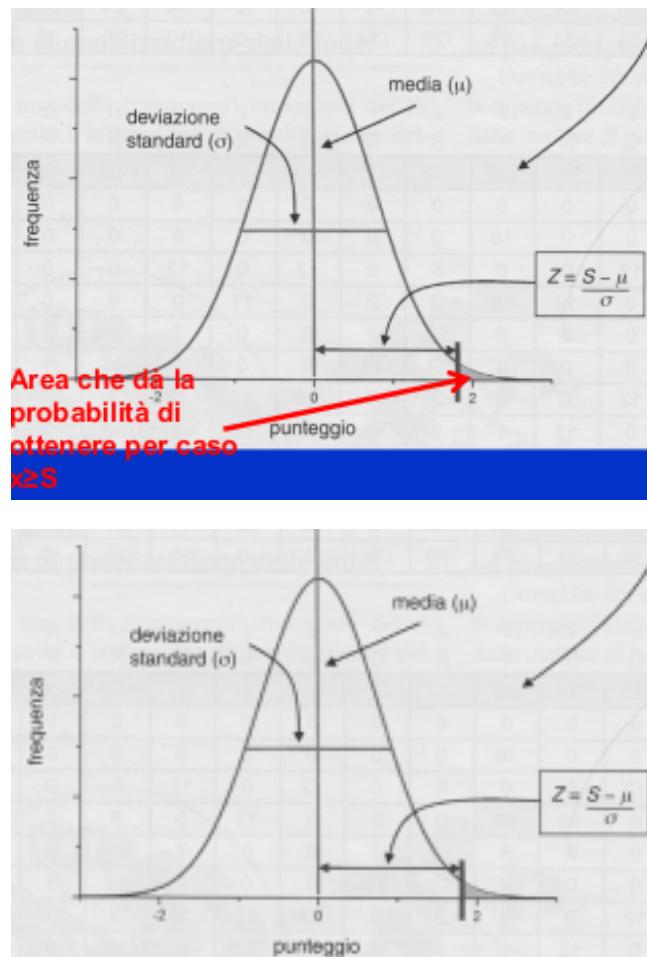
### 12.1.1 Significatività allineamento globale: lo Z score

La seq A è mantenuta fissa; la B è "anagrammata" n volte ed ogni volta globalmente allineata ad A, calcolando lo score  $S_i$  per l'allineamento  $i$ .



$S_i$  si distribuisce su una curva di cui si calcola la media  $\mu$  e la deviazione standard  $\sigma$ . Si definisce allora la distanza Z del punteggio S dell'allineamento dalla media in termini di dev. standard:

$$Z = \frac{S - \mu}{\sigma}$$



- Uno Z-score 0 = significa che la somiglianza osservata non è migliore rispetto alla media di permutazioni casuali della sequenza, e può anche essere casuale.
- Problema con Z-score: si assume una distribuzione normale, ma ciò può non esser corretto. Perciò Z deve essere considerato come una soglia di significatività.

### 12.1.2 Significatività allineamento locale

Teoria abbastanza complessa, sviluppata da Karlin e Altschul partendo da questa osservazione:

Date due sequenze casuali, di lunghezza  $m$  ed  $n$ , il numero atteso  $E$  di sottosequenze allineate localmente senza indel che ottengono un punteggio  $S \geq x$  è:

$$E(S \geq x) = Kmne^{-\lambda x}$$

$m, n$  : lunghezze delle due sequenze.

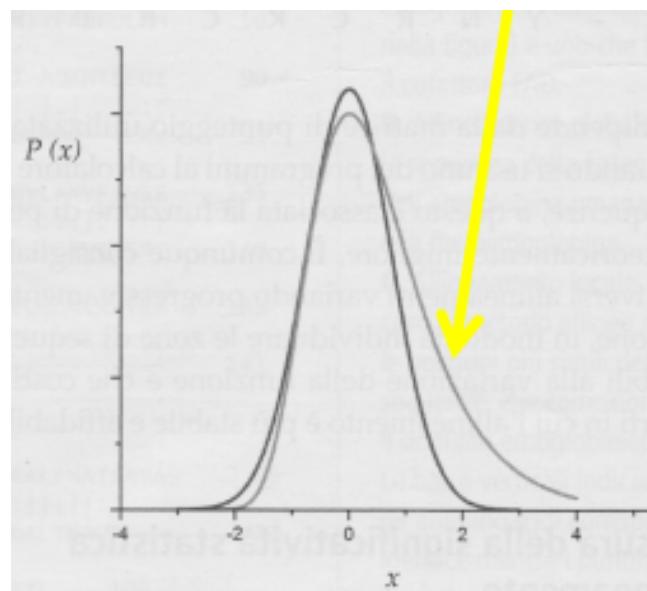
$K$ : dipende dalla matrice di punteggio.

$\lambda$ : dipende dalla composizione amminoacidica.

Dalla definizione di  $E$  si può calcolare la probabilità di osservare un allineamento locale con punteggio  $S \geq x$ :

$$p(S \geq x) = -\exp(Kmne^{-\lambda x})$$

Distribuzione del valore estremo o di Gumbel: è diversa dalla gaussiana



In pratica:

- allineamo localmente due seq
- otteniamo il punteggio  $x$
- calcoliamo  $p(S \geq x)$ , la probabilità di ottenere un punteggio maggiore di  $x$  nell'ipotesi: le due seq NON sono omologhe
- se  $p <$  soglia (es.  $0.01 = 1\%$ ) siamo confidenti che siano omologhe.
- SEMPRE: serve significatività BIOLOGICA oltre che statistica

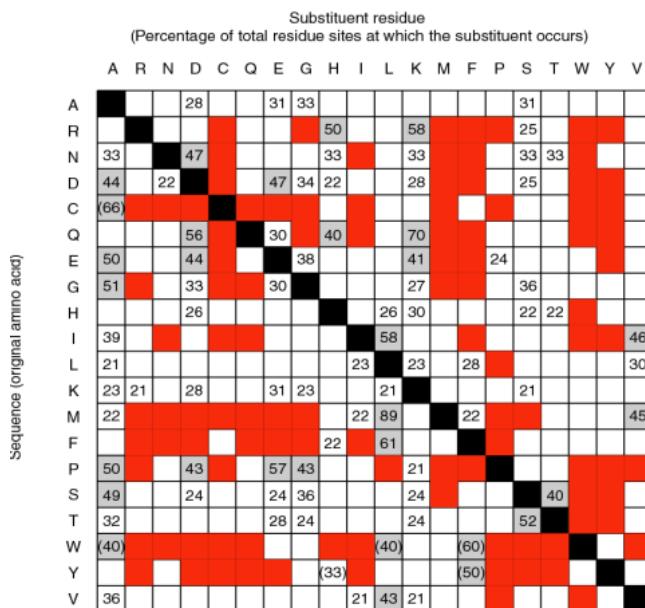
## 13 Matrici di punteggio

Abbiamo già visto che per dare un punteggio a un allineamento dobbiamo misurare la similitudine fra aa. Usiamo perciò matrici di punteggio o di sostituzione: saranno matrici  $20 \times 20$ . Sono matrici simmetriche:  $A \rightarrow B = B \rightarrow A$  (non sappiamo evolutivamente chi si è trasformato dei due).

**Come quantificare la somiglianza degli aminoacidi** Difficile stabilire criteri oggettivi per le somiglianze fisico-chimiche degli amino acidi. Non è possibile sapere a priori quali delle varie caratteristiche fisico-chimiche sono più importanti per le proteine.

Emile Zuckerkandl e Linus Pauling (1965) considerarono frequenze di sostituzione in 18 globine (mioglobine e emoglobine da uomo a lampreda).

- Nero: identità
- Grigio: sostituzione molto conservativa (occorrenza  $> 40\%$ )
- Bianco: sostituzione abbastanza conservativa (occorrenza  $> 21\%$ )
- Rosso: non è possibile osservare sostituzioni



**Cosa vogliamo ottenere?** una matrice (PAM250) di score. Matrice di calcolo che assegna i punteggi e tollera le discordanze. Inoltre ... tutta una serie di matrici di punteggio, (fino a PAM10) che è via via meno tollerante con i disallineamenti.

### 13.1 PAM: Point Accepted Mutation

Mutazione puntuale accettata.

- È l'evento in cui il DNA subisce una mutazione che produce il cambiamento di un aminoacido
- Tale mutazione diviene prevalente in una specie

Dayhoff ha osservato famiglie di sequenze identiche all'85% (omologhe e molto simili). Le ha allineate e ha creato alberi di sequenze in cui ha dedotto le sequenze dei progenitori. Piccoli passi evolutivi, per osservare l'evoluzione e dedurne le caratteristiche.

Le matrici PAM sono basate su allineamenti globali di proteine strettamente correlate. Il PAM1 è la matrice calcolata dal confronto di sequenze con non più di 1% di divergenza. Ad un intervallo evolutivo di PAM1, un cambiamento si è verificato su una lunghezza di 100 aminoacidi.

Altre matrici PAM sono estrapolate da PAM1 ( PAM1 non ha utilità pratica ). Per PAM250, 250 sostituzioni si sono verificate tra due proteine su una lunghezza di 100 aminoacidi, nel passo evolutivo che essa rappresenta. *Nota bene:* Tutti i dati PAM provengono da proteine strettamente correlate (> 85% di identità degli aminoacidi).

## Dayhoff: 34 superfamiglie di proteine

Proteina	PAMs per 100 milioni di anni
Ig kappa chain	37
Kappa casein	33
luteinizing hormone b	30
lactalbumin	27
complement component 3	27
epidermal growth factor	26
proopiomelanocortin	21
pancreatic ribonuclease	21
haptoglobin alpha	20
serum albumin	19
phospholipase A2, group IB	19
prolactin	17
carbonic anhydrase C	16
Hemoglobin $\alpha$	12
Hemoglobin $\beta$	12
apolipoprotein A-II	10
lysozyme	9.8
gastrin	9.8
myoglobin	8.9
nerve growth factor	8.5
myelin basic protein	7.4
thyroid stimulating hormone b	7.4
parathyroid hormone	7.3
parvalbumin	7.0
trypsin	5.9
insulin	4.4
calcitonin	4.3
arginine vasopressin	3.6
adenylate kinase	3.2
triosephosphate isomerase 1	2.8
vasoactive intestinal peptide	2.6
glyceraldehyde phosph. dehydrogease	2.2
cytochrome c	2.2
collagen	1.7
troponin C, skeletal muscle	1.5
alpha crystallin B chain	1.5
glucagon	1.2
glutamate dehydrogenase	0.9
histone H2B, member Q	0.9
ubiquitin	0

**Dayhoff e i numeri di “point accepted mutations”:  
Quali sostituzioni si verificano nelle proteine?**

	A Ala	R Arg	N Asn	D Asp	C Cys	Q Gln	E Glu	G Gly
A								
R	30							
N	109	17						
D	154	0	532					
C	33	10	0	0				
Q	93	120	50	76	0			
E	266	0	94	831	0	422		
G	579	10	156	162	10	30	112	
H	21	103	226	43	10	243	23	10

Conteggio delle mutazioni osservate (PAM1) Dayhoff (1978) p.346.

### 13.2 La mutabilità relativa degli amminoacidi

Quanto spesso mutano nelle proteine? Definiamo la Frequenza relativa di mutazione.

AA	Freq	AA	Freq
Asn	134	His	66
Ser	120	Arg	65
Asp	106	Lys	56
Glu	102	Pro	56
Ala	100	Gly	49
Thr	97	Phe	41
Ile	96	Phe	41
Gln	93	Cys	20
Val	74	Trp	18

Frequenze normalizzate

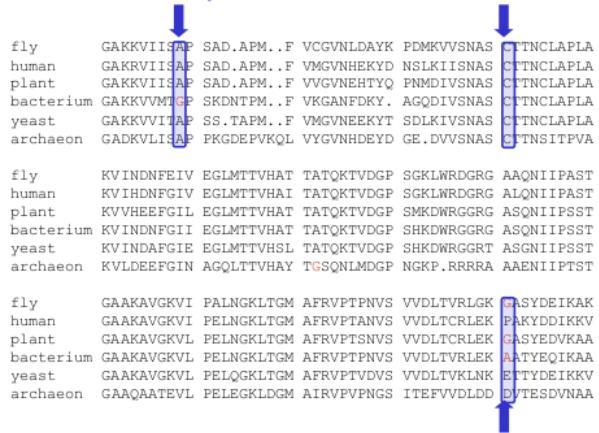
### Ogni quanto occorrono nelle proteine?

<b>Gly</b>	<b>8.9%</b>	<b>Arg</b>	<b>4.1%</b>
<b>Ala</b>	<b>8.7%</b>	<b>Asn</b>	<b>4.0%</b>
<b>Leu</b>	<b>8.5%</b>	<b>Phe</b>	<b>4.0%</b>
<b>Lys</b>	<b>8.1%</b>	<b>Gln</b>	<b>3.8%</b>
<b>Ser</b>	<b>7.0%</b>	<b>Ile</b>	<b>3.7%</b>
<b>Val</b>	<b>6.5%</b>	<b>His</b>	<b>3.4%</b>
<b>Thr</b>	<b>5.8%</b>	<b>Cys</b>	<b>3.3%</b>
<b>Pro</b>	<b>5.1%</b>	<b>Tyr</b>	<b>3.0%</b>
<b>Glu</b>	<b>5.0%</b>	<b>Met</b>	<b>1.5%</b>
<b>Asp</b>	<b>4.7%</b>	<b>Trp</b>	<b>1.0%</b>

- blu=6 codoni; rosso=1 codone
- Le frequenze  $f_i$  si sommano a 100

**Esempio** Prendiamo un allineamento multiplo di sequenze, ad esempio della deidrogenasi gliceraldeide 3-fosfato.

**OSSERVAZIONE:** le colonne di residui possono avere conservazione alta o bassa.



### 13.2.1 Matrice PAM1 (probabilità) di Dayhoff

Aminoacido originale										
Aminoacido mutato	A	R	N	D	C	Q	E	G	H	I
	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile
A	98,6	0,02	0,09	0,01	0,03	0,08	0,17	0,21	0,02	0,06
R	0,01	99,1	0,01	0	0,01	10	0	0	10	0,03
N	0,04	0,01	98,2	0,36	0	0,04	0,06	0,06	0,21	0,03
D	0,06	0	0,42	98,5	0	0,06	0,53	0,06	0,04	0,01
C	0,01	0,01	0	0	99,7	0	0	0	0,01	0,01
Q	0,03	0,09	0,04	0,05	0	98,7	0,027	0,01	0,23	0,01
E	0,10	0	0,07	0,56	0	0,35	98,6	0,04	0,02	0,03
G	0,21	0,01	0,12	0,11	0,01	0,03	0,07	99,3	0,01	0
H	0,01	0,08	0,18	0,03	0,01	0,20	0,01	0	99,1	0
I	0,02	0,02	0,3	0,01	0,02	0,01	0,02	0	0	98,7

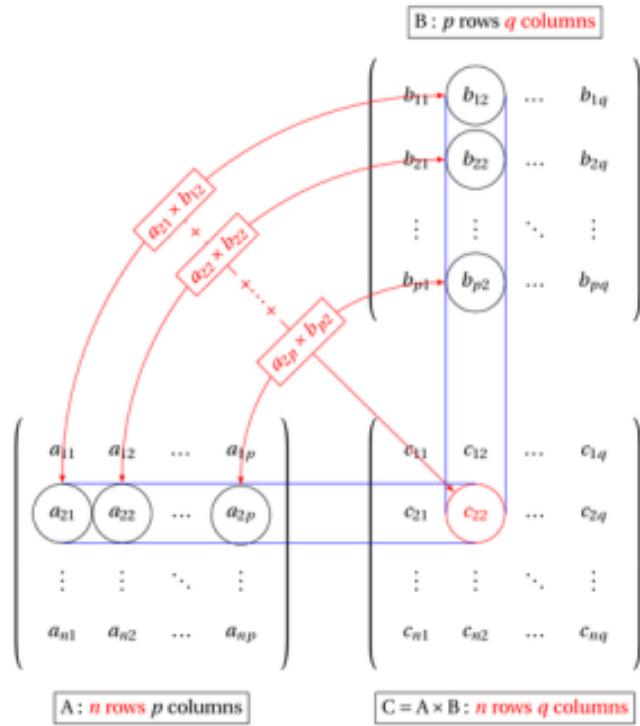
Ogni elemento della matrice mostra la probabilità che un amminoacido (in alto) venga sostituito da un altro amminoacido (a lato).

## 14 Matrici di sostituzione

Una matrice di sostituzione contiene valori proporzionali alla probabilità che l'amminoacido  $i$  muti nell' amminoacido  $j$  per tutte le coppie possibili di aminoacidi. Le matrici di sostituzione sono costruite assemblando un campione ampio e diversificato di allineamenti a coppie (o allineamenti multipli di sequenza) di aminoacidi. Le matrici di sostituzione dovrebbero riflettere la probabilità reale di mutazione in un periodo di evoluzione. I due principali tipi di matrici di sostituzione: PAM e BLOSUM.

### 14.1 Moltiplicare le matrici

Con  $(PAM1)^n$  si può simulare il passaggio di  $n$  passi di evoluzione.



**Matrice di sostituzione PAM0 (probabilità)** Ovvero: nulla cambia.

		Aminoacido originale								
		A	R	N	D	C	Q	E	G	
Aminoacido mutato	PAM0	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	
	A	100%	0%	0%	0%	0%	0%	0%	0%	
	R	0%	100%	0%	0%	0%	0%	0%	0%	
	N	0%	0%	100%	0%	0%	0%	0%	0%	
	D	0%	0%	0%	100%	0%	0%	0%	0%	
	C	0%	0%	0%	0%	100%	0%	0%	0%	
	Q	0%	0%	0%	0%	0%	100%	0%	0%	
	E	0%	0%	0%	0%	0%	0%	100%	0%	
	G	0%	0%	0%	0%	0%	0%	0%	100%	

Si sono verificati 0 passi di evoluzione: non è cambiato nulla!

**Matrice di sostituzione PAM2000 (probabilità)** PAM1<sup>2000</sup>, ovvero: il caso.

		Aminoacido originale								
		A	R	N	D	C	Q	E	G	
Aminoacido mutato	PAM $\infty$	Ala	Arg	ASN	ASP	Cys	Gln	Glu	Gly	
	A	8,7%	8,7%	8,7%	8,7%	8,7%	8,7%	8,7%	8,7%	
	R	4,1%	4,1%	4,1%	4,1%	4,1%	4,1%	4,1%	4,1%	
	N	4,0%	4,0%	4,0%	4,0%	4,0%	4,0%	4,0%	4,0%	
	D	4,7%	4,7%	4,7%	4,7%	4,7%	4,7%	4,7%	4,7%	
	C	3,3%	3,3%	3,3%	3,3%	3,3%	3,3%	3,3%	3,3%	
	Q	3,8%	3,8%	3,8%	3,8%	3,8%	3,8%	3,8%	3,8%	
	E	5,0%	5,0%	5,0%	5,0%	5,0%	5,0%	5,0%	5,0%	
	G	8,9%	8,9%	8,9%	8,9%	8,9%	8,9%	8,9%	8,9%	

Moltiplicando PAM1 per 2000 (passi di evoluzione) si arriva ad una situazione in cui la probabilità converge alla frequenza osservata

**Matrice di sostituzione PAM250 (probabilità) di mutazione**

		Aminoacido originale																			
		A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
Aminoacido mutato		13	6	9	9	5	8	9	12	6	8	6	7	7	4	11	11	11	2	4	9
	R	3	17	4	3	2	5	3	2	6	3	2	9	4	1	4	4	3	7	2	2
N	4	4	6	7	2	5	6	4	6	3	2	5	3	2	4	5	4	2	3	3	3
D	5	4	8	11	1	7	10	5	6	3	2	5	3	1	4	5	5	1	2	3	3
C	2	1	1	1	52	1	1	2	2	2	1	1	1	1	2	3	2	1	4	2	3
Q	3	5	5	6	1	10	7	3	7	2	3	5	3	1	4	3	3	1	2	3	3
E	5	4	7	11	1	9	12	5	6	3	2	5	3	1	4	5	5	1	2	3	3
G	12	5	10	10	4	7	9	27	5	5	4	6	5	3	8	11	9	2	3	7	7
H	2	5	5	4	2	7	4	2	15	2	2	3	2	2	3	3	2	2	2	3	2
I	3	2	2	2	2	2	2	2	2	10	6	2	6	5	2	3	4	1	3	9	9
L	6	4	4	3	2	6	4	3	5	15	34	4	20	13	5	4	6	6	7	13	13
K	6	18	10	8	2	10	8	5	8	5	4	24	9	2	6	8	8	4	3	5	5
M	1	1	1	1	0	1	1	1	2	3	2	6	2	1	1	1	1	1	2	2	2
F	2	1	2	1	1	1	1	1	3	5	6	1	4	32	1	2	2	4	20	3	3
P	7	5	5	4	3	5	4	5	5	3	3	4	3	2	20	6	5	1	2	4	4
S	9	6	8	7	7	6	7	9	6	5	4	7	5	3	9	10	9	4	4	6	6
T	8	5	6	6	4	5	5	6	4	6	4	6	5	3	6	8	11	2	3	6	6
W	0	2	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	55	1	0
Y	1	1	2	1	3	1	1	1	3	2	2	1	2	15	1	2	2	3	31	2	2
V	7	4	4	4	4	4	4	5	4	15	10	4	10	5	5	5	7	2	4	17	17

PAM 250 è un caso interessante: ottenuta da PAM1<sup>250</sup> prevede che circa il 20% della sequenza sia conservato. A → A ha probabilità del 13%. Da notare W e C che anche dopo 250 mutazioni hanno il 50% di probabilità di non mutare.

## 14.2 Approccio Dayhoff

per l'assegnazione di punteggi per ogni due residui di aminoacidi allineati Dayhoff et al. hanno definito il punteggio (score) per due generici residui  $i, j$ :

- $q_{i,j}$  = probabilità che l'aminoacido  $i$  venga sostituito da  $j$  (probabilità di omologia in base alle sostituzioni osservate)
- $p_{i,j}$  = Probabilità di trovare casualmente l'appaiamento  $i, j$  (prodotto della probabilità di trovare un “ $i$ ” e quella di trovare un “ $j$ ” in una qualunque sequenza, cioè prodotto delle frequenze)

Il loro rapporto serve a tenere conto che l'evento  $q_{i,j}$  sia casuale.

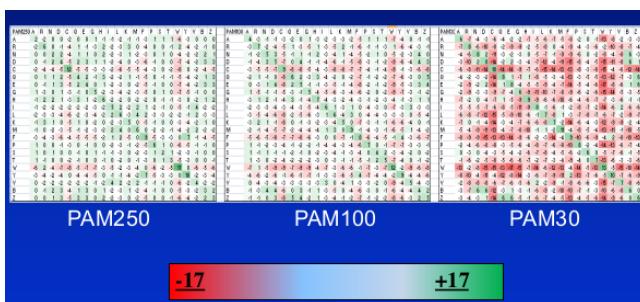
Il valore è poi convertito al log (per rendere due valori sommabili) e moltiplicato per 10 (così che, prendendo la parte intera del valore si conserva la prima cifra decimale). Gli score sono utili negli allineamenti a coppie (e in algoritmi di ricerca come BLAST)

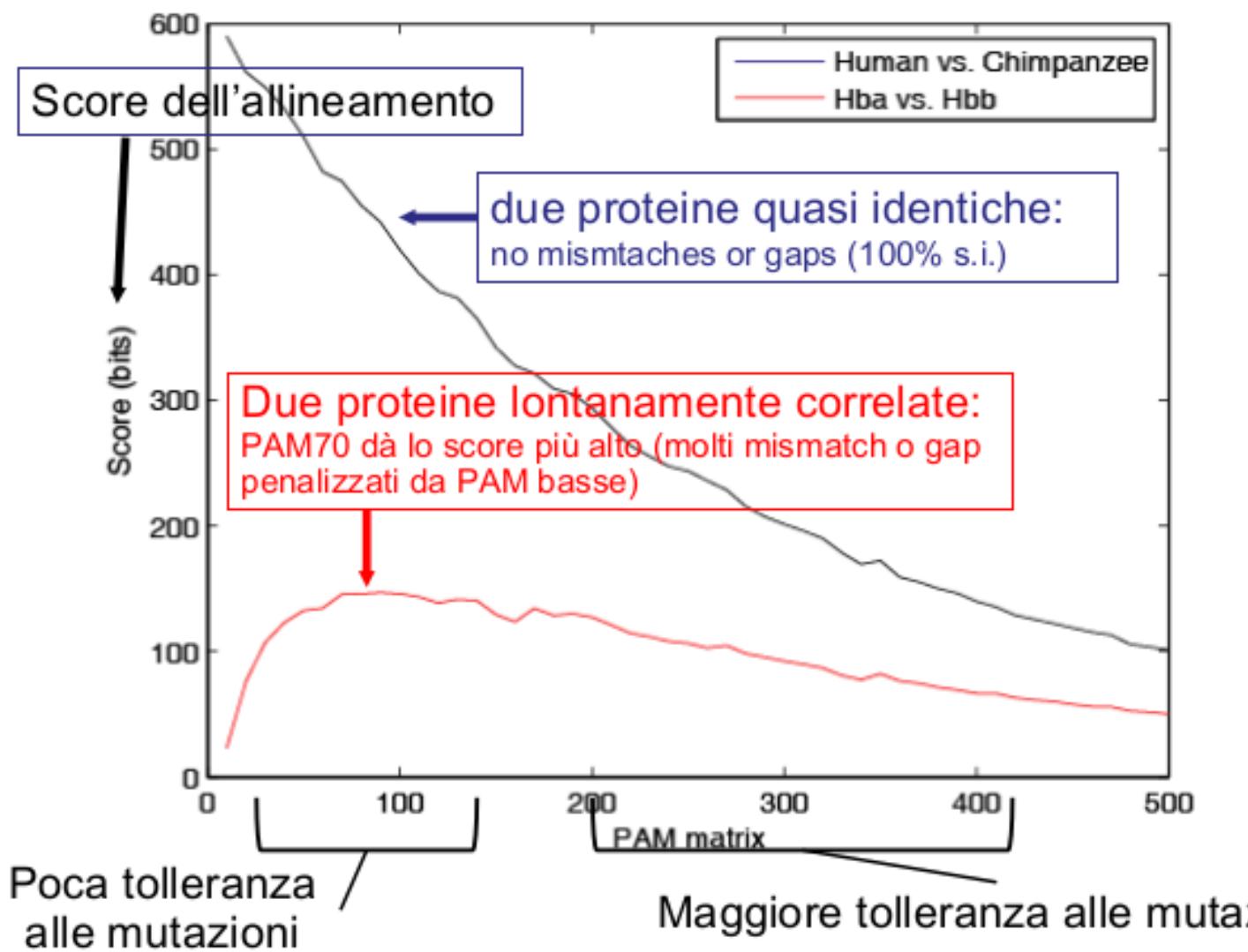
$$s_{i,j} = 10 \times \log\left(\frac{q_{i,j}}{p_{i,j}}\right)$$

$S(trp, trp) = 10 \log(\frac{0.55}{0.010}) = 17,4$  significa che la probabilità di trovare un  $W$  conservato è 50 volte maggiore della probabilità che un  $W$  sia a caso nelle due posizioni considerate.

Un -10 è un  $-1(\log)$  quindi  $\frac{1}{10}$  e indica che la probabilità che quell'allineamento si verifichi è  $\frac{1}{10}$  della frequenza di quegli amminoacidi in posizioni corrispondenti.

## Matrici PAM a confronto



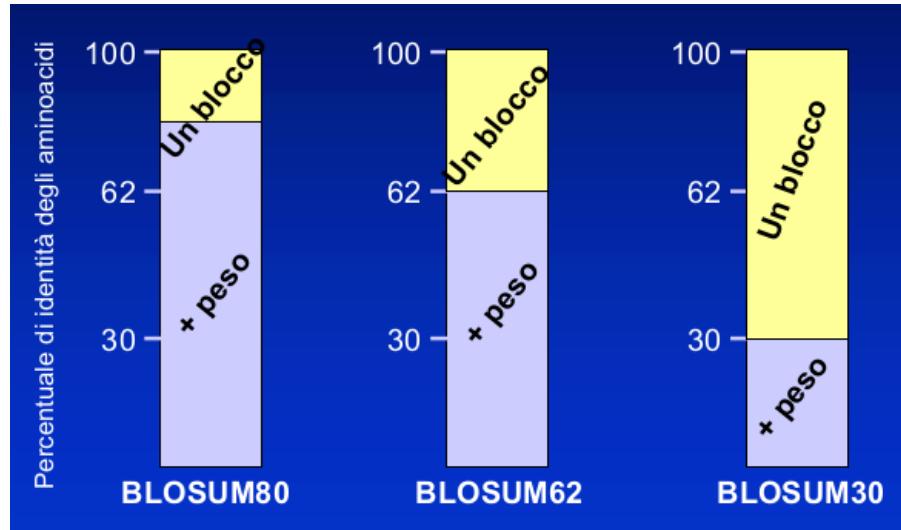


## 15 Matrici BLOSUM

Le matrici BLOSUM sono basate su allineamenti locali, tratti dal database BLOCKS che raggruppa blocchi (regioni) di allineamenti locali di sequenze lontanamente correlate. BLOSUM sta per BLOck SUbstitution Matrix.

BLOSUM62 è una matrice calcolata a partire da sequenze con divergenza minore del 62%. Default per BLAST . Il metodo di calcolo degli score è poi simile a quello per le PAM, ma si usa  $\lambda = 2$  al posto di 10 (infatti per BLOSUM il range è 90 – 45 VS 30 – 250 per le PAM)

$$S_{i,j} = \left( \frac{1}{\lambda} \right) \log \left( \frac{p_{i,j}}{q_i * q_j} \right)$$



The figure shows a sequence alignment and its corresponding score calculation. On the left is a 20x20 BLOSUM62 scoring matrix. On the right, two sequences are aligned:

		Score allineamento: 15																			
		Seq1		Seq2																	
		V D S - C Y		V E S L C Y																	
		Score		Score																	
		4 2 4		4 2 4 -11 9 7																	

The alignment is shown as:

```

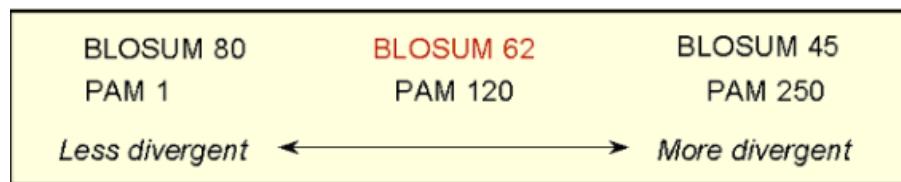
Seq1: V D S - C Y
Seq2: V E S L C Y
Score: 4 2 4 -11 9 7
  
```

Arrows point from the alignment scores in the green box to the corresponding entries in the BLOSUM62 matrix.

$$\text{Punteggio}_{\text{totale}} = \sum \text{somiglianze} - \sum \text{penalità gap}$$

## 16 BLOSUM vs. PAM

PAM si basa su principi evoluzionistici, mentre BLOSUM si basa più sull'osservazione di allineamenti reali, senza fare assunzioni di omologia.

**Più conservativo**

es.Globina: topo vs ratto

Nella BLOSUM 80 le sequenze identiche per l'80% finiscono in un unico blocco e gli score sono applicati considerando gli altri allineamenti → score adatti per proteine simili (come per la PAM10)

**Meno conservativo**

es.Globina: topo vs batterio

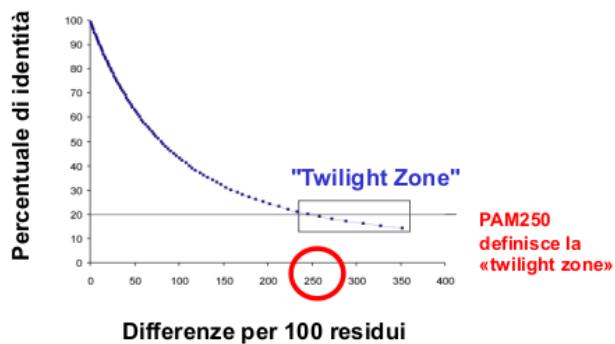
Nella BLOSUM 45 le sequenze identiche per il 45% finiscono in un unico blocco e gli score sono applicati considerando gli altri allineamenti → score adatti per proteine diverse (come per la PAM250)

**16.1 La "Twilight zone" nell'allineamento di proteine**

**Quando compariamo due sequenze proteiche simili, quante mutazioni possono accumularsi prima che le differenze le rendano irriconoscibili?** L'identità tra due sequenze ciascuna di 100 aminoacidi cala come un esponenziale negativo all'accumularsi delle mutazioni.

- A PAM1, due proteine sono al 99% identiche
- A PAM10.7, ci sono 10 differenze ogni 100 residui
- A PAM80, ci sono 50 differenze ogni 100 residui
- A PAM250, ci sono 80 differenze ogni 100 residui

Oltre (20-25% identità) non è più distinguibile una similitudine (Allineamenti multipli, modeling)



# **Lezione 5: BLAST**

## **Basic Local Alignment Search Tool**

## 17 BLAST

### 17.1 Problema con gli algoritmi dinamici

Gli algoritmi visti (WS, NW) sono precisi ma lenti. Servono metodi euristici: trovano soluzioni approssimate ma vicine a quella ottimale in tempi brevi.

Spesso la domanda è: *data una sequenza query, quali sequenze simili sono note e già presenti nei databases?*  
Tra i metodi euristici più usati per la ricerca di singole sequenze in banche dati troviamo FASTA e BLAST.

### 17.2 Cos'è Blast?

BLAST (Basic Local Alignment Search Tool) permette un confronto rapido tra una sequenza query e il contenuto di un database. L'algoritmo di BLAST è:

- veloce
- accurato
- web-accessibile

BLAST è fondamentale per capire la relazione di una sequenza query con altre proteine o sequenze di DNA note.

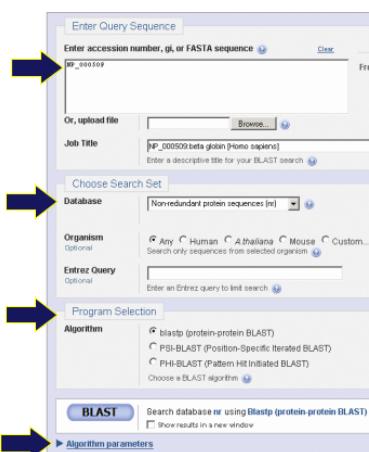
I suoi utilizzi comprendono:

- individuare ortologhi e paraloghi
- scoperta di nuovi geni o proteine
- scoperta di varianti di geni o proteine noti
- lo studio delle expressed sequence tags (EST)
- lo studio di struttura e funzione delle proteine

### 17.3 Ricerca su blast

Le quattro fasi di una ricerca BLAST:

1. Scegliere la sequenza (query)
2. Selezionare il tipo di programma BLAST
3. Selezionare il database per la ricerca
4. Scegliere i parametri opzionali
5. Quindi fare clic su "BLAST"



**Passo 1: Scelta della sequenza** La sequenza può essere inserita in formato FASTA o come accession number (RefSeq).

Esempio:

The screenshot shows the NCBI Protein search results for the hemoglobin subunit beta from Homo sapiens. The sequence is shown in FASTA format:

```
>NP_000509.11|ref|NP_000509.11| hemoglobin subunit beta [Homo sapiens]
>MHLTPEKXCAATLNGKRVRLRLEVGGALGRLLVVTPSTQFTEESFGQLSLTPDPAVEGRPVYKANGKGLG
AFSDGLAHLNLQKTFATLSLHCDELHRDPENFLLCGHYLVCVILAHMFGEETTPFVQAAVGRVVAAGVAN
ALAHMYH
```

## Passo 2: Scegli il programma BLAST

- blastn (nucleotide BLAST)
- blastp (protein BLAST)
- blastx (BLAST tradotto n → P)
- tblastn (BLAST tradotto p → N)

Ci sono anche altri tools disponibili.

The screenshot shows the NCBI Web BLAST homepage. It features a "Basic Local Alignment Search Tool" section with a red box around "Nucleotide BLAST". To its right are "blastx" (translated nucleotide to protein), "tblastn" (protein to translated nucleotide), and "Protein BLAST" (protein to protein). A blue box highlights "Protein BLAST".

## Passo 3: scegliere il database

- nr**

Non ridondante (database più generale, ritorna una sequenza e diversi riferimenti in database in cui la stessa è presente)

- refseq**

Solo sequenze con codice refse q

- dbest**

Database di EST

- dbsts**

Database di sequenze localizzate

- gss**

Genome sequence surveys ( BAC, Yac, ecc )

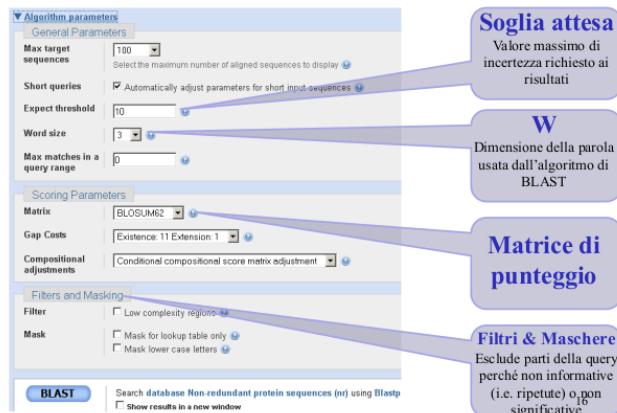
- altri**

Pdb, genomi completi, solo sequenze oggetto di brevetti (pat), ...

**Fase 4: parametri opzionali** Si può:

- Scegliere l'organismo di ricerca
- Attivare filtri
- Modificare la matrice di punteggio
- Cambiare il valore minimo di affidabilità dei risultati
- Modificare la dimensione della parola W

Allo stesso modo in blastp e blastn:



L'output

The screenshot shows the 'Summary della ricerca' (Search Summary) page. It displays search details and filtering options. Red arrows point to the 'programma' (program) and 'database' (database) fields in the search parameters.

## 17.4 L'algoritmo

**BLAST: le basi dell'allineamento di sequenze** L'allineamento può essere:

### 1. Globale

(Needleman & Wunsch)

Usa la programmazione dinamica per trovare i migliori allineamenti tra due sequenze. (Anche se gli allineamenti sono ottimali, la ricerca non è esaustiva). I gap sono ammessi e l'intera lunghezza delle sequenze è allineata (da cui "globale").

### 2. Locale

(Smith & Waterman, 1980 - modifica del primo (per all. globale))

L'allineamento può riguardare solo una parte di una delle sequenze, ed è adatto alla ricerca di caratteristiche locali comuni alle sequenze (domini, siti attivi, ecc.).

BLAST è una approssimazione euristica per l'allineamento locale. Esamina solo una parte dello spazio di ricerca.

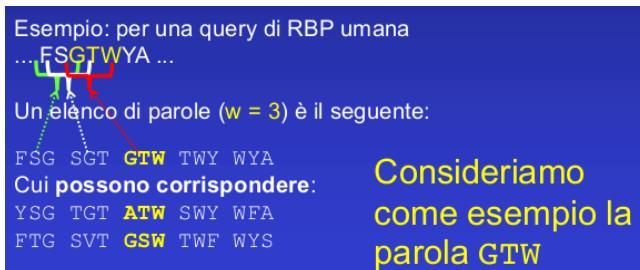
## Come funziona una ricerca BLAST?

L'idea centrale dell'algoritmo di BLAST è di limitare l'attenzione a coppie di segmenti in cui si allineano parole di lunghezza  $w$  con un punteggio di almeno  $T$ .

*Altschul et al. (1990)*

### 17.4.1 Le 3 fasi

**Fase 1** compilare una lista di coppie di parole (di lunghezza  $w = 3$ ) con un punteggio oltre la soglia  $T$ .



Se pongo la soglia  $T = 11$ , gli allineamenti con punteggio maggiore della soglia sono:

- GTW  $6 + 5 + 11 \rightarrow 22$
- GSW  $6 + 1 + 11 \rightarrow 18$
- ATW  $0 + 5 + 11 \rightarrow 16$
- NTW  $0 + 5 + 11 \rightarrow 16$
- GTY  $6 + 5 + 2 \rightarrow 13$

Gli allineamenti con punteggio minore della soglia:

- GNW  $\rightarrow 10$
- GAW  $\rightarrow 9$

Dunque saranno esclusi dai passi successivi.

**Fase 2** Scansione del database per le voci che corrispondono alla lista compilata (con punteggio maggiore della soglia  $T$ ). È un passo veloce e relativamente semplice: i database sono indicizzati per la ricerca di parole di lunghezza  $W$ .

**Fase 3** Quando si riesce a trovare una corrispondenza Fase 3 : nella versione originale di BLAST (1990) (abbinamento tra una “parola” con punteggio maggiore  $T$  al database), estendere l’allineamento in entrambe le direzioni.

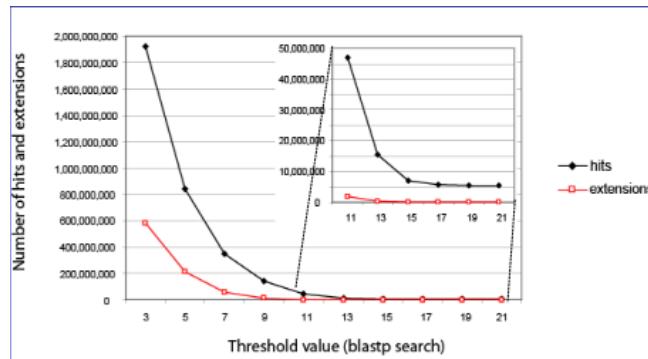
- Tenere traccia del punteggio (utilizzando la matrice di punteggio)
- Stop quando il punteggio è inferiore a una certa soglia.



Nella versione originale di BLAST (1990) ciascun hit è esteso in entrambe le direzioni.

Nella versione migliorata, dal 1997, sono necessari due hit vicini (entro una distanza  $A$ ). In questo modo le estensioni avvengono meno frequentemente ma si ottiene un notevole risparmio di tempo

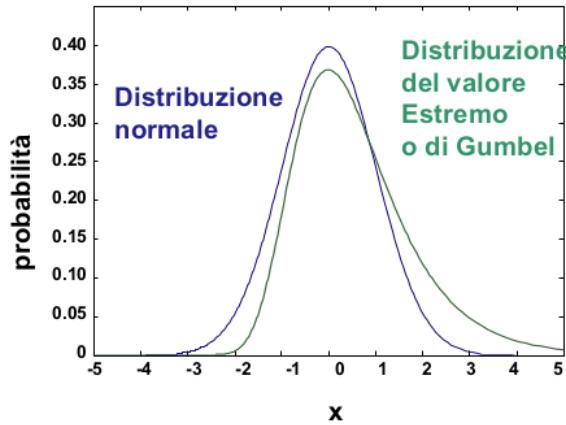
**I parametri** È possibile personalizzare i parametri di BLAST, sia per le soglie menzionate che per le dimensioni delle parole w (default 3 per le proteine, 6 negli algoritmi più recenti, e 11 per le sequenze nucleotidiche). E' importante valutare la significatività statistica dei risultati della ricerca. Ecco che effetto ha la scelta di diversi valori della soglia  $T$ . Parole più grandi producono meno hits, velocizzando la ricerca, a scapito della sensibilità.



## 17.5 Come interpretare una ricerca BLAST

È importante valutare la significatività statistica dei risultati della ricerca. Per gli allineamenti globali, gli aspetti statistici non sono molto approfonditi.

Per gli allineamenti locali (compresi i risultati di ricerca BLAST), le statistiche sono più solide. I punteggi seguono una distribuzione del valore estremo (EVD), piuttosto che una distribuzione normale. La densità di probabilità del valore estremo ( $u$  valore caratteristico = 0 e costante di decadimento  $\lambda = 1$ ) È spostata a destra rispetto alla distribuzione normale.



### 17.5.1 Il valore atteso $E$

È il numero di allineamenti (high scoring segment pairs o HSP) con punteggio maggiore o uguale a un punteggio  $S$  che dovrebbero verificarsi per caso in quella ricerca sul database. Si può pensarlo come un indice di incertezza. Un valore  $E$  è correlato a un valore di probabilità  $p$ .

L'equazione fondamentale che descrive un valore  $E$  è:

$$E = KMNe^{\lambda S}$$

Con:

- $M$ , lunghezza della query
- $N$ , lunghezza della sequenza nel database
- $S$ , score
- $K, \lambda$ , parametri che dipendono dallo scoring system ( $\lambda$ ) dal database usato ( $K$ )

### Alcune proprietà

- Il valore di  $E$  decresce esponenzialmente con l'aumentare  $S$ . Valori più elevati di  $S$  corrispondono a migliori allineamenti e infatti hanno  $E$  values più bassi.
- Ottenere un allineamento con  $E = 1$  significa che esiste un altro allineamento con lo stesso score  $S$  che è risultato per caso.
- La stessa ricerca, su un database più piccolo o più grande, anche se restituisce lo stesso allineamento deve avere un valore di  $E$  diverso (ciò dipende da  $K$ ).

**Punteggio, dallo score grezzo (S) in bit** Ci sono due tipi di punteggi:

- punteggi grezzi (calcolato da una particolare matrice di sostituzione)
- punteggi bit (punteggio normalizzato)

I punteggi in bit sono paragonabili tra diverse ricerche (es. diverse matrici di sostituzione o diverse banche dati) perché sono normalizzati, adatti quindi al paragone anche se originati tramite diverse matrici di punteggio e database di dimensione diversa

$$S' = \text{bitScore} = \frac{(\lambda S - \ln(K))}{\ln(2)}$$

#### 17.5.2 E-value e p-value

Il valore atteso  $E$  è il numero di allineamenti con punteggio maggiore o uguale a un punteggio  $S$  che dovrebbero verificarsi per caso in un database di ricerca.

Il *p-value* è un modo diverso (ma equivalente) di rappresentare la significatività di un risultato.

$$p = 1 - e^{-E}$$

Per valori molto piccoli E e p sono molto simili. Tuttavia tra 1 e 10 il valore di E è più chiaro (poichè riflette un numero di hits).

E		p
10		0.99995460
5		0.99326205
2		0.86466472
1		0.63212056
0.1		0.09516258 (circa 0.1)
0.05		0.04877058 (circa 0.05)
0.001		0.00099950 (circa 0.001)
0.0001		0.0001000

#### 17.5.3 Panoramica

- **W:** Dimensione della parola usata dall'algoritmo di Blast
- **Soglia attesa:** Valore massimo di incertezza richiesto ai risultati
- **Costo gap:** Apertura ed estensione
- **Matrice di punteggio**
- **Soglia T di punteggio**
- **Dimensione del database**
- **Parametri per il calcolo di E**

**In quali casi ha senso una soglia E elevata?** Si supponga di eseguire una ricerca con una query corta (ad esempio 9 aminoacidi). Non ci sono residui sufficienti per accumulare un punteggio elevato (o un valore di E piccolo). Infatti, una corrispondenza di tutti i 9 i residui potrebbe produrre un punteggio basso con un valore di E =100 o 200. E tuttavia, questo risultato potrebbe essere "reale" e di vostro interesse. In casi particolari, impostando il valore di cut-off per E a 20.000 non si cambia il modo in cui è stata fatta la ricerca, ma cambiano i risultati riportati.

## 17.6 Strategia per la ricerca con BLAST

- Concetti generali
- Come valutare la significatività dei risultati
- Come gestire troppi risultati
- Come gestire troppo pochi risultati
- Blast e la valutazione dell'omologia

A volte un vero match ha un valore di E > 1

Sequences producing significant alignments:	Score	E
	(bits)	Value
gi 5000139 ref NP_00735..1  retinol-binding protein 4, integrin-linked kinase-like protein 1	378	e-105
gi 2302841 gb 1LRBP1  Retinol Binding Protein >gi 4928971 p..	371	e-103
gi 80384 gi 1A2786  plasma retinol-binding protein - human	370	e-103
gi 458179 gb 1QAB1E  Chain E, The Structure Of Human Retinol-Binding Protein	363	e-100
gi 7770173 gb 1AAF59G21..1 1 AF19917_30  (AF19968) PRO2222 (Homo sapiens)	324	5e-89
gi 1294551 ref NP_009089..1  retinol-binding protein 4, integrin-linked kinase-like protein 1	323	6e-88
gi 2302841 gb 1LRBP1  retinol-binding protein 4, integrin-linked kinase-like protein 1	320	6e-84
gi 54129992 emb CA84e499..1  (AF02824) RBP (aa 101-172) (Homo sapiens)	249	2e-16
gi 12989503 gb 1AKC02945..1  (AF025334) mutant retinol binding...	240	2e-18
gi 12989504 gb 1AKC02946..1  (AF025335) mutant retinol binding...	73	2e-13
gi 456141 gb 1AF01638..1  apolipoprotein D precursor [Homo sapiens]	55	4e-08
gi 6193963 gb 1AA82220..1  apolipoprotein D, apob [human, plasmid]	55	5e-08
gi 12440994 gb 1AA82219..1  (AF025336) apolipoprotein D, apob (Homo sapiens)	43	3e-09
gi 12440995 gb 1AA82221..1  (AF025337) apolipoprotein D, apob (Homo sapiens)	35	3e-01
gi 1294541 gb 1CA84205..1  (AL050169) hypothetical protein	35	0.043
gi 13143932 gb 1XP_005360..3  61620 [Homo sapiens] >gi 13639..	35	0.043
gi 4502047 ref NP_001624..1  alpha-1-microglobulin/bikunin precursor	35	0.048
gi 1473782 ref NP_001964..1  progestagen-associated endonuclease	35	0.070
gi 4557393 ref NP_001697..1  complement component 8, gamma polypeptide	34	0.14
gi 4555583 ref NP_001562..1  progestagen-associated endonuclease	32	0.49
gi 13393945 ref NP_005430..2  complement component 8, gamma ...	31	1.1

← reale  
match?

Posso provare un BLAST reciproco per confermare.

A volte un valore E simile si verifica sia per un match esatto corto che per uno lungo.

<pre>&gt;gi 12989504 gb 1AKC02945..1  (AF025334) mutant retinol binding protein [Homo sapiens] Length = 36 Score = 72.0 bits (17%), Expect = 2e-13 Identities = 34/36 (94%), Positives = 35/36 (96%) Query: 82 NHVQCLADMVGVFTDTEPAPKEMEDYVQAVASFLQHN 117           VVDDVCAADMVTVTTTTEPAPKEMEDYVQAVASFLQHN+ Sbjct: 1 NHVQCLADMVGVFTDTEPAPKEMEDYVQAVASFLQHN+ 36 </pre>	{ <span style="color: green;">corto, quasi identico</span>
<pre>&gt;gi 12989504 gb 1AKC02945..1  (AF025334) mutant retinol binding protein [Homo sapiens]           apolipoprotein D precursor [Homo sapiens] &gt;gi 11334407 ref NP_003047..1  apolipoprotein D precursor [Homo sapiens]           apolipoprotein D precursor [Homo sapiens] &gt;gi 11340241 gb 1PO50991 PDB_HUMAN_APOLYPOROTEIN_D_PRECUCOS           apolipoprotein D precursor [Homo sapiens] - human           apolipoprotein D precursor [Homo sapiens]           apolipoprotein D precursor [Homo sapiens]           apolipoprotein D precursor [Homo sapiens]           apolipoprotein D [Homo sapiens]           apolipoprotein D [Homo sapiens] Length = 189 Score = 55.0 bits (13%), Expect = 4e-08 Identities = 47/189 (25%), Positives = 78/189 (41%), Gaps = 59/189 (32%) </pre>	{ <span style="color: blue;">lungo, ma solo 31% identità, valore di E simile</span>
<pre>Query: 27 VVDDVCAADMVTVTTTTEPAPKEMEDYVQAVASFLQHN 86           V+ENFD ++ 0 WY + K P I A +H+ E ++++++NL ++-----GEKINLMD-ELR 82 Sbjct: 33 QCMEWVWVNLGEVTEI-EKTTTFENGRCIOAHTSLRME-----GEKINLMD-ELR 82 </pre>	

Query: 87 ADWWVFTDTEPAPKEMEDYVQAVASFLQHN 126
 DPAFFKEM-WWVAVASFLQHNODDNIVVTPVTTTAPOVSC 126
 +FAK ++K- V + S +W+I- TSV4 TAA YSC
 Sbjct: 83 AD-GTVNGIXKEATPVNLTEPALEEVFSWMPG-----APWVILATSYNTALAUVYSC 124

Query: 137 -----PLMLNLTQTCADSTPVSFVGSPPWLPPE 165
 +L+ +E+ +--+ +--+ +W+FW LPPE
 Sbjct: 135 TCIIQLFLRHPV-----FAMILARHHM-LPPE 158

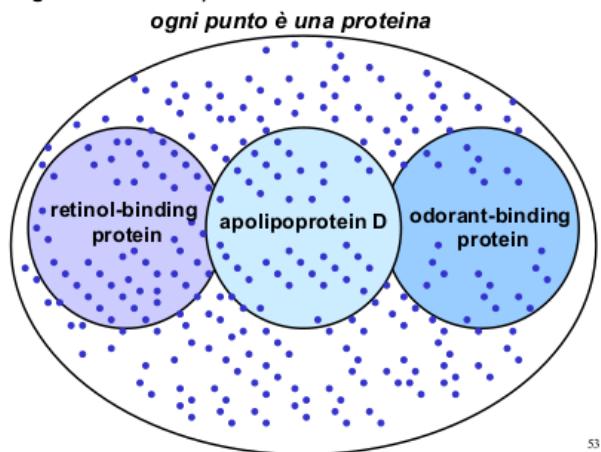
I risultati delle ricerche, ordinati per E, si valutano poi nel dettaglio!

### 17.6.1 Valutare se le proteine sono omologhe

*Es. lipocaline: hanno bassa s.i. ma struttura 3D simile*

**RBP4 (query) e Salivary Lipocalin from boar:** Basso punteggio bit, valore E=0.40, il 21 (“twilight zone”). Ma sono davvero omologhe. Prova un’altra ricerca BLAST con la seconda come query, per trovare molte altre lipocaline.

L’universo di lipocaline (proteine che trasportano piccoli ligandi idrofobici).



53

1. La ricerca con BLAST con salivary lipocalin [Sus scrofa] come query trova molte altre lipocaline
2. Utilizzando la beta globina umana come una query, ecco i risultati blastp contro le proteine umane RefSeq (PAM30). Dove si trova mioglobina? È assente!
3. Abbiamo bisogno di usare PSI-BLAST

#### 17.6.2 Due esempi di problemi che BLAST standard non può risolvere

1. Usando la beta globina umana come query contro proteine umane con RefSeq, blastp non “trova” la mioglobina umana. Questo perché le due proteine sono troppo distanti. PSI-BLAST presso NCBI così come i modelli nascosti di Markov possono facilmente risolvere questo problema.
2. Come possiamo cercare con 10000 paia o addirittura milioni di paia di basi come query?  
Molti strumenti tipo-BLAST per il DNA genomico sono disponibili come PatternHunter, Megablast, Blat, e BLASTZ (NON li vedremo nel corso).

## 18 PSI-BLAST (position specific iterated)

Lo scopo di PSI-BLAST è quello di cercare più in profondità nel database match alla sequenza della proteina query, utilizzando una matrice di punteggio che è adattata dinamicamente (in modo iterativo) per la ricerca in corso.

### 18.1 Fasi di esecuzione

1. Selezionare una query di ricerca (diverse sequenze) contro un database proteico e lanciare BLASTP
2. PSI-BLAST costruisce un allineamento multiplo di sequenze a partire dagli hit migliori e crea quindi un "profilo" detto anche Matrice di calcolo posizione-specifica (PSSM, position- specific scoring matrix)
3. Il PSSM (e non più la sequenza iniziale) è usato come query sul database: mi aspetto ora di trovare più sequenze sui cui ricalcolare l’allineamento multiplo
4. PSI-BLAST stima la significatività statistica (valori di E)
5. Ripetere i passaggi 3 e 4 in modo iterativo, tipicamente 5 volte. Ad ogni nuova ricerca, un nuovo profilo viene utilizzato nella query.

Un PROFILO di lunghezza L è una matrice L X 20 (per proteine, o L X 4 per DNA) di elementi s<sub>i,j</sub> rappresentanti lo score per allineare la lettera j alla posizione i. Un profilo può essere allineato ad una sequenza individuale esattamente allo stesso modo di una normale sequenza

### 18.1.1 Come costruire il profilo?

Esaminare l'output di blastp per identificare "regole" empiriche di punteggio per gli aminoacidi tollerati in ogni posizione: non tutte le posizioni ammettono la stessa variabilità. Il punteggio ne terrà conto.

### 18.1.2 Funzionamento di Psi-Blast

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1 M	-1	-2	-2	-3	-2	-1	-2	-3	-2	1	2	-2	6	0	-3	-2	-1	-2	-1	1
2 K	-1	1	0	1	-4	2	4	-2	0	-3	-3	3	-2	-4	-1	0	-1	-3	-2	-3
3 W	-3	-3	-4	-5	-3	-2	-3	-3	-3	-3	-2	-3	-2	1	-4	-3	-3	12	2	-3
4 V	0	-3	-3	-4	-1	-3	-3	-4	-4	3	1	-3	1	-1	-3	-2	0	-3	-1	4
5 W	-3	-3	-4	-5	-3	-2	-3	-3	-3	-3	-2	-3	-2	1	-4	-3	-3	12	2	-3
6 A	5	-2	-2	-2	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0	
7 L	-2	-2	-4	-4	-1	-2	-3	-4	-3	2	4	-3	2	0	-3	-3	-1	-2	-1	1
8 L	-1	-3	-3	-4	-1	-3	-3	-4	-3	2	2	-3	1	3	-3	-2	-1	-2	0	3
9 L	-1	-3	-4	-4	-1	-2	-3	-4	-3	2	4	-3	2	0	-3	-3	-1	-2	-1	2
10 L	-2	-2	-4	-4	-1	-2	-3	-4	-3	2	4	-3	2	0	-3	-3	-1	-2	-1	1
11 A	5	-2	-2	-2	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0	
12 A	5	-2	-2	-2	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0	
13 W	-2	-3	-4	-4	-2	-2	-3	-4	-3	1	4	-3	2	1	-3	-3	-2	7	0	0
14 A	3	-2	-1	-2	-1	-1	-2	4	-2	-2	-2	-1	-2	-3	-1	1	-1	-3	-3	-1
15 A	2	-1	0	-1	-2	2	0	2	-1	-3	-3	0	-2	-3	-1	3	0	-3	-2	-2
16 A	4	-2	-1	-2	-1	-1	3	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	-1	
...																				
37 S	2	-1	0	-1	-1	0	0	0	-1	-2	-3	0	-2	-3	-1	4	1	-3	-2	-2
38 G	0	-3	-1	-2	-3	-2	-2	6	-2	-4	-4	-2	-3	-4	-2	0	-2	-3	-3	-4
39 T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-2	-1	1	5	-3	-2	0	
40 W	-3	-3	-4	-5	-3	-2	-3	-3	-3	-3	-2	-3	-2	1	-4	-3	-3	12	2	-3
41 Y	-2	-2	-2	-3	-3	-2	-2	-3	2	-2	-1	-2	-1	3	-3	-2	-2	2	7	-1
42 A	4	-2	-2	-2	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0	

- In riga: 20 amminoacidi
- In colonna: tutti gli aa della posizione 1 alla fine (L) della proteina query per PSI-BLAST
- Notare che dato un amminoacido (come ad esempio lalanina) nella sequenza query, si possono ottenere diversi punteggi per il match, in relazione alla posizione nella sequenza query e alla frequenza in cui la ritrovo nell'allineamento multiplo.
- Lo stesso vale anche per il triptofano: posizioni diverse hanno, per lo stesso match, punteggi diversi

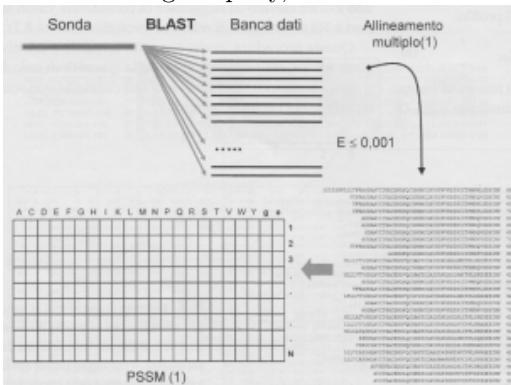
**In pratica:** il profilo posizione-specifico (PSSM) cattura il pattern di conservazione nell'allineamento multiplo ottenuto dai migliori hits di BLASTP e lo immagazzina sottoforma di matrice di score → Posizioni altamente conservate ottengono punteggi alti; posizioni poco conservate ottengono punteggi bassi.

Il profilo è quindi una specie di nuova query in cui ogni posizione ha un "peso" differenziato: questa informazione può essere utilizzata per estendere la ricerca.

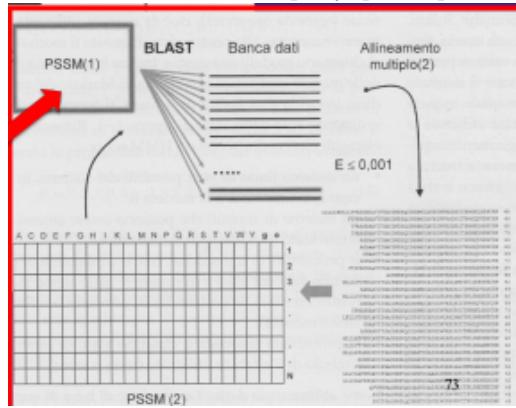
### 18.1.3 Esempio di Cicli

Schematicamente, questi gli step per i primi due cicli:

- CICLO 1: singola query, 1 run di BLAST



- CICLO 2: PSSM come query, più sequenze trovate



Iterazioni	#Hits	#Hits > Soglia
1	104	49
2	173	96
3	236	178
4	301	240
5	344	283
6	342	298
7	378	310
8	382	320

Nota: (come prima, globina beta, PAM30, RefSeq, Homo sapiens). Gli E-values cambiano a seconda delle interazioni, e possono migliorare drasticamente. Seguiamo iterativamente l'allineamento con la subunità mu.

- Iterazione 1:

## hemoglobin subunit mu [Homo sapiens]

Sequence ID: [ref|NP\\_001003938.1|](#) Length: 141 Number of Matches: 1

Range 1: 2 to 141 GenPept Graphics							Next Match		Previous M
Score	Expect	Method	Identities	Positives	Gaps				
92.7 bits(211)	5e-20	Compositional matrix adjust.	51/146(35%)	56/146(38%)	8/146(5%)				
Query 4	LTPPEEKSAVTALWGKVNDEV--GGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKV	L +E + W + E G E L RL VYP T +F	61						
Sbjct 2	LSAQERAQIAQVWDLIAGHEAQFGAELLRLFTVYPSTKVYFPHL-----SACQDATQL		55						
Query 62	KAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGK	HG L A H DNL L +LH L VDP NF LL VLA H	121						
Sbjct 56	LSHGQRMLAAVGAAVQHVVDNLRAALSPADLHALVLRVDPANFPPLLICQCFHVVLASHLQD		115						
Query 122	EFTPVQAAQKVVAGVANALAHKYH	EFT QAA K GVA L KY	147						
Sbjct 116	EFTVQMQAADKFILTGVAVVLTEKYR		141						

- Iterazione 2:

## hemoglobin subunit mu [Homo sapiens]

Sequence ID: [ref|NP\\_001003938.1|](#) Length: 141 Number of Matches: 1

Range 1: 2 to 141 GenPept Graphics							Next Match		Previous M
Score	Expect	Method	Identities	Positives	Gaps				
184 bits(426)	2e-50	Composition-based stats.	51/146(35%)	56/146(38%)	8/146(5%)				
Query 4	LTPPEEKSAVTALWGKVNDEV--GGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKV	L +E + W + E G E L RL VYP T +F	61						
Sbjct 2	LSAQERAQIAQVWDLIAGHEAQFGAELLRLFTVYPSTKVYFPHL-----SACQDATQL		55						
Query 62	KAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGK	HG L A H DNL L +LH L VDP NF LL VLA H	121						
Sbjct 56	LSHGQRMLAAVGAAVQHVVDNLRAALSPADLHALVLRVDPANFPPLLICQCFHVVLASHLQD		115						
Query 122	EFTPVQAAQKVVAGVANALAHKYH	EFT QAA K GVA L KY	147						
Sbjct 116	EFTVQMQAADKFILTGVAVVLTEKYR		141						

- Iterazione 3:

## hemoglobin subunit mu [Homo sapiens]

Sequence ID: [ref|NP\\_001003938.1|](#) Length: 141 Number of Matches: 1

Range 1: 2 to 141 GenPept Graphics							Next Match		Previous M
Score	Expect	Method	Identities	Positives	Gaps				
162 bits(373)	5e-43	Composition-based stats.	51/146(35%)	56/146(38%)	8/146(5%)				
Query 4	LTPPEEKSAVTALWGKVNDEV--GGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKV	L +E + W + E G E L RL VYP T +F	61						
Sbjct 2	LSAQERAQIAQVWDLIAGHEAQFGAELLRLFTVYPSTKVYFPHL-----SACQDATQL		55						
Query 62	KAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGK	HG L A H DNL L +LH L VDP NF LL VLA H	121						
Sbjct 56	LSHGQRMLAAVGAAVQHVVDNLRAALSPADLHALVLRVDPANFPPLLICQCFHVVLASHLQD		115						
Query 122	EFTPVQAAQKVVAGVANALAHKYH	EFT QAA K GVA L KY	147						
Sbjct 116	EFTVQMQAADKFILTGVAVVLTEKYR		141						

- Iterazione 4:

hemoglobin subunit mu [Homo sapiens]  
 Sequence ID: [ref|NP\\_001003938.1|](#) Length: 141 Number of Matches: 1

Range 1: 2 to 141 GenPept Graphics			▼ Next Match	▲ Previous Mi	
Score	Expect	Method	Identities	Positives	Gaps
161 bits(371)	1e-42	Composition-based stats.	51/146(35%)	56/146(38%)	8/146(5%)
Query 4	LTPEEKSAVTALWGKVNDEV--GGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKV	61			
	L +E + W + E G E L RL VYP T +F A				
Sbjct 2	LSAQERAQIAQVWDLIAGHEAQFGAELLRLFTVYPSTKVYFPHL-----SACQDATQL	55			
Query 62	KAHGKKVLGAFSDGLAHLDNLKGTPATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGK	121			
	HG L A H DNL L +LH L VDP NF LL VLA H				
Sbjct 56	LSHGQRMLAAVGAAVQHVDNLRALARSPADLHALVLRVDPANFPPLIQCFCVVVLASHLQD	115			
Query 122	EFTPQVQAAYQKVVAAGVANALAHKYH	147			
	EFT QAA K GVA L KY				
Sbjct 116	EFTVQMQAADKFLTGVAVVLTEKYL	141			

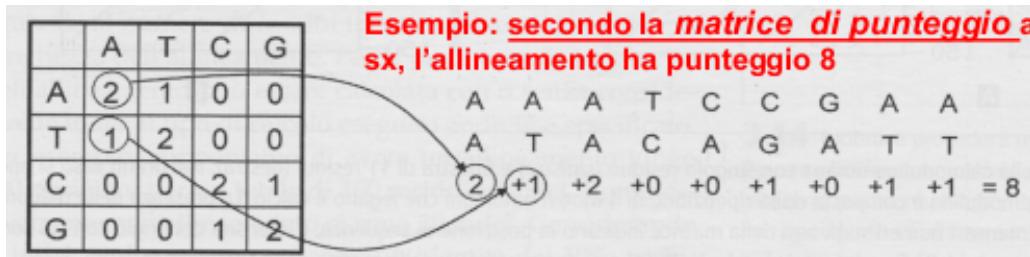
- Iterazione 5:

hemoglobin subunit mu [Homo sapiens]  
 Sequence ID: [ref|NP\\_001003938.1|](#) Length: 141 Number of Matches: 1

Range 1: 2 to 141 GenPept Graphics			▼ Next Match	▲ Previous Mi	
Score	Expect	Method	Identities	Positives	Gaps
161 bits(371)	1e-42	Composition-based stats.	51/146(35%)	56/146(38%)	8/146(5%)
Query 4	LTPEEKSAVTALWGKVNDEV--GGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKV	61			
	L +E + W + E G E L RL VYP T +F A				
Sbjct 2	LSAQERAQIAQVWDLIAGHEAQFGAELLRLFTVYPSTKVYFPHL-----SACQDATQL	55			
Query 62	KAHGKKVLGAFSDGLAHLDNLKGTPATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGK	121			
	HG L A H DNL L +LH L VDP NF LL VLA H				
Sbjct 56	LSHGQRMLAAVGAAVQHVDNLRALARSPADLHALVLRVDPANFPPLIQCFCVVVLASHLQD	115			
Query 122	EFTPQVQAAYQKVVAAGVANALAHKYH	147			
	EFT QAA K GVA L KY				
Sbjct 116	EFTVQMQAADKFLTGVAVVLTEKYL	141			

- E non migliora più (e nemmeno il bit-score)

Le matrici di score permettono di campionare una regione relativamente limitata dell'universo delle sequenze:



PSI-BLAST genera matrici di punteggio più potenti delle PAM o BLOSUM

## 18.2 Psi-Blast: la corruzione

PSI-BLAST è utile per rilevare relazioni deboli ma biologicamente significative tra le proteine. La principale fonte di falsi positivi è la falsa amplificazione di sequenze non correlate alla query.

*Esempio:* una query con un motif coiled-coil può rilevare migliaia di altre proteine con questo motivo che non sono omologhe.

Una volta che anche una singola proteina spuria è inclusa in una ricerca PSI-BLAST entro la soglia, non ne uscirà e influenzerà le iterazioni successive.

La **corruzione** è definita come la presenza di almeno un falso allineamento positivo con un valore  $E < 10^{-4}$  dopo cinque iterazioni. Tre approcci per arrestare la corruzione:

1. Applicare filtri alle regioni con composizione poco specifica
2. Aggiustare il valore di  $E$  da 0.001 (default) ad uno più basso, come  $E = 0.0001$ .
3. Controllare visivamente l'output di ogni iterazione. Rimuovere gli hit sospetti deselezionando la casella.

## Lezione 6: Allineamenti Multipli di Sequenze

## 19 Allineamento multiplo di sequenze

### 19.1 Visione Generale

#### 19.1.1 Una definizione

Un allineamento multiplo è una collezione di tre o più sequenze proteiche (o nucleotidiche) parzialmente o completamente allineate

- I residui e le zone omologhe sono allineate in colonne per tutta la lunghezza delle sequenze
- Il senso dell'omologia dei residui è evoluzionario
- Il senso dell'omologia dei residui è strutturale

Si tratta di un argomento di ricerca attivo dagli anni '90.

#### 19.1.2 Alcuni fatti

Non c'è necessariamente un allineamento "corretto" per una famiglia di proteine.

**Perchè?**

- Le sequenze di proteine evolvono
- Le corrispondenti strutture tridimensionali evolvono, anche se più lentamente
- Può essere particolarmente difficile identificare i residui che si sovrappongono nello spazio (strutturalmente) in un allineamento multiplo di sequenze.

Due proteine che condividono il 30% di identità di sequenza avranno circa il 50% dei residui sovrapponibili nelle due strutture

#### 19.1.3 Caratteristiche utili per realizzarlo

Alcuni residui allineati, come cisteine che formano ponti disolfuro, o i triptofani, possono essere altamente conservati

- Ci possono essere motivi conservati come un dominio transmembrana
- Alcune caratteristiche come le strutture secondarie, siti attivi e di legame per ligandi o complessi sono spesso conservate
- Ci possono essere regioni con inserimenti o delezioni propagati in parte della famiglia.
- I principi che vedremo sono focalizzati sulle proteine ma sono validi in generale anche per sequenze nucleotidiche.

#### 19.1.4 Utilizzi e Vantaggi

- Il MSA è più sensibile di quello a coppie nel rilevamento di omologie, per questo è uno strumento essenziale nella costruzione di modelli strutturali per omologia
- L'output di BLAST può assumere la forma di un MSA, e possono essere individuati residui conservati o motivi
- In un MSA si possono analizzare i dati di una popolazione
- Una singola query può essere cercata contro un database di MSA (ad esempio Pfam)
- Le regioni regolatorie dei geni sono spesso identificabili da MSA

## 19.2 Metodi Euristici

I metodi esatti non vengono trattati in questa sede: non ci sono soluzioni efficienti e già con 5 sequenze il tempo di computazione è eccessivo (esponenziale) **Metodi progressivi**: usano un albero guida (analogo ad un albero filogenetico) per determinare come combinare uno per uno allineamenti a coppie (progressivamente) per creare un allineamento multiplo.

Esempi: CLUSTAL OMEGA (W), MUSCLE (usato da HomoloGene)

## 20 Clustal Omega

Usa Clusta Omega per fare un MSA progressivo.

### 20.1 Fasi di MSA

**Il MSA progressivo di Feng-Doolittle (1987) alla base di Clustal (W) avviene in 3 fasi**

1. Realizzare una serie di allineamenti a coppie globali (Needleman e Wunsch, algoritmo di programmazione dinamica) di cui si calcola la distanza (matrice delle distanze)
2. Creare un albero guida a partire dalla matrice delle distanze
3. Allineare progressivamente le sequenze

#### 20.1.1 Feng Doolittle fase 1: generare allineamenti

Generare allineamenti a coppie globali

Esempio: allineare 5 globine (1, 2, 3, 4, 5).

**Primo step: a due a due e valutare gli score di ogni possibile allineamento a coppie**

**Numero di allineamenti a coppie necessari per coprire tutte le possibili combinazioni**

- Per n sequenze,  $(n-1) \cdot (n) / 2$
- Per 5 sequenze,  $(4) \cdot (5) / 2 = 10$
- Per 200 sequenze,  $(199) \cdot (200) / 2 = 19.900$

... Quindi per molte sequenze ClustalW è molto lento ed è preferibile usare metodi più veloci (MUSCLE è molto veloce).

#### 20.1.2 Feng-Doolittle fase 2: albero guida

**Convertire i punteggi di similitudine in punteggi di distanza:** è matematicamente più semplice, oltre che più intuitivo, lavorare con le distanze. Una semplice definizione di distanza è data dalla percentuale di residui diversi (100-SI in %) che viene inserita nella matrice delle distanze.

- Dalla matrice delle distanze si calcola l'albero guida con il metodo di clustering neighbor joining che vedrete nel modulo 2.
- Vediamo un semplice esempio di clustering e costruzione di albero guida

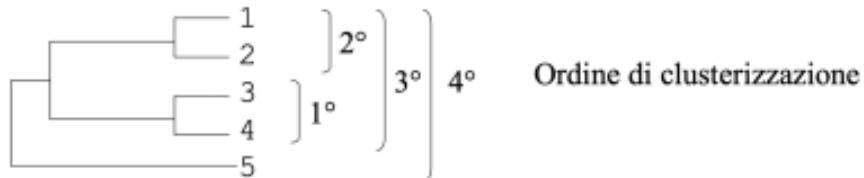
**Il clustering alla base di CLUSTAL(W)** È una matrice di distanze, minore è il numero, maggiore è la similitudine. *Nota: tutte le distanze tra la I e la IV riga sono minori di quelle riportate nella V*

## L'albero guida e la clusterizzazione

1 Hbb_human		2°			
2 Hbb_horse	.17	-			
3 Hba_human	.59	.60	-	1°	
4 Hba_horse	.59	.59	.13	-	
5 Myg_whale	.77	.77	.75	.75 -	
	b_hu	b_ho	a_hu	a_ho	M_w

E' una matrice di distanze, minore è il numero, maggiore è la similitudine...

Nota: tutte le distanze tra la I e la IV riga sono minori di quelle riportate nella V



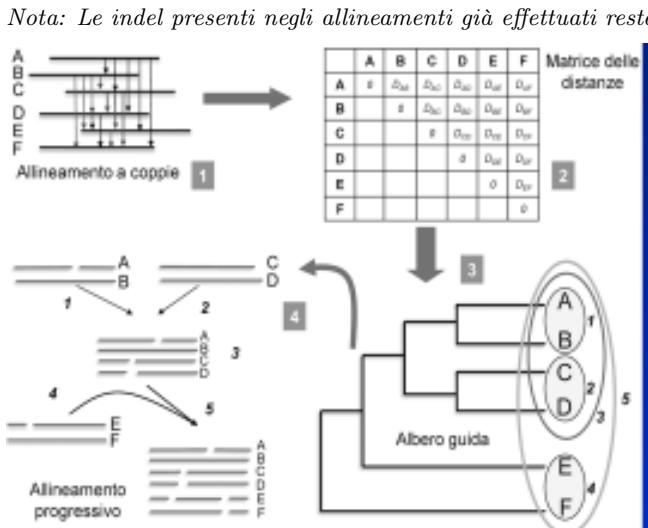
```

PEEKSAVTALWGKVN--VDEVGG Hbb_human
GEEKAAVLALWDKVN--EEEVGG Hbb_horse
PADKTNVKAAGKVGGAHGEYGA Hba_human
AADKTNVKAAWSKVGGHAGEYGA Hba_horse
EHEWQLVLHVWAKVEAGVAGHGQ Myg_whale

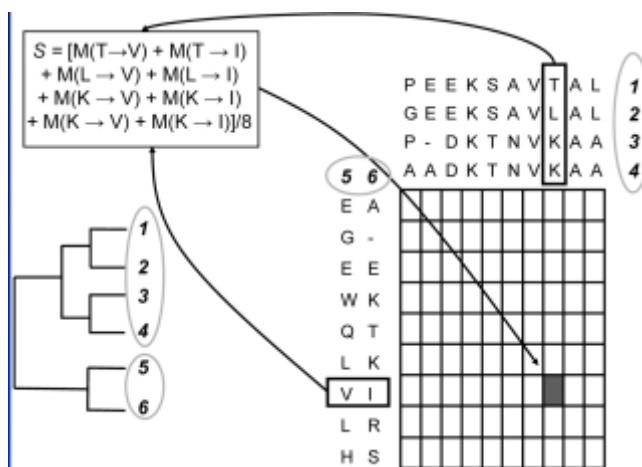
```

### Allineamento finale

- Tutte le sequenze vengono poi allineate progressivamente, seguendo le indicazioni dell'albero guida: prima si allineano le più simili (vicine) e poi progressivamente le più distanti.
- A ogni passaggio si utilizza un algoritmo dinamico di allineamento molto efficiente che accoppia sequenze o gruppi di sequenze
- Il MA è composto a partire a tanti allineamenti a coppie, anche fra gruppi di sequenze.



- Come allineare progressivamente due gruppi di sequenze? Si usa sempre una matrice. È simile all'allineamento dinamico di due sequenze visto.
- Lo score S in ogni casella è la media degli score ottenuti confrontando tutte le possibili coppie di a.a. nella riga e colonna corrispondenti (secondo ad es. BLOSUM62)
- Quando la matrice è stata totalmente calcolata, si calcola il percorso con il migliore allineamento (unisce gli S più alti)



### 20.1.3 Feng-Doolittle fase 3: allineamento progressivo

Fare un MSA in base all'ordine nell'albero guida:

- Iniziare con le due sequenze più strettamente correlate
- Quindi aggiungere la sequenza (o il gruppo) successiva più vicina
- Continuare fino a quando tutte le sequenze vengono aggiunte al MSA
- Regola: "Un gap è per sempre"

### Metodi euristici per l'allineamento multiplo di sequenze

- Cinque globine lontanamente correlate (umano a pianta)
- Cinque globine beta strettamente correlate (specie differenti)

**CLUSTAL W (1.83) multiple sequence alignment**

```

human_NP_000509      MVLHTPEEKSAVTALWUGKVNVDVGGEALGRLLVVYPUTQRFFESFGDLS 50
Pan_troglodytes_XP_508242  MVLHTPEEKSAVTALWUGKVNVDVGGEALGRLLVVYPUTQRFFESFGDLS 50
Canis_familiaris_XP_537902  MVLHTAEEKSLVSGLWUGKVNVDVGGEALGRLLIVYPUTQRFFDSFGDLS 50
Mus_musculus_NP_058652    MVLHTDAEKSAVSCLWAKVMPDEVGGEALGRLLVVYPUTQRYFDSFGDLS 50
Gallus_gallus_XP_444648    MHWHTAAEKQLITGLWUGKVNVAECCGAEALARLLIVYPUTQRFFASFGNLS 50
*** * **. : : **.*** * *.***.***:*****: ***:**

human_NP_000509      TPPDAVMGMPKVKAHGGKVKLGAFSDGLAHLDNLKGTFATLSELHCDKLHV 100
Pan_troglodytes_XP_508242  TPPDAVMGMPKVKAHGGKVKLGAFSDGLAHLDNLKGTFATLSELHCDKLHV 100
Canis_familiaris_XP_537902  TPPDAVMSAKVKAHGGKVKLNSFDGLENLDNLKGTFAKLSELHCDKLHV 100
Mus_musculus_NP_058652    SASAINGGMPPKVKAHGGKVKITAFAENEGLENLDNLKGTFASLSELHCDKLHV 100
Gallus_gallus_XP_444648    SPTAILGMPPMVRAGGGKVLTSFGDAVENLDMIRKNTFSQLSSELHCDKLHV 100
*: *;:*. *:*****: ;*:..: :***:*,**: *****:****

human_NP_000509      PENFRLLGNVILVCVLAAHIFGKEFTPPVQAAYQKVVAGVANALAHKYH 147
Pan_troglodytes_XP_508242  PENFRLLGNVILVCVLAAHIFGKEFTPPVQAAYQKVVAGVANALAHKYH 147
Canis_familiaris_XP_537902  PENFKLLGNVILVCVLAAHIFGKEFTPPVQAAYQKVVAGVANALAHKYH 147
Mus_musculus_NP_058652    PENFRLLGNAIVIVLGHHLGKDFTPAAAQAFQKVVAGVATALAHKYH 147
Gallus_gallus_XP_444648    PENFRLLGNDILIVLAAHFSKDFTPPECQAAVQKLVRRVVAHALAHKYH 147
*****:***: : : **. *;:*** * * *;:***
```

E' un allineamento globale.  
 \* : . e spazio nell'ordine, indicano la bontà dell'appaiamento. \* = identità

### 20.1.4 Perché un GAP è per sempre?

Ci sono molti modi possibili per fare un MSA. Dove aggiungere i gaps è una questione cruciale : i gaps sono spesso aggiunti tra le prime due sequenze (le più vicine).

Cambiare la scelta iniziale di un gap successivamente significherebbe dare più peso, nella definizione dell'evoluzione della famiglia a sequenze più distanti tra loro. Mantenere le scelte dei gap iniziale significa credere che quell'evento sia più affidabile, quindi se serve un gap, prima si valuta dove ne sono già stati visti e poi si considerano altre posizioni.

### 20.1.5 Esempio sulla variabilità dei MSA

**Allineamenti di 5 globine utilizzando 5 diversi programmi** Consideriamo un allineamento multiplo di sequenze (MSA) di cinque globine. Useremo cinque tra i migliori programmi per il MSA: ClustalW, Praline, MUSCLE (utilizzato da HomoloGene), ProbCons e TCoffee. Ogni programma offre punti di forza particolari.

Ci concentreremo su una istidina (H) residuo che ha un ruolo critico nel legare l'ossigeno nelle globine, e dovrebbe quindi essere allineata; tuttavia spesso non è allineata. Ciascuno dei cinque programmi può dare risposte diverse.

La nostra **conclusione** è che non esiste un approccio ottimale al MSA. Decine di nuovi programmi sono stati introdotti negli ultimi anni.

## 21 Confronto

### 21.1 ClustalW

Il più usato, è uscito nel 1994 e non viene più aggiornato. È anche quello di riferimento per i nuovi programmi che spesso hanno prestazioni migliori.



Nota come la regione in cui una istidina è conservata (▼) vari a seconda di quale dei cinque algoritmi è utilizzato

### 21.2 Praline

Introdotto di recente, ha buone prestazioni

## (a) Praline multiple sequence alignment

```

beta globin      .....MVHLTPEEKSAVTALNGKVNVD--EVGGEALGRILLVVYPWTQRFES-FG
myoglobin       ....MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDK-FK
neuroglobin     .....MERPEPELIRQSNRAVRSRSPLEHGTVLFARLFALEPDLLPLFQYNCR
soybean          .....MVAFTEKQDALVSSSFEAFKANIPQYSVVFYTSILEKAPAAKDLFS..FL
rice             MALVEDNNNAVAVSFSEEALVLKSWAILKKDSANIALRFFLKIPEVAPSASQMFS..FL
Consistency     000000000014265438257934573463364343624453686433*35344*50063

beta globin      DLSTPDAMGNPKVKANGKKVLGAFSDG-LAHLDNLKGTFATLSEL..HCDKLH...VDP
myoglobin       HLKSEDEMKAESDLKKHGATVLTALGGILKKKGHHHEAEIKPLAQ..HATKHK...IPV
neuroglobin     QFSSPEDCLSSPEFLDHIRKVMLVIAAATNVEDLSSLSEYLAISLGRKHRAVG...VKL
soybean          A.NGVDP..TNPKLTGHAEKLFLALVRD8AGOLKASGTVVADAA...LGSVHAKAVTD
rice             R.NSDVPPLEKNPKLKTHAMSVEFVMTCEAAQCL.RPKAGKVTVRDITLKRKGATHLKYGVGD
Consistency     3166354224776653*43686354244545133563433354200333540000922

beta globin      ENFRLLGNNVLVCVLAHHF.GKEPTPPVQAAYQKVVAGVANALAHKYH...
myoglobin       KYLEFISECIIQVILQSKH..PGDFGADAQGANNKALELFRKIDMASNYKELGFQG
neuroglobin     SSFSTVGESLLYMLEKCL.GPAFTPATRAAWSQLYGAVVQAMSRGWDE..GK
soybean          PQFVVVKKEALLKTIKAAV.GDFWSDELSRAWEVAYDELAAAIKKAA.....
rice             ARFEVVVKFALLDTIKEEVPAADMWSPAMKSAWSEAYDHLVAAIKQEMKPAE...
Consistency     43744844498258542305336554454*55465426446754322001000

```

Punteggio di affidabilità

Qui  
l'allineamento  
dell'H non è  
buono

Nota come la regione in cui una istidina è conservata (▼) vari a seconda di quale dei cinque algoritmi è utilizzato

## 21.3 MUSCLE

Nuovo programma (molto buono e molto veloce) che sta avendo più successo di ClustalW

## (b) MUSCLE (3.6) multiple sequence alignment

```

beta globin      .....MVHLTPEEKSAVTALNGKVNVD--EVGGEALGRILLVVYPWTQRFES-FG
myoglobin       ....MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDK-FK
neuroglobin     .....MERPEPELIRQSNRAVRSRSPLEHGTVLFARLFALEPDLLPLFQYNCR
soybean          .....MVAFTEKQDALVSSSFEAFKANIPQYSVVFYTSILEKAPAAKDLFS-LA
rice             MALVEDNNNAVAVSFSEEALVLKSWAILKKDSANIALRFFLKIPEVAPSASQMFS-LR

beta globin      DLSTPDAMGNPKVKANGKKVLGAF--SDGLAHLDNLKGTFATLSELHCDKLH--VDP
myoglobin       HLKSEDEMKAESDLKKHGATVLTAL--GGILKKKGHHHEAEIKPLAQSHATKHK--IPV
neuroglobin     QFSSPEDCLSSPEFLDHIRKVMLV--DAAATNVEDLSSLSEYLAISLGRKHRAVGVL
soybean          NGVDP---TNPKLTGHAEKLFLALVRD8AGOLKASGTVVAD---AALGSVHAKAVTD
rice             NSDVP---LEKNPKLKTHAMSVEFVMTCEAAQCL.RPKAGKVTVRDITLKRKGATHLKYGVDA

beta globin      NFRLLGNNVLVCVLAHHFGKE-FTPPTVQAAYQKVVAGVANALAHKYH-----
myoglobin       YLEFISECIIQVILQSKHPGD-FGADAQGANNKALELFRKIDMASNYKELGFQG
neuroglobin     SFSTVGESLLYMLEKCLGP-A-FTPATRAAWSQLYGAVVQAMSRGWDE-----
soybean          QFVVVKKEALLKTIKAAV.GDK-WSDELSRAWEVAYDELAAAIKKAA-----
rice             HFEVVVKFALLDTIKEEVPAADMWSPAMKSAWSEAYDHLVAAIKQEMKPAE-----

```

Qui  
l'allineamento  
dell'H non è  
buono

## 21.4 Probcons

Usa le HMM e con MUSCLE sta guadagnando consensi fra gli utilizzatori



## 21.5 TCoffee

Incorpora informazioni sulla struttura secondaria nel realizzare l'allineamento.



## 22 Metodi Iterativi

**Metodi iterativi:** consistono nel calcolare una soluzione sub-ottimale e modificarla ripetutamente con metodi della programmazione dinamica o altri metodi fino a quando la soluzione converge (non migliora ulteriormente).

*Esempi:* MAFFT, MUSCLE, IterAlign, Praline... Non vengono visti nel dettaglio

### 22.1 La consistenza

Algoritmi potenti, veloci e accurati basati sulla consistenza: valutano la probabilità di allineamento sulla base di database di allineamenti a coppie locali ad alto punteggio e allineamenti a coppie globali a lungo raggio per creare un allineamento completo.

**Concetto:** se abbiamo 3 sequenze A, B e C, e allineiamo A con B e B con C, implicitamente abbiamo allineato anche A con C. Ma magari l'allineamento隐式 (implicit) è incoerente (o inconsistente) rispetto al diretto allineamento A con C: cerco un MSA che ottimizzi la consistenza.

- Sequenza x:  $x_i$
- Sequenza y:  $y_j$
- Sequenza z:  $z_k$

Se  $x_i$  si allinea con  $z_k$  e  $z_k$  su akkubea cib  $y_j$  allora  $x_i$  dovrebbe allinearsi con  $y_j$ . Il programma determina vincoli che cercano di soddisfare questa necessità di coerenza.

ProbCons incorpora elementi di prova da più sequenze per guidare la creazione di un allineamento a coppie.

## 23 T-Coffee (2000)

*Tree-based Consistency Objective Function for Alignment Evaluation*

Attraverso un procedimento progressivo simile a Clustal W si determina un allineamento multiplo che sia il più possibile coerente con “vincoli” esterni.

### 23.1 Vincoli

derivano dall’insieme di allineamenti a coppie tra le seq. da allineare (locali e globali). Il MA finale deve soddisfare il piu’ possibile quei vincoli:

- Primari: danno come peso a ciascun appaiamento di a.a. AB il valore di seq. sim. ottenuta dal relativo allineamento a coppie.
- Estesi: quante volte l’appaiamento AB si ritrova coerentemente in altri allineamenti possibili.

T-Coffee è una collezione di tools per MSA. Ci sono anche vari tools per valutare gli MSA ottenuti.



TCoffee può integrare le informazioni strutturali in un MSA.

## 24 Metodi

*Come facciamo a decidere quale programma utilizzare?*

Ci sono dataset di benchmarking per gli allineamenti multipli allineati meticolosamente a mano, in base a somiglianza strutturale, o con algoritmi automatici ma molto accurati (dispendiosi in termini di tempo e memoria). Alcuni programmi dotati di interfacce che sono più user-friendly rispetto ad altri. La maggior parte dei programmi sono molto buoni, dunque dipende dalla vostra preferenza. Se le vostre proteine hanno una struttura 3D, usatela per aiutarvi a giudicare l’allineamento.

## 24.1 Strategia per la valutazione degli algoritmi per l'allineamento multiplo (benchmarking)

1. Creare o utilizzare un database di sequenze proteiche per le quali la struttura 3D è nota. Così possiamo definire i "veri" omologhi in base ai criteri strutturali.
2. Prova a fare allineamenti multipli di sequenza con molti gruppi diversi di proteine (molto correlati, molto lontani, pochi gap, molti gap, inserzioni, outliers).
3. Confrontare le risposte e scegliere il migliore algoritmo per quel set

**BAlibase:** A benchmark alignments database for the evaluation of multiple sequence alignment programs (database di allineamenti affidabili)

**In conclusione:** Test di valutazione suggeriscono che ProbCons (algoritmo basato sulla consistenza/progressivo), dà i migliori risultati con BAlibase, anche se MUSCLE (algoritmo progressivo) è un programma estremamente veloce e preciso.

ClustalW è il programma più usato. Ha una bella interfaccia (in particolare con ClustalX) ed è facile da usare. Ma diversi programmi sono migliori. Non c'è IL programma migliore: gli output possono essere certamente diversi (soprattutto se si allineano proteine divergenti o sequenze di DNA).

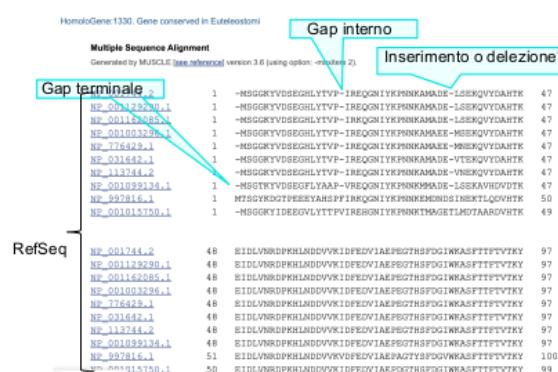
**Pfam** Il database per Pfam (famiglie di proteine) è una risorsa importante per l'analisi delle famiglie di proteine.

## 25 Homologene

Un modo rapido per accedere ad allineamenti multipli; facciamo un esempio: la caveolina (proteina presente nelle caveole della membrana plasmatica, "raft" lipidici ricchi di colesterolo e sfingolipidi).

- Fase 1: in NCBI cambiare il menu a tendina per HomoloGene e immettere caveolin nella casella di ricerca
- Fase 2: verificare i risultati. Prendiamo la prima serie di caveolin 1. Visualizzare l'allineamento multiplo.

- Fase 3: controllare l'allineamento multiplo. Si noti che queste dieci proteine si allineano bene, anche se devono essere inclusi alcuni gaps.



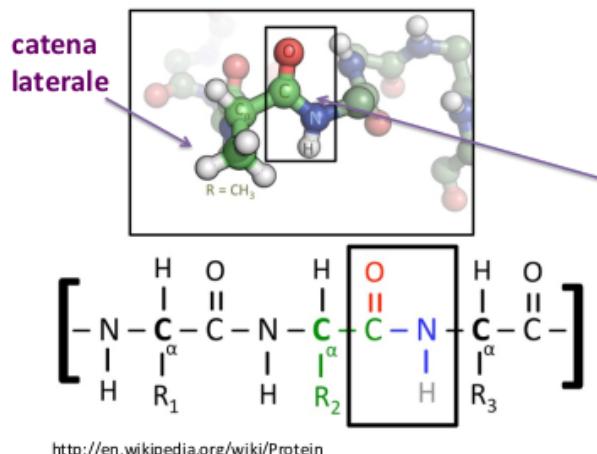
**Lezione 7:**  
**Introduzione alla Bioinformatica Strutturale**

## 26 Introduzione alle Banche dati di proteine

### 26.1 Le proteine: complessi polimeri di amminoacidi

Il legame peptidico lega covalentemente amminoacidi adiacenti costituendo il “backbone” della proteina. Una proteina è formata da una o più catene polipeptidiche.

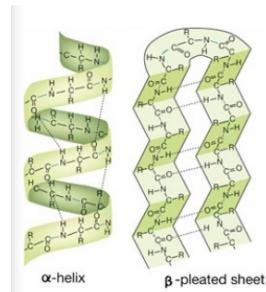
**Legame peptidico** l’alternanza di legami peptidici costituisce il backbone e determina la struttura secondaria della proteina.



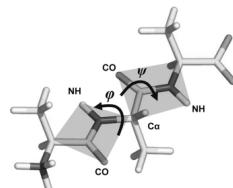
### 26.2 Struttura secondaria

$\alpha$ -eliche e foglietti  $\beta$  si formano a seguito di ben precisi pattern di legami H tra carbonili e gruppi NH

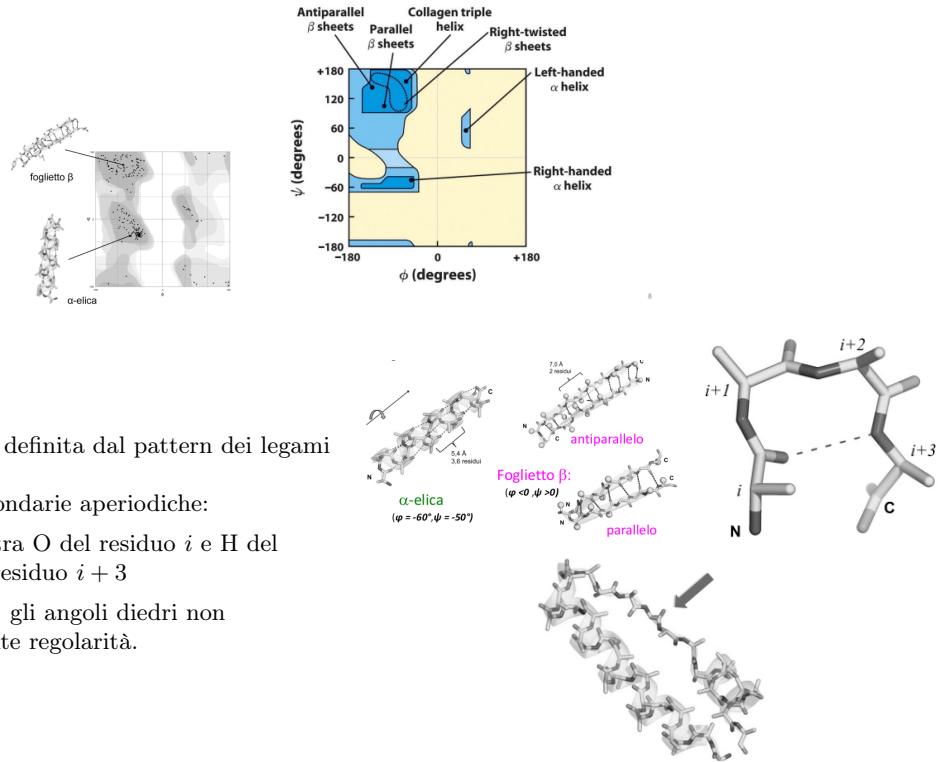
- **$\alpha$ -elica:** legame a H tra residuo  $i$  e  $i + 4$  (C=O dell’uno e N-H dell’altro). L’elica che si forma ha un giro completo ogni 3.6 a.a e la distanza media è 0.54 nm
- **$\beta$ -sheet:** più filamenti  $\beta$  disposti uno accanto all’altro e collegati tra loro da tre o più legami H che formano una struttura planare molto compatta



- Gli angoli  $\varphi$  e  $\psi$  sono definiti dai legami singoli che uniscono il  $C_\alpha$  al gruppo NH e CO contigui e dai due legami peptidici (planari) coinvolti. Essi non possono assumere qualunque valore per ragioni steriche.



- Le coppie  $(\varphi, \psi)$  definiscono il plot di Ramachandran (conformazioni permesse sulle aree ombreggiate).



**La struttura secondaria** è definita dal pattern dei legami a idrogeno.

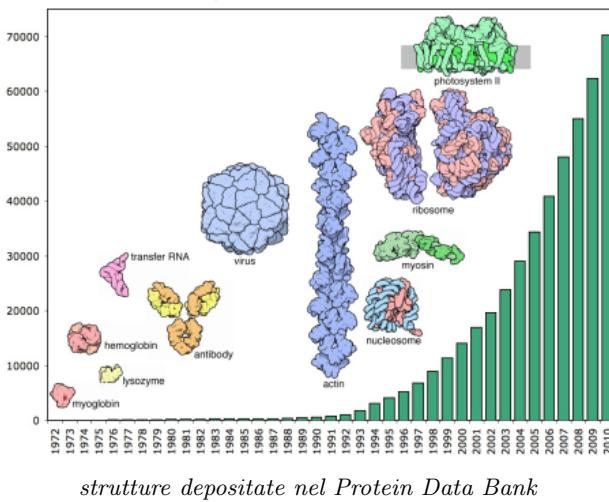
Esistono anche strutture secondarie aperiodiche:

- **$\beta$ -turn:** il legame H è tra O del residuo  $i$  e H del gruppo amminico del residuo  $i + 3$
- Random coil o "ansa": gli angoli diedri non presentano generalmente regolarità.

### 26.3 Struttura terziaria e quaternaria

La **struttura terziaria** e **quaternaria** dipendono dalle interazioni fra le catene laterali, che hanno proprietà chimico-fisiche molto diverse.

La struttura 3D di una proteina è molto complessa (1958, John Kendrew, prima struttura della mioglobina)



l'organizzazione strutturale delle proteine è ancora più complessa: Si identificano motivi strutturali e domini, inoltre cofattori, gruppi prostetici. *Esempio: il motivo EF-hand e la calmodulina*

## 27 Protein Data Bank

Ad oggi (22 novembre 2021) ci sono 184407 strutture depositate.

- PDB: È la principale risorsa per le strutture di macromolecole (Proteine, Supercomplessi, Acidi nucleici)
- Le strutture sono determinate mediante cristallografia, NMR e microscopia crioelettronica (cryoEM)

- Oltre ai file di struttura (.PDB) sono presenti informazioni sulle sequenze e molti strumenti per:  
Analisi di struttura, Visualizzazione di ligandi, Determinazione delle similitudini

## 27.1 Il file .PDB

Si può scaricare il file .pdb dal link a destra, e aprirlo con un qualsiasi editor di testo.

Lo stesso file aperto con un visualizzatore molecolare (es. PyMol o VMD) permette di visualizzare la struttura 3D della macromolecola.

**Il file .PDB: formato testuale per la descrizione di strutture 3D di macromolecole biologiche.**

Contiene la descrizione e l'annotazione di strutture di proteine e acidi nucleici tra cui: coordinate atomiche, rotameri di catene laterali osservati, assegnazione a particolari strutture secondarie, e connettività atomica. Altre molecole come acqua, ioni, acidi nucleici, ligandi e così via possono essere descritti nel formato pdb.

**Un breve peptide con sequenza ripetuta** I vari campi hanno significati precisi.

```

HEADER      EXTRACELLULAR MATRIX          22-JAN-98   1A31
TITLE       X-RAY CRYSTALLOGRAPHIC DETERMINATION OF A COLLAGEN-LIKE
TITLE       2 PEPTIDE WITH THE REPEATING SEQUENCE (PRO-PRO-GLY)
...
EXPDTA    X-RAY DIFFRACTION
AUTHOR    R.Z.KRAMER,L.VITAGLIANO,J.BELLA,R.BERISIO,L.MAZZARELLA,
AUTHOR    2.B.BRODSKY,A.ZAGARI,H.M.BERMAN
...
REMARK 350 BIOMOLECULE: 1
REMARK 350 APPLY THE FOLLOWING TO CHAINS: A, B, C
REMARK 350 BIOMT1  1  1.000000  0.000000  0.000000      0.00000
REMARK 350 BIOMT2  1  0.000000  1.000000  0.000000      0.00000
...
SEQRES   1 A    9  PRO PRO GLY PRO PRO GLY PRO PRO GLY
SEQRES   1 B    6  PRO PRO GLY PRO PRO GLY
SEQRES   1 C    6  PRO PRO GLY PRO PRO GLY
...
ATOM     1 N   PRO A  1    8.316  21.206  21.530  1.00 17.44      N
ATOM     2 CA  PRO A  1    7.608  20.729  20.336  1.00 17.44      C
ATOM     3 C   PRO A  1    8.487  20.707  19.092  1.00 17.44      C
ATOM     4 O   PRO A  1    9.466  21.457  19.005  1.00 17.44      O
ATOM     5 CB  PRO A  1    6.460  21.723  20.211  1.00 22.26      C
...
HETATM 130 C   ACY  401    3.682  22.541  11.236  1.00 21.19      C
HETATM 131 O   ACY  401    2.807  23.097  10.553  1.00 21.19      O
HETATM 132 OXT ACY  401    4.306  23.101  12.291  1.00 21.19      O
...

```

## 27.2 Pfam e Prosite

- Utili per studiare e catalogare le strutture proteiche.
- Le similitudini (domini, fold, ponti disolfuro, ecc) possono essere usate per inferire la funzione.
- Utile tra proteine simili e omologhe in specie diverse.

### 27.2.1 Pfam

Suddivide le proteine e ne descrive le caratteristiche in famiglie in base a metodi statistici:

- Allineamenti
- HMM

### 27.2.2 Prosite

- Individua, data una sequenza, le possibili famiglie di appartenenza.
- Determina possibili caratteristiche funzionali; domini; cofattori; siti attivi; amminoacidi strutturalmente importanti; livello di conservazione

### 27.2.3 CATH

- Database che definiscono famiglie strutturali.
- Aiutano a predire le strutture e a caratterizzarle (secondo un'idea evoluzionistica della funzione).

**CATH:** a hierarchical domain classification of protein structures in the Protein Data Bank.

Classificazione in modo curato sulla base di:

- CLASSE (contenuto e tipo di strutture secondarie)

- ARCHITETTURA (descrizione dell'orientamento delle strutture secondarie senza tener conto delle connessioni)
- TOPOLOGIA (tiene conto delle connessioni che caratterizzano le strutture secondarie)
- (H)OMOLOGIA (raggruppa proteine con strutture e funzioni simili)

**È possibile prevedere proprietà strutturali di proteine a partire dalla sequenza?** In molti casi (ma non in tutti...) la struttura 3D assunta da una proteina è determinata totalmente dalla propria sequenza. Questo è alla base della scoperta di Christian Anfinsen (1957) e del paradigma del protein folding.  
La RNasi A denaturata chimicamente è in grado di ripiegarsi in forma nativa e cataliticamente attiva se vengono gradualmente a mancare le condizioni denaturanti: l'informazione sul corretto folding deve essere contenuta nella sequenza.

## 28 Reti Neurali

Le reti neurali: un metodo efficace per predizioni di elementi strutturali proteici.

### Premesse:

- È possibile riprodurre artificialmente alcune funzioni cognitive del cervello e “allenarle”, come i bambini apprendono e riconoscono.
- Si possono usare architetture di calcolo dette reti neurali artificiali (ANN)
- Sulla base di caratteristiche distintive di un oggetto (osservazione) la rete deve essere in grado di classificarlo
- Il numero e le modalità di interconnessione di equazioni di una rete ne definiscono l'architettura, che viene definita e organizzata durante l'apprendimento

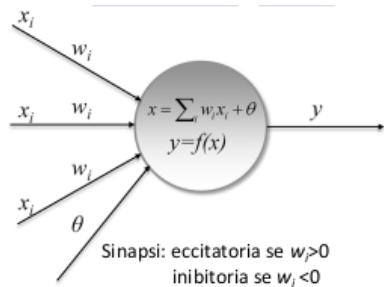
### 28.1 Struttura di una rete neurale

Ogni rete deve possedere:

1. un'area in cui i dati distintivi dell'oggetto entrano;
2. una in cui sono elaborati;
3. un'altra in cui viene emesso il risultato.

l'unità elementare di calcolo di una NN è il neurone: la connessione tra i vari neuroni è detta sinapsi. Ogni neurone riceve uno o più input attraverso valori numerici  $x_i$ , e restituisce dopo l'elaborazione l'output  $y$ .

- $x$  è l'ingresso e si può esprimere come somma pesata dei singoli input  $x_i$  provenienti dai neuroni  $i$ , ciascuno con un peso  $w_i$ , e da un parametro di modulazione  $\theta$ .
- $y$  è la funzione di attivazione del neurone.



l'output  $y$  di un neurone può costituire l'input di un altro neurone a valle, e così via, fino ai neuroni di output che emettono un valore che tiene conto di tutta l'informazione transitata nel network.

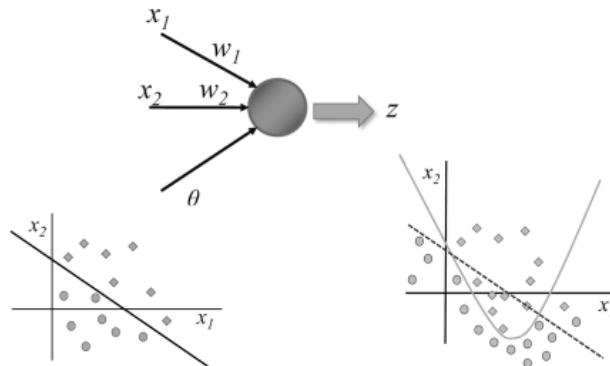
I neuroni possono essere connessi a strati (layers) completamente connessi fra loro: ogni neurone di uno strato è connesso a tutti i neuroni dello strato successivo. Il flusso dell'informazione procede dal primo all'ultimo strato (feed forward): il primo strato è l'“occhio”, l'ultimo la “bocca”, in mezzo tutti gli strati di elaborazione.

*Esempio semplice: il perceptrone, NN a 2 strati. Qui riceve in ingresso 5 valori e produce in output il valore 0 o 1 a seconda della soglia t e del risultato della funzione di attivazione. Le sinapsi amplificano o attenuano il segnale in base ai pesi w*

- Ingressi:  $x_1, x_2, x_3, x_4, x_5$
- con  $w_1, w_2, w_3, w_4, w_5$  pesi
- Funzione di attivazione:  $y = \frac{1}{e+e^{-\sum_i x_i w_i}}$
- Uscita: se  $y > t$  allora emetti 1, altrimenti 0

## 28.2 Architettura e apprendimento di reti neurali

Una rete semplicissima riesce a classificare gli oggetti in base a pochi descrittori ( $x_1, x_2, \dots, x_N$ ) linearmente separabili.



Se il “perceprone apprende” l’equazione della retta, esso decide a quale parte di piano appartiene un (nuovo) oggetto caratterizzato dalle coordinate  $(x_1, x_2)$

Se gli oggetti di due classi non sono separabili linearmente serve una rete in grado di apprendere funzioni di ordine superiore a quello lineare (retta)

Il problema della percezione di relazioni di ordini superiori al primo si risolve mediante reti con strati nascosti, che richiedono algoritmi più complicati.

Prima di usare la rete come classificatore di oggetti sulla base di un assetto di descrittori numerici, la rete deve essere guidata all’apprendimento, cioè deve imparare a riconoscere quegli oggetti.

**Apprendimento:** si presenta alla rete una serie di oggetti che appartiene o meno alla classe in esame e si interroga la rete. L’apprendimento ha successo se la rete riesce a riconoscere gli elementi essenziali che definiscono l’oggetto e può generalizzare.

Durante l’apprendimento si modulano i parametri  $w_j$ , eventualmente anche  $\theta$  e  $t$  per ottimizzare l’accuratezza in un apprendimento supervisionato.

**Esempio:** si forniscono alla NN in sequenza alcuni vettori di  $n$  elementi  $x_i$  descrittivi di oggetti noti da riconoscere (peso, lunghezza, colore...). La rete deve emettere 1 per ogni oggetto riconosciuto. Se non lo classifica bene, si modificano i parametri fino ad ottenere 1. I parametri si ottimizzano (fino ad un certo limite) con la retropropagazione (cambiando  $w_i$ ).

## 28.3 Collegamento con le proteine

Le reti neurali possono riconoscere la presenza di particolari segnali di sequenza e prevedere proprietà strutturali e funzionali associate al segnale. Esempio: la struttura secondaria.

**Storicamente:** approcci statistici basati sull’osservazione (frequenza di residui in particolari ss; “propensione”; Chou e Fasman, 1974). Il problema però è molto adatto alle NN.

**L’idea:** La rete può “leggere” una porzione di sequenza e decidere se il residuo centrale appartiene alle due principali ss periodiche ( $\alpha$ -elica,  $\beta$ -sheet) o a nessuno delle due. Usando poi finestre scorrevoli si può estendere la ricerca a tutta la sequenza

### La rete di Holley & Karplus (1989) per la previsione di ss

- Ha uno strato di ingresso, uno strato nascosto e uno di output formato da due neuroni
- Input: 17 neuroni (lunghezza ideale) che “leggono” una finestra di 17 aa
- La rete prevede la conformazione del residuo centrale (il nono)
- La sequenza scorre dall’N- al C-terminale (residui 1-17, 2-18, ecc.) e l’info si propaga feed-forward
- Output dato dai due neuroni in uscita: (1,0)  $\alpha$ -elica; (0,1)  $\beta$ -sheet; (0,0) random coil

Per allenare la rete occorrono due set di dati: un training set (campione di apprendimento) e un test set (verifica della performance). Questi non devono mai coincidere.

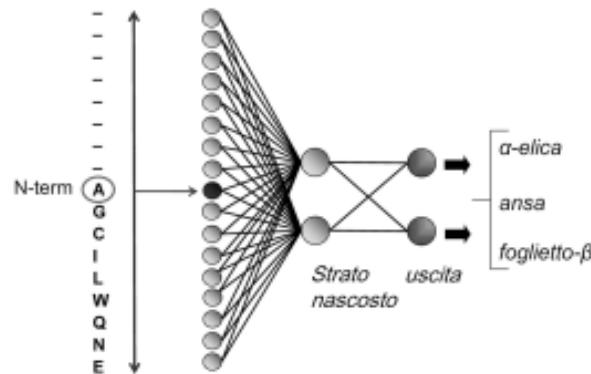
- Training set: per modulare i parametri della rete
- Test set: per valutare l'accuratezza e l'errore della NN

Ese. due set di proteine non omologhe a struttura nota per allenare e validare la rete.

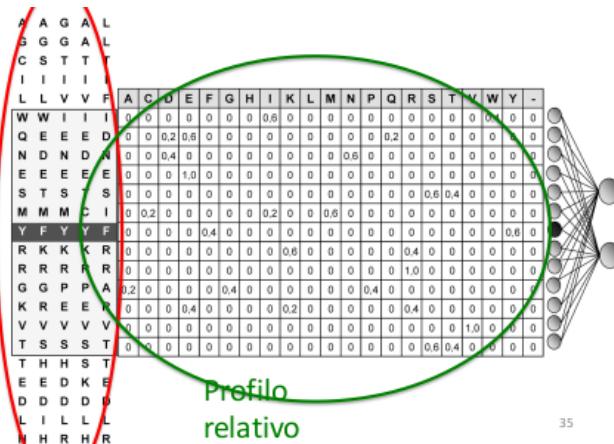
**Codifica di un amminoacido:** Come può la rete “vedere” un amminoacido?

In genere, un aa è codificato con un vettore di 21 componenti, tutti uguali a zero tranne quello la cui posizione identifica l'aa. l'ultima posizione è dedicata a caratteri riempitivi (N- e C- terminale).

I caratteri riempitivi servono ad esempio per predire la conformazione del residuo all'N terminale. Se esso diventa centrale, devo riempire le posizioni precedenti



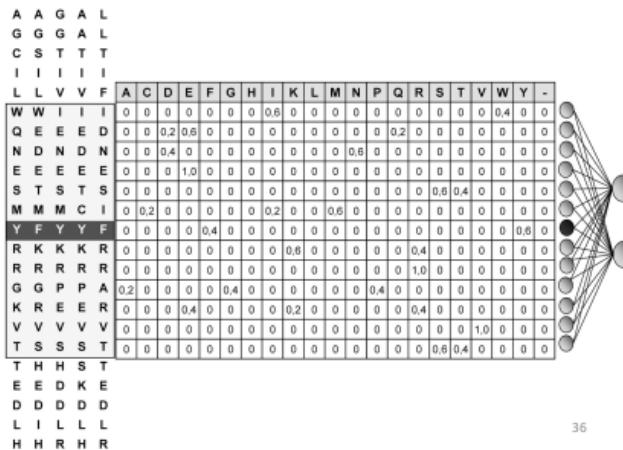
**Codifica di un allineamento multiplo:** Notevole aumento di accuratezza della previsione di strutture secondarie.  
Nei singoli neuroni non entrano più singoli residui in input, ma profili contenenti le frequenze del residuo.



35

Nei singoli neuroni non entrano più singoli residui in input, ma profili contenenti le frequenze del residuo.

- Ciascun neurone in ingresso contiene 21 unità corrispondenti alle 21 colonne del profilo (ultima colonna: eventuali indel)
- Esempio: finestra scorrevole con N=13



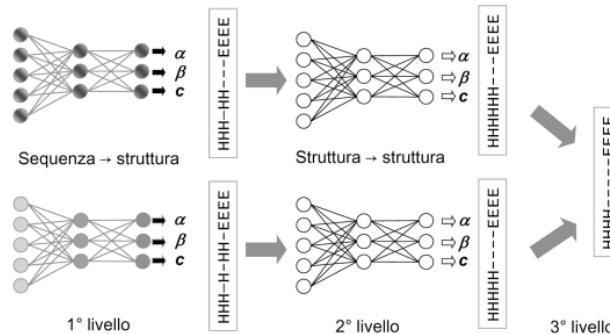
36

### 28.3.1 Storicamente: Profile Network from Heidelberg (PHD)

Sviluppato da Rost e Sander (1993) è il primo sistema di previsione che sfrutta questa strategia. È basato su tre livelli, ciascuno contenente varie reti alternative:

- Livello 1: input è MA sotto forma di profilo. 13 gruppi di 21 neuroni ciascuno, uno strato nascosto e 3 unità di uscita (una per  $\alpha$ -elica, una per  $\beta$ -sheet e una per random coil). Il neurone che emette il valore più elevato indica la ss prevista per il residuo al centro della finestra (mappatura sequenza-struttura)
- Livello 2: l'output del livello 1 può contenere incongruenze. Es., singoli residui in elica. C'è una seconda rete allenata a risolvere questi inghippi (17 gruppi di 3 neuroni, ciascuno riceve in ingresso i tre valori prodotti dal livello precedente) (mappatura struttura-struttura)
- Livello 3: media risultati leggermente diversi ottenuti ripetendo le previsioni utilizzando reti con parametri leggermente diversi; la "giuria" assegna la predizione in base alla media delle varie elaborazioni e fornisce l'attendibilità della previsione

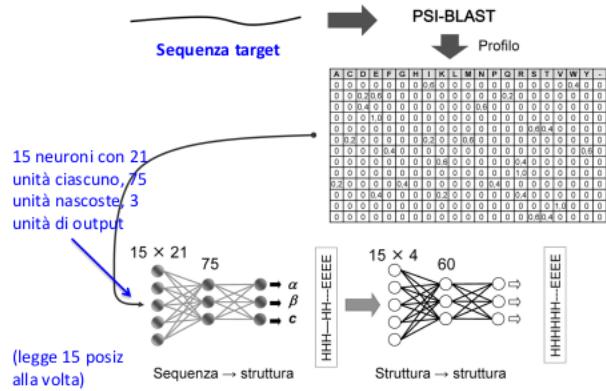
#### Schema del funzionamento di PHD



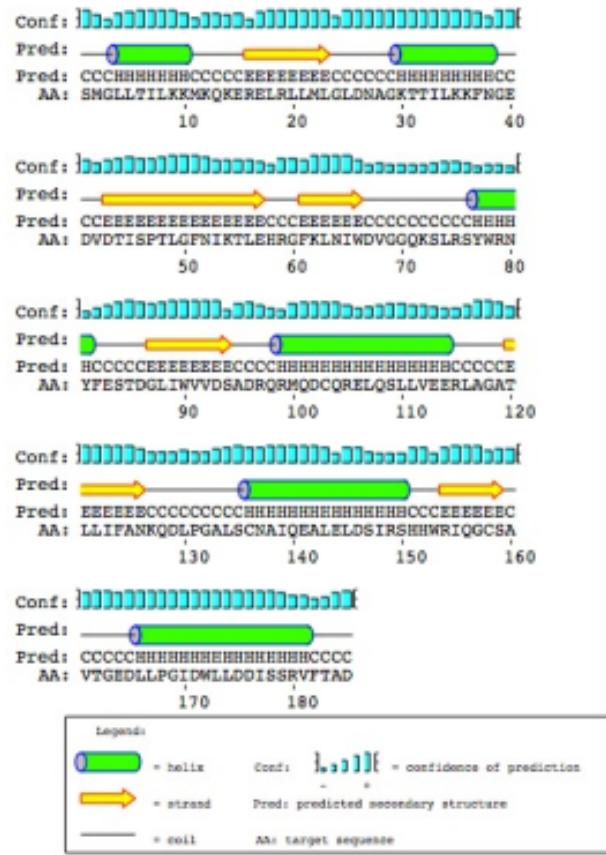
*Nota: il numero di neuroni dei vari strati non è fedelmente riportato qui! Inoltre non sono riportate tutte le reti alternative per ciascun livello: qui solo due, nella realtà fino a 8-9*

### 28.3.2 Schema del funzionamento di PSIPRED

Sviluppato da David Jones (1999), utilizza 3 iterazioni di PSI-BLAST per costruire il profilo, raccogliendo le sequenze omologhe alla query e calcola il profilo mediante BLOSUM62.

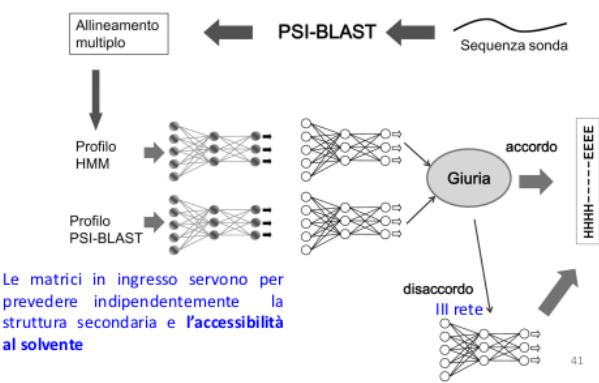


Esempio di output:



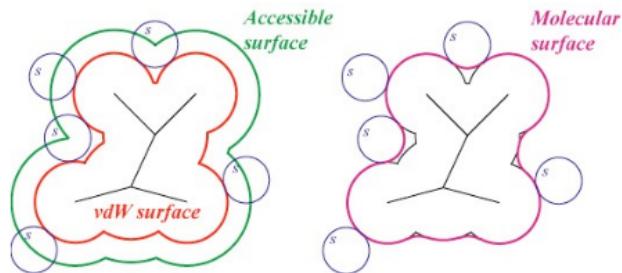
### 28.3.3 Schema del funzionamento di JPRED

Sviluppato da Cuff & Barton (2000), al primo livello sfrutta un MA codificato mediante una matrice PSSM calcolata da PSI-BLAST o un HMM. Esiste un secondo livello per rimuovere incongruenze ed una giuria che eventualmente coinvolge una terza rete. Accuratezza 81.5%

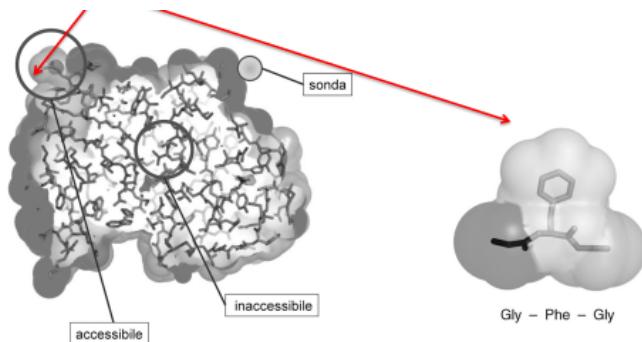


## 28.4 Altre proprietà

Le reti neurali possono essere utilizzate per predire anche altre proprietà delle proteine. Ad esempio: l'**accessibilità al solvente**: si fa “rotolare” una sfera di 1.4 Å sulla superficie della proteina e si determina la SAS (solvent- accessible surface). La SAS è diversa dalla superficie di Van der Waals (unione delle superfici ottenute dai raggi di VdW) e dalla superficie molecolare (inviluppo inferiore generato dal probe).

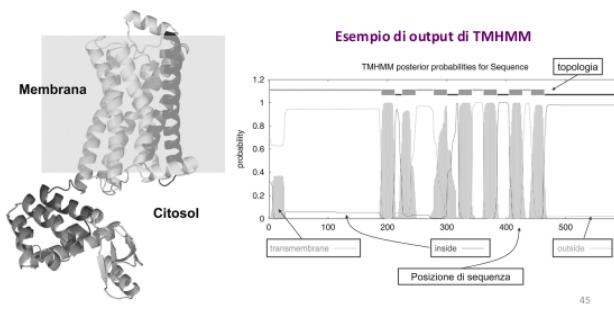


Le proteine strutturate in soluzione presentano alcuni residui esposti al solvente, ed altri “sepolti” (buried) al loro interno. Può essere utile valutare l'accessibilità relativa del residuo X per confronto con il tiripeptide libero G-X-G. Domanda: che frazione di questa Phe è effettivamente esposta? Risposta:  $24/210\text{Å}^2 = 11.4\%$



Jpred permette anche di predire l'accessibilità al solvente. Esempio: sol 25 - buried (B) se l'accessibilità < 25%. Nell'output è specificata la predizione delle singole matrici usate (hmm o pssm).

**Topologia** La topologia di membrana si può predire mediante NN. Esistono vari tools che migliorano quelli storici funzionanti su indici di idrofobicità. I più moderni ed efficaci usano le NN (es. PHDhtm, Rost et al., 1995) o modelli nascosti di Markov. Un esempio è il programma TMHMM di Krogh et al. (2001) che ha un'accuratezza dell'80%



45

## Lezione 8: Introduzione alla Biologia dei Sistemi

## 29 Cos'è la Biologia dei Sistemi?

La scienza è spesso un fatto di scala, quello che vediamo dipende da quello che *stiamo guardando*. Ingrandire o rimpicciolire un'immagine, spesso può portare a scenari differenti.

La **Biologia dei Sistemi** è un approccio olistico di comprensione della biologia.

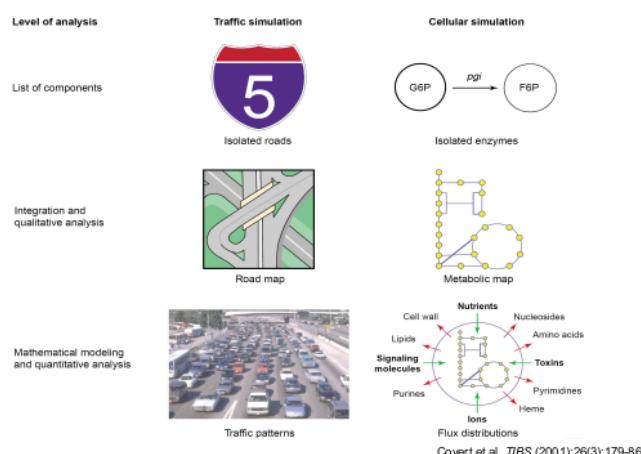
- Si occupa delle interazioni tra i componenti dei sistemi biologici, e come portano alle funzioni e comportamenti conosciuti.
- Assume che molte proprietà della vita emergono solo a livello di sistema, poiché il comportamento di un sistema come un tutt'uno non può essere spiegato solo attraverso i suoi componenti.
- Si applica sia alle cellule che agli organismi.

*"... il pluralismo di cause ed effetti nelle reti biologiche è affrontato meglio tramite l'osservazione, attraverso misure quantitative, componenti multipli simultaneamente e integrando rigorosamente i dati con i modelli matematici."*

Sauer et al., Science 316:550 (2007)

### 29.1 Dai componenti ai sistemi

*Attraverso le computer simulations*



### 29.2 Nascita della Biologia dei Sistemi

La Biologia dei Sistemi non è una scienza del tutto nuova, trova infatti radici in:

- Modellamento quantitativo della cinetica degli enzimi
- Simulazione di processi neurofisiologici
- Teoria cibernetica e di controllo

Esempi storici:

1. **Hodgkin AL, Huxley AF (1952):** A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol*, 117: 500- 544.
2. **Noble D (1960):** Cardiac action and pacemaker potentials based on the Hodgkin- Huxley equations. *Nature*, 188: 495-497

Gli anni 60 e 70 hanno visto un'evoluzione degli approcci allo studio di sistemi molecolari complessi: Analisi del controllo metabolico e Teoria dei sistemi biochimici.

### 29.3 È necessaria?

Nell'era *-omica* la biologia molecolare e la biochimica sono evolute molto rapidamente portando a un grosso ammontare di dati da analizzare.

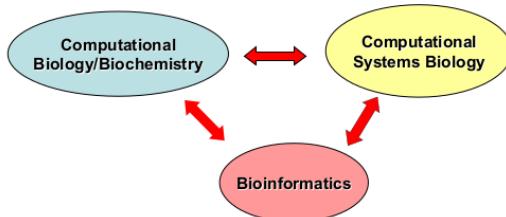
### Screening ad alto rendimento

milioni di test veloci biochimici, genetici e farmacologici.



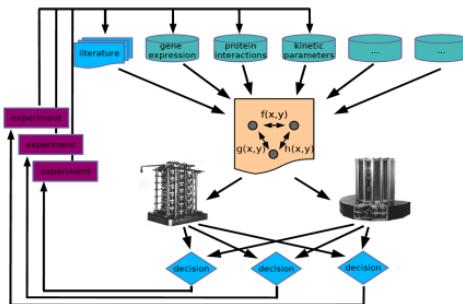
Identificazione di composti attivi, anticorpi o geni che modulano un particolare pathway.

Molti dati da analizzare equivalgono all'uso estensivo di matematica e modellamento in generale. Ad oggi è possibile grazie al potere computazionale migliorato.

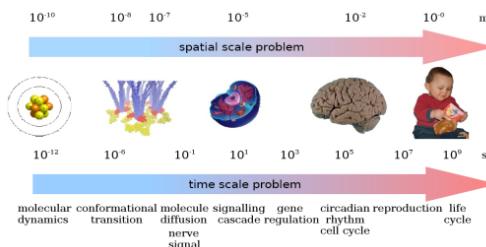


Il framework teoretico è la Teoria dei Sistemi: descrive ogni gruppo di oggetti che lavora concertatamente per produrre risultati, spesso un comportamento complesso.

### Modelli matematici: il cuore della Biologia dei Sistemi



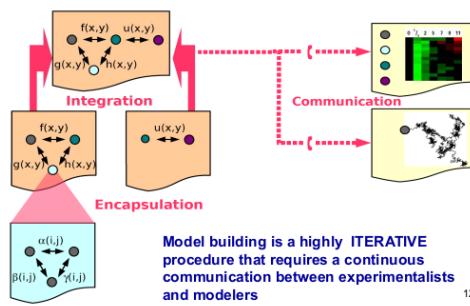
### Modellare: un problema multiscala



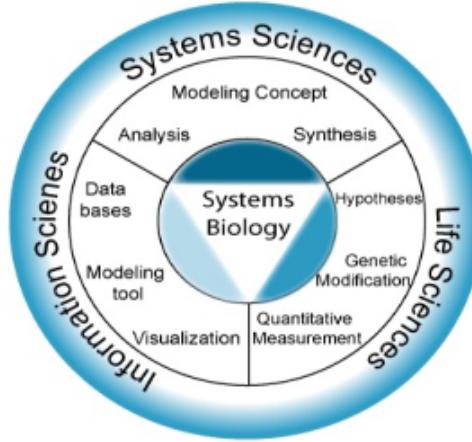
Abbiamo bisogno di trasferire e integrare l'informazione da differenti scale spaziali e temporali per arrivare al modello **predittivo** finale.

### 29.4 La costruzione di modelli computazionali dev'essere sistemico

- Lo sviluppo di modelli quantitativi di anche semplici sistemi viventi richiede una conoscenza estensiva della biologia, da reazioni biochimiche a quelle fisiologiche
- Molti approcci diversi devono essere usati: ad esempio per le reti biochimiche si possono usare le time-series (continue o discrete), steady-state analyses (MCA, FBA), descrizioni logiche, ecc.
- Pathways molto grandi non possono essere costruiti in una prova sola. Parliamo di centinaia di migliaia di interazioni.
- Possiamo pensare di usare e riusare modelli fatti di moduli differenti.



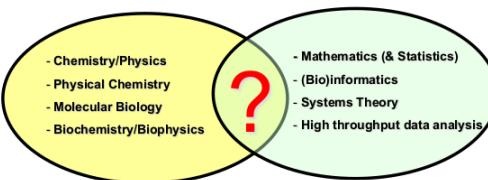
## 30 Scienze riduzionistiche



Cosa sono la chimica? La fisica? La biologia molecolare?

### 30.0.1 Descrizione di sistemi/processi biochimici

**Obiettivo:** capire i sistemi/processi biochimici. Da dove iniziamo?



#### Biologia strutturale/chimica

Approccio riduzionista; alta risoluzione; scale spaziali e temporali piccole

#### Biologia di Sistema

Approccio sistemico; nessun limite di scale spazio-temporali; bassa risoluzione

### 30.1 Qual è l'intersezione?

Se intersechiamo la Biologia Strutturale e la Biologia di Sistema ... ?

Design di proteine/farmaci per uso biomedico/industriale. Più in generale ... descrizione meccanica della rete molecolare nei suoi dettagli più sottili per il fine della capibilità predittiva.

### 30.2 Esempio pratico: Rete di Proteine

#### 1. La rete

Cosa sappiamo del sistema intero? (dati -omici, analisi matematica o di modello)

## 2. I componenti

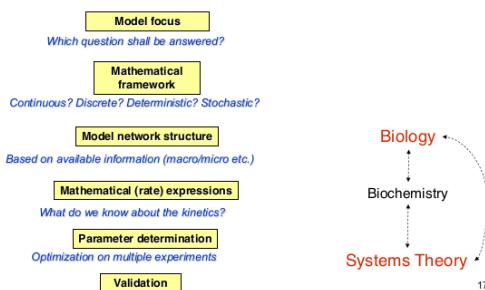
Cosa sappiamo delle due proteine? (dati strutturali, dati biochimici/biofisici)

**Cosa succede ai sistemi se queste particolari interazioni si perturbano?** Ad esempio a causa di mutazioni, modificazioni chimiche, ...

L'intersezione è particolarmente chiara quanto lo scopo è la Biologia Sintetica: combina scienza e ingegneria per sintetizzare nuove funzioni e sistemi.

# 31 Modellare la struttura della rete

## Modellamento dei sistemi biochimici: Framework dei Modelli Sistemici



### 31.1 Struttura

- Quali sono gli elementi del modello?
- Quali sono le interazioni tra questi elementi?
- Quali sono i link all'esterno non modellato?
- Il modello intero può essere diviso in moduli che possono essere trattati indipendentemente?

#### Definizione:

- La struttura del modello della rete in termini di "rete di elementi interconnessi"
- L'esterno (può essere costante o dinamico)
  - Un esterno non modellato non è l'esterno di una cellula, ecc. ma più astratto e generico
- Moduli che possono essere modellati indipendentemente

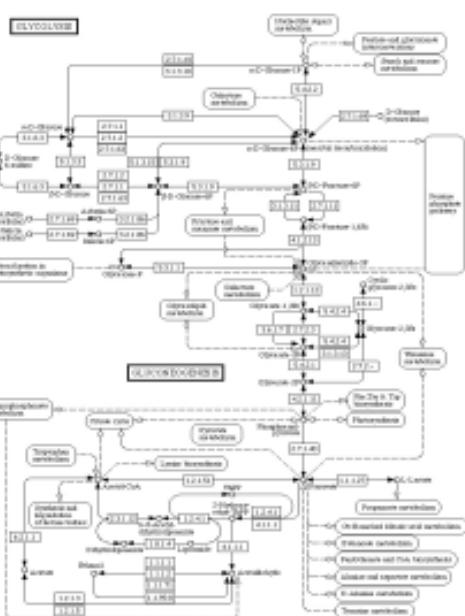
### 31.2 Da dove partiamo?

**Conoscenza biologica** Letteratura, database, ecc.

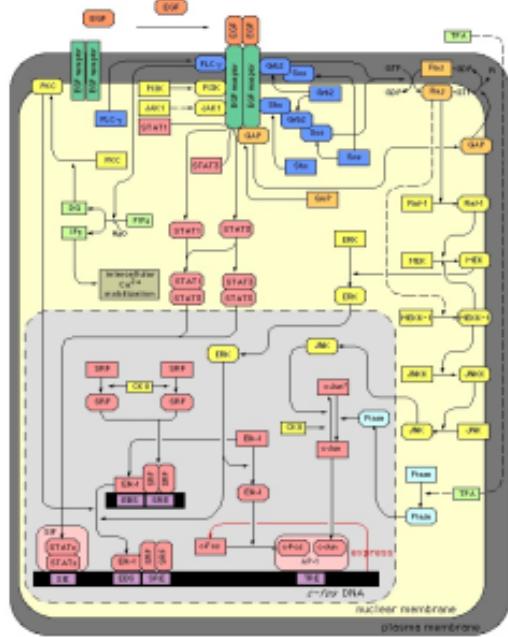
**Model Network Structure** *Struttura del modello della rete*

Diagramma di processi (Kitano, Science, 2003)

## Esempi di database

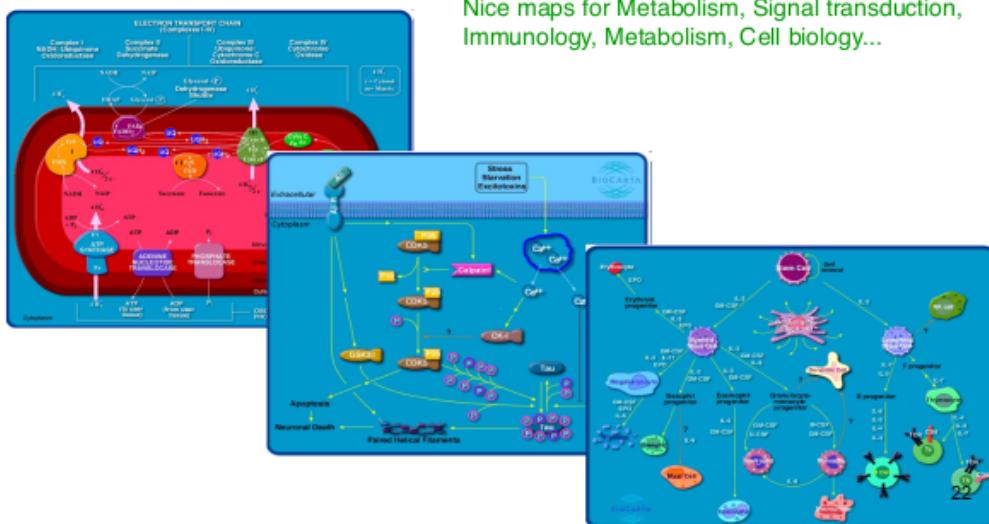


Kyoto Encyclopedia of Genes and Genomes



Signaling Pathway Database

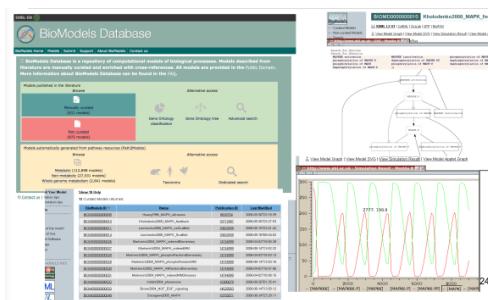
**BioCarta** <http://www.biocarta.com/>  
Nice maps for Metabolism, Signal transduction, Immunology, Metabolism, Cell biology...



### 31.2.1 I Database di Biomodels

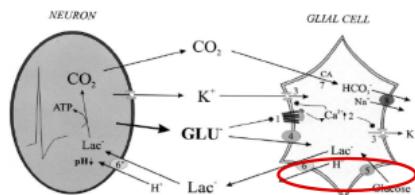
L'idea dietro i Biomodels Database:

- Immagazzinare e fornire modelli quantitativi di interesse biomedicale
- Solo modelli descritti nella letteratura scientifica peer-reviewed
- I modelli sono curati: dei software fanno il controllo della sintassi, mentre un essere umano cura la semantica
- I modelli sono simulati per controllare che i riferimenti corrispondono
- I componenti dei modelli sono annotati, per migliorare l'identificazione e il recupero
- I modelli sono accettati in vari formati, e essere adatti a tanti altri.



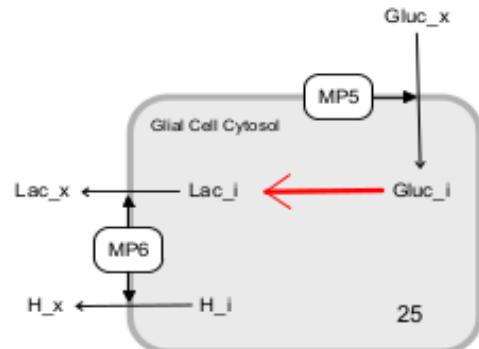
### 31.3 Esempio di Struttura del Modello di Rete

Trasporto di glucosio e energetici in cellule gliali.



Descrizione schematica del meccanismo fisiologico

Ci concentriamo solo sull'assorbimento del glucosio, la produzione ed estrusione del lattato e lo scambio di protoni. Il trasportatore per H e Lac è il Monocarbossilato Cotrasportatore.



Struttura del modello di rete

Struttura:

- Rappresentazione grafica della rete
- Definire: reagenti, prodotti, modificatori, compartimenti

**Prossimo step: Aggiungere le informazioni stechiometriche** Ad esempio:

R\_MP5: 1 Gluc\_x → 1 Gluc\_i

R\_Glyc: 1 Gluc\_i → 1 Lac\_i

R\_MP6: 1 Lac\_i → 1 Lac\_x

2 H\_i → 2 H\_x d/dt (concentraation vector) = stoichiometric matrix \* reaction rates vector

- In maniera formale:
- $\dot{x} = Nv(x)$

$$\frac{d}{dt} \begin{pmatrix} Gluc_x \\ Lac_x \\ H_x \\ Gluc_i \\ Lac_i \\ H_i \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & 2 \\ 0 & -2 & 0 \end{pmatrix} \begin{pmatrix} R_{MP5} \\ R_{MP6} \\ R_{Glyc} \end{pmatrix}$$

- Modello stechiometrico - spesso chiamato 'modello statico'
- Informazioni addizionali

$$\begin{aligned} R_{MP5} &= R_{MP5}(Gluc_x, MP5) \\ R_{MP6} &= R_{MP6}(Lac_i, H_x, H_i, MP6) \\ R_{Glyc} &= R_{Glyc}(Gluc_i, \dots) \end{aligned}$$

**Espressioni matematiche:**

Definizione delle espressioni matematiche per le interazioni tra gli elementi (trascrizioni, traslazioni, degradazioni, reazioni, trasporti, ecc.). Bisogna definire la matematica dietro il modello.

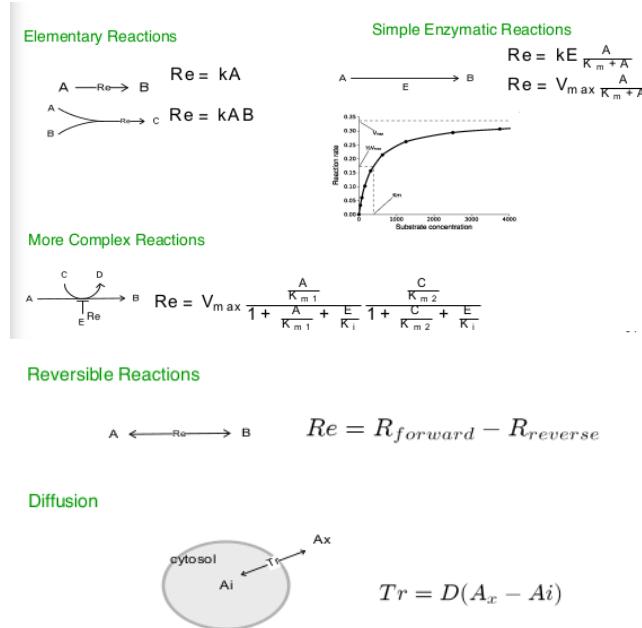
### 31.4 Model Tasks

#### 31.4.1 Espressioni matematiche nel modellamento dinamico

**Dynamic model**  $\begin{cases} \dot{x} = Nv(x, p) \\ v_i(x, p) = ??? \end{cases}$  Come determiniamo  $v_i(x, p)$ ?

- Intuizione meccanistica/biochimica
- Comportamento qualitativo (saturazione, veloce, lento)
- Tipo di processo (trasporto, reazione enzimatica, reazione concentrata, diffusione, ...)
- Regione di operazione (lineare, sempre saturata, ...)
- Reversibile, irreversibile

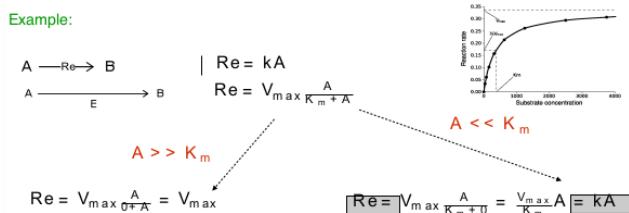
#### 31.4.2 Espressioni matematiche



La complessità di una reazione cinetica è definita da:

- lo scopo della reazione
- lo scopo del modello
- la conoscenza biochimica

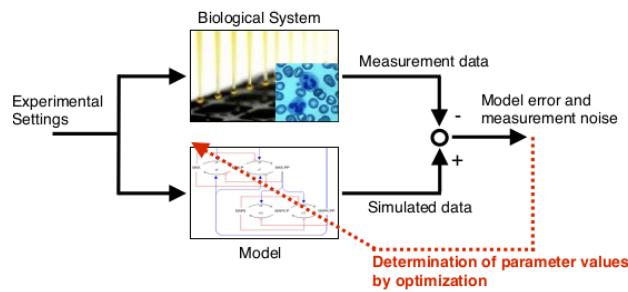
Condizioni di operazione del modello



#### 31.4.3 Determinazione dei parametri

Assegnare un valore ai parametri del modello. Devo definire i valori.

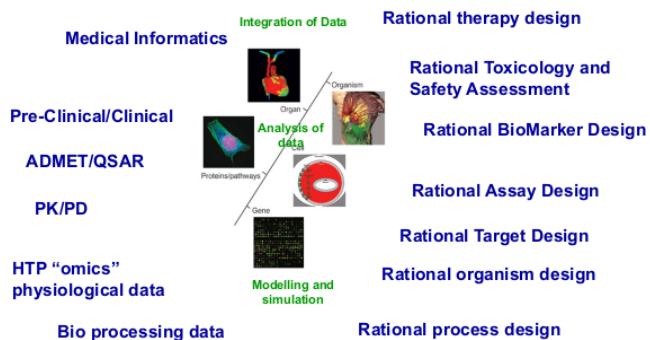
## Principio



## 32 Modelli standard nella Biologia dei sistemi

- Metabolismo  
Cinetica degli enzimi e termodinamica, controllo metabolico della rete
- Trasduzione del segnale  
Comunicazione intra-extracellulare, adattamento e regolazione dinamica
- Ciclo cellulare  
Oscillazioni glicolitiche, invecchiamento e processi relativi
- Espressione genetica  
Reti Bayesiane e Booleane, processi di regolazione

### Futuro della Biologia dei Sistemi



### 32.1 Concetti generali e proprietà

Nessun modello SB è unico, ma è sempre definito da i suoi:

- Stati:  
un'istantanea che contiene abbastanza informazioni da permettere predizioni del futuro  
Stazionario asintotico, oscillatorio, caotico, ...
- Parametri, variabili, costanti:  
costanti di tasso e di equilibrio, dipendenza/relazioni tra stati, ...
- Classificazione del processo:  
reversibilità, periodicità, framework deterministico o stocastico, variabili continue o discrete, ...

**Necessità di standard** I modelli SB hanno bisogno di essere condivisi e riutilizzati: abbiamo quindi bisogno di formati standard

- cellML  
Basato sui moduli, scalabile;

- Neuroml  
Flessibile (set espandibile di classi/schemi)
- BrainML  
I modelli sono schemi XML
- BioPAX  
No cinetica, semantica profonda, OWL;
- SBGN  
Rappresentazione semantica delle interazioni;
- SBML (Systems Biology Markup Language)  
Cinetica ricca, semantica debole, XML;

## 32.2 Lingaggio comune: SBML

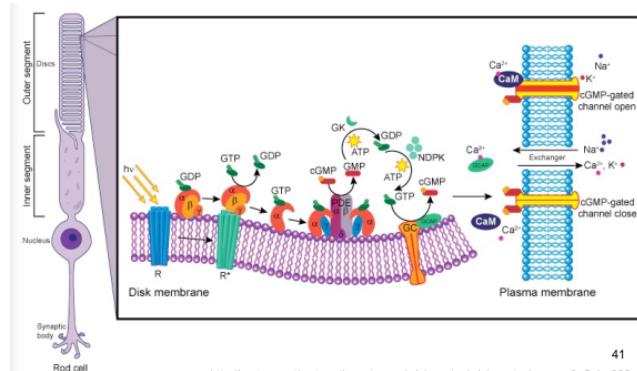
Il Systems Biology Markup Language (SBML), nato nell'anno 2000, è un linguaggio leggibile dalla macchina, derivato da XML, per rappresentare modelli di reti di reazioni biochimiche e:

- Consente l'uso di più strumenti software senza riscrivere i modelli per ogni strumento;
- Consente ai modelli di essere condivisi e pubblicati in una forma diversa, così che i ricercatori possono utilizzarli anche in un ambiente software diverso;
- Garantisce la sopravvivenza dei modelli oltre la durata del software usato per crearli

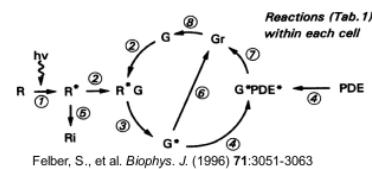
SBML è definito in livelli (specifiche compatibili con le versioni successive che aggiungono caratteristiche e potenza espressiva). I nuovi livelli non sostituiscono i vecchi livelli. Tuttavia, ogni livello può avere più versioni.

Al momento più di 100 sistemi software supportano SBML.

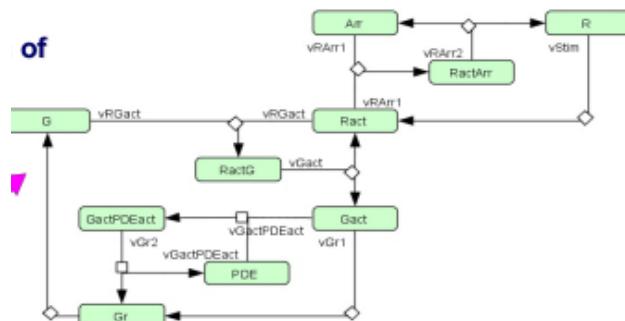
**Esempio:** Trasduzione nel ciclo della GTPasi (nella cascata della fototrasduzione)



41



Due possibili rappresentazione del modello di struttura



L'implementazione grafica di CellDesigner:  
mentazioni e simulazioni qui sono fatte con IntiQuan (IQM Tool Repository).

Tutte le imple-

Possiamo implementare questo sistema a 8 reazioni nell'IQMTools e testare la piattaforma sia con framework deterministico che stocastico.

## Implementazione

```
***** MODEL PARAMETERS
k1 = 100
k2 = 1
k3 = 7000
k4 = 0.3
k5 = 2.0
k6 = 0.05
k7 = 8
k8 = 2

***** MODEL NAME
Hofmann1

***** MODEL NOTES
Biophysical Journal Volume 71, 1996, 3051-3063

***** MODEL STATE INFORMATION
G(0) = 3000
Gact(0) = 0
GactPDEact(0) = 0
Gr(0) = 0
PDE(0) = 300
R(0) = 1
Ract(0) = 1
RactG(0) = 0
Ri(0) = 0

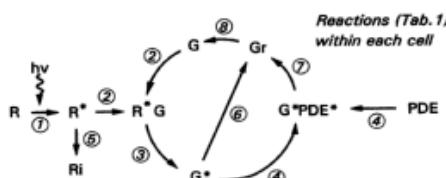
***** MODEL VARIABLES
R => Ract : R1
vf = k1*R
G+Ract => RactG : R2
vf = k2*Ract*G
RactG => Gact+Ract : R3
vf = k3*RactG
Gact+PDE => GactPDEact : R4
vf = k4*Gact*PDE

***** MODEL REACTIONS
R1 = k1*R
R2 = k2*Ract*G
R3 = k3*RactG
R4 = k4*Gact*PDE
R5 = k5*Ract
R6 = k6*Gact
R7 = k7*GactPDEact
R8 = k8*Gr
```

Una descrizione alternativa: equazioni ordinarie differenziali:

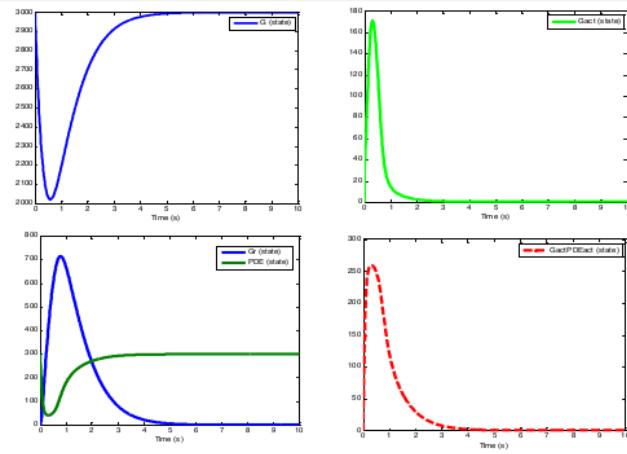
```
***** MODEL STATES
d/dt(G) = -R2+R8
d/dt(Gact) = +R3-R4-R6
d/dt(GactPDEact) = +R4-R7
d/dt(Gr) = +R6+R7-R8
d/dt(PDE) = -R4+R7
d/dt(R) = -R1
d/dt(Ract) = +R1-R2+R3-R5
d/dt(RactG) = +R2-R3
d/dt(Ri) = +R5
G(0) = 3000
Gact(0) = 0
GactPDEact(0) = 0
Gr(0) = 0
PDE(0) = 300
R(0) = 1
Ract(0) = 1
RactG(0) = 0
Ri(0) = 0

***** MODEL REACTIONS
R1 = k1*R
R2 = k2*Ract*G
R3 = k3*RactG
R4 = k4*Gact*PDE
R5 = k5*Ract
R6 = k6*Gact
R7 = k7*GactPDEact
R8 = k8*Gr
```



Le due rappresentazioni sono equivalenti

## Simulazione deterministica del modello



### 32.3 Framework deterministico: è realistico?

In termini generali:

- $N$  specie chimiche  $S_1, S_2, \dots, S_N$  e  $M$  possibili reazioni  $R_1, R_2, \dots, R_M$
- Il sistema ha una costante di volume  $\Omega$  e temperatura  $T$ , ed è ben mescolato (spazialmente omogeneo)
- $X_i$  è il numero di molecole di  $S_i$

$$X = (X_1(t), X_2(t), \dots, X_N(t))$$

è lo STATO del sistema al tempo  $t$ .

Il problema è: dato  $X(t_0)$  trova  $X(t)$ .

L'approccio tradizionale è scrivere un insieme di primo ordine accoppiato equazioni differenziali ordinarie, dove  $f_i$  dipende da ciascuna reazione cinetica:

$$\frac{dX_i}{dt} = f_i(X_1, \dots, X_N) \\ i = 1, \dots, N$$

Queste sono chiamate equazioni della velocità di reazione (RRE) e di solito lo sono scritte in termini di concentrazioni  $C_i = X_i/\Omega$  quindi  $X(t)$  è apparentemente un processo continuo e deterministico.

### Ma nei fatti ...

- $X(t)$  non è continuo; è discreto
- Le molecole si presentano in numero intero e le popolazioni molecolari cambiano solo per importi interi
- $X(t)$  non è deterministico; è stocastico
- Le reazioni chimiche si verificano come eventi discreti, come risultato di collisioni molecolari che non possono essere previste con precisione

Nel migliore dei casi, possiamo predire solo la *probabilità* che una reazione avvenga.

### Dalla cinetica chimica deterministica a quella STOCASTICA

**Framework cinetico chimico stocastico** Ogni relazione elementare  $R_j$  è definita da due quantità:

- una funzione di propensione  $a_j(x)$ , dove  $a_j(x)dt$  fornisce la probabilità, dato  $X(t) = x$ , che una reazione  $R_j$  occorrerà in  $[t, t + dt]$
- un vettore di cambio di stato  $v_j = (v_{1j}, \dots, v_{Nj})$  dove  $v_{ij}$  è il cambiamento nella popolazione  $S_j$  causato da una reazione  $R_j$
- Implicazione:  $X(t)$  è un **jump Markov process** (un processo stocastico, a tempo continuo, a stati discreti e che dimentica il passato)

Per un sistema ben miscelato possiamo ottenere una soluzione analitica esatta della funzione di densità di probabilità.

### 32.4 Equazione chimica principale

La funzione di densità di probabilità della variabile aleatoria  $X(t)$  è definita come:

$$P(x, t|x_0, t_0) = \text{Prob}\{X(t) = x, \text{ dato } X(t_0) = x_0\}$$

Si può dimostrare che l'evoluzione temporale di  $P$  segue la seguente EQUAZIONE MASTER CHIMICA:

$$\frac{\partial P(x, t|x_0, t_0)}{\partial t} = \sum_{j=1}^M [a_j(x - v_j)P(x - v_j, t|x_0, t_0) - a_j(x)P(x, t|x_0, t_0)]$$

Determina completamente  $X(t)$  ed è impossibile risolvere analiticamente tranne per quelli più semplici. Con  $M$  indichiamo tutte le reazioni possibili.

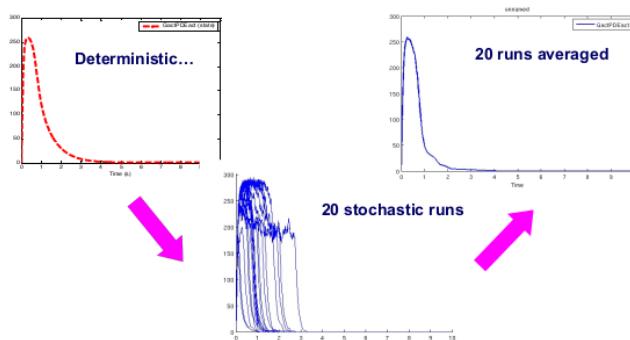
### 32.5 Quando stocastica e quando deterministica?

Per i sistemi chimici più pratici la popolazione molecolare è molto grande, e la RRE fenomenologica:

$$\frac{dX(t)}{dt} = \sum_{j=1}^M [v_j a_j(X(t))]$$

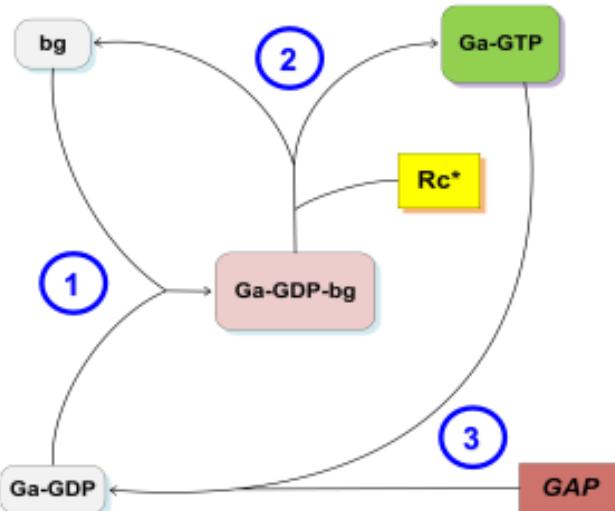
è accurata e veloce. In piccoli sistemi biochimici però, in cui sono presenti specie reagenti critiche in numeri bassi (ad esempio nell'espressione genica), il sistema è intrinsecamente stocastico e deterministico RRE può essere impreciso e fuorviante. Per tali sistemi, dobbiamo invece usare il CME o altre approssimazioni.

#### Comparazione di approcci deterministici e stocastici



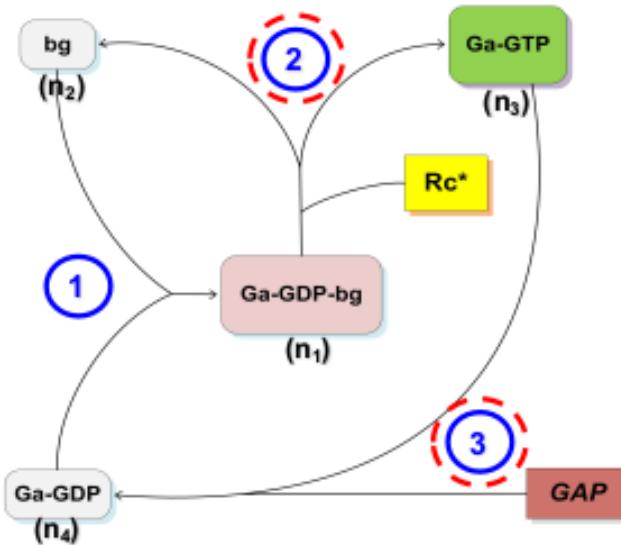
#### 32.5.1 Un altro sistema semplice

Focus sul ciclo di segnalazione della proteina G



Quattro stati tempo-dipendenti  
Due quantità indipendenti (Rc\* e GAP considerate costanti)

$$\begin{cases} \frac{d[G\alpha^{GDP}\beta\gamma]}{dt} = \frac{dn_1}{dt} = V_1 - V_2 \\ \frac{d[\beta\gamma]}{dt} = \frac{dn_2}{dt} = V_2 - V_1 \\ \frac{[G\alpha^{GTP}]}{dt} = \frac{dn_3}{dt} = V_2 - V_3 \\ \frac{[G\alpha^{GDP}]}{dt} = \frac{dn_4}{dt} = V_3 - V_1 \end{cases}$$



Una possibile descrizione cinetica delle reazioni biochimiche:

$$V_1 = k_{ass}[G\alpha^{GDP}][\beta\gamma] = k_1 n_4 n_2$$

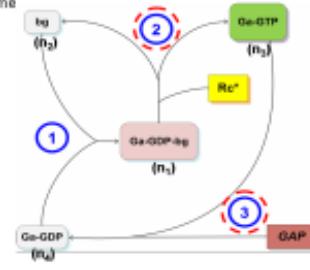
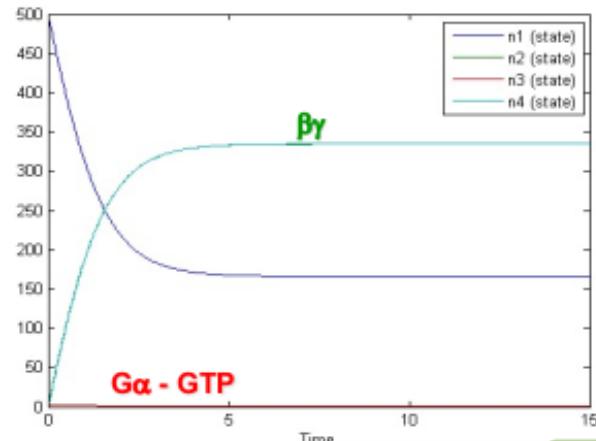
$$V_2 = \frac{k_{diss}[G\alpha^{GDP}\beta\gamma][Rc^*]}{K_2 + [G\alpha^{GDP}\beta\gamma]} = k_2 \frac{n_1}{K_2 + n_1}$$

$$V_3 = \frac{k_{hydr}[G\alpha^{GTP}][GAP]}{K_3 + [G\alpha^{GTP}]} = k_3 \frac{n_3}{K_3 + n_3}$$

Simulazione per 15 secondi

```
***** MODEL PARAMETERS
K2 = 500
K3 = 2
kass = 0.001
kdiss = 15
khydr = 15
RCact = 30
GAP = 25
k2 = 450
k3= 375
k1 = 0.001

***** MODEL STATES
n1(0) = 500
n2(0) = 1
n3(0) = 1
n4(0) = 1
```



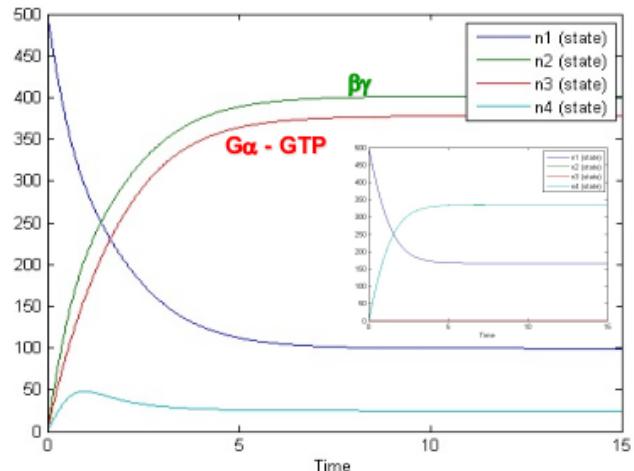
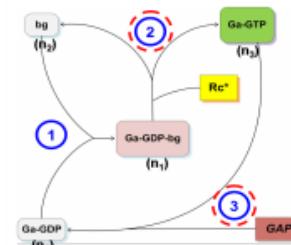
## Cambiamo alcuni parametri

### \*\*\*\*\* MODEL PARAMETERS

```
K2 = 500
K3 = 2
kass = 0.01
kdiss = 20
khydr = 5
RCact = 30
GAP = 20
k2 = 450
k3= 375
k1 = 0.001
```

### \*\*\*\*\* MODEL STATES

```
n1(0) = 500
n2(0) = 1
n3(0) = 1
n4(0) = 1
```



Nessuna delle condizioni iniziali cambia: solo il numero di GAP!

Elevate quantità allo stato stazionario di  $G\alpha - GTP$  e  $\beta\gamma$ : questa è l'immagine cinetica classica del ciclo della proteina G.

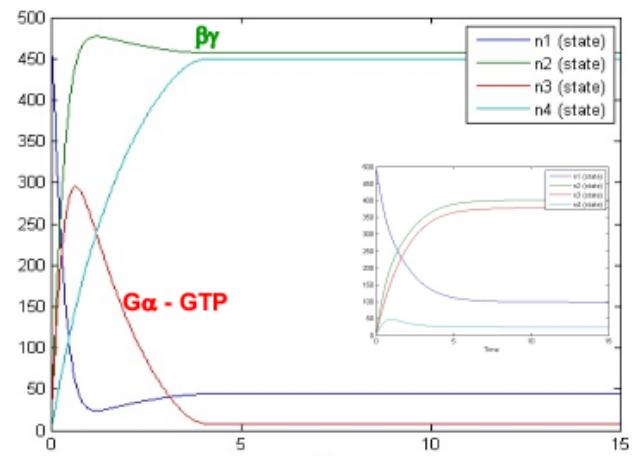
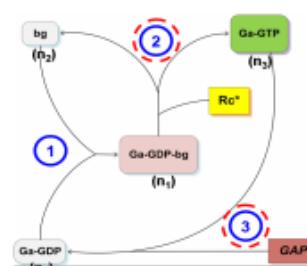
## Cambiamo altri parametri ...

### \*\*\*\*\* MODEL PARAMETERS

```
K2 = 500
K3 = 2
kass = 0.001
kdiss = 25
khydr = 5
RCact = 100
GAP = 52
```

### \*\*\*\*\* MODEL STATES

```
n1(0) = 500
n2(0) = 1
n3(0) = 1
n4(0) = 1
```



Ora

$G\alpha - GTP$  mostra un comportamento transitorio, mentre  $\beta\gamma$  raggiunge un importo stazionario elevato: l'immagine cinetica del ciclo della proteina G è completamente cambiato!

- Include un feedback negativo per l'internazionalizzazione del recettore  $G\alpha - GTP$ -indotto.

Il recettore  $Rc$  potrebbe essere rimosso dalla superficie cellulare. Quindi,  $[Rc^{act}]$  non è più costante e potremmo pensare di cambiarlo come:

$$\frac{d[Rc^{act}]}{dt} = V_{del} - V_{rem} - k_{rem}[Rc^{act}] \frac{1 + (An_3/K_A)}{1 + (n_3/K_4)}$$

$V_{del}$  = velocità di consegna del recettore alla superficie cellulare

$k_{rem}$  = costante di velocità per la rimozione del recettore

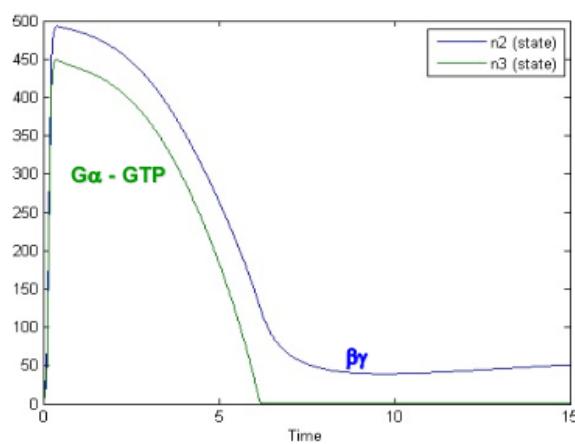
$A, K_A$  = costanti cinetiche

- Include un feedback positivo per l'attività GPCR  $G\alpha - GTP$ -migliorata

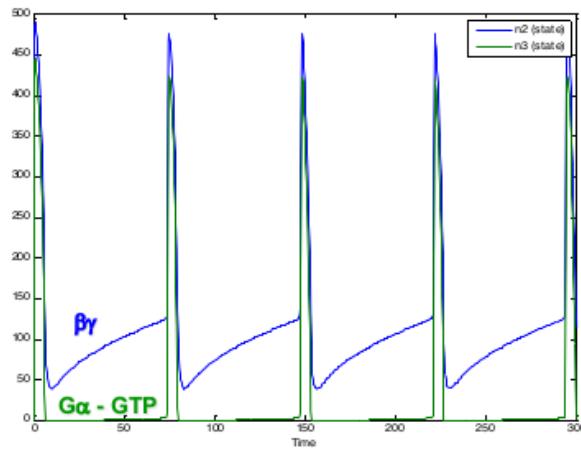
$$V_2 = k_2 \frac{n_1}{K_2 + n_1} \frac{1 + (Bn_3/K_B)}{1 + (n_3/K_B)}$$

$B, K_B$  = costanti cinetiche

Simulando per 15 secondi:



Se provo a simulare per un tempo più lungo (300 secondi)



L'aggiunta di anse regolatrici comporta oscillazioni nel concentrazione dei componenti del ciclo delle proteine G.

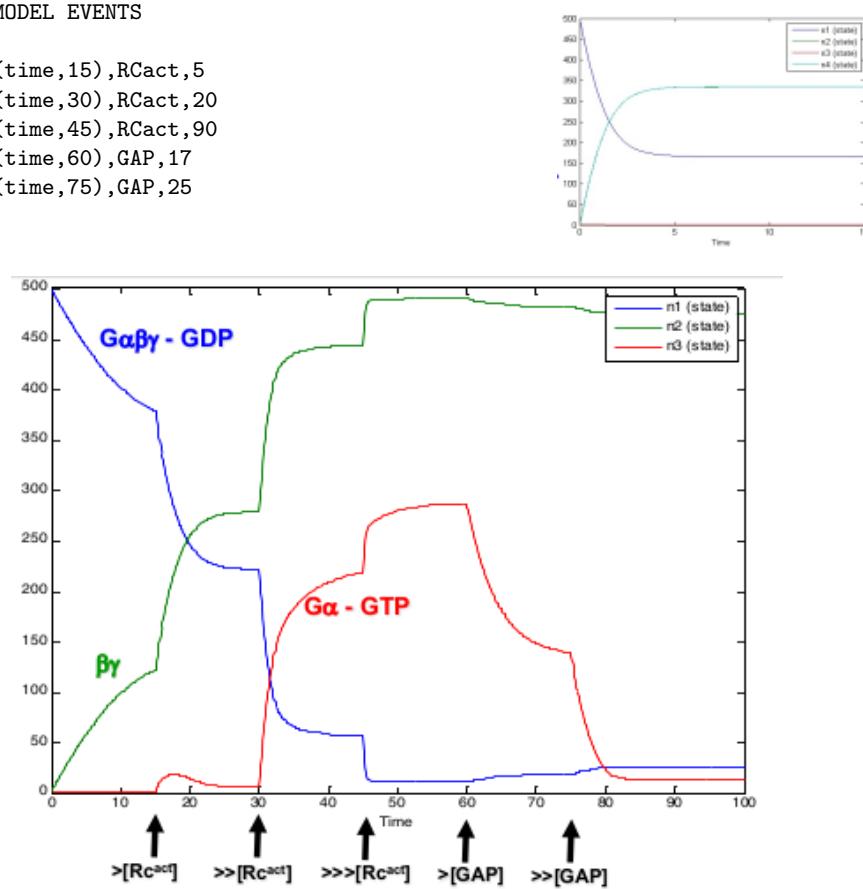
**Potremmo anche aggiungere alcuni eventi che facciano sì che il sistema salti tra modi differenti**  
Il sistema di segnalazione della proteina G trimerica può passare da una modalità a un'altra e tornare ai cambiamenti iniziali delle condizioni.

Il sistema potrebbe essere stimolato con diverse quantità di GPCR e/o GAP in tempo.

Stessa semplice cinetica della prima modalità analizzata + nuovi EVENTI che si verificano nel tempo:

\*\*\*\*\* MODEL EVENTS

```
event1 = gt(time,15),RCact,5
event2 = gt(time,30),RCact,20
event3 = gt(time,45),RCact,90
event4 = gt(time,60),GAP,17
event5 = gt(time,75),GAP,25
```



### Un esempio di modello SBML

```
<?xml version="1.0" encoding="UTF-8"?>
<smlb xmlns="http://www.sbml.org/sbml/level2" level="2" version="1">
  <model name="G_protein_Cycle_dd">
    <notes>
      <html xmlns="http://www.w3.org/1999/xhtml">
        <body>Excercise form the paper by Katanec and Chornomorets
          (Biochem J.2007)</body>
      </html>
    </notes>
    ...
  </model>
</smlb>
```