UNIVERSITÀ DEGLI STUDI DI BOLOGNA

Facoltà di Scienze Matematiche, Fisiche e Naturali

Dottorato di Ricerca in Biotecnologie Cellulari e Molecolari

XIX CICLO

METODI BIOINFORMATICI PER LA CARATTERIZZAZIONE DELL'INSTABILITÀ PROTEICA, SNPs E MALATTIE

SETTORE SCIENTIFICO DISCIPLINARE: BIO10

Tesi di:
Remo Calabrese

Relatore:
Chiar.ma Prof. Rita Casadio

Coordinatore:
Chiar.mo Prof. Lanfranco Masotti

Anno Accademico: 2005-2006

Capitolo 4

Machine Learning e Support Vector Machines

Capitolo 5

Nuova metodologia per la predizione della variazione di stabilità dei mutanti delle proteine

Capitolo 6

Nuova metodologia per la predizione dell'insorgenza di malattie genetiche umane dovute a mutazioni proteiche puntiformi

Capitolo 7

Capitolo 1

Il materiale genetico: Geni e Mutazioni

1.1 I Geni

La vita dipende dalla capacità delle cellule di immagazzinare e tradurre le istruzioni genetiche necessarie alla costruzione ed al perpetuarsi degli organismi. Questa informazione ereditaria viene trasmessa dalla cellula madre alle cellule figlie e, per un organismo, da una generazione alla successiva attraverso la sua linea germinale riproduttiva. Gli elementi che contengono tali informazioni e che determinano le caratteristiche di una certa specie, e degli invidi al suo interno, sono i ***geni***. L'informazione genetica è contenuta in una sequenza lineare di nucleotidi, il DNA. Ogni molecola di DNA è una doppia elica formata da due filamenti complementari di nucleotidi tenuti insieme da legami ad idrogeno tra le coppie di basi azotate (Adenina, Timina, Guanina e Citosina) A-T e G-C (vedi fig 1.1).

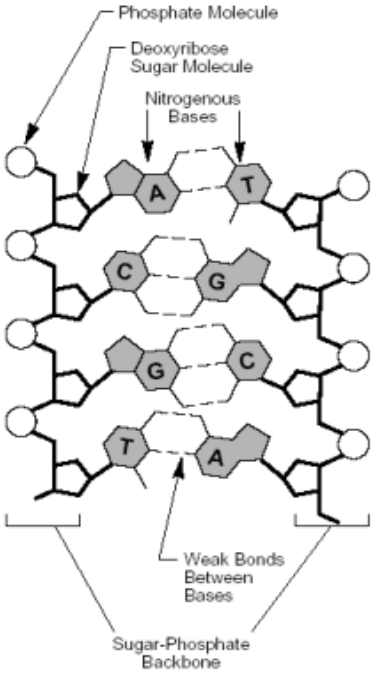


Figura 1.1 Schema dell'appaiamento dei filamenti di DNA in cui si mettono in evidenza i ponti ad idrogeno tra le coppie di basi A-T (due legami idrogeno), G-C (tre legami idrogeno) e lo scheletro esterno formato dai gruppi fosfato e dalle molecole di zucchero (deossiribosio). Le basi puriniche sono Adenina e Guanina, mentre Timina e Citosina sono le basi pirimidiniche (Uracile nell'RNA). http://web.mit.edu/esgbio/www/lm/nucleicacids/dna_hbonds.gif

Pertanto un gene non è altro che l'unità funzionale dell'ereditarietà che corrisponde ad una sequenza di DNA che codifica per una proteina (o per gli RNA-ribosomiali e RNA-transfer). I geni si trovano in particolari siti cromosomici detti ***locus*** ed assumono due forme alternative, dette ***alleli.*** In altre parole l'allele è responsabile della particolare modalità con cui si manifesta un carattere ereditario controllato da un dato gene. Ogni organismo diploide, possiede per ciascun carattere due alleli, cioè due copie; ognuno dei due alleli è presente su uno stesso locus, su ciascuno dei due cromosomi che costituiscono, nella cellula, una coppia di omologhi. I cromosomi sono strutture altamente compatte in cui si organizza la ***cromatina*** (costituita da DNA avvolto sulle proteine istoniche,formando il nucleosoma, e proteine non-istoniche) durante le ultime fasi della mitosi (o della meiosi).

Il numero dei cromosomi umani è pari a 23 (22 coppie di omologhi, ***autosomi***, ed una coppia di cromosomi diversi che determina il sesso).Se sui cromosomi omologhi è presente una duplice copia dello stesso allele, si dice che l'individuo è omozigote per quel carattere; se gli alleli sono differenti, l'individuo è detto eterozigote.

Ogni carattere, all'interno di una popolazione, può essere rappresentato anche da molti alleli, in questo caso si parla di allelia multipla (sebbene ogni individuo ne possa portare solo due). L'insieme degli alleli presenti in una popolazione è detto "*pool*" genico. Non tutti gli alleli determinano un effetto visibile nell'individuo che ne è portatore. Se il carattere da essi controllato si manifesta, si parla di alleli dominanti; in caso contrario si parla di alleli recessivi. Un individuo può essere quindi omozigote dominante, se possiede due alleli dominanti; eterozigote, se possiede due alleli differenti; omozigote recessivo, se possiede entrambi gli alleli recessivi (vedi fig 1.2). Un allele dominante sarà espresso sempre, anche se l'individuo è eterozigote, un allele recessivo potrà essere espresso solo in individui omozigoti recessivi. Esistono anche fenomeni di dominanza incompleta, in cui il fenotipo di un individuo avente un allele recessivo ed uno a dominanza incompleta sarà una via di mezzo tra i due; e fenomeni di codominanza, in cui entrambi gli alleli presenti nel genotipo sono dominanti. Il ***genotipo*** non è niente altro che lo specifico set di alleli che formano il genoma di un individuo,mentre il fenotipo è la manifestazione visibile dei caratteri genetici dell'individuo.

Si parla di geni (o fenotipi) selvatici ("wild-type") quando abbiamo a che fare con i geni (o fenotipi) che normalmente si trovano nelle popolazioni naturali, mentre parliamo di geni (o fenotipi) mutanti quando abbiamo a che fare con elementi che differiscono da quelli "wild-type" a causa di eventi mutazionali che ne hanno cambiato la sequenza genetica.

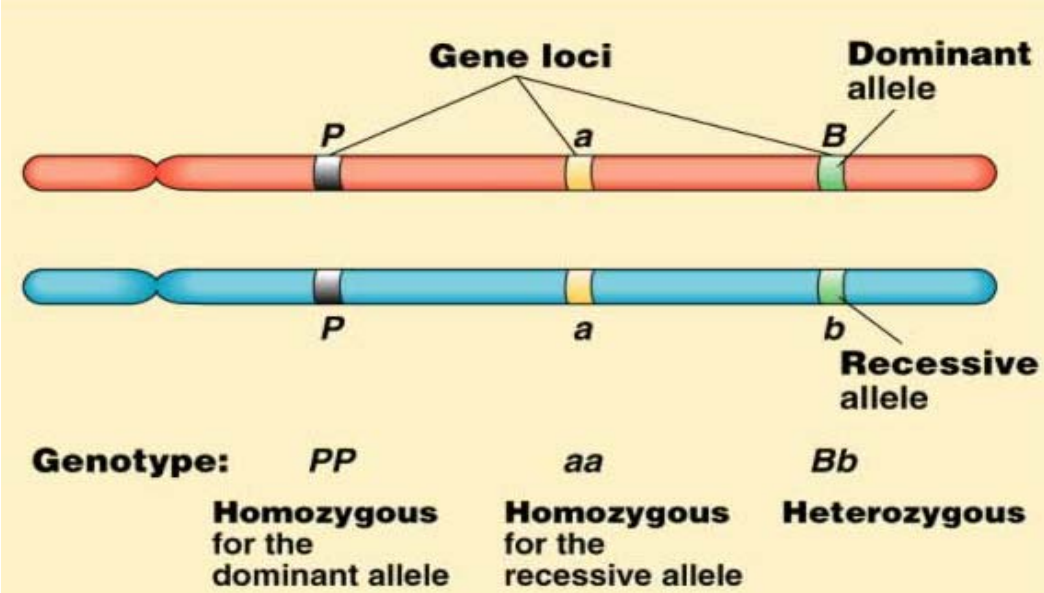


Figura 1.2 Schema dell'organizzazione dei locus genici sui cromosomi omologhi. Gli alleli "P" e "B" hanno carattere dominante, mentre gli alleli "a" e "b" hanno carattere recessivo. Il genotipo dei vari locus è Omozigote Dominante ("PP") per il gene "P" , Omozigote recessivo ("aa") per il gene "a" ed Eterozigote ("Bb") per il gene "B". www.anselm.edu/homepage/jpitocch/genbio/locus.JPG

1.2 Le Mutazioni

L'informazione genetica contenuta nel DNA deve soddisfare due aspetti tra loro antitetici, da una parte deve essere stabile limitando gli eventi che ne perturbano l'integrità, dall'altra deve poter evolvere per permettere la sopravvivenza dell'organismo al mutare delle condizioni ambientali selezionando gli individui che hanno le migliori capacità di adattamento. Il DNA sia durante il processo di duplicazione che in altri momenti del ciclo cellulare può subire dei danni o andare incontro ad errori di replicazione che ne alterano la sequenza causando quelle che si chiamano comunemente ***mutazioni***. Quindi quando una cellula si divide, le due cellule figlie non sono perfettamente identiche né tra di loro né con la cellula madre. In alcuni casi questi errori apportano dei vantaggi, in altri casi non hanno nessun effetto significativo, oppure in molti casi questi errori causano seri danni come ad esempio la distruzione di una sequenza codificante per una proteina importante per la funzione cellulare. Cambiamenti dovuti ad errori del primo genere verranno perpetuati, fissati nel genoma, in quanto le cellule alterate hanno una maggiore probabilità di riprodursi e quindi ne traggono un vantaggio evolutivo, questo fenomeno si chiama *selezione positiva*. Cambiamenti dovuti ad errori del secondo tipo possono essere perpetuati o meno date, ad esempio, particolari condizioni ambientali, in quanto non apportano alcun vantaggio significativo e sono controllate da un tipo di *selezione neutra*. Cambiamenti dovuti ad errori che causano seri danni alla cellula non verranno perpetuati attraverso le generazioni successive e verranno eliminate, cioè si attuerà un tipo di *selezione negativa*. Attraverso questo ciclo infinito di errori e tentativi, di mutazioni e selezione naturale, gli organismi si evolvono.

Le loro specifiche genetiche cambiano dando loro nuovi modi di sfruttare in maniera più efficiente le risorse ambientali e di riprodursi con successo essendo in competizione con altri organismi. Ovviamente alcune parti del genoma cambiano più facilmente rispetto ad altre lungo il corso dell'evoluzione. Ad esempio una sequenza di DNA che non codifica per alcuna proteina e non ha alcuna funzione regolatoria specifica, può accumulare mutazioni ad un tasso elevato, limitato solo dalla frequenza degli errori casuali. Al contrario un gene che codifica per una proteina o per una molecole di RNA (ribosomiale o transfer) non può subire cambiamenti così facilmente e quando avvengono delle mutazioni dannose le cellule alterate vengono eliminate.

In generale il materiale grezzo dell'evoluzione sono le sequenze di DNA già esistenti, in questo senso nessun gene è completamente nuovo, tuttavia le innovazioni possono avvenire tramite:

i) mutazioni intrageniche: un gene esistente può essere modificato dalle mutazioni nella sua sequenza di DNA

ii) duplicazione genica: un gene esistente può essere duplicato in modo da creare una coppia di geni altamente correlati all'interno della cellula

iii) riarrangiameto genico: due o più geni esistenti possono essere spezzati e riuniti in modo da creare un gene ibrido costituito da segmenti originariamente appartenenti a geni diversi

iv) trasferimento orizzontale: un pezzo di DNA può essere trasferito da un genoma di una cellula a quello di un'altra o di un altro organismo. Questo processo è in contrasto con quello usale del trasferimento verticale dell'informazione genetica da un genitore alla progenie.

Un aspetto particolarmente interessante del fenomeno della duplicazione genica è quello che una delle due copie del gene può accumulare mutazioni e diventare specializzato nell'adempiere una nuova funzione. Numerosi cicli di questo processo di duplicazione e divergenza, attraverso milioni di anni, hanno fatto in modo che da un gene si potesse originare una famiglia genica e che quindi gli individui di una stessa specie possano essere provvisti di molteplici varianti del gene primordiale. Geni correlati che sono il risultato di eventi di duplicazione, e che probabilmente hanno funzioni divergenti, all'interno di uno stesso genoma si chiamano ***Paraloghi***. Geni correlati, che derivano dallo stesso gene ancestrale dell'ultimo antenato comune e che appartengono a specie diverse, si chiamano ***Ortologhi***.

Esistono diverse tipologie di mutazioni che possono alterare l'integrità del materiale genetico:

i) Delezioni: in cui segmenti di DNA di un cromosoma vengono deleti e quindi persi

ii) Traslocazioni: in cui porzioni di un cromosoma vengono rotte e unite su un cromosoma diverso dal precedente

iii) Inversioni: in cui un segmento di DNA di un cromosoma viene invertito

iv) Mutazioni Puntiformi: in cui si ha la mutazione di una singola coppia di nucleotidi o una piccola porzione di un gene nel genoma.

Ulteriormente le mutazioni posso essere classificate da un punto di vista fenotipico/funzionale in:

i) Mutanti letali: provocano la morte prematura di un organismo in via di sviluppo

ii) Mutanti condizionali: manifestano l'effetto fenotipico solo sotto determinate condizioni, dette restrittive, mentre sotto altre condizioni, dette permissive, gli effetti non sono manifesti. Si pensi a mutazioni sensibili alla temperatura in cui le condizioni restrittive sono tipicamente rappresentate dalle alte temperature ,mentre le condizioni permissive sono rappresentate dalle basse temperature

iii) Mutanti con perdita di funzione: in cui viene ridotta o abolita la funzionalità del gene. Sono la classe più comune di mutazioni, sono di solito di carattere recessivo e pertanto di solito le funzionalità dell'organismo sono normali fino a quando lo stesso riesce a mantiene almeno una copia funzionante del gene alterato

iv) Mutanti nulli: mutanti con perdita di funzione che aboliscono completamente l'attività del gene

v) Mutanti con guadagno di funzione: in cui l'attività del gene viene aumentata o viene resa attiva in circostanze opportune, di solito queste mutazioni sono di tipo dominante

vi) Mutanti dominanti negativi: mutazioni dominanti che bloccano l'attività genica causando un fenotipo con perdita di funzione anche in presenza di una copia normale e funzionante del gene. Questo fenomeno avviene quando il prodotto del gene mutato interferisce con la funzione del prodotto del gene normale.

vii) Mutanti soppressori: sopprimono l'effetto fenotipico di un'altra mutazione in modo tale che i doppio mutante sembra normale. Una mutazione soppressore intragenica si trova sullo stesso gene alterato dalla prima mutazione; una mutazione soppressore extragenica si trova su un secondo gene il cui prodotto di solito interagisce direttamente col prodotto del primo

1.2.1 I Polimorfismi di Singolo Nucleotide

La classe di mutazioni che interessano questo lavoro di tesi solo le mutazioni proteiche puntiformi derivanti da polimorfismi del DNA che coinvolgono il cambiamento di singolo nucleotide (Single Nucleotide Polymorphism) e sono la classe di mutazioni più abbondante nel genoma umano. Per polimorfismo si intende che in una popolazione esiste più di un allele per un dato locus genico con frequenza superiore all'1% (vedi fig. 1.3).

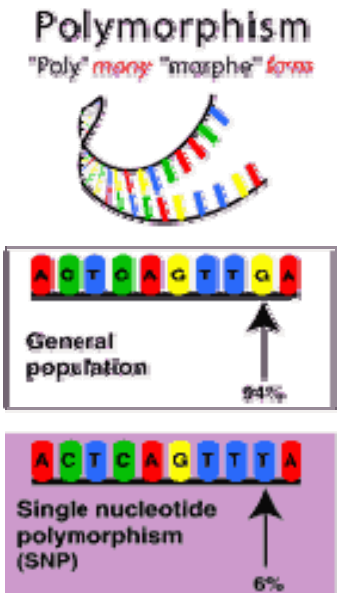


Figura 1.3 Esempio di Polimorfismo di Singolo Nucleotide in frammento di DNA. Si vede come nella popolazione generale nel 94% dei casi abbiamo una "G" mentre nel restante 6% troviamo una "T". Questo è un esempio di **trasversione,** cioè una purina ("G") viene mutata in pirimidina ("T"). Si parla di **transizione** quando una purina viene mutata in un'altra purina, ad esempio "A" → "G" , oppure una pirimidina viene cambiata in pirimidina. ("T" → "C"). http://www.laskerfoundation.org/rprimers/gnn/wagn/gvariations.html

Gli SNPs pertanto sono un tipo di variazione che si presenta tra individui della stessa specie in cui la differenza nella sequenza di DNA è a carico di una singolo nucleotide e possono presentarsi ovunque, sia in sequenze intergeniche che all'interno della struttura di un gene, ovvero li possiamo trovare nelle regioni: codificanti, intronic, promotrici e non tradotte (vedi fig 1.4).

Un primo tipo di classificazione riguarda proprio la loro presenza in regioni codificanti o meno. Dal momento che solomente una piccola percentuale del genoma umano codifica per proteine (un valore che si attesta attorno al 5% del genoma), la maggior parte degli SNPs si trova nelle regioni non codificanti ("non-coding SNPs"), tuttavia quelli che si trovano nelle regioni effettivamente espresse ("coding SNPs") sono di particolare interesse in quanto hanno una maggiore probabilità di alterare la funzione biologica di una proteina.

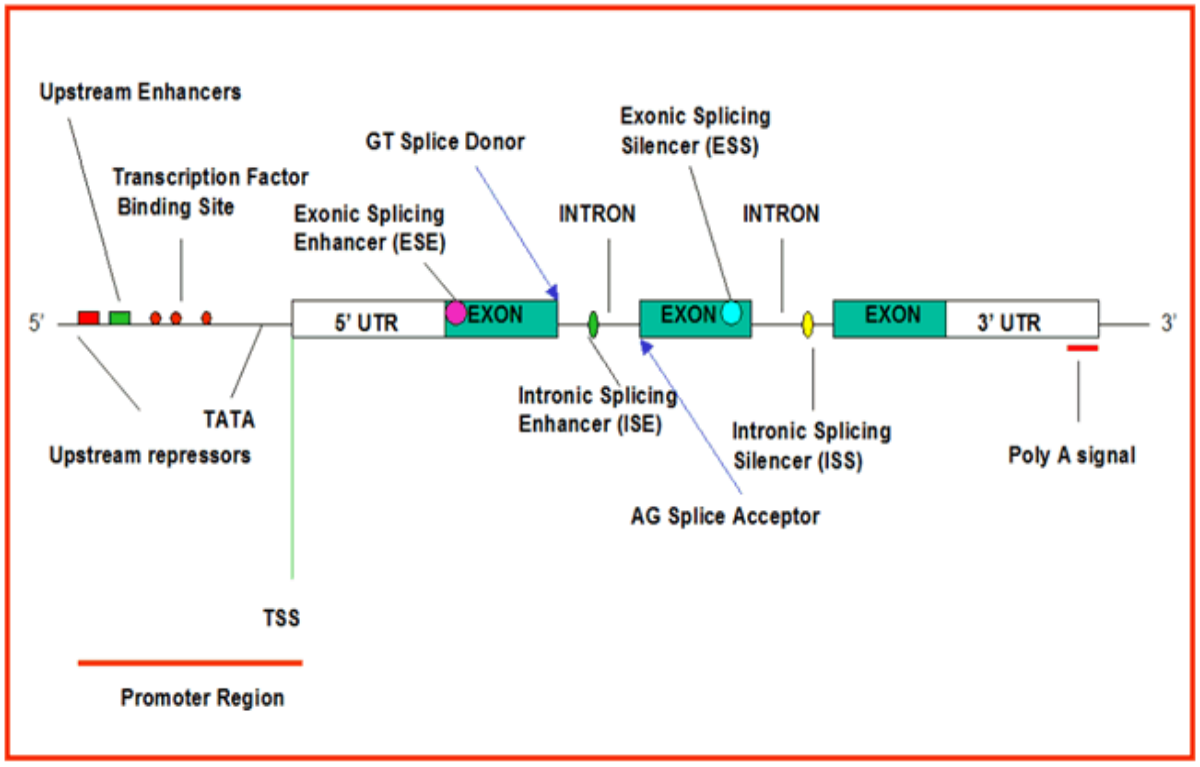


Figura 1.4 Descrizione dei vari elementi strutturali di un gene e delle possibili localizzazioni di uno SNP. In fatti gli SNPs si possono trovare sia nelle regioni codificanti (cerchietto azzurro e viola), che in quelle non codificanti (cerchietto giallo e verde) e posso essere sia sinonimi che non-sinonimi. http://www.ncbi.nlm.nih.gov/

Gli SNPs all'interno di un gene, in ogni caso, non necessariamente modificano la sequenza amminoacidica codificata, dal momento che il codice genetico è degenerato. Uno SNP che genera in tutte le sue forme lo stesso peptide è detto sinonimo ("synonymous coding SNP"); in caso contrario è detto non-sinonimo ("non-synonymous coding SNP"). Gli SNPs che non si trovano in una sequenza codificante possono, in ogni caso, avere degli effetti negativi sullo splicing o sul legame dei fattori di trascrizione o , in definitiva, avere un relazione con la predisposizione alle malattie genetiche e con la risposta ai trattamenti farmacologici.

Capitolo 2

Le Proteine: Struttura, Interazioni, Folding e Stabilità

2.1 Le proteine dal punto di vista chimico-fisico

Le proteine costituiscono una classe di macromolecole organiche di importanza fondamentale per i processi biologici, essendo parte integrante della struttura e della funzione cellulare. Le proteine sono dei polimeri costituiti da molecole organiche più semplici, gli amminoacidi, così chiamati perché contenenti nella loro struttura una base amminica ($-NH_3^+$) e un gruppo carbossilico ($-COO^-$) (vedi Fig. 2.1). Gli amminoacidi di una proteina sono uniti tra loro da legami covalenti, chiamati legami peptidici. (vedi Fig. 2.2).

Figura 2.1 Struttura generale di un L amminoacido

Il numero di residui di una proteina è una proprietà molto

variabile: infatti si possono avere polimeri composti da una cinquantina di amminoacidi ed altri più complessi che possono raggiungere oltre le 2000 unità. A questa diversità di tipo quantitativo si aggiunge quella qualitativa data dalla varietà di composizione delle proteine dati i 20 amminoacidi di base (tabella 2.1).

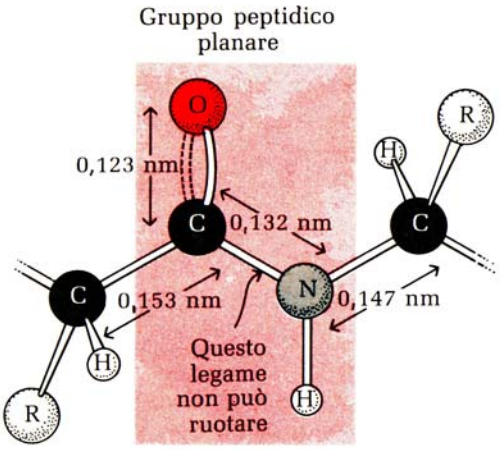


Figura 2.2 Legame peptidico che si forma tra un gruppo -COO- e un gruppo -NH3+

Le forze di interazione (di tipo covalente) tra gli amminoacidi determinano la struttura complessiva della proteina, detta anche “backbone”. Tuttavia gran parte delle forze stabilizzanti la struttura proteica, e che permettono le interazioni tra diverse catene polipetidiche, sono di tipo non covalente (o di non legame) e vengono generalmente indicate come forze di legame debole, in quanto richiedono una quantità di energia molto inferiore rispetto al legame di tipo covalente per essere distrutte.

Le principali forze di non legame sono: legami ionici, ponti idrogeno, forze di van der Waals ed interazioni di tipo idrofobico. I legami ionici sono forze di attrazione di tipo elettrostatico tra atomi aventi cariche opposte; queste forze sono abbastanza forti in assenza di acqua (circa 80 kcal/mole), tuttavia le molecole polari dell'acqua formano dei cluster sia attorno alle cariche ioniche sia alle molecole polari che hanno un dipolo permanente in modo tale da ridurre enormemente la potenziale attrattività delle specie cariche determinando una forza di interazione dell'ordine delle 3 kcal/mole.

I legami ad idrogeno (anche detti ponti ad idrogeno) sono una forma particolare di interazione polare in cui un atomo di idrogeno (elettropositivo) è parzialmente condiviso tra due atomi elettronegativi. L'atomo di idrogeno può essere visto come un protone che è parzialmente dissociato da un atomo donatore permettendogli di essere condiviso da un atomo accettore; a differenza di una tipica interazione elettrostatica, questo legame è altamente direzionale, essendo più forte quando i tre atomi coinvolti si trovano lungo uno stesso piano (circa 4 kcal/mole). Come detto sopra, per i legami ionici, le molecole di acqua indeboliscono questi tipi di legame formando delle interazioni che competono con gli atomi direttamente coinvolti nel ponte ad idrogeno riducendone la forza di interazione che in questo modo si attesta sull'ordine di 1 kcal/mole.

Nelle forze di attrazione di van der Waals la nuvola elettronica attorno ogni atomo non polare fluttua, producendo un momentaneo dipolo indotto. Tali dipoli indurranno in maniera transiente un opposto dipolo polarizzato in un atomo vicino. Questa interazione genera un'attrazione atomica molto debole (circa 0.1 kcal/mole), ma dal momento che molti atomi posso contemporaneamente essere in contatto quando due superfici sono molto vicine, allora il risultato netto è spesso significativo.

Queste forze di attrazione di van der Waals non sono indebolite dalla presenza dell'acqua.

Il quarto effetto che gioca un ruolo importante nella stabilità di una proteina è la forza di interazione idrofobica. Tale forza agisce in modo da non permettere l'esposizione al solvente polare di elementi non polari. Infatti le molecole non polari interferiscono negativamente con la rete di interazioni di tipo ad idrogeno, molto favorevoli, che si stabiliscono tra le molecole di acqua del solvente. Dal momento che raggruppare vicino due superfici non polari riduce il loro contatto con l'acqua, questo tipo di forza è di natura non specifica, tuttavia è una forza di primaria importanza per il corretto ripiegamento delle proteine.

Aminoacidi	*Abbreviazioni a tre lettere*	*Simbolo a una lettera*
Acido aspartico	Asp	D
Acido glutammico	Glu	E
Alanina	Ala	A
Arginina	Arg	R
Asparagina	Asn	N
Cisteina	Cys	C
Fenilalanina	Phe	F
Glicina	Gly	G
Glutammina	Gln	Q
Isoleucina	Ile	I
Istidina	His	H
Leucina	Leu	L
Lisina	Lys	K
Metionina	Met	M
Prolina	Pro	P
Serina	Ser	S
Tirosina	Tir	Y
Treonina	Thr	T
Triptofano	Trp	W
Valina	Val	V

Tabella 2.1 Nomi ed abbreviazioni dei 20 amminoacidi essenziali.

2.2 Struttura spaziale delle proteine

La struttura delle proteine è organizzata secondo una gerarchia di livelli che riguardano caratteristiche spaziali sempre più estese. In particolare si distinguono quattro livelli di organizzazione:struttura primaria, secondaria, terziaria e quaternaria. La struttura primaria fornisce un'informazione di tipo prettamente chimico e indica la sequenza di residui che compongono la macromolecola secondo il codice riportato in tabella 2.1.

La struttura secondaria fa riferimento alla disposizione spaziale dei residui amminoacidici adiacenti nella sequenza lineare. Gli elementi di struttura secondaria principali sono: l'α-elica, ed i foglietti ß . Entrambi questi elementi di struttura secondaria derivano dalla formazione di ponti ad idrogeno tra i gruppi N-H e C=O del backbone della catena polipeptidica, senza coinvolgere le catene laterali degli amminoacidi in modo tale che possono essere formati da qualsivoglia sequenza amminoacidica. Le α-eliche sono generate dalla formazione di ponti ad idrogeno tra i guppi N-H e C=O ogni quattro residui amminoacidici lungo la sequenza polipeptidica lineare; questo fatto dà luogo ad un'elica regolare che compie un giro completo attorno al proprio asse ogni 3.6 amminoacidi. I foglietti ß possono essere formati sia da catene polipetidiche vicine che hanno la stessa orientazione (foglietti ß paralleli), sia da catene polipetidiche vicine aventi orientazione l'una opposta all'altra (foglietti ß antiparalleli) ed entrambi sono tenuti insieme da legami idrogeno che coinvolgono i gruppi N-H e C=O di residui appartenenti ai diversi filamenti che compongono il foglietto (vedi fig. 2.3).

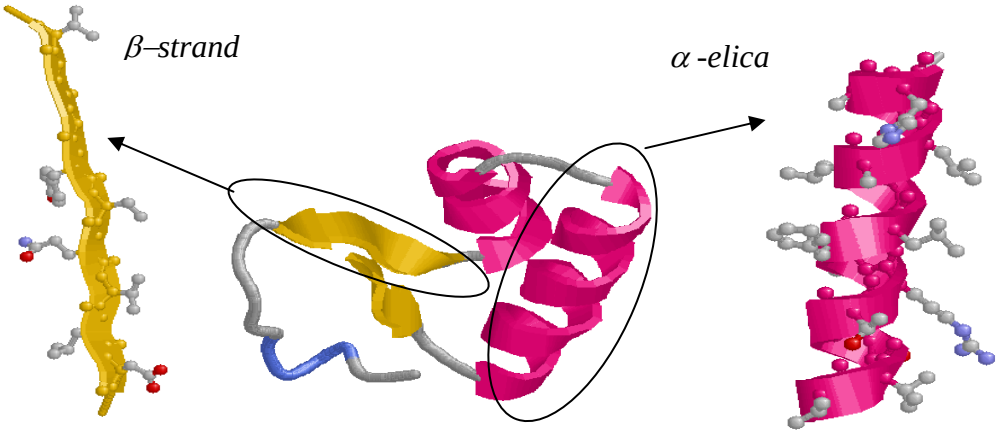


Figura 2.3. I principali motivi di struttura secondaria.

Le strutture secondarie si organizzano tra loro a formare la conformazione stabile e funzionale della proteina, detta struttura terziaria. Tale struttura deriva unicamente, dall'interazione tra gli aminoacidi e tra questi e il solvente. I principali tipi di interazione sono i seguenti:

i) Interazioni elettrostatiche che si generano a causa della distribuzione non uniforme della densità elettronica. Gli amminoacidi carichi, quelli polari e l'acqua, portatrice di un considerevole momento di dipolo, interagiscono fortemente in questo modo. In accordo con le leggi coulombiane, l'interazione elettrostatica è a lungo raggio. Un tipo particolare di interazione elettrostatica è quella tra due residui portatori di carica opposta che si legano con un legame ionico, detto ponte salino.

ii) Il cosiddetto legame a idrogeno che consiste nella "condivisione" di un atomo di idrogeno da parte di due atomi molto elettronegativi. Si è detto nella sezione precedente che legami a idrogeno tra gli atomi della catena principale stabilizzano i motivi di

struttura secondaria. Molti residui possono formare legami a idrogeno anche tramite la catena laterale; inoltre l'acqua può comportarsi sia da donatore che da accettore di legami ad idrogeno.

iii) Le interazioni di dispersione, che derivano dalla polarizzazione reciproca di atomi neutri. Sono forze attrattive a corto raggio (scalano con r^{-6}, se con r si indica la distanza tra i due atomi)

iv) Le interazioni di repulsione, che derivano dalla repulsione tra gli orbitali elettronici di due atomi e che descrivono l'ingombro sterico.

v) I ponti disolfuro, che costituiscono l'unico legame covalente tra amminoacidi di una proteina al di fuori dei legami peptidici. Si formano tra i due atomi di zolfo di due residui di cisteina, in ambiente ossidante. I due residui di cisteina sono tipicamente lontani in sequenza.

vi) L'effetto idrofobico che non costituisce una vera e propria interazione tra coppie di atomi, bensì un comportamento collettivo delle molecole idrofobiche in soluzione polare: queste tendono ad organizzarsi in modo da minimizzare la loro superficie di esposizione al solvente. In una proteina i gruppi apolari tendono ad addensarsi all'interno della struttura globulare, preclusi all'interazione con il solvente dai gruppi polari che invece costituiscono la superficie di esposizione della proteina. Quando una proteina è formata come complesso di più catene polipetidiche, allora la struttura definitiva viene indicata come struttura quaternaria.

2.3 Il processo di folding

Il processo di folding è quella sequenza di transizioni conformazionali mediante cui le proteine appena sintetizzate dai ***Ribosomi***, da stringhe inattive e destrutturate di amminoacidi, si ripiegano su se stesse e acquistano la struttura terziaria funzionale (struttura nativa) (Frauenfelder e Wolynes, 1994; Bryngelson et al., 1995). L'informazione necessaria per il completamento del folding è contenuta nella struttura primaria purché siano verificate le condizioni chimico-fisiche ambientali adeguate (pH, temperatura, forza ionica). Infatti se una proteina viene tratta con agenti denaturanti che ne distruggono la struttura nativa, andando a distruggere la rete di interazioni di tipo non covalente, e successivamente tali agenti vengono rimossi, allora la proteina è in grado di rinaturare spontaneamente nella conformazione originale, indicando così che tutta l'informazione necessaria per il corretto ripiegamento delle proteine è contenuta nella sola sequenza amminoacidica. Sulla base di questo dato empirico è stato formulato il principio di Anfinsen (Anfinsen et al., 1961; Anfinsen, 1973; Anfinsen e Scheraga 1975) che stabilisce che ad ogni sequenza polipeptidica naturale corrisponde una struttura nativa unica e stabile. Tuttavia nelle cellule ci sono degli specifici "macchinari", chiamati "chaperon" molecolari, che aiutano le proteine ad assumere la corretta conformazione nativa. Queste proteine si legano alle catene polipetidiche parzialmente ripiegate in modo da prevenire l'esposizione temporanea di regioni idrofobiche che potrebbero interagire tra di loro a formare degli aggregati proteici particolarmente dannosi.

Da circa un decennio il problema del folding è diventato uno dei problemi cruciali della biologia molecolare, sul quale si sono concentrati numerosissimi studi di tipo sperimentale resi possibili dai recenti sviluppi di tecniche quali l'NMR in più dimensioni, la spettroscopia di fluorescenza risolta nel tempo, etc. Sino ad oggi si è registrato un discreto successo nello studio di quelle fasi preliminari del folding che riguardano la formazione di strutture secondarie e terziarie delle proteine. Il problema che rimane tuttora aperto è quello della simulazione completa di tutte le fasi del folding che portano alla formazione della struttura nativa.

Gli approcci utilizzati per lo studio del sistema complesso proteina si distinguono in due categorie essenziali:

i) approcci probabilistici e derivanti dalle teorie di analisi dei segnali: la struttura primaria è codificata sotto forma di stringhe simboliche o numeriche a partire dalle quali si tenta di estrarre informazioni strutturali; su tale approccio sono basati la maggior parte dei metodi predittivi per la struttura proteica;

ii) approcci molecolari: gli amminoacidi della struttura primaria vengono considerati come oggetti che interagiscono tra loro e con l'ambiente esterno, che generano potenziali di vario tipo e che subiscono l'influenza dei potenziali generati dagli altri amminoacidi; la struttura terziaria e la funzionalità della proteina sono i risultati di tali interazioni.

Tali approcci si differenziano tra loro essenzialmente per il grado di risoluzione con cui viene trattata la proteina che può andare dal livello quantistico, per piccolissimi sistemi, al considerare ogni amminoacido come un unico pseudoatomo o addirittura considerare elementi di struttura secondaria.

In generale tali approcci non sono in grado di determinare la struttura terziaria a partire dalla sola struttura primaria ma forniscono informazioni importanti sulla stabilità, sul rapporto struttura-funzione e sugli aspetti dinamici delle proteine.

Dati sperimentali confermano che le proteine hanno una stabilità marginale, la variazione di energia libera durante il processo di folding varia tipicamente tra 5-20 Kcal/mole (Pace, 1975; Privalov, 1979). La stabilità marginale delle proteine a temperatura ambiente, rende qualsiasi interazione importante per il raggiungimento dello stato nativo (Alber, 1989a,b; Matthews, 1987a,b)

2.4 Stabilità delle proteine

Per descrivere quantitativamente il processo di folding è necessario costruire dei modelli del processo e confrontare i risultati ottenuti con i dati sperimentali a disposizione. In linea di principio, se le l'effetto idrofobico e l'entropia conformazionale sono i contributi principali al processo di folding, allora l'energia libera può essere calcolata semplicemente come variazione dei residui non polari esposti, dallo stato denaturato a quello nativo, moltiplicata per l'energia libera di trasferimento dei residui non polari in superficie e sommata alla differenza di entropia conformazionale. L'energia libera di folding e' stata calcolata con alcune delle diverse assunzioni esemplificative di seguito riportate:

i) nello stato nativo tutti i residui idrofobici sono sepolti all'interno del core non polare della proteina
ii) nello stato denaturato i residui idrofobici sono totalmente esposti al solvente.

Se cosi fosse, il contributo idrofobico dovrebbe essere uguale all'energia di trasferimento di residui non polari in superficie. Considerando inoltre l'unicità dello stato nativo, l'equivalenza tra stato denaturato e una conformazione random e che l'entropia fornisce un contributo locale, allora la variazione entropica dovrebbe valere $nk\ln(z)$ dove n rappresenta il numero legami rotazionali e z il numero di conformazioni accessibili per di-peptide. Questo tipo di approccio non ha avuto molto successo per le seguenti ragioni: le proteine hanno anche residui non polari in superficie e inoltre l'idea che lo stato denaturato sia completamente esposto al solvente è una approssimazione.

E’ più realistico descrivere lo stato denaturato come una serie di conformazioni con le seguenti proprietà:

i) lo stato denaturato ha spesso una densità confrontabile con quella dello stato nativo, avendo solo 1.3-2 volte il volume dello stato stabile invece di 10-100 volte, come previsto in alcuni modelli (Privalov et al., 1986; Ptitsyn, 1987; Goto et al., 1990)

ii) la variazione netta di superficie esposta al solvente, nello stato denaturato, può essere il 14 % della superficie massima (Tanford, 1968; Tanford, 1970; Ahmad e Bigelow, 1986; Schrier e Schrier, 1976)

iii) l’energia libera dello stato denaturato dipende dalla composizione e dalla lunghezza della catena polipeptidica e dalle condizioni sperimentali tra cui: temperatura; pH e salinità (Tanford, 1968; Goto et al. 1990; Shortle et al., 1988; Privalov et al., 1986)

iv) lo stato denaturato può presentare diverse configurazioni non native identificabili sperimentalmente e tra loro transienti; (Evans et al., 1987; Goto e Fink, 1989; Goto et al., 1990; Shortle e Meeker, 1989)

v) in alcuni casi lo stato denaturato presenta una discreta presenza di strutture secondarie formate (Ptitsyn, 1987; Baum et al., 1989; Shortle e Meeker, 1989; Shortle et al., 1988).

Appare chiaro da quanto detto che l’entropia conformazionale non dipende soltanto da fattori locali ma dall’intera catena polipeptidica e dalle proprietà del solvente.

In generale, possiamo affermare che un modello efficiente del meccanismo di folding, deve prevedere anche una descrizione dettagliata dello stato denaturato, in quanto anch'esso concorre alla stabilità della proteina.

I primi modelli di folding riguardavano omopolimeri in soluzioni povere di solvente (Lifschitz, 1968; de Gennes, 1975; Post e Zimm, 1979; Sanchez, 1979). Difficoltà maggiori si hanno nello studio di proteine reali, per le quali il folding comporta la riduzione del raggio della proteina, e la minimizzazione dei residui apolari esposti (vedi fig. 2.4).

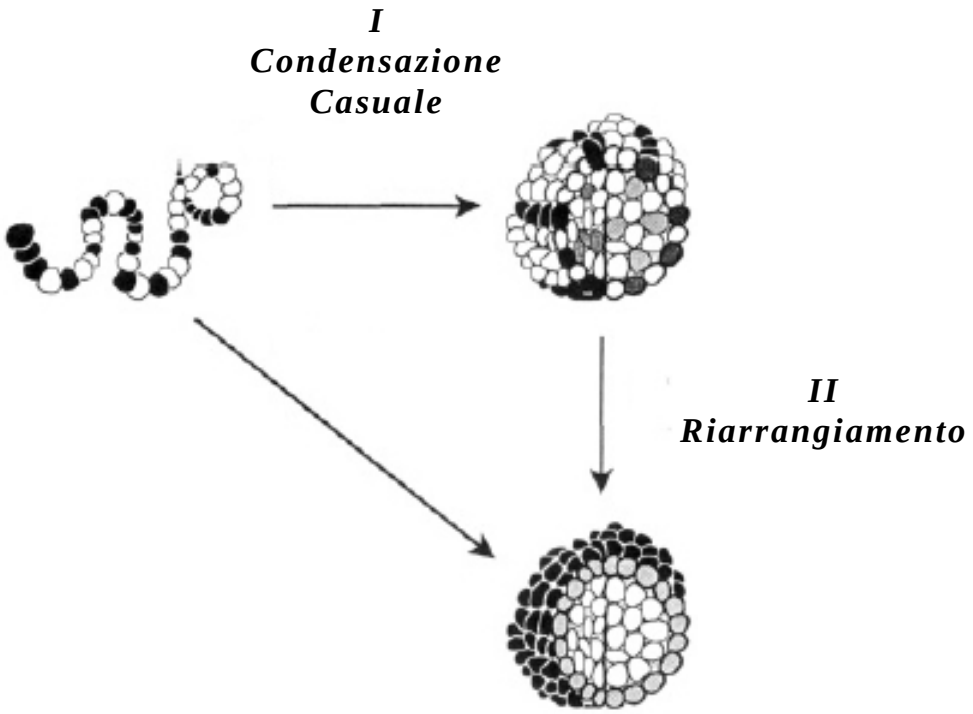


Figura 2.4 In figura è rappresentata una schematizzazione del folding delle proteine in due fasi: la prima, che porta alla formazione di una struttura compatta (I); la seconda, comporta un riordinamento dei residui in modo da portare in superficie i residui polari (disegnati in nero) e spostare quelli non polari (disegnati in chiaro) all'interno (II). Il folding degli omopolimeri presenta solo la fase I.

Gli omopolimeri quindi arrivano allo stato stabile ricercando la conformazione più compatta, mentre le proteine reali hanno anche il problema della minimizzazione dei residui non polari in superficie.

Nella seconda meta degli anni 80 furono sviluppate nuove teorie che prendevano in considerazione le precedenti assunzioni (Dill, 1985; Dill et al., 1989). I nuovi modelli descrivono la proteina come una serie di monomeri connessi con la possibilità di ruotare attorno ai loro legami. Nella rappresentazione precedente la proteina è caratterizzata da due minimi di energia libera: il primo minimo, con una struttura tridimensionale compatta con i residui non polari all'interno del core della proteina (stato foldato) e il secondo, con una struttura meno compatta, la cui stabilità dipende dalle condizioni sperimentali (stato denaturato). Le due classi di possibili conformazioni della proteina sono divise da una barriera di energia libera, così in media, tutti gli stati intermedi sono caratterizzati da una bassa popolazione. Questo risultato è in accordo con dati sperimentali disponibili (Lumry et al., 1966; Privalov e Kechinashvili, 1974; Privalov, 1979).

Il nuovo modello proposto conferma diversi fenomeni osservati sperimentalmente, come ad esempio la transizione del prim'ordine tra lo stato denaturato e quello nativo, a basse concentrazioni di solvente (Dill, 1985; Dill et al., 1989). E' possibile, inoltre, prevedere problemi a raggiungere lo stato stabile per quelle proteine con un basso numero di residui idrofobici e per quelle con la sequenza troppo corta. In media, l'energia libera dovuta all'effetto idrofobico per una proteina piccola a 25°C è circa 60 Kcal/mole e l'energia libera conformazionale, nelle stesse condizioni, vale circa 50 Kcal/mole.

Dai dati precedenti possiamo comprendere come una piccola variazione dei residui idrofobici lungo la catena possa rendere instabile la proteina. Uno dei maggiori successi di questa teoria è quello di riuscire a spiegare il paradosso di Levintal, ovvero come sia possibile che una proteina, che presenta un numero di conformazioni enormemente alto, possa raggiungere lo stato nativo in tempi molto brevi. Un'altra importante conferma fornita da questa teoria è la dipendenza della stabilità della proteina dalla temperatura e dalla natura del solvente. Nel capitolo 5 sarà discusso un nuovo algoritmo per lo studio della stabilità delle proteine utilizzando un metodo basato su tecniche di apprendimento automatico quali le Support Vector Machines.

Capitolo 3

Banche dati e confronto di sequenze

3.1 Le banche dati

La biologia grazie alle innovazioni tecnologiche degli ultimi anni, di cui il progetto del sequenziamento del genoma umano è stato uno dei promotori fondamentali, è diventata una scienza che produce una notevole quantità di dati. Pertanto è sempre più importante informatizzare questa enorme mole di dati in modo che la comunità scientifica internazionale possa accedere in maniera rapida ed efficiente a qualsivoglia tipo di informazione. A tale fine le banche dati rivestono un ruolo di primaria importanza il mantenimento e la cura dei dati riguardanti ad esempio le sequenze biologiche (sia di acidi nucleici che di proteine), le strutture tridimensionali delle proteine, e per la consultazione di tutto il materiale bibliografico di interesse medico-biologico. Mi limiterò qui ad una sommaria descrizione delle banche dati di primaria importanza per questo lavoro di tesi.

3.1.1 Banche dati di sequenze.

Esistono varie banche dati in cui sono depositate le sequenze primarie delle proteine. Le informazioni in esse contenute sono in continuo aumento, grazie al continuo sequenziamento di interi genomi. Le sequenze primarie possono derivare dal sequenziamento di proteine isolate e purificate o dalla traduzione automatica di sequenze di acidi nucleici, possono appartenere a proteine pienamente caratterizzate o essere definite come proteine putative. La banca dati contenente i dati più “puliti” e meglio annotati è SWISS-PROT (http://www.expasy.org/sprot; Bairoch e Apweiler, 2000) che raccoglie le sequenze di 259034 proteine (febbraio 2007). Questa banca dati di sequenze viene curata manualmente da un certo numero di esperti che mantengono un alto livello di annotazione per le proteine. Annotare una sequenza significa discriverne la funzione molecolare, i suoi domini strutturali, eventuali modificazioni post-traduzionali, localizzazione di siti attivi e di binding , localizzazione sub-cellulare, etc etc. Inoltre i curatori della banca dati cercano di mantenere un livello minimo di ridondanza, catalogando in un'unica “entry” le sequenze derivanti da uno stesso gene, e forniscono un alto livello di integrazione con altre banche dati di interesse biologico-molecolare. La banca dati TrEMBL contiene invece 3826359 (febbraio 2007) sequenze proteiche tradotte da acidi nucleici e annotate automaticamente con metodi computazionali. Una banca dati che somma tutte le sequenze proteiche disponibili derivanti dall’unione di diverse banche dati (GenPept+PDB+SwissProt+PIR+RefSeq) è la NR (Non Redundant), scaricabile dal sito ftp dell’NCBI, (ftp://ftp.ncbi.nih.gov/blast/db/) che contiene circa 4,692,000 sequenze.

3.1.2 Banche dati strutturali

La banca dati PDB (Protein Data Bank: http://www.rcsb.org/pdb/; Berman et al., 2000) contiene le coordinate atomiche di 38504 (febbraio 2007) proteine a struttura nota. Gli esperimenti che permettono di risolvere la struttura di una proteina sono essenzialmente la diffrazione a raggi X di una proteina cristallizzata e la risonanza magnetica nucleare di (piccole) proteine in soluzione. I curatori del PDB hanno sviluppato un sistema automatico mediante cui avviene l'acquisizione dei dati ed la loro validazione. Questi sono costruiti secondo il dizionario mmCIF, che è un'ontologia di circa 1700 termini che definiscono la struttura macromolecolare e gli esperimenti cristallografici. I dati depositati vengono considerati dati primari ed oltre alle coordinate atomiche questi contengono informazioni generali e quelle relative al metodo di risoluzione della struttura. Un file PDB è quello che generalmente viene chiamato "flat file" ed è il risultato dell'integrazione delle informazioni contenute nelle diverse componenti della banca dati; è un file di testo che ha la caratteristica di essere organizzato secondo dei campi ben definiti. Il numero di strutture note è relativamente piccolo, se confrontato a quello delle sequenze, a causa delle difficoltà sperimentali. Un altro problema è quello della ridondanza, vale a dire, che sono depositate più strutture relative alla medesima sequenza, in quanto sono state risolte con diverse condizioni sperimentali oppure hanno diversi gradi di risoluzione e quindi di qualità (minor risoluzione implica maggiore qualità).

3.2 L’allineamento di sequenze

L’allineamento di sequenze è una procedura che viene utilizzata per confrontare le diverse sequenze biologiche. Il suo uso nel campo della biologia molecolare è ormai una pratica di routine. L’origine della cosiddetta informazione evolutiva risiede nella constatazione che sequenze simili condividono la medesima struttura tridimensionale (Sander e Schneider, 1991). Questo fatto viene ricondotto all’idea che tutte le sequenze siano separabili in classi, dette di omologia, ognuna delle quali costituisce l’insieme delle sequenze evolutesi a partire da una unica sequenza ancestrale. Questa, in seguito a mutazioni casuali e riarrangiamenti genetici, ha dato origine a sequenze differenti che però si strutturano in proteine aventi la stessa conformazione tridimensionale e molto spesso conservano anche la stessa funzione. Alla data di febbraio 2007 il sito dell’NCBI (http://www.ncbi.nlm.nih.gov/) mette a disposizione i genomi completi di 36 Archaeabatteri, 432 Eubatteri, 26 Eucarioti. Da questi semplici dati si capisce come l’utilizzo di tecniche per il confronto tra sequenze, costituisca una procedura necessaria per molte indagini biologiche. Diversi sono i problemi in cui l’uso dell’allineamento di sequenze può costituire una valida soluzione:

i) La ricostruzione della struttura tridimensionale di proteine non cristallizzate. E’ ormai risaputo che due proteine di lunghezza superiore a 100 residui che hanno un’identità di sequenza superiore al 30 % hanno strutture simili (Rost B, 1999). Se una nuova proteina è sequenziata, ed è disponibile la struttura di una proteina con una identità di sequenza superiore del 30%, posso ricostruire la struttura della prima proteina a partire da quella cristallizzata.

ii) Costruendo un pattern di residui che appartengono ad una famiglia di proteine con una certa funzionalità, posso tramite l'allineamento di sequenze, dire se una nuova proteina può appartenere o meno a quella classe, quindi assegnare una funzione ad una proteina di cui conosco solo la sequenza.

iii) Gli allineamenti di sequenze sono inoltre utili per andare a confrontare le proteine con stessa funzionalità in organismi differenti e costruire un albero filogenetico.

3.2.1 La procedura di allineamento

L'analisi di sequenza è utile per comprendere se due sequenze siano o meno correlate. La soluzione del problema è solitamente fornita dalla valutazione dall'allineamento delle due sequenze. I punti chiave sono: 1) quali ordini di allineamento devono essere considerati; 2) il sistema di scoring usato per effettuare gli allineamenti; 3) l'algoritmo utilizzato per ottimizzare gli allineamenti; 4) i metodi per la valutazione della bontà dell'allineamento.

Sono mostrate tre paia di allineamenti della stessa regione dell'alfa emoglobina (P01922) umana confrontata con: i) emoglobina delta umana (P02042); ii) emoglobina di *Paramecium*; iii) idrolasi di E.Coli (vedi fig 3.1).

```
i)    P01922   KKVADALTN AVAHVDDMPNALSALSDLHAHKL
             :: :   :. . . .. : :. : ..  . . .: : :.: :   ::
      P02042   KKVLGAFSDGL AHLDN LKGTFSQLSELHCDKL

ii)   P01922   KKVADALTNAVAHVDDM PNAL SA  - - LSDLHAHKL
            .:. :    :   :. .          :: :   ..    :. . .:: .  .
      P15160   NKTAAF LCAALGG - - - -  PNAWTGRN LKEVHAN–M

iii)  P01922   KKV - - - - - ADALTNAVAHV DDMPNALSALSDLHAHKL
             . .       .. . . .  .::    ::  :.      ::  :. :
      P77731   TQAFI PMKGEVFVVVVA LGDDKPD - - - - LSTLRAF IT
```

Figura 3.1: Esempio di allineamento a coppie di una sequenza (P01922) contro tre sequenze. Le sequenze allineate sono: regione dell'alfa emoglobina (P01922) umana confrontata con: i) emoglobina delta umana (P02042); ii) emoglobina di Paramecium; iii) idrolasi di E.Coli.

Nella linea centrale di ciascun allineamento si indica con ":" le posizioni identiche e con "." le posizioni "simili". Paia di residui "simili" ed identiche sono quelle che danno uno score positivo nelle matrici di sostituzione. Si può osservare nell'allineamento i) molte posizioni nelle quali gli aminoacidi corrispondenti sono identici o funzionalmente conservati. Nell'allineamento ii) abbiamo due sequenze di due specie diversi con la stessa struttura tridimensionale e la stessa funzione (legare le molecole di ossigeno). In questo caso abbiamo un numero minore di identità e sono stati inseriti meno gap nelle sequenze. L'allineamento iii) mostra un simile numero di identità e di cambiamenti conservativi rispetto al ii). In quest'ultimo caso si tratta di un allineamento privo di senso, in quanto è ottenuto con una proteina che è completamente differente sia nella funzione che nella struttura. Distinguere l'allineamento ii) dal iii) è la sfida per i metodi di allineamento di sequenze. Non sempre è possibile distinguere con certezza se un dato allineamento sia sensato o meno.

3.2.2 Il modello del punteggio

Comparando le sequenze, cerchiamo l'evidenza che esse derivino, mediante un processo di mutazione e selezione, da un comune antenato. I processi di mutazione considerati sono: sostituzioni, con le quali vengono sostituiti dei residui aminoacidici in una sequenza; inserzioni e delezioni, con le quali vengono aggiunti o rimossi degli aminoacidi. Inserzioni o delezioni sono insieme definite come indels. La selezione naturale ha un effetto su questo processo attraverso il controllo delle mutazioni. In conclusione la somiglianza tra le due sequenze aminoacidiche può derivare da una relazione di omologia strutturale e/o funzionale evolutasi attraverso un meccanismo di progressiva differenziazione da un comune progenitore. Il livello di similarità globale dipende in questo caso dal grado di divergenza delle due sequenze lungo la linea filogenetica e può costituire il criterio guida per la ricostruzione dell'albero filogenetico della proteina.

Ad un allineamento assegneremo un punteggio ("score") totale che sarà una somma di termini per ciascun paio di residui allineati, più i termini per ciascun gap. Daremo una interpretazione probabilistica a questo punteggio, mediante il calcolo del logaritmo della probabilità relativa che le due sequenze siano correlate rispetto alla probabilità che non lo siano. Ci aspettiamo che identità o sostituzioni conservative siano più probabili in allineamenti che quelle attese per caso e quindi avremo un contributo in termini di score positivo; cambiamenti non conservativi sono attesi meno frequentemente in allineamenti veri rispetto a quelli attesi per caso e questi daranno un contributo negativo.

Utilizzando uno schema di scoring aggiuntivo equivale ad assumere di poter considerare le mutazioni in diversi siti della sequenza aminoacidica come eventi indipendenti (trattando un gap di lunghezza arbitraria come una mutazione singola). Su tale schema di scoring si sono ideati algoritmi per trovare gli allineamenti ottimali. L'indipendenza delle mutazioni appare essere una ragionevole approssimazione per il DNA e per le sequenze proteiche, nonostante le interazioni tra residui abbiano un ruolo importante nella determinazione della struttura proteica.

3.2.3 Matrici di sostituzione

Abbiamo bisogno di termini di score per ogni coppia di residui allineati. Si possono dedurre gli score di sostituzione da un modello probabilistico. Consideriamo un paio do sequenze x e y, di lunghezza rispettivamente n e m. Sia x_i il simbolo i-esimo di x e y_j il simbolo j-esimo di y. Questi simboli potranno essere le quattro basi del DNA {A,G,C,T} e nel caso delle proteine i venti amminoacidi. Prendendo in considerazione allineamenti globali senza gap, come nell'allineamento i), dato un paio di sequenze allineate vogliamo assegnare uno score all'allineamento che dia una misura della probabilità relativa che le due sequenze siano correlate o meno. Dobbiamo avere due modelli che assegnino una probabilità random ed una mach e poi considerarne il rapporto. Il modello "random" o a sequenze non correlate assume che il residuo a avvenga indipendentemente con frequenza q_a e la probabilità delle due sequenze è il prodotto delle probabilità di ogni aminoacidi:

$$P(x,y \mid R) = \prod_i q_{x_i} \prod_j q_{y_j} \qquad (3.1)$$

Nel modello M match i residui allineati si verificano con una probabilità congiunta p_{ab}. Questo valore p_{ab} può essere pensato come la probabilità che i residui a e b siano entrambi derivati indipendentemente uno dall'altro dallo stesso residuo c non conosciuto, loro comune ancestrale (c può essere lo stesso a e/o b). Questo fornisce una probabilità per l'intero allineamento

$$P(x,y \mid M) = \prod_i p_{x_i y_i} \qquad (3.2)$$

Il rapporto di queste due probabilità è conosciuto come “odds ratio”:

$$\frac{P(x,y|M)}{P(x,y|R)} = \frac{\prod_i p_{x_i y_i}}{\prod_i q_{x_i} \prod_i q_{y_i}} = \prod_i \frac{p_{x_i y_i}}{q_{x_i} q_{y_i}} \qquad (3.3)$$

Per arrivare ad un sistema di scoring aggiuntivo si considera la sommatoria su tutte le coppie allineate dei logaritmi di tale rapporto (log-odds ratio):

$$S = \sum_i s(x_i, y_i) \qquad (3.4)$$

dove

$$s(a,b) = \log\left(\frac{p_{ab}}{q_a q_b}\right) \qquad (3.5)$$

è il rapporto delle probabilità che di un paio di residui (a,b) siano allineati piuttosto che non lo siano. Si può notare come la formula (3.4) sia una somma di score individuali $s=(a,b)$ per ciascun paio di residui allineati, questi score possono essere inseriti in una matrice. Nel caso delle proteine questa matrice è 20X20 con $s=(a_i,b_j)$ in posizione i,j nella matrice, dove a_i,b_j indicano gli aminoacidi i-esimo ed j-esimo. Ciò è conosciuto come score matrix o matrice di sostituzione. Il concetto che il grado di analogia tra aminoacidi differenti sia quantificabile attraverso la probabilità, e quindi delle corrispondenti frequenze di mutazione osservate, ha permesso la costruzione di matrici di largo impiego: le *Mutation Data Matrix* (Dayhoff et al., 1978).

Fin dalla prima versione di Dayhoff l'idea base è costituita dalla percentuale di mutazioni accettate, PAM (Percent Accepted Mutation). Una unità PAM viene definita come la frequenza di mutazione di un aminoacido in un altro che si osserva in un set di proteine omologhe quando 1 residuo su 100 sia andato incontro a mutazione. Per una stima accurata delle probabilità di mutazione sono state allestite varie matrici utilizzando gruppi sempre più estesi di proteine sino a raggiungere matrici PAM 120,250,300, ecc. Le MDM (*Mutation Data Matrix*) hanno alcune limitazioni che vari tentativi successivi hanno cercato di annullare. Le frequenze di mutazioni nelle MDM risultano calcolate su di un set esiguo di famiglie proteiche. Un altro limite delle MDM è l'assunzione che la probabilità di mutazione si distribuisca in modo uniforme lungo la sequenza. Ciò è in contrasto con l'osservazione che si possono individuare zone con maggiore o minore possibilità di mutazione (Taylor, 1986). Un ulteriore miglioramento sarebbe di non limitare il set impiegato per il calcolo delle frequenze di mutazione alle proteine globulari, bensì di estenderlo alle sequenze proteiche filamentose e di membrana. Un modello alternativo alle MDM è costituito dalla matrice di tipo BLOSUM, Blocks Substitution Matrix (Henikoff e Henikoff, 1992) (vedi figura 3.2). Anziché sugli allineamenti globali di proteine omologhe, BLOSUM è stata derivata dall'assemblaggio di blocchi locali ininterrotti di sequenze di proteine altamente conservate. Le varianti di BLOSUM 50, 62, 75, 80, ecc. corrispondono a differenti valori di percentuali di identità prescelti come soglia per la selezione dei blocchi.

	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W
C	9	-1	-1	-3	0	-3	-3	-3	-4	-3	-3	-3	-3	-1	-1	-1	-1	-2	-2	-2
S	-1	4	1	-1	1	0	1	0	0	0	-1	-1	0	-1	-2	-2	-2	-2	-2	-3
T	-1	1	4	1	-1	1	0	1	0	0	0	-1	0	-1	-2	-2	-2	-2	-2	-3
P	-3	-1	1	7	-1	-2	-1	-1	-1	-1	-2	-2	-1	-2	-3	-3	-2	-4	-3	-4
A	0	1	-1	-1	4	0	-1	-2	-1	-1	-2	-1	-1	-1	-1	-1	-2	-2	-2	-3
G	-3	0	1	-2	0	6	-2	-1	-2	-2	-2	-2	-2	-3	-4	-4	0	-3	-3	-2
N	-3	1	0	-2	-2	0	6	1	0	0	-1	0	0	-2	-3	-3	-3	-3	-2	-4
D	-3	0	1	-1	-2	-1	1	6	2	0	-1	-2	-1	-3	-3	-4	-3	-3	-3	-4
E	-4	0	0	-1	-1	-2	0	2	5	2	0	0	1	-2	-3	-3	-3	-3	-2	-3
Q	-3	0	0	-1	-1	-2	0	0	2	5	0	1	1	0	-3	-2	-2	-3	-1	-2
H	-3	-1	0	-2	-2	-2	1	1	0	0	8	0	-1	-2	-3	-3	-2	-1	2	-2
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5	2	-1	-3	-2	-3	-3	-2	-3
K	-3	0	0	-1	-1	-2	0	-1	1	1	-1	2	5	-1	-3	-2	-3	-3	-2	-3
M	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5	1	2	-2	0	-1	-1
I	-1	-2	-2	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4	2	1	0	-1	-3
L	-1	-2	-2	-3	-1	-4	-3	-4	-3	-2	-3	-2	-2	2	2	4	3	0	-1	-2
V	-1	-2	-2	-2	0	-3	-3	-3	-2	-2	-3	-3	-2	1	3	1	4	-1	-1	-3
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6	3	1
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-2	-2	-1	-1	-1	-1	3	7	2
W	-2	-3	-3	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11

Figura 3.2 Matrice di sostituzione blosum62

Un perfezionamento dei sistemi di punteggio è necessario per una più precisa rivelazione delle correlazioni deboli. Vi è un accordo generale nel considerare MDM e BLOSUM come i modelli di scelta per il calcolo dei punteggi di similarità (Schwartz e Dayhoff, 1978).

3.2.4 Penalità dei gap

Ci aspettiamo di penalizzare i gap. Il costo standard associato ad un gap di lunghezza g è dato o mediante uno score lineare

$$\gamma(g) = -ge \tag{3.6}$$

o con score affine

$$\gamma(g) = -d - (g-1)e \tag{3.7}$$

dove d è chiamata penalità gap-opening ed e è definita penalità gap-extension. La penalità di estensione del gap è di solito meno penalizzante dell'apertura del gap stesso con il costo lineare. Questo è auspicabile quando i gap di pochi residui sono attesi con la medesima frequenza dei gap di un singolo residuo. Anche le penalità per i gap si rifanno ad un modello probabilistico di allineamento. Si assume la probabilità di un gap, presente in un particolare sito in una data sequenza, come il prodotto della funzione $f(g)$ della lunghezza del gap per la probabilità combinata del set di residui inseriti:

$$P(\text{gap}) = f(g) \prod_{i \text{ in gap}} q_{x_i} \tag{3.8}$$

Dalla formula si evince che la lunghezza del gap non è correlata ai residui che contiene. I valori naturali per la probabilità q_a sono come quelli utilizzati nel modello random, poiché corrispondono ambedue a residui indipendenti non combinati.

Quando otteniamo la "odds ratio" la penalità per i gap si semplifica in $\gamma(g) = \log(f(g))$ il logaritmo della probabilità di un gap di quella lunghezza, per cui assumendo il gap affine o lineare $f(g) = e^{\gamma(g)}$.

3.2.5 Valutazione degli allineamenti

La fase finale della procedura di allineamento tra sequenze, consiste nella valutazione del risultato ottenuto. Il primo valore importante nel giudicare un allineamento e sicuramente il punteggio associato. Questa parametro pur essendo indicativo della qualità dell'allineamento ottenuto non ha comunque validità generale, basti pensare al fatto che la scelta della matrice di sostituzione è del tutto arbitraria e dettata dall'esperienza. Infatti, il problema principale associato all'allineamento di sequenze è rappresentato dalla possibilità di ottenere falsi positivi. I falsi positivi sono costituiti da tutti quegli allineamenti che, pur dando un risultato accettabile in termini di punteggio, mettono in relazione due sequenze con proprietà completamente differenti. Un altro fattore importante nel giudizio degli allineamenti è la percentuale di identità, che date sue sequenze di caratteri A_i e A_j allineati e di lunghezza l, può essere definita come:

$$\%ID = \frac{\sum_{k=1}^{l} \delta(A_i(k), A_j(k))}{l} \cdot 100 \qquad (3.9)$$

dove $\delta(A_i(k), A_j(k))$ è uguale a 1 quando i due caratteri allineati sono uguali e 0 altrimenti. In generale però anche questo parametro, se considerato da solo, potrebbe portare a valutazioni errate. Si è pensato quindi di giudicare gli allineamenti utilizzando come metro gli allineamenti casuali. La qualità degli allineamenti attualmente è valutata calcolando lo Z-score. Questo parametro è ottenuto statisticamente generando un numero abbastanza grande di allineamenti casuali, sui quali si calcola la media dei punteggi degli allineamenti $\langle S \rangle$ e la deviazione standard σ.

Se S è il punteggio ottenuto dall'allineamento che voglio valutare, lo Z-score associato sarà:

$$Z - score = \frac{S - \langle S \rangle}{\sigma} \qquad (3.10)$$

Questo valore ci dice a che distanza (misurata in σ) il nostro punteggio si trova rispetto al punteggio medio dagli allineamenti casuali generati $\langle S \rangle$.

3.3 Algoritmi per allineamento di sequenze

Nell'analisi dell'allineamento di sequenze un ruolo fondamentale è svolto dagli algoritmi deputati alla ricerca del miglior allineamento tra esse. Considerando due sequenze della stessa lunghezza n è possibile un solo allineamento globale per le sequenze complete, ma una volta permessi i gap tutto diviene molto più complesso. Ci sono

$$\binom{2n}{n} = \frac{(2n)!}{(n!)^2} \cong \frac{2^{2n}}{\sqrt{2\pi n}} \qquad (3.11)$$

possibili allineamenti globali tra due sequenze di lunghezza n. Visto il numero elevato non è fattibile enumerare tutti gli accoppiamenti anche per valori di n molto piccoli. Gli algoritmi per trovare gli allineamenti ottimali dato un score di allineamento aggiuntivo del tipo descritto sopra sono chiamati dynamic programming. Gli algoritmi più semplici sono quelli per l'allineamento di sequenze pairwise (a coppia), sia in maniera globale che locale. Introdotto lo schema di scoring come log-odds ratio, il miglior allineamento avrà lo score più alto e quindi dobbiamo cercare di massimizzare tale score. E' utile ricordare che lo scopo di un algoritmo di allineamento è quello di incorporare nella matrice di allineamento più valori positivi di score possibili per le varie coppie di residui conservati, minimizzando il costo per i residui non conservati, per i gap e per altri vincoli. Date due sequenze si cercherà di ottenere il miglior allineamento globale. L'algoritmo di programmazione dinamica per questo problema è conosciuto nella biologia di analisi di sequenze come l'algoritmo Needleman-Wunsch (Needleman e Wunsch, 1970), anche se la versione più efficiente fu introdotta da Gotoh (Gotoh, 1982).

L’idea base di questo algoritmo è di ottenere il miglior allineamento usando precedenti soluzioni per allineamenti ottimali di brevi sottosequenze. Finora abbiamo parlato di confrontare le sequenze considerandone l’allineamento migliore da una estremità all’altra. Più comune è la situazione in cui cerchiamo l’allineamento ottimale tra due sottosequenze di x e y ad esempio si sospetta che due sequenze proteiche condividano un dominio comune. La versione locale dell’algoritmo di allineamento di sequenze di programmazione dinamica è stato sviluppato negli anni ’80 ed è conosciuto come algoritmo di Smith-Waterman (Smith e Waterman, 1981; Gotoh, 1982). Questa procedura risulta la migliore anche per individuare similarità tra sequenze altamente divergenti, ma che condividono un’origine evolutiva comune lungo l’intera loro lunghezza. Questo perché in tali casi solo parte della sequenza è stata soggetta ad una forte selezione preservando zone di similarità; il resto della sequenza avrà accumulato “rumore” attraverso mutazioni e non sarà allineabile. L’allineamento di sottosequenze di x e y con lo score più alto è definito il miglior allineamento locale.

3.3.1 Notazione O-grande per la complessità algoritmica

Nello studio degli algoritmi è di notevole importanza conoscerne la complessità. Gli algoritmi di programmazione dinamica descritti finora hanno una complessità dell'ordine di $O(nm)$, il prodotto delle lunghezze delle sequenze. Un algoritmo impiega $O(nm)$ tempo e $O(nm)$ memoria, dove n e m sono le lunghezze delle sequenze; $O(nm)$ è una notazione standard chiamata O-grande che significa "di ordine nm". Questo vuol dire che il tempo computazionale e la memoria richiesti per risolvere il problema sono proporzionali al prodotto delle lunghezze delle sequenze nm. Poiché n ed m sono di solito simili l'algoritmo è solito dirsi $O(n^2)$.

Poiché il tempo di esecuzione di un programma è proporzionale alla complessità computazionale di quest'ultimo $(O(f(n)))$, maggiore è il grado della funzione $f(n)$ e meno applicabile a casi reali è l'algoritmo. In casi in cui la funzione $f(n)$ è "Non Polinomiale" (es. $n!$) l'algoritmo è di solito intrattabile e si dice che è un problema NP-completo.

3.4 Algoritmi di allineamento per la ricerca in banche dati

Gli algoritmi di programmazione dinamica citati in precedenza sono considerati “corretti” nel senso che sono implementati per trovare lo score migliore secondo lo schema di scoring specificato. Questi metodi di appaiamento di sequenze non sono i più veloci ed in molti casi la lentezza diviene un problema. Per superare l’ostacolo della lentezza si sono susseguiti tentativi per ideare algoritmi più veloci della programmazione dinamica diretta. Lo scopo di questi metodi è cercare la più piccola frazione possibile delle celle nella matrice di programmazione dinamica, mentre allo stesso tempo ricercare tutti gli allineamenti che danno un alto punteggio. Sono disponibili varie tecniche di ricerca di questo tipo: l’algoritmo più utilizzato è BLAST (Basic Local Alignment Search Tool).

3.4.1 BLAST

BLAST, Basic Local Alignment Search Tool, (Altschul et al., 1990) permette di ricercare similarità tra una sequenza "query" ed un database di sequenze (possono essere sequenze proteiche o DNA). BLAST cerca allineamenti di sequenza ad elevato score tra la sequenza query e quelle contenute in una certa banca dati, utilizzando un approccio euristico che approssima l'algoritmo di Smith-Waterman, che sarebbe troppo lento per condurre una ricerca in banche dati di sequenze di grandi dimensioni. Tuttavia l'approccio euristico di BLAST è leggermente meno accurato rispetto all'algoritmo di Smith-Waterman, ma è 50 volte più veloce. Queste due caratteristiche fanno di BLAST uno degli strumenti più utilizzati e fondamentali per la ricerca bioinformatica. Concettualmente possiamo dividere l'algoritmo di BLAST in tre fasi:

i) come prima cosa BLAST cerca appaiamenti ("match") esatti e senza gap, di piccoli pezzi di lunghezza W tra la sequenza query e quelle della banca dati, detti "seeds" e che abbiano almeno un certo valore soglia di score. Ad esempio, date le sequenze AGTTAC ed ACTTAG e definita una stringa di lunghezza W=3, BLAST identificherà la sottostringa comune TTA che dà un match esatto. Per gli acidi nucleici di default W=11, per le proteine di default W=3.

ii) successivamente BLAST cerca di estendere il match in entrambe le direzioni a partire dai "seeds" identificati nella prima fase. Il processo di allineamento senza gap estende i seeds iniziali di lunghezza W in entrambe le direzioni nel tentativo di incrementare lo score dell'allineamento in modo che non scenda al di sotto del valore soglia di score definito.

In questa fase gli eventi di inserzioni e delezioni non sono presi in considerazione. Per esempio l'allineamento senza gap per le due sequenze di cui sopra, centrate attorno al seed TTA sarebbe:

..AGTTAC..
| |||
..ACTTAG..

se viene trovato un allineamento privo di gap e con score elevato, si passa alla fase successiva

iii) nella terza fase BLAST opera la costruzione di un allineamento con gap tra la sequenza query e quelle della banca dati utilizzando una versione modificata dell'algoritmo Smith-Waterman. Infine vengono mostrati gli allineamenti statisticamente significativi

La significatività di un allineamento generato da BLAST è stabilito dal valore di un parametro statistico chiamato "Expectation value" (E-value). L'E-value rappresenta il numero di allineamenti differenti con score maggiori o uguali allo score S, ottenuto per mio allineamento, che mi aspetto di trovare per caso in una ricerca di sequenza in banca dati. Tanto più è piccolo l'E-value (prossimo a 0) tanto più è significativo lo score. In fig 3.1 abbiamo riportato la pagina web di protein-protein BLAST collegata alla pagina principale della suite di BLAST (http://www.ncbi.nlm.nih.gov/BLAST/).

Figura 3.3: Pagina web dell'algoritmo protein-protein BLAST disponibile, tramite un link, presso il sito dell'NCBI all'indirizzo http://www.ncbi.nlm.nih.gov/BLAST.

3.4.2 Profili di sequenza

L'informazione evolutiva contenuta in un allineamento di N sequenze viene spesso riassunta in un profilo di allineamento, cioè in una matrice 20xL, dove L indica il numero di posizioni dell'allineamento, contenente la composizione amminoacidica percentuale delle sequenze allineate in ogni posizione. La figura 3.5 esemplifica il passaggio da un allineamento multiplo ad un profilo. E' evidente che il profilo di allineamento contiene meno informazione di quanta ne sia contenuta nell'allineamento multiplo, in quanto rappresenta una sorta di sequenza vettoriale media. Dato un profilo non è perciò possibile ricostruire l'allineamento da cui esso deriva. Tuttavia le informazioni concernenti il grado di conservazione di un residuo della sequenza o le mutazioni compatibili con una struttura corretta sono immediatamente leggibili nei valori del profilo che sono, posizione per posizione, tanto più vicini a 100 quanto più un residuo è conservato. Il primo passo per la costruzione del profilo evolutivo di una sequenza è la ricerca nella banca dati delle sequenze ad essa omologhe. Su queste viene condotto un allineamento multiplo da ci viene estratto il profilo. Questa è la strategia adottata dal programma MaxHom nella costruzione della banca dati HSSP contenente allineamenti e profili di tutte le sequenze risolte tridimensionalmente (Sander e Schneider, 1991). Un'altra strategia è quella di generare i profili dagli allineamenti locali calcolati dall'algoritmo di ricerca. Questa tecnica ha acquisito particolare importanza da quanto è stato sviluppato il programma PSI-BLAST (Altschul et al., 1997).

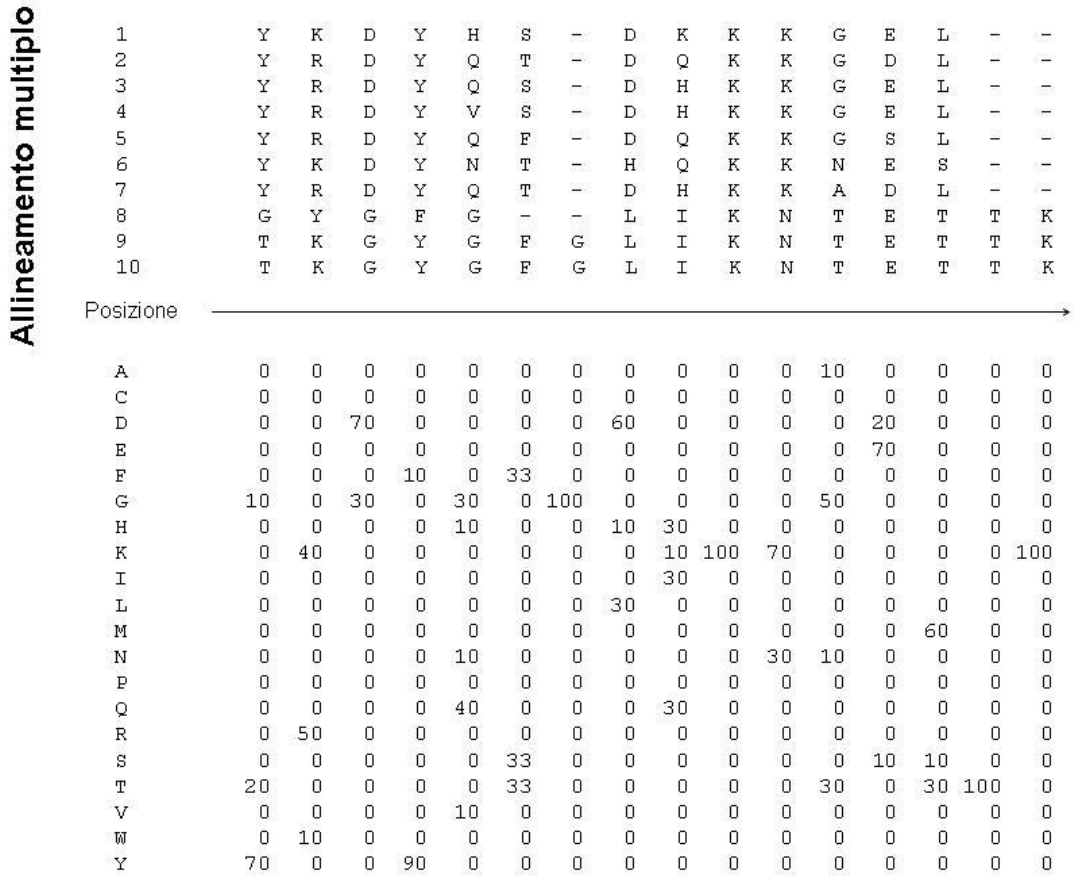


Figura 3.4 Costruzione di un profilo di sequenza a partire da un allineamento multiplo

Questo esegue la ricerca nella banca dati tramite un processo iterativo che prevede una prima ricerca effettuata con BLAST, la costruzione del profilo sulla base degli allineamenti locali rintracciati e quindi una nuova ricerca nella banca dati delle sequenze che meglio si allineano al profilo. Questo schema può essere iterato fino a quando il profilo non muta dopo un ciclo di ricerca-costruzione del profilo.

Capitolo 4

Machine Learning e Suport Vector Machines

4.1 Machine Learning

I metodi di apprendimento automatico ("Machine Learning") si occupano della realizzazione di sistemi che si basano su osservazioni, o esempi noti, come dati per la sintesi di nuova conoscenza (classificazioni, generalizzazioni). Sono numerose le situazioni di difficile soluzione mediante algoritmi tradizionali, cioè mediante una serie finita e non ambigua di istruzioni formulate sulla base di un modello matematico del fenomeno preso in analisi. Le principali limitazioni per l'utilizzo di un procedimento algoritmico tipicamente sono dovute alla presenza di uno o più dei seguenti fattori:

i) Difficoltà di formalizzazione.
ii) Elevato numero di variabili in gioco
iii) Mancanza di teoria
iv) Necessità di personalizzazione

Gli algoritmi di apprendimento automatico si possono dividere in due tipologie principali:

i) Apprendimento supervisionato: un istruttore fornisce esempi (positivi e negativi) di quello che si deve apprendere.

ii) Apprendimento non supervisionato: parte da osservazioni non preclassificate

L'Apprendimento automatico di tipo supervisionato cerca di istruire un sistema informatico in modo da consentirgli di risolvere dei compiti in automatico. Bisogna definire i dati in ingresso, di solito in fora vettoriale, come un insieme di esempi I; poi si definisce l'insieme dei dati in uscita come insieme di output O (gli output possono essere un'etichetta numerica, ad esempio ±1); infine bisogna definire una funzione *f* che associa ad ogni dato in ingresso (I) la sua risposta corretta (O). Tutti gli algoritmi di apprendimento supervisionato partono dal presupposto che se forniamo all'algoritmo un numero adeguato di esempi l'algoritmo sarà in grado di creare una funzione *f*1 che approssimerà la funzione *f*. Se l'approssimazione di *f* risulterà adeguata, quando proporremo ad *f*1 dei dati in ingresso, mai analizzati in precedenza, la funzione dovrebbe essere in grado di fornire delle risposte in uscita simili a quelle fornite da *f* e quindi corrette o quasi. Molti di questi algoritmi in sostanza lavorano in un mondo lineare, presupponendo che a ingressi simili necessitino di uscite simili. Nel nostro mondo questo in generale non è vero, basta vedere le dinamiche caotiche legate al tempo, ma esistono molte condizioni in cui questa semplificazione è accettabile. Si può facilmente intuire che il buon funzionamento di questi algoritmi dipende in modo significativo dai dati in ingresso; se si forniscono pochi ingressi l'algoritmo potrebbe non aver abbastanza elementi per apprendere, mentre molti dati in ingresso potrebbero renderlo eccessivamente lento, dato che la funzione *f*1 generata dagli ingressi potrebbe essere molto complicata. Questi algoritmi sono molto sensibili al rumore, anche pochi dati errati potrebbero rendere l'intero sistema non affidabile e condurlo a decisioni errate.

Le tecniche di apprendimento automatico di tipo non supervisionato cercano di estrarre in modo automatico dalle basi di dati delle regole utili per generare nuova conoscenza. Questa conoscenza viene estratta senza una specifica descrizione dei contenuti che si dovranno analizzare. Un esempio tipico di questi algoritmi lo si ha nei motori di ricerca. Questi programmi, data una o più parole chiave, sono in grado di creare una lista di link rimandanti alle pagine che l'algoritmo di ricerca ritiene attinenti alla ricerca effettuata. La validità di questi algoritmi è legata alla utilità delle informazioni che riescono ad estrarre dalla base di dati. Questi algoritmi lavorano confrontando i dati e ricercando similarità o differenze. Sono molto efficienti con elementi di tipo numerico, dato che possono utilizzare tutte le tecniche derivate dalla statistica, ma sono molto meno efficienti con dati non numerici. Se i dati sono dotati di un ordinamento intrinseco gli algoritmi riescono comunque ad estrarre informazioni, ma se i dati in ingresso non sono dotati di un qualche tipo di ordinamento spesso gli algoritmi falliscono.

Questi algoritmi in conclusione lavorano correttamente in presenza di dati contenenti un ordinamento o un raggruppamento netto e chiaramente identificabile. Uno dei problemi principali per le tecniche di apprendimento automatico è dato dall'***overfitting***. Si parla di overfitting (eccessivo adattamento) quando un modello statistico si adatta ai dati osservati (il campione) usando un numero eccessivo di parametri. Come detto un algoritmo di apprendimento viene addestrato usando un certo insieme di esempi (il training set appunto), su situazioni di cui è già noto il risultato che interessa prevedere (output). Si assume che l'algoritmo di apprendimento (il learner) raggiungerà uno stato in cui sarà in grado di predire gli output per tutti gli altri esempi che ancora non ha visionato, cioè si assume che il modello di apprendimento sarà in grado di generalizzare.

Tuttavia, soprattutto nei casi in cui l'apprendimento è stato effettuato su uno scarso numero di esempi, il modello potrebbe adattarsi a caratteristiche che sono specifiche solo del training set, ma che non hanno riscontro nel resto dei casi; perciò, in presenza di overfitting, le prestazioni (cioè la capacità di adattarsi/prevedere) sui dati di addestramento aumenteranno, mentre le prestazioni sui dati non visionati saranno peggiori. Per evitare l'overfitting, è necessario adottare particolari tecniche, come la ***cross-validation***. Ci sono diverse strategie di cross-validation, ma in questa tesi verrà usata sempre la "K-fold cross-validation". Questa procedura consiste nel partizionare i dati del training set in un certo numero *K* di sotto insiemi. Dei *K* sotto insiemi, uno viene utilizzato come set di validazione per testare il modello, mentre i restanti *K-1* sotto insiemi sono utilizzati come training set per la costruzione del modello. La procedura di cross-validation viene poi ripetuta *K* volte, con ognuno dei *K* sotto insiemi utilizzato esattamente una volta come set di validazione. In fine i *K* risultati ottenuti nella fase di test vengono mediati in modo da ottenere una stima unica del modello generato.

4.2 Support vector Machines

Le macchine a supporto vettoriale (Support Vector Machines, SVMs) sono un'insieme di algoritmi per la regressione e la classificazione di pattern, sviluppati da Vladimir Vapnik (Vapnik et al., 1971, Vapnik et al., 1991, Schölkopf et al., 2002). Le SVM possono essere pensate come una tecnica alternativa alle reti neurali che permettono l'apprendimento mediante classificatori polinomiali; la tecnica di addestramento di una SVM permette di ottenere i parametri caratteristici del modello da generare mediante la soluzione di un problema di ottimizzazione che prevede un unico minimo globale. Al contrario, l'addestramento di una rete neurale mediante la risoluzione di un problema di ottimizzazione ritorna un numero indeterminato di minimi relativi. In parole semplici, una SVM è un classificatore binario che apprende il confine fra esempi appartenenti a due classi diverse. Funziona proiettando gli esempi in ingresso, aventi una certa dimensione nello spazio, in uno spazio multidimensionale, mediante una funzione che opera una mappatura dei dati nel nuovo spazio, e cercando un iperpiano di separazione ottimale in questo spazio (vedi fig 4.1) .

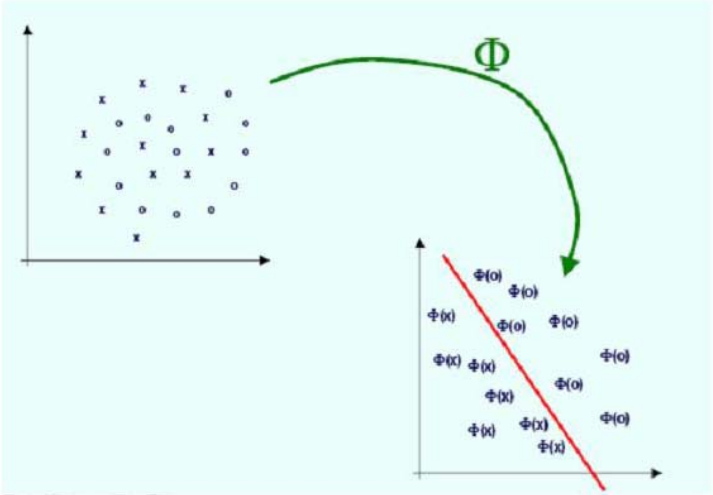


Figura 4.1 In alto a sinistra vediamo come sono mischiate le due classi di esempi (classe x e classe o) nello spazio di ingresso; mediante la funzione di "mapping" ,Φ, i due insiemi vengono proiettati in uno spazio a dimensionalità maggiore dove è possibile trovare un iperpiano ottimale di separazione. http://www3.csr.unibo.it/~maniezzo/didattica/SoftComputing/SVM.pdf

L'iperpiano di separazione massimizza la sua distanza (il "margine") dagli esempi di training più vicini. Le proprietà generali delle SVM sono:

i) improbabile l'overfitting

ii) capacità di gestire dati con molte caratteristiche descrittive

iii) compattamento dell'informazione contenuta nel data set in input.

Nel caso di classificazione di dati appartenenti a due sole classi, il processo di apprendimento può essere formulato come segue: dato un insieme di funzioni di soglia:

$$\{f_\lambda(x): \lambda \in \Lambda\},\ f_\lambda: R^N \rightarrow \{-1, +1\} \qquad (4.1)$$

dove Λ è un insieme di parametri reali, e dato un insieme di esempi preclassificati:

$$(x_1,y_1), \ldots, (x_m,y_m),\ x_i \in R^N,\ y_i \in \{-1, +1\} \qquad (4.2)$$

presi da una distribuzione sconosciuta P(x,y), si vuole trovare una funzione f_λ^* che minimizzi l'errore teorico:

$$R(\lambda) = \int |f_\lambda(x)-y| P(x,y)\, dxdy \qquad (4.3)$$

L'insieme di parametri reali Λ genera una macchina in grado di risolvere un particolare problema (ad esempio Λ può corrispondere ai pesi delle sinapsi di una rete neurale). Le funzioni f_λ sono chiamate ipotesi, e l'insieme $\{ f_\lambda (x): \lambda \in \Lambda\}$ viene chiamato spazio delle ipotesi e si indica con H. L'errore teorico rappresenta una misura di quanto sia buona un'ipotesi nel predire la classe y_i di un punto x. L'insieme delle funzioni può essere ad esempio un insieme di Radial Basis. La distribuzione di probabilità P(x,y) non è nota, quindi non è possibile calcolare l'errore teorico $R(\lambda)$. Tuttavia è disponibile un campione di P(x,y), il training set: si può calcolare un'approssimazione di $R(\lambda)$, l'errore empirico $R_{emp}(\lambda)$:

$$R_{emp}(\lambda) = 1/m \sum_{i=1}^{m} |f_\lambda(x_i)-y_i| \qquad (4.4)$$

La legge dei grandi numeri garantisce che l'errore empirico converge in probabilità all'errore teorico, per cui si cerca di minimizzare l'errore empirico piuttosto che quello teorico. La dimensione VC dello spazio di ipotesi H (o la dimensione VC del classificatore f_λ) è un numero naturale che corrisponde al più grande numero di punti che possono essere separati in tutti i modi possibili dall'insieme di funzioni f_λ. Cioè, dato un insieme di m punti, se per ognuna delle 2^m possibili classificazioni (-1,+1) esiste una funzione f_λ che assegna correttamente le classi, allora si dice che l'insieme di punti viene separato dall'insieme di funzioni. La dimensione VC è una misura della complessità dell'insieme H.

La teoria della convergenza uniforme in probabilità, sviluppata da Vapnik e Chervonenkis, fornisce anche un limite alla deviazione dell'errore empirico dall'errore teorico; fissato η con $0 \leq \eta \leq 1$ vale la seguente disuguaglianza:

$$R(\lambda) \leq R_{emp}(\lambda) + \frac{\sqrt{h(\log\frac{2m}{h}) - \log(\frac{\eta}{4})}}{m} \qquad (4.5)$$

Dove h è la dimensione VC di f_λ. Per ottenere l'errore teorico minimo, bisogna minimizzare sia l'errore empirico sia il rapporto tra la dimensione VC e il numero di punti(h/m).L'errore empirico è solitamente una funzione decrescente di h, quindi, per ogni dato numero di punti, esiste un valore ottimale della dimensione VC (trade-off R_{emp} e h/m). L'algoritmo SVM risolve efficacemente questo problema minimizzando contemporaneamente la dimensione VC e il numero di errori sul training set.

Ipotesi: insieme di dati linearmente separabili. Si vuole trovare il miglior iperpiano che li separa. Un insieme di dati è linearmente separabile, quando è possibile trovare una coppia (w,b) tale che:

$$w \cdot x_i + b \geq +1 \text{ con } x_i \in \text{Classe1} \qquad (4.6)$$

$$w \cdot x_i + b \leq -1 \text{ con } x_i \in \text{Classe 2} \qquad (4.7)$$

Lo spazio delle ipotesi in questo caso è formato dall'insieme di funzioni:

$$h = f_{w,b} = \text{sign}(w \cdot x + b) \qquad (4.8)$$

(sign: discriminatore binario: +/-, 0/1, vero/falso, ...)

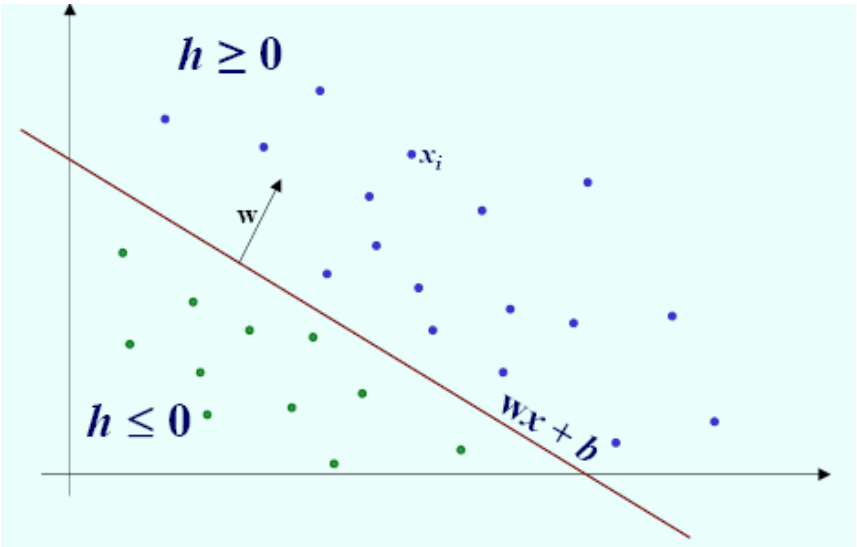


Figura 4.2 Esempio di dati linearmente separabili.
http://www3.csr.unibo.it/~maniezzo/didattica/SoftComputing/SVM.pdf

Se i dati sono linearmente separabili, lo scopo dell'SVM è di trovare tra tutti gli iperpiani che classificano correttamente il training set quello che ha norma minima, cioè margine massimo rispetto ai punti del training set. Ad esempio come vediamo in figura (fig. 4.3), le classi dei cerchi e dei quadrati sono separate dal piano tratteggiato con un margine piccolo (a), o grande (b). Nel caso (b) ci si aspetta un minor rischio di overfitting(migliore generalizzazione).

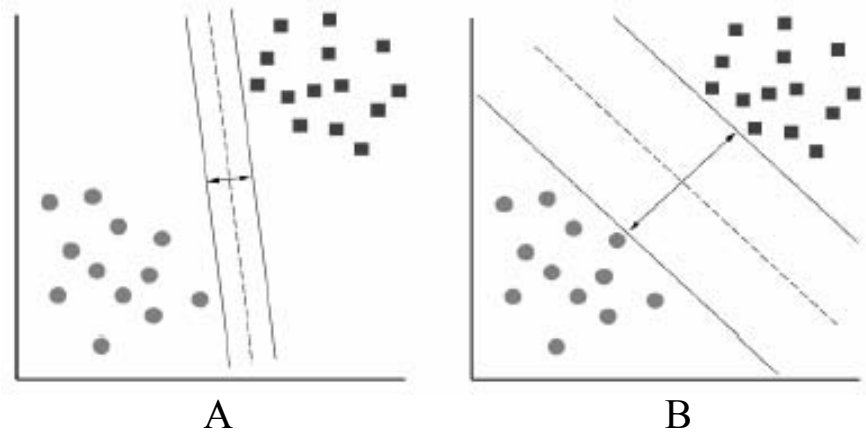


Figura 4.3 Due possibili esempi di margine con cui le classi dei cerchi e dei quadrati possono essere separati. Tanto è maggiore il margine tra le due classi tanto è più improbabile che si verifichino casi di over-fitting. http://www3.csr.unibo.it/~maniezzo/didattica/SoftComputing/SVM.pdf

L'iperpiano ottimo è quello che massimizza il margine, cioè la distanza tra se stesso e i punti più vicini dell'insieme di dati. Per costruire l'iperpiano ottimale, bisogna classificare correttamente i punti del training set nelle due classi (ad esempio etichettate come ±1) usando la più piccola norma di coefficiente w. Il problema può essere formulato come segue:

Minimizzare

$\Phi(w) = \frac{1}{2}||w||^2$ (4.9)

con w,b soggetti al vincolo

$y_i(w \cdot x_i + b) \geq 1,\ i = 1, \ldots, m$ (4.10)

L'iperpiano ottimo può essere scritto come una combinazione lineare dei vettori del training set:

$$f(x)=\text{sign}\left(\sum_{i=1}^{m} y_i \lambda_i^* (x \cdot x_i) + b^*\right) \quad (4.11)$$

per ogni vettore xi

Nella soluzione, tutti i punti x_i per cui il corrispondente moltiplicatore λ_i è strettamente maggiore di zero vengono detti support vector e si trovano su uno dei due iperpiani H1, H2 . Tutti gli altri punti del training set hanno il corrispondente λ_i uguale a zero e non influenzano il classificatore. I support vector sono i punti critici del training set e sono i più vicini all'iperpiano di separazione (vedi fig 4.4); se tutti gli altri punti venissero rimossi o spostati senza oltrepassare i piani su H1 e H2 e l'algoritmo di apprendimento venisse ripetuto, darebbe esattamente lo stesso risultato.

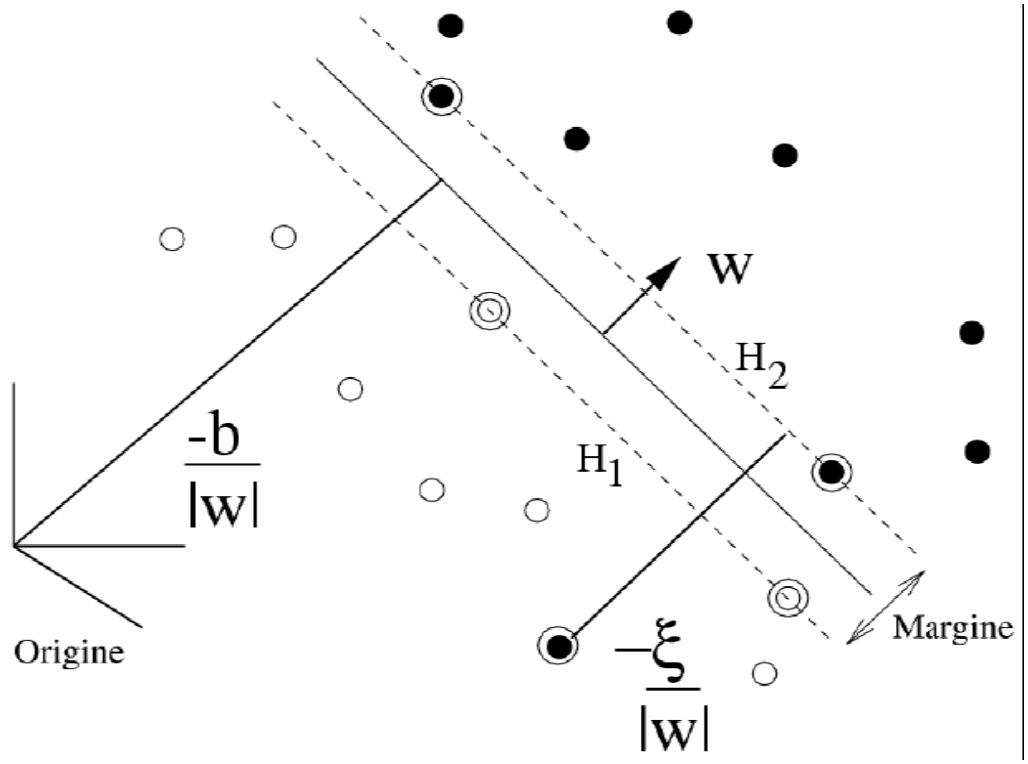


Figura 4.4 Piano separatore per un insieme di punti non linearmente separabili; il piano ha distanza –b/||w|| dall'origine e viene determinato dai support vector (i punti cerchiati). Il punto in posizione anomala e dista -ξ/||w|| dalla sua classe. http://www3.csr.unibo.it/~maniezzo/didattica/SoftComputing/SVM.pdf

Esistono punti in posizione anomala rispetto agli altri punti della stessa classe. Si considera una costante di scarto ξ tanto maggiore quanto più lontani sono i punti anomali.

La 4.10 diventa quindi:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0\ \ i = 1, \ldots, l \qquad (4.12)$$

Il vincolo ora ammette una certa tolleranza (ξ_i) agli errori. Perché un punto del training set venga mal classificato, il corrispondente ξ_i deve superare l'unità. La $\Sigma_i \xi_i$ è un limite superiore al numero massimo di errori possibili sul training set. Il problema può essere quindi riformulato così:

Minimizzare

$$\Phi(w,E) = \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{m}\xi_i\right)^k \qquad (4.13)$$

con w, b e E vincolati da:

$y_i(w \cdot x_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0\ i = 1, \ldots, m$

dove C e k sono parametriche devono essere determinati a priori: ad un alto valore di C corrisponde un'alta penalità dovuta agli errori. In pratica l'algoritmo SVM cerca di minimizzare ||w|| e allo stesso tempo separare i punti dati, commettendo il minimo numero di errori possibile.

Le due classi rappresentate dai cerchi e dalle croci in figura 4.1, nello spazio di input non sono linearmente separabili ma attraverso la funzione Φ i punti vengono mappati in uno spazio in cui diventano linearmente separabili

Supponiamo di mappare i dati iniziali non linearmente separabili in uno spazio di dimensione superiore usando una funzione di mapping $\Phi: R^d \rightarrow H$ in cui essi siano linearmente separabili. In questa situazione l'algoritmo di apprendimento dipende dai dati solamente tramite il prodotto delle loro immagini attraverso Φ in H, cioè tramite funzioni della forma $\Phi(x_i) \cdot \Phi(x_j)$.

Uno spazio di dimensione maggiore causa però seri problemi di calcolo, perché l'algoritmo di apprendimento deve lavorare con vettori di grandi dimensioni. Per ovviare a questo problema si può introdurre una funzione Kernel che restituisce il prodotto delle immagini dei suoi due argomenti, $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$: è possibile evitare di eseguire il prodotto esplicito tra le immagini dei vettori. Una funzione kernel è quindi una funzione che ritorna il valore del prodotto interno fra le immagini di due argomenti:

$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$

Sostituendo $x_i \cdot x_j$ con $K(x_i, x_j)$ ovunque nell'algoritmo, si genera una Support Vector Machine che "lavora" in H e fornisce il risultato nella stessa quantità di tempo che impiegherebbe se lavorasse con i dati originali non mappati.

In pratica l'estensione a superfici di decisione complesse avviene in una maniera abbastanza semplice, mappando la variabile in input x in uno spazio di dimensione maggiore e lavorando poi con una classificazione lineare in questo nuovo spazio.

La funzione Kernel va scelta accuratamente per un tipo di problema: è sempre possibile mappare l'input in uno spazio di dimensione maggiore del numero di punti del training set e produrre un classificatore perfetto; tuttavia questi generalizzerebbe malissimo su dati nuovi, per via dell'overfitting.

Tipi di Kernel comunemente usati sono i seguenti:

Lineare $K(x,y)=x\cdot y$

Polinomiale $K(x,y)= (1+x\cdot y)^d$

Radial Basis function $K(x,y)= \exp(-\gamma||x-y||^2)$

Gaussian Radial Basisfunction $K(x,y)=\exp(- (x-y^2)/2\sigma^2)$

4.2.1 LIBSVM

Libsvm (Chih-Chung Chang et al., 2001; http://www.csie.ntu.edu.tw/~cjlin/libsvm/) è il pacchetto software utilizzato per costruire i modelli basati su Support Vector Machines di classificazione e regressione utili ai fini della mia tesi di dottorato. Oltre ai programmi necessari per la costruzione dei modelli di classificazione e di regressione utilizzati nella fase di training/testing, *libsvm* mette a disposizione un'utility molto importante per determinare due parametri fondamentali per l'ottimizzazione delle fasi di apprendimento/generalizzazione da parte della SVM quando si usa un kernel di tipo RBF. Questi due parametri sono C e γ. Il parametro C (C>0) rappresenta la penalità da adottare affinché la SVM possa permettere o meno errori di classificazione nella fase di training/testing. In altre parole questo parametro permette che il margine di separazione tra due classi di esempi non sia "rigido" ma in un certo qual modo flessibile, cercando di controbilanciare la necessità di ammettere un certo numero di errori di training ("misclassifications") con la necessità di avere un margine netto. Aumentando il valore di C si aumenta il costo degli esempi non classificati correttamente forzando così la creazione di un modello più accurato che potrebbe non essere molto generale. Il parametro gamma, invece, è un coefficiente intrinseco del kernel RBF che determina la larghezza del vettore di supporto.

Capitolo 5

Nuova metodologia per la predizione della variazione di stabilità dei mutanti delle proteine

5.1 Stabilità e proteine mutanti

La comprensione delle regole che determinano la stabilità delle proteine è una delle problematiche che potrebbe aiutare nell'analisi delle strutture proteiche (Daggett e Fesht, 2003). Questi studi sono molto utili per la ingegnerizzazione di nuove proteine. Per queste motivazioni, esistono differenti metodi descritti in letteratura per la determinazione della variazione di stabilità in proteine mutanti. Tali metodi sono basati su differenti approcci ma il loro limite è rappresentato dalla loro difficoltà computazionale. Questo fatto rende proibitivo l'uso di questi metodi per analisi su larga scala (Guerois et al., 2002). Pochi infatti sono gli algoritmi disponibili che possono essere applicati, su un grande insieme di dati, per la determinazione della variazione di stabilità delle proteine (Gills e Rooman, 1997; Funahashi et al., 2001; Guerois et al., 2002; Zhou e Zhou, 2002). I metodi elencati in genere mostrano una buona correlazione tra i dati sperimentali e quelli calcolati, ma il suo valore dipende dalla scelta del set di dati iniziale (Guerois et al., 2002). In generale le metodologie esistenti non hanno una procedura di valutazione che permetta di compararle tra loro. Diventa quindi difficile sapere realmente quale sia l'affidabilità di questi metodi e la validità statistica delle predizioni della variazione di stabilità, indicata con $\Delta\Delta G$ (vedi eq. 5.1), per singolo residuo mutato.

Ai fini pratici, diventa quindi molto più importante riuscire a predire correttamente il segno di ΔΔG. Il valore di tale funzione ci indica se in seguito ad una mutazione la nuova proteina e più stabile (ΔΔG>0) o meno stabile (ΔΔG<0). Un altro problema termodinamicamente importante nella mutagenesi delle proteine sono le condizioni sperimentali, come ad esempio il pH e la temperatura (Gromiha et al., 2000, Bava et al., 2004). Le procedure classiche basate su funzioni energetiche non considerano esplicitamente questi parametri. La possibilità di includere le condizioni sperimentali e la grande quantità di dati disponibili sulle varie proteine mutate (Gromiha et al., 2000) rende ora utilizzabili tecniche di ***machine learning*** (vedi capitolo 4).

In questo capitolo della tesi sarà presentato un progetto sviluppato negli anni di dottorato che ha portato allo realizzazione di un predittore per determinare la variazione di stabilità delle proteine mutate. I dati ottenuti sono stati inviati alla selezione della IV edizione dell' European Conference on Computational Biology (ECCB).

Nel secondo capitolo sono state descritte le principali interazioni che intervengono durante il processo di folding. L'importanza delle interazioni è stata rivelata grazie agli esperimenti di mutagenesi (Alber et al., 1987; Yutani et al., 1987; Wetzel et al., 1988; Shortle et al., 1990; Chen et al., 1993; Takano et al., 1995; Takano et al., 1997; Tissot et al., 1996; Akasako et al., 1997). Questa tecnica consiste nel mutare solo uno o pochi amminoacidi della proteina in esame e analizzare i cambiamenti indotti. Per riuscire ad ottenere un mutante di una proteina dobbiamo mutare il rispettivo DNA codificante dell'organismo, cambiando opportunamente determinati codoni (tripletta di acidi nucleici).

La fase successiva consiste nel determinare il livello di espressione della proteina, che potrebbe non essere espressa o anche se espressa, distrutta dalle proteasi dell'organismo. Una volta ottenuta le proteina e purificata si vanno ad analizzare i possibili cambiamenti conformazionali e le proprietà chimico-fisiche del nuovo polipeptide. E' stato dimostrato che il cambiamento di stabilità indotto dalle mutazioni può avere effetto sulle diverse interazioni che governano il processo di folding. Gli effetti delle mutazioni sull'idrofobicità di alcune proteine sono stati discussi in diversi lavori (Yutani et al., 1987; Matsumura et al., 1988; Shortle et al., 1990; Takano et al., 1995; Takano et al., 1997; Akasako et al., 1997; Xu et al., 1998). La mutagenesi di uno o più residui influisce anche sui ponti salini e i legami ad idrogeno presenti nella struttura terziaria come dimostrato da alcune pubblicazioni (Chen et al., 1993; Tissot et al., 1996). L'interesse per lo studio delle proteine mutate ha dato impulso allo sviluppo di metodiche per la predizione dei cambiamenti di stabilità. Questi metodi sono basati su diversi approcci: modelli atomici accoppiati con potenziali semi-empirici (Bash et al. 1987; Dang et al. 1989; Tidor e Karplus, 1991; Simonson e Brunger, 1992); semplici criteri energetici (Lee e Levitt 1991; van Gusteren e Mark 1992); metodi empirici che tengono conto della variazione di energia libera tra stato nativo e stato denaturato (Miyazawa e Jernigan, 1994); modelli con potenziali derivati da database (Gills e Rooman, 1996; Gills e Rooman, 1997) e dipendente dalla struttura (Topham et al. 1997). L'importanza di questi esperimenti per la determinazione dei meccanismi di folding e l'incremento dei dati a disposizione ha portato alla creazione di una banca dati specifica per i mutanti delle proteine chiamata ProTherm (Gromiha et al., 1999; Gromiha et al., 2000), http://gibk26.bse.kyutech.ac.jp/jouhou/Protherm/protherm.html.

Il grande numero di informazioni a disposizione ci ha spinto ad utilizzare un algoritmo di apprendimento automatico implementato sulla base di una libreria di Support Vector Machines (http://www.csie.ntu.edu.tw/~cjlin/libsvm/) per estrapolare le principali caratteristiche che concorrono al cambiamento della stabilità nelle proteine sulla base di mutazioni puntiformi. In precedenza il gruppo di biocomputing, presso cui ho svolto il mio periodo di dottorato, aveva sviluppato un metodo basato sulle reti neurali (Capriotti et al., 2004) per la predizione del segno della variazione di $\Delta\Delta G$ (+ incremento della stabilità, - decrescita della stabilità) a partire da informazioni di tipo strutturale. Questa predizione è sufficiente per valutare l'effetto che una mutazione proteica puntiforme ha sulla stabilità. Tuttavia il metodo precedente era limitato dal fatto che era indispensabile conoscere la struttura della proteina, invece in questo capitolo verrà presentato una nuova metodica basata su Support Vector Machines (SVM) per la predizione della variazione di stabilità, in questo caso sia il segno che il valore di $\Delta\Delta G$, a partire solo dall' informazione in sequenza. Questo è particolarmente interessante in quanto permette di eseguire predizioni su larga scala e per di più è possibile valutare se una mutazione proteica puntiforme possa essere messa in relazione a malattie legate al misfolding (Dobson, 2003).

5.2 Data Sets

Il data set utilizzato per addestrare la SVM è stato estratto dalla banca dati ProTherm (Dicembre 2004) in base alle seguenti specifiche:

i) il valore di $\Delta\Delta G$ è determinato sperimentalmente

ii) i dati sono relativi a mutazioni proteiche puntiformi

Dopo questa fase di filtraggio dei dati, l'insieme finale di dati consiste di 2087 mutazioni proteiche puntiformi relative a 65 sequenze proteiche. Al fine di testare il predittore rispetto al compito di predire se malattie indotte da una mutazione proteica puntiforme possano destabilizzare il ripiegamento proteico, ho raccolto un certo numero di mutazioni per due proteine molto ben caratterizzate sperimentalmente: la proteina prionica umana (PRIO_HUMAN) e la transtiretina (TTHY_HUMAN). In pratica ho recuperato tutte le mutazioni correlate a malattie di cui è noto l'effetto destabilizzante sul folding e per le quali sono disponibili dati termodinamici in letteratura e di cui sia depositata la struttura nel Protein Data Bank. Alla fine di questa fase di recupero e filtraggio dei dati, ho costruito una lista di mutazioni, per le due proteine di cui sopra, che risultano essere associate a malattie come le sindromi di Creutzfeldt-Jacob e Gerstmann-Strussler ed amiloidosi (vedi tabella 5.1). Questi dati sono utlizzati come "blind test" per il predittore.

5.3 Caratteristiche della SVM

Il progetto sviluppato consiste nell'implementazione di una Support Vector Machines (SVM), utilizzando diverse funzioni kernel, che svolge due diversi compiti: la predizione del segno della variazione del ΔG di stabilità dei mutanti delle proteine (ΔΔG) e la predizione del valore stesso di ΔΔG. Il primo compito è un semplice problema di classificazione tra due possibili classi, cioè la classe che rappresenta un aumento di stabilità e una che riguarda una diminuzione di stabilità in base alla mutazione proteica puntiforme presa in considerazione. Il secondo compito, riguardante la predizione del valore di ΔΔG, è un problema di regressione e di fitting. Il valore di ΔΔG può essere calcolato come

$$\Delta\Delta G = \Delta G_{mut} - \Delta G_{wt} = (G_{mut}^{(u)} - G_{mut}^{(f)}) - (G_{wt}^{(u)} - G_{wt}^{(f)}) \qquad (5.1)$$

dove G è l'energia libera, *mut* indica la proteina mutata, *wt* (wild-type) la proteina iniziale, *u* e *f* rispettivamente lo stato denaturato e quello nativo. In più è stato confrontato questo nuovo sistema di SVM con il precedente predittore basato su reti neurali, in entrambi i casi la struttura dell'input è la medesima. L'architettura della rete neurale consiste di uno strato di input, due nodi nascosti ed un nodo di uscita (vedi fig 5.1) che discrimina tra un aumento di stabilità proteica ($\Delta\Delta G \geq 0$, output desiderato settato uguale ad 1) o una diminuzione della stabilità ($\Delta\Delta G < 0$, output desiderato settato uguale ad 0); la soglia di decisione per assegnare una mutazione ad una delle due classi è 0.5. La stessa strategia di assegnazione delle classi e della soglia di decisione è utilizzata per la SVM (vedi fig. 5.2).

Il nostro metodo è quindi in grado di predire se dopo la mutazione di un singolo amminoacido la proteina che ne scaturisce sia più o meno stabile della proteina iniziale. L'idea che ci ha guidato nella codifica delle informazioni a nostra disposizione è che la mutazione di un singolo amminoacido influisce in maniera diretta solo su un certo intorno in sequenza entro una finestra di una certa lunghezza *w* centrata sul residuo mutato. Sono state provate diverse finestre di lunghezza variabile, compresa tra 7 e 23 residui, in modo tale da avere sempre un uguale numero pari di residui sia a destra che a sinistra del residuo mutato. Oltre alle caratteristiche in sequenza della proteina, la variazione di stabilità dipende dalle condizioni sperimentali (pH e temperatura). A seguito di un'operazione di ottimizzazione abbiamo progettato un vettore di input con 42 valori che rappresentano i seguenti contributi :

i) i primi venti valori di input corrispondono ognuno ad un dei possibili 20 amminoacidi. Il vettore è uguale a 0 per tutte le posizioni tranne quella corrispondente dell'amminoacido sostituito, posta uguale a -1, e quella del nuovo amminoacido posta uguale a 1;

ii) i secondi venti valori corrispondono ai 20 amminoacidi essenziali e ogni posizione contiene il numero di residui di quella specie che si trovano entro una finestra di specifica lunghezza *w* centrata sull'amminoacido sostituito;

iii) gli ultimi due valori, che prendono in input la temperatura e il pH, descrivono le condizioni sperimentali in cui viene misurata la variazione di energia libera.

I vari tipi di kernel testati sono:

Lineare $K(x_i,x_j) = x_i T x_j$;

Polinomiale $K(x_i,x_j) = (G\, x_i T x_j + r)^d$;

Sigmoidale $K(x_i,x_j) = \tanh(G\, xi T x_j + r)$;

RBF $K(x_i,x_j) = \exp (-G \| x_i - x_j \|^2)$;

Mediante la rete neurale è possibile eseguire solo il compito di classificazione, mentre con l'approccio basato su SVM possiamo sia determinare il segno del ΔΔG che il suo valore. Sia i risultati ottenuti con la rete neurale che con la SVM sono valutati mediante una procedura di cross-validation sul data set (vedi capitolo 4). I dati sono stati ripartiti in 20 gruppi di cross-validation in modo tale da riflettere la stessa distribuzione degli esempi positivi e negativi del data set intero; per di più le stesse mutazioni, riportate a diverse condizioni sperimentali, sono state raggruppate nello stesso set di cross-validation al fine di prevenire una sovrastima dei risultati. Per ogni metodo testato sono stati utilizzati gli stessi set di cross-validation in modo che i vari metodi possono essere confrontati direttamente dal momento che la fase di test è stata condotta nelle medesime condizioni.

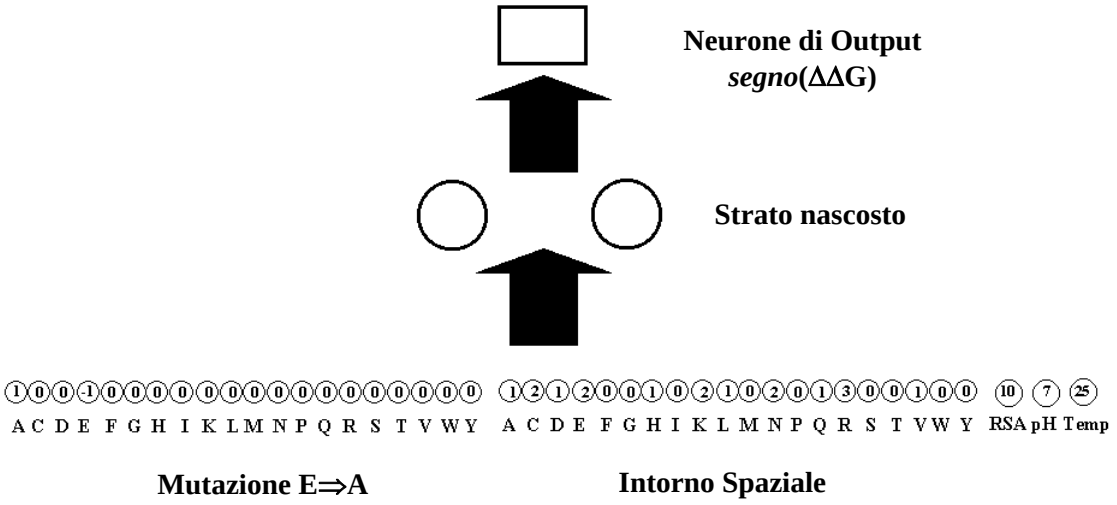


Figura 5.1 Codifica dell'input della rete neurale per la predizione del segno di ΔΔG delle proteine mutate. L'input è composto da: il tipo di mutazione (vettore dei primi 20 elementi); l'intorno spaziale (vettore dei successivi 20 elementi); il pH e la temperatura (Temp). Il parametro RSA (area relativa accessibile al solvente) si riferisce all'input del metodo basato sulla 3D, per la sequenza non viene calcolato ed utilizzato

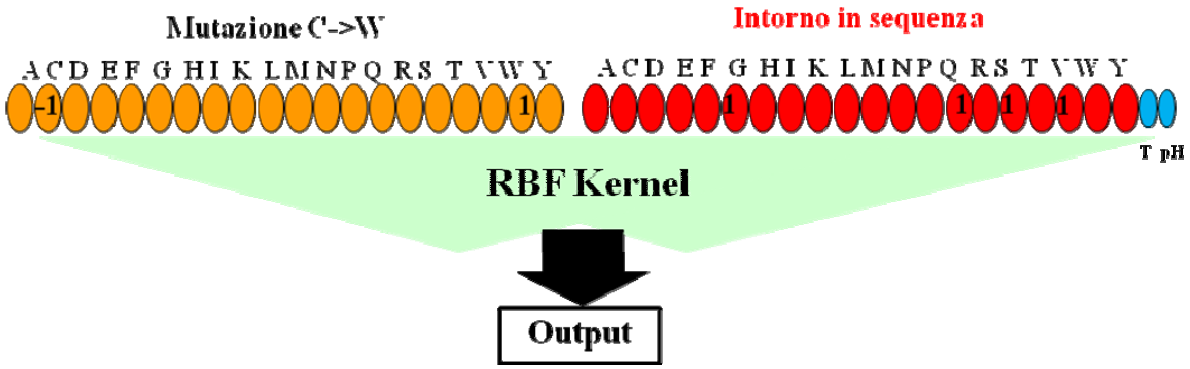


Figura 5.2 Codifica dell'input della SVM per la predizione del segno e del valore di ΔΔG delle proteine mutate. L'input è composto da: il tipo di mutazione (vettore dei primi 20 elementi); l'intorno in sequenza (vettore dei successivi 20 elementi); il pH e la temperatura (T) con valori di default 7 e 25 °C rispettivamente

5.4 Risultati e discussione

In precedenza il metodo basato su reti neurali assegnava correttamente più dell'80% delle mutazioni del data set contenente 1615 mutazioni di cui era nota la struttura tridimensionale (Capriotti et al., 2004). In questo lavoro ci siamo focalizzati sulla sequenza proteica al fine di stabilire se una mutazione puntiforme lungo la sequenza possa far aumentare o diminuire la stabilita della proteina senza la necessita di conoscerne la struttura 3D. I risultati ottenuti con diversi predittori presi in esame sono riportati in tabella 5.1. È interessante notare che sebbene l'informazione sfruttata venga estratta solo dalla sequenza, la SVM con kernel RBF raggiunge un'accuratezza generale del 77% con un coefficiente di correlazione pari al 42%. Questo risultato indica che un pezzo di informazione rilevante della stabilità del folding proteico può essere rintracciata a livello dei residui vicini all'amminoacido interessato nell'evento mutazionale.

Tabella 5.1 Confronto tra NN e SVM

Metodo	Q2	P(+)	Q(+)	P(-)	Q(-)	C
NeuralNet	0.73	0.39	0.56	0.77	0.87	0.30
SVM-Linear	0.67	0.41	0.28	0.73	0.84	0.13
SVM-Polynomial	0.73	0.58	0.38	0.77	0.88	0.30
SVM-Sigmoid	0.68	0.44	0.27	0.73	0.85	0.15
SVM-RBF	0.77	0.69	0.44	0.79	0.91	0.41

+ and – : rappresentano le classi "aumento stabilità" e "diminuzione stabilità" rispettivamente; per gli indici si veda l'appendice A. la lunghezza della finestra è pari a 19 per ambo i metodi di machine learning

Il kernel RBF e' quello che ha le migliori prestazioni per quanto concerne la classificazione delle mutazioni proteiche puntiformi. Questo indica che il kernel RBF riesce a catturare in maniera efficace le proprietà sottese dal residuo mutato e dal suo intorno locale che possono determinare la stabilita/instabilità della proteina in relazione anche alle condizioni di pH e temperatura. Nella tabella 5.2 si vede che le migliori prestazioni sono ottenute quando utilizziamo una finestra lunga 19 residui. Per di piu abbiamo testato anche l'informazione dell'intorno contenuta in una finestra infinita in modo da includere l'effetto dato dall'intera sequenza.

Tabella 5.2 Confronto tra finestre di diversa lunghezza usando un kernel RBF

Window	Q2	P(+)	Q(+)	P(-)	Q(-)	C
7	0.74	0.58	0.36	0.77	0.89	0.30
11	0.73	0.85	0.12	0.73	0.99	0.25
15	0.76	0.64	0.38	0.78	0.91	0.35
19	0.77	0.69	0.44	0.79	0.91	0.41
23	0.76	0.64	0.44	0.79	0.90	0.38
whole sequence	0.73	0.59	0.32	0.76	0.90	0.28

Per le notazioni si veda la tabella 5.1

Come si vede l'accuratezza diluisce e questo indica che la composizione dell'intera sequenza non e' così specifica come quella dell'intorno locale nel determinare il segno della variazione di stabilità. L'analisi dell'accuratezza della SVM in funzione delle caratteristiche chimico-fisico delle mutazioni (tabella 5.3), mette in evidenza come la variazione di stabilità proteica che coinvolge le mutazione del tipo carico/carico, polare/carico e carico/apolare, hanno delle prestazioni più basse di quelle che coinvolgono le coppie apolare/apolare; questo suggerisce che per i residui carichi e polari presenti sulla superficie o per i residui carichi che formano ponti salini sono necessarie più informazioni oltre quella data dall'intorno locale in sequenza al fine di migliorare la capacità predittiva del metodo.

Tabella 5.3 Q2 in funzione del tipo di residuo mutato

Native*new*	Charged	Polar	Apolar
Charged	0.65 (4%)	0.72 (7%)	0.69 (12%)
Polar	0.57 (5%)	0.76 (5%)	0.77 (13%)
Apolar	0.80 (5%)	0.88 (9%)	0.80 (40%)

Ogni cella rappresenta un particolare tipo di mutazione in accordo alla sua proprietà chimico-fisiche. Le righe si riferiscono per il residuo wild-type, mentre le colonne si riferiscono al nuovo residuo nelle proteine mutanti (*new*). Tra parentesi è riportata la frazione relativa dei vari tipi di residui nel data set (2087).

L'accuratezza generale,Q2, in funzione del "Reliability Index" è riportata in figura 5.3. Il valore di Rel in relazione all'accuratezza delle predizioni può essere utile per selezionare quelle mutazioni che più probabilmente influenzano la stabilità proteica, questo è utile in un'ottica di mutagenesi mirata a livello genomico.

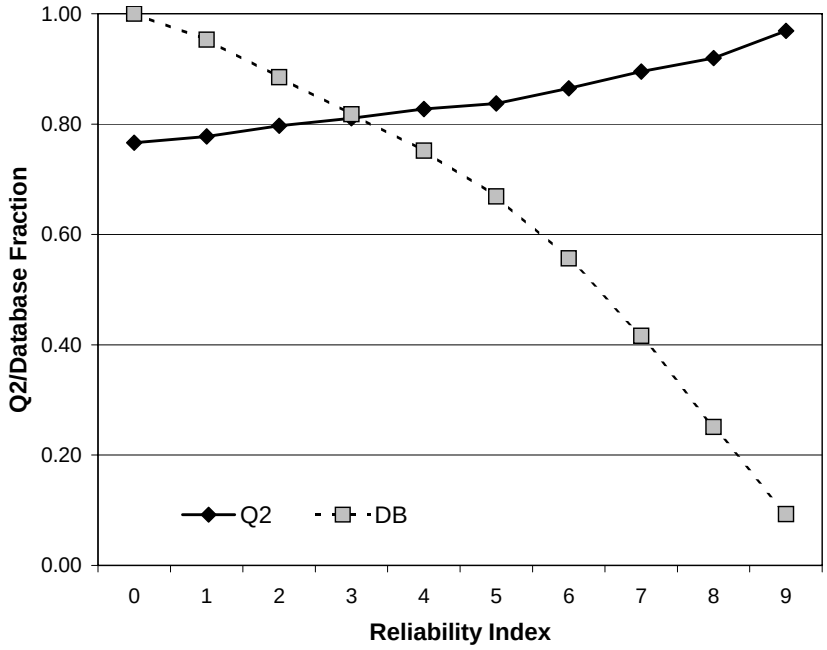


Fig ura 5.3 Q2 della SVM-RBF in funzione del " reliability index" (*Rel*) delle predizioni (appendice A). DB è la frazione del data set con valori di Rel maggiori od uguali ad una certa soglia.

In casi specifici, non solo il segno della variazione del ΔΔG ma anche il suo effettivo valore, può essere necessario per selezionare il tipo di mutazione desiderata. Questa possibilità è ottenuta utilizzando una SVM che svolge un compito di regressione utilizzando sempre un kernel RBF. In figura 5.4 vediamo il grafico di regressione tra il valore predetto ed atteso della variazione di ΔΔG. Le predizioni sono sempre ottenute utilizzando 20 set di cross-validation. Il valore R per la regressione e' uguale a 0.62 con un errore quadratico medio pari a 1.45 Kcal/mole. Questa è stato il primo lavoro che permette di calcolare una correlazione così elevata partendo solo dall'informazione in sequenza.

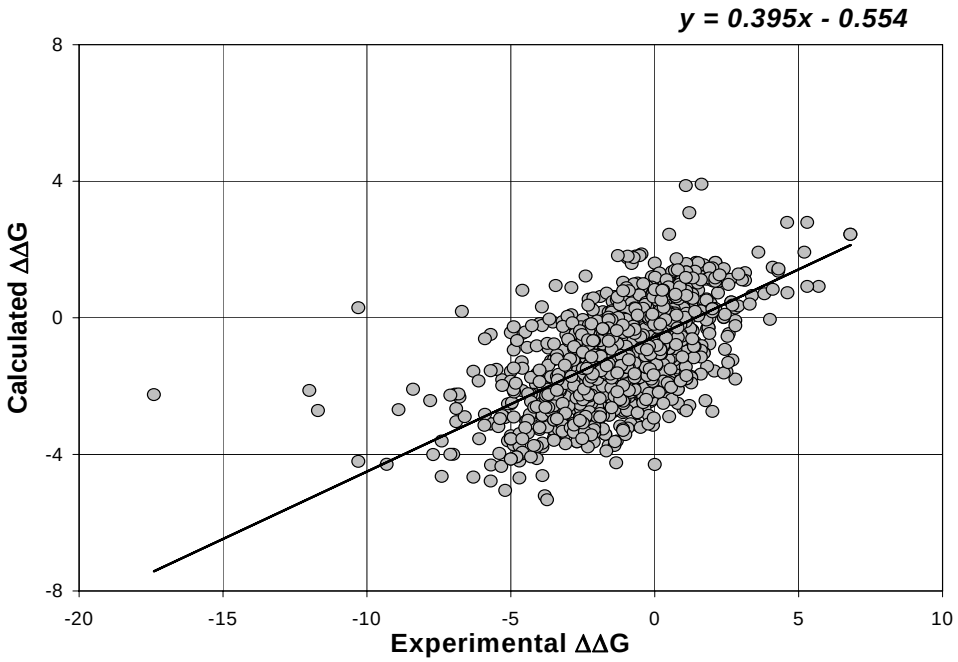


Figura 5.4. Regressione tra I valori predetti e sperimentali della variazione ΔΔG (R=0.60, RMSE= 1.47 (0.395*x-0.554) Kcal/mol).

Numerose evidenze, negli ultimi anni, si sono accumulate in merito a mutazioni proteiche puntiformi coinvolte nell'insorgenza di malattie e che influenzano negativamente il corretto ripiegamento proteico (Wang and Moult, 2001; Wang and Moult, 2003; Dobson 2003; Selkoe 2003) . Un' applicazione interessante del nostro metodo e' quella di poterlo applicare per la predizione della variazione della stabilità proteica quando sono note mutazioni correlate a malattie legate al folding. In alcuni casi le proteine che hanno problemi legati al folding vengono eliminate dal complesso di degradazione cellulare, il *proteosoma*. Pertanto le malattie legate al misfolding di queste proteine dipendono dall'assenza di tali elementi funzionali; esempi di questa classe di malattie sono: la *Fibrosi Cistica*, la *Sindrome di Marfan* e la *Retinite Pigmentosa*. Tuttavia molte malattie legate al misfolding sono caratterizzate dal deposito di aggregati insolubili all'interno della cellula, chiamati *amiloidi*. Malattie causate dal deposito di aggregati proteici sono: l' *Alzheimer*, il *Parkinson*, il morbo di *Creutzfeldt-Jakob* e vari tipi di *Amiloidosi*. Il tratto comune di tutte queste malattie è il deposito delle fibre amiloidee che sono strutturalmente molto simili tra di loro anche se derivano da proteine strutturalmente diverse. È ancora materia di discussione se le fibre amiloidee siano la causa di queste malattie o ne siano solo un sintomo e se ne siano una causa non è chiaro se la patologia sia legata al mal funzionamento della proteina coinvolta nella formazione delle fibre o alla tossicità di tali aggregati. In alcuni casi il misfolding è direttamente legato a mutazioni che destabilizzano la struttura proteica; più queste mutazioni sono destabilizzanti prima si verifica l'insorgenza di malattie, come avviene ad esempio per la transtiretina che determina la FAP ("familial Amyloidotic Polyneuropathy"). È da tenere in considerazione che queste mutazioni destabilizzanti molto spesso non alterano drammaticamente la struttura 3D o la funzione della proteina, ma semplicemente favoriscono conformazioni che più facilmente

siano prone a formare aggregati proteici; in altri casi le mutazioni possono promuovere direttamente la formazione degli amiloidi. Infine la formazione di aggregati amiloidei può essere dovuta solo all'invecchiamento e non avere un'associazione genetica chiara come ad esempio nell'Alzheimer "late onset". Nella tabella 5.4 sono riportate le predizioni dei dati termodinamici per 20 mutazioni relative alla proteina prionica umana ed alla transtiretina e messe a confronto con i valori sperimentali di $\Delta\Delta G$, quando disponibili, o con eventuali dati riguardanti i cambiamenti conformazionali quando sono note le strutture3D. Il segno della variazione di stabilità è predetto correttamente a parte due casi, con un coefficiente di correlazione di 0.42. pertanto le prestazioni su questo blind test sono simili a quelle del data set di training/testing. Per di più è interessante notare che se ci concentriamo solo sul sotto set di variazioni di $\Delta\Delta G \geq 0.5$ Kcal/mole, tutte le mutazioni correlate a malattie, sono predette come destabilizzanti tranne che in un caso. I risultati di questo test sono in accordo con l'idea generale che difetti nel corretto processo di ripiegamento proteico possono essere le cause di alcune malattie umane. Dunque è possibile applicare il nostro metodo per la predizione di mutazioni puntiformi in relazione a malattie legate all'instabilità derivante da un non corretto funzionamento del processo di folding.

Protein	Mutation	Effect	Predicted Stability change	RI	Experimental ΔΔG (Kcal/mol)
Human prion (PRIO_HUMAN)					
	P102L	GSD	Increase	2	0.2±0.6
	M129V	Polymorphism	Decrease	6	-0.3±0.5
	V180I	**GSD**	**Decrease**	**2**	**-0.5±0.4**
	T183A	**CJD**	**Decrease**	**6**	**-4.6±0.7**
	T190V	Polymorphism	Decrease	2	0.2±0.6
	F198S	**GSD**	**Decrease**	**7**	**-2.5±0.4**
	E200K	CJD	Decrease	5	-0.1±0.6
	R208H	**CJD**	**Decrease**	**7**	**-1.4±0.6**
	V210I	CJD	Decrease	2	-0.3±0.6
	Q217R	**GSD**	**Increase**	**1**	**-2.1±0.4**
	M166V	Polymorphism	Decrease	6	S.C.(1E1J)
	S170N	Polymorphism	Increase	1	S.C.(1E1P)
	R220K	Polymorphism	Decrease	7	S.C.(1FKC)
Transthyretin (TTHY_HUMAN)					
	V50M	**Amyloidosis**	**Decrease**	**6**	**-2.2±2.4**
	L75P	**Amyloidosis**	**Decrease**	**5**	**-1.5±2.3**
	T139M	Unclassified	Decrease	0	-0.1±2.8
	T80A	Amyloidosis	Decrease	6	S.C.(1TSH)
	S97Y	Amyloidosis	Increase	2	S.C.(2TRY)
	Y134C	Amyloidosis	Increase	0	S.C.(1IIK)
	V142I	Unclassified	Decrease	2	S.C.(1TTR)

GSD=Gerstmann-Straussler disease. CJD=Creutzfeldt-Jakob disease. RI= reliability index (vedi appendice A). S.C. = cambiamento conformazionale nella struttura tra proteina nativa e mutante (1QLX, proteina prionica e 1BM7 transtiterina) le strutture 3D mutate sono riportate tra parentesi. **In grassetto** sono indicate il sotto set di mutazioni con valori di ΔΔG maggiori di 0.5 Kcal/mol.

Capitolo 6

Nuova metodologia per la predizione dell'insorgenza di malattie genetiche umane dovute a mutazioni proteiche puntiformi.

6.1 Mutazioni puntiformi e malattie

Nel capitolo precedente abbiamo trattato del problema della variazione della stabilità proteica ($\Delta\Delta G$) ed abbiamo visto che per situazioni in cui si conoscono dati relativi al misfolding che determinano poi l'insorgenza di malattie umane, il modello della stabilità adottato riesce a predire in maniera efficace quali mutazioni siano responsabili di tali patologie. Sfortunatamente il modello della stabilità si può applicare ad un numero limitato di casi possibili in quanto i dati presenti in letteratura che mettono in evidenza la relazione tra misfolding e malattie non sono in numero tale da giustificare un approccio basato su apprendimento automatico. Pertanto abbiamo esteso il modello della stabilità in maniera che potesse essere più generale per quello che riguarda la predizione dell'insorgenza di malattie genetiche umane dovute a mutazioni proteiche puntiformi. Questo modello, che ho chiamato *modello funzionale,* prende in esame le mutazioni che sono annotate come essere responsabili di malattie genetiche umane o come polimorfismi neutri che si trovano naturalmente nella popolazione. Tale modello ha il vantaggio di non essere necessariamente legato a fenomeni di misfloding che influenzano la stabilità proteica e pertanto ha una valenza ed un'applicabilità molto più vasta.

Come detto nel primo capitolo primo i polimorfismi di singolo nucleotide (SNPs) sono la classe di variazione genetica più comune nell'uomo e rappresentano circa il 90% delle differenze che troviamo nelle sequenze di individui appartenenti alla stessa specie (Collins et al., 1998). È stato stimato che gli SNPs si trovano ogni 1000 paia di basi lungo le sequenze del genoma. L'importanza degli SNPs negli studi genetici è dovuta a diverse ragioni, tra cui:

i) Essendo ereditati da una generazione alla successiva, gli SNPs, caratterizzano l'evoluzione della specie umana (Goldstein and Cavalleri, 2005)

ii) Studiando la distribuzione degli SNPs nei diversi gruppi della popolazione umana mondiale, può senza dubbio portare a formulare importanti considerazioni sulla storia della nostra specie (Barbujani and Goldstein, 2004; Edmonds et al., 2004)

iii) Gli SNPs sono responsabili di malattie genetiche (and Henikoff, 2002; Bell, 2004)

Le nuove tecniche sperimentali per l'identificazione su larga scala degli SNPs nella popolazione (Wang et al., 1998), ha fatto crescere esponenzialmente la quantità di dati presente nella banca dati dbSNP (http://www.ncbi.nlm.nih.gov/SNP) (Sherry et al., 2001) che nella versione 126 (dbSNP 126) conteneva circa sei milioni di SNPs validati. Di recente numerose banche dati, diversi server e tools sono stati sviluppati al fine di studiare gli effetti che gli SNPs hanno sull'uomo (Wang and Moult, 2001; Ramensky et al., 2002; Riva and Kohane, 2002; Ng and Henikoff, 2003; Stenson et al., 2003; Conde et al., 2004 Reumers et al., 2005; Karchin et al. 2005; Yue and Moult 2006). Un aspetto importante è capire quale delle varianti genetiche che si trovano nella popolazione siano effettivamente correlate all'insorgenza delle malattie genetiche umane.

Una regola generalmente accettata è quella che le mutazioni che avvengono nelle regioni geniche codificanti hanno un impatto maggiore sulla funzionalità del prodotto proteico. Tuttavia non va dimenticato che gli SNPs presenti in regioni regolatorie importanti, come promotori ed enhancer, possono alterare in maniera negativa i livelli di espressione genica, il che si può riflettere in un danno irreparabile per l'organismo. In questo capitolo prenderemo in considerazione quella classe di SNPs che si trovano nelle regioni geniche codificanti e che causano mutazioni puntiformi nella sequenza del prodotto proteico, gli SNPs non-sinonimi (nsSNPs).

6.2 Data Sets

I dati riguardanti le mutazoni proteiche puntiformi sono stati estratti dalla banca dati SWISS-PROT (versione 48, Dicembre 2005) (Boeckmann et al., 2003). La classificazione di polimorfismi neutri o con effetto deleterio derivano dalla lista delle varianti geniche contenuta in SWISS-PROT di cui è riportato anche il link alla banca dati OMIM (Online Mendelian Inheritance in Man, http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM) quando la mutazione ha un effetto patologico. Sono stati costruiti tre data sets:

i) il primo utilizzato nella fase di training/testing di una SVM basata solo sull'informazione in sequenza(HumVar data set)

ii) il secondo utilizzato nella fase di training/testing di una SVM basata su informazioni derivanti dal profilo di sequenza (HumVarProf)

iii) il terzo, contenente dati non "visti" dal sistema, utilizzato per testare la robustezza del metodo (NewHumVar)

L'intero data set è stato derivato dalla banca dati secondo i seguenti criteri:

i) le proteine sono esclusivamente di *Homo Sapiens*

ii) le mutazioni sono correlate a malattie o polimorfismi neutri (i casi "Unclassified " non sono considerati)

iii) i dati sono relativi a mutazioni proteiche puntiformi (delezioni ed inserzioni non sono presi in considerazione)

Dopo una fase di filtraggio il data set consiste di 21185 mutazioni di cui 12944 sono coinvolte nell'insorgenza di malattie genetiche umane, mentre 8241 sono classificate come polimorfismi neutri. Queste mutazioni si riferiscono a 3587 proteine umane, che sono state raggruppate in "clusters" utilizzando il programma ***blastclust*** presente nella suite di programmi di BLAST (Altschul et al., 1997) (vedi capitolo 3). In seguito ogni sequenza del data set è stata allineata con le sequenze presenti nella banca dati nr95 (versione di Giugno 2005), vale a dire che dalla banca dati nr sono state eliminate tutte quelle sequenze con una percentuale di identità superiore al 95% mediante l'utilizzo del programma *cd-hit*, disponibile alla pagina web http://bioinformatics.org/cd-hit (Li et al., 2001). Considerando ogni profilo di sequenza, sono state selezionate solo quelle mutazioni dell'intero data set per cui la frequenza del residuo wild-type e del nuovo residuo sia diversa da 0. Questo sottoinsieme è quello chiamato HumVarProf e comprende 8718 mutazioni di cui 3852 sono classificate come "Disease" mentre 4866 sono classificate come "Neutral". Il terzo set di nsSNPs comprende mutazioni proteiche puntiformi appartenenti a sequenze umane che sono riportate nella versione di Giugno 2006 della banca dati SWISS-PROT. Queste mutazioni derivano da nuove sequenze che non appartengono a nessun gruppo di omologia del precedente data set HumVar su cui è stata condotta la fase di apprendimento. Il data set derivante consiste di 935 mutazioni, di cui 149 sono classificate come "Disease" e 786 sono etichettate come "Neutral", appartenenti a 469 sequenze diverse.

6.3 I predittori

Il nostro obbiettivo è quello di predire se una data mutazione proteica puntiforme, generata da uno SNP non-sinonimo, possa essere classificata come “Disease” o come “Neutral”. Per affrontare tale problema sono stati implementati diversi metodi : un predittore di base (ProbMeth) preso come riferimento da superare, una SVM (SVM-Sequence) basata solo su informazioni derivanti dall’intorno sequenziale della mutazione presa in considerazione ed una SVM (SVM-Profile) che prende in input informazioni di tipo evolutivo derivanti dal profilo. Infine la SVM-Sequence e la SVM-Profile sono state accoppiate in un sistema unico mediante un albero decisionale (HybridMeth) che permette di adottare una delle due SVM in base alla presenza o meno dei parametri derivati dal profilo per la mutazione in esame (vedi fig. 6.1)

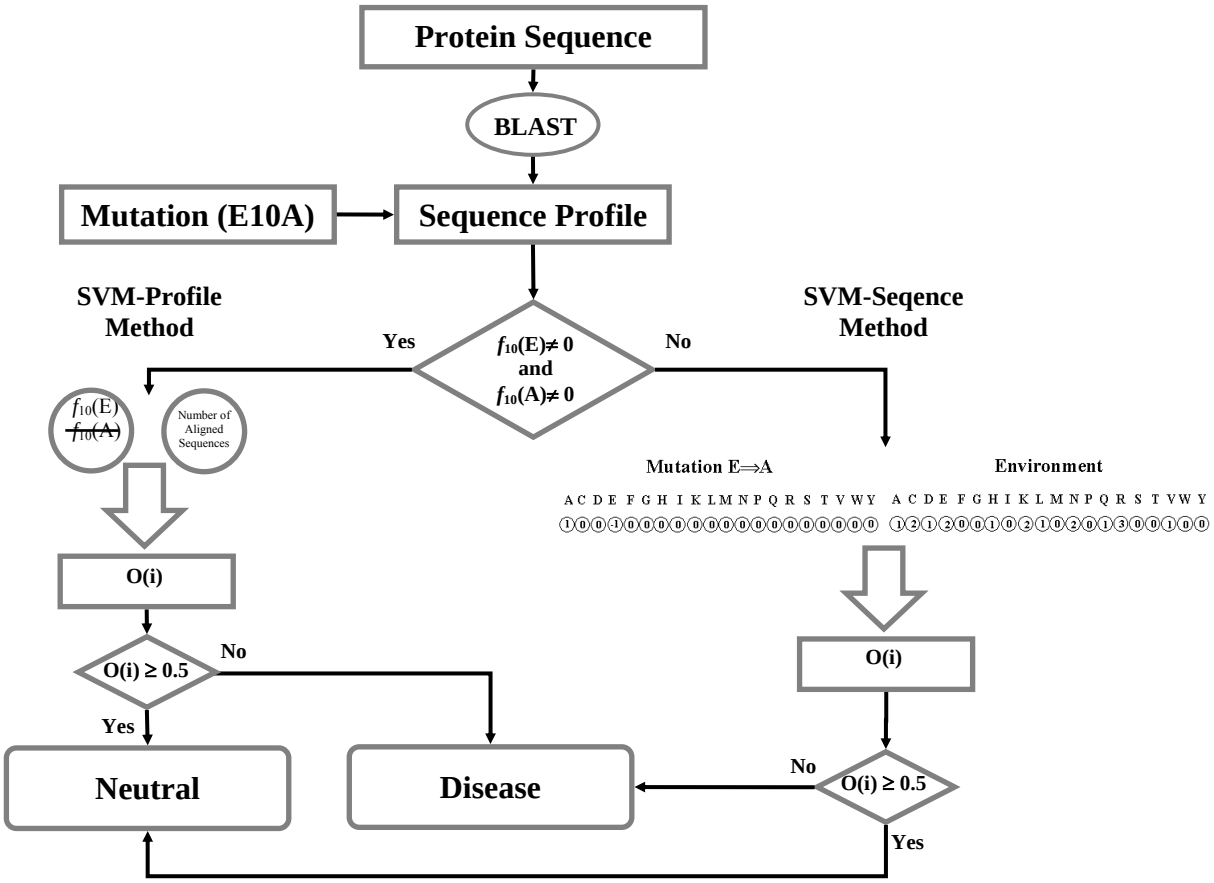


Figura 6.1 Diagramma di flusso del metodo ibrido (HybridMeth). Per una data sequenza proteica il metodo predice se una certa mutazione (es il residuo E in posizione 10 è mutato in A) possa essere correlata ad una malattia genetica o meno. Come prima cosa viene costruito il profilo della sequenza mediante BLAST. Il secondo step, viene valutato il profilo nella posizione mutata. Se $f_{10}(E)\neq 0$ and $f_{10}(A) \neq 0$, la predizione viene effettuata seguendo il ramo sinistro del diagramma, grazie al metodo SVM-Profile , prendendo in input il rapporto $f_{10}(E)/f_{10}(A)$ ed il numero di sequenze allineate nella posizione in esame. Altrimenti se $f_{10}(E)=0$ e/o $f_{10}(A)=0$ si utilizza la SVM-Sequence per effettuare le predizioni.

6.3.1 Il metodo probabilistico (ProbMeth)

Il predittore di base è costruito considerando il numero di occorrenza delle mutazioni (coppie di residui wild-type/mutato) nel nostro data set. Per la classe "Disease" (D) e la classe "Neutral" (N), sono state derivate le probabilità $M(D)_{i,j}$ ed $M(N)_{i,j}$ in forma di matrici 20x20. Il generico elemento della matrice, $M_{i,j}$, rappresenta l'occorrenza della mutazione per il residuo *i* nel residuo *j*, calcolata come:

$$M_{i,j} = f(i,j)/[f(i)\cdot f(j)] \qquad (6.1)$$

Dove *f(i,j)* è la frequenza di occorrenza della mutazione del residuo *i* nel residuo *j*; *f(i)* è la frequenza di occorrenza del residuo *i* nel data set ed *f(j)* è la frequenza di occorrenza di ogni mutazione corrispondente al residuo *j* nel data set. Calcolando il valore massimo tra $M(D)_{i,j}$ e $M(N)_{i,j}$ per una certa mutazione del residuo i nel residuo j, viene predetto se la mutazione possa essere classificata come "Disease" o come "Neutral". In altre parole se $M(D)_{i,j} > M(N)_{i,j}$ la mutazione proteica puntiforme è classificata come appartenente alla classe D , mentre se $M(D)_{i,j} \leq M(N)_{i,j}$ allora la mutazione è classificata come polimorfismo neutro.

6.3.2 Il metodo SVM basato sull'informazione in sequenza (SVM-Sequence)

Questa SVM classifica le mutazioni come appartenenti alla classe D (output desiderato uguale a 0) oppure alla classe N (output desiderato uguale a 1), con la soglia di decisione posta uguale a 0.5, in base solo ad informazioni estratte dalla sequenza proteica. Il vettore di input consiste di 40 elementi; i primi 20 elementi definiscono la mutazione assegnando al residuo wild-type il valore di -1, al residuo mutato il valore di 1 e mantenendo a 0 tutti gli altri elementi; gli altri 20 elementi codificano l'intorno della mutazione. È stata utilizzata una finestra di lunghezza pari a 19, centrata sulla posizione interessata dalla mutazione, in modo da avere un numero pari di residui sia a destra che a sinistra. I 20 elementi del vettore riguardante l'intorno della mutazione, corrispondenti quindi ai 20 amminoacidi, rappresentano il numero di volte che un residuo compare all'interno della finestra utilizzata (vedi fig. 6.2).

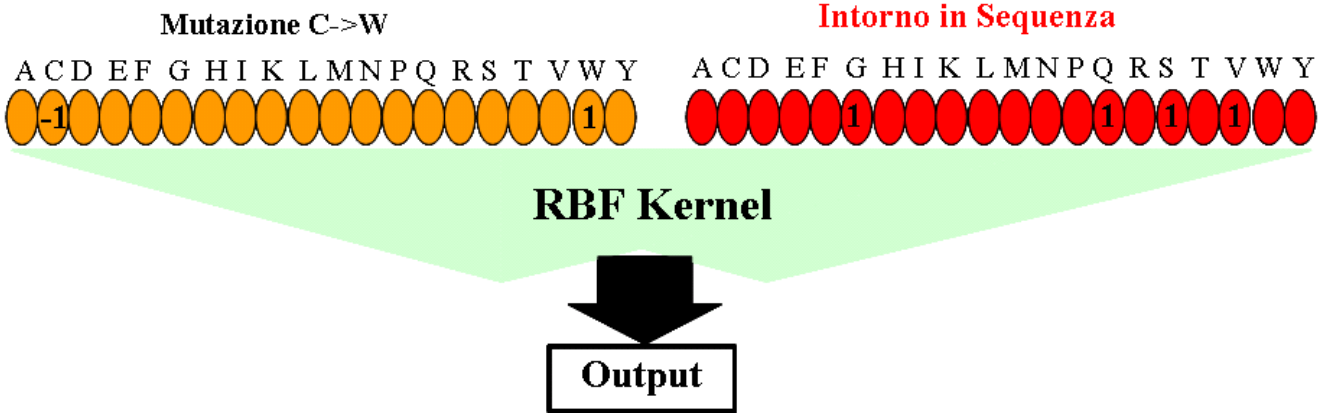


Figura 6.2 Codifica dell'input della SVM-Sequence. L'input è composto da: il tipo di mutazione (vettore dei primi 20 elementi), l'intorno in sequenza (vettore dei successivi 20 elementi)

La SVM implementata (derivante dal pacchetto software LIBSVM, http://www.csie.ntu.edu.tw/~cjlin) utilizza un kernel RBF ($K(x_i,x_j)=\exp(-G\ ||x_i - x_j||^2)$).

6.3.3 Il metodo SVM basato sull'informazione evolutiva (SVM-Profile)

Il secondo metodo basato su SVM classifica la mutazioni proteiche puntiformi prendendo come input un vettore composto da due soli elementi derivanti dal profilo di sequenza. Questo viene costruito dall'output del programma BLAST (Altschul et al., 1997) (vedi capitolo 3) che cerca le sequenze simili a quelle del data set, nella banca dati nr95 (con soglia di E-value pari a 10^{-9} e numero di run=1). Il primo elemento del vettore di input rappresenta il rapporto tra la frequenza di occorrenza nel profilo del residuo wild-type e la frequenza di occorrenza del residuo mutato, mentre il secondo elemento si riferisce al numero di sequenze allineate nella posizione della mutazione. Il software e il kernel sono gli stessi menzionati in precedenza.

6.3.4 Il metodo ibrido (HybridMeth)

Il metodo ibrido è basato su una strategia ad albero decisionale che sfrutta le SVM (SVM-Sequence ed SVM-Profile) descritte sopra, organizzate in modo da sfruttare o l'una o l'altra in base ad un punto di decisione (vedi fig 6.1). Il metodo ibrido consta dei seguenti passi:

i) per una data proteina, il suo profilo è costruito in accordo alla procedura descritta in precedenza. Fatto ciò si valutano sia le frequenze di occorenza del residuo wild-type[f_k(wt)] e mutato [f_k(mut)] nel profilo nella posizione *k*

ii) quando f_k(wt) e f_k(mut) sono entrambi diversi da 0, il rapporto f_k(wt)/f_k(mut) viene calcolato ed unitamente al numero di sequenze allineate nella posizione *k* viene fornito alla SVM-Profile, addestrata sull'insieme HumVarProf

iii) quando non è possibile determinare il profilo per una data sequenza oppure dal profilo, nella posizione *k*, f_k(wt)=0 o f_k(mut)=0, allora la predizione viene effettuata mediante la SVM-Sequence.

Tutti i risultati ottenuti con i metodi basati su SVM sono stati valutati utilizzando una procedura di cross-validation (vedi capitolo 4), dividendo sia l'insieme HumVar che HumVarProf in 20 set di cross-validation in modo che la distribuzione delle due classi al loro interno riflettesse quella del set interno. Per di più tutte le proteine presenti nel set HumVar ed HumVarProf sono state raggruppate in insiemi di omologia in base al programma *blastclust* come descritto in precedenza. Tutte le mutazioni appartenenti a proteine dello stesso cluster sono state inserite nello stesso insieme di cross-validation al fine di evitare problemi di sovrastima dei risultati. Per le definizioni delle misure di accuratezza si veda l'appendice A.

6.4 Risultati e confronto con altri metodi

L'informazione derivante dal profilo di sequenza è molto importante per determinare se una mutazione può influenzare negativamente la funzionalità dell'organismo e di conseguenza la sua salute (Ramensky et al., 2002). Spinti da questa considerazione abbiamo fatto un'analisi statistica del nostro data set per mettere in evidenza le caratteristiche più importanti al fine di discriminare tra SNP non-sinonimi correlati alla classe "Disease" o alla classe "Neutral". Dopo un'attenta ricerca si è dimostrato che la migliore funzione discriminate tra le due classi è il rapporto [f_k(wt)/f_k(mut)] (vedi fig. 6.3).

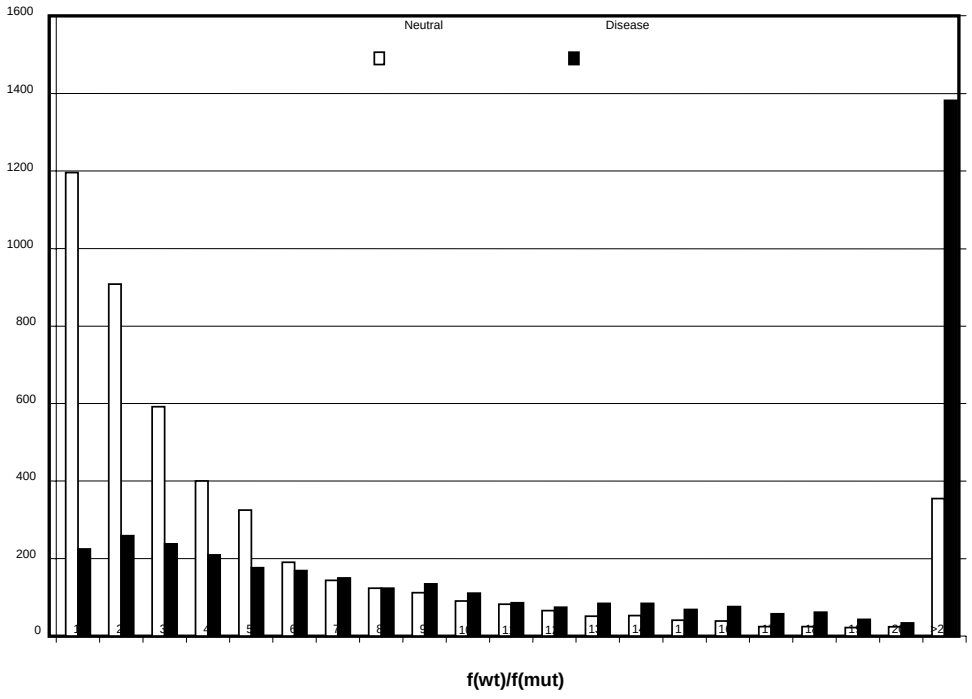


Figura 6.3 Distribuzione dei polimorfismi neutri e delle mutazioni correlate a malattia a valori differenti del rapporto tra la frequenza del residuo wild-type e del mutato ((*f*(wt)/*f*(mut)) nel profilo di sequenza. Questi dati sono calcolati sul data set HumVarProf set contenente 8718 mutazioni (3852 etichettate come "Disease" e 4866 come "Neutral").

Questa distribuzione rimane pressoché inalterata quando, per costruire il profilo dagli allineamenti, dalla banca dati nr95 vengono eliminate tutte le sequenze umane. Questo suggerisce che la funzione utilizzata non è influenzata da "bias" dovuti alla presenza di sequenze paraloghe, ma riesce ad estrarre l'informazione necessaria principalmente dalle sequenze ortologhe. Come detto, siccome non è sempre possibile calcolare il rapporto $f_k(wt)/f_k(mut)$ si è reso necessario utilizzare una strategia ad albero decisionale per poter essere sempre in grado di avere una risposta predittiva grazia alla SVM-Sequence. I risultati dei vari metodi implementati sono riportati in tabella 6.1.

Tabella 6.1 Prestazioni dei diversi metodi sul set HumVar

Method	Q2	P[D]	Q[D]	P[N]	Q[N]	C
ProbMeth	0.62	0.63	0.91	0.56	0.18	0.13
SVM-Sequence	0.70	0.71	0.84	0.65	0.46	0.34
HybridMeth	0.74	0.80	0.76	0.65	0.70	0.46

Gli indici D ed N rappresentano rispettivamente la classe "Disease" e "Neutral". Per le definizioni delle misure si veda l'appendice A

Come atteso la SVM-Sequence ha prestazioni migliori del metodo ProbMeth, si veda in particolare l'incremento di 0.21 per quello che riguarda la correlazione. Il metodo SVM-Sequence mostra delle buone prestazione nel classificare le mutazioni appartenenti alla classe "Disease", mentre pecca nell'assegnare le mutazioni alla classe "Neutral". Accoppiando la SVM-Sequence con la SVM-Profile nel metodo ibrido, l'accuratezza generale sale al 74% ed il coefficiente di correlazione sale a 46%.

Un'altra stima delle prestazioni della SVM-Sequence in confronto col metodo ibrido è riportato in figura 6.4 dove viene riportato l'andamento del tasso di veri positivi (TPR) in funzione del tasso di falsi postivi (FPR), questa curva viene chiamata ROC (Baldi P., et al 2000).

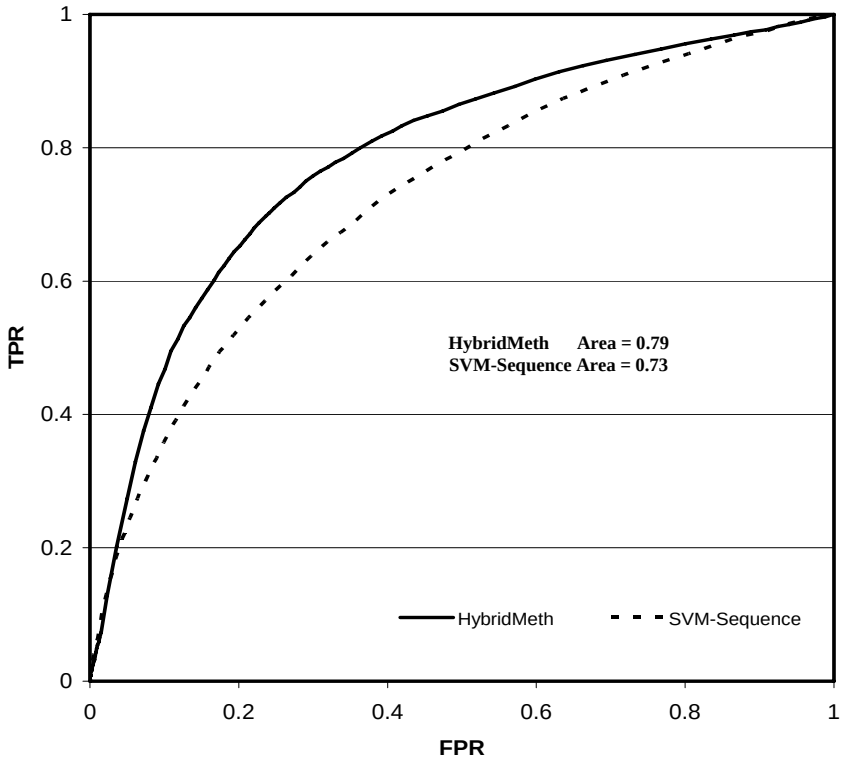


Figura 6.4 curve ROC per l'HybridMeth e la SVM-Sequence ottenute graficando il tasso di falsi positivi (FPR=1-P(s)) vs il tasso dei veri positivi (TPR=Q(s)).

È chiaro come il metodo ibrido riesce ad incrementare l'area della ROC di sei punti percentuali. Questo si riflette nel fatto che se accettiamo un tasso di falsi positivi pari al 5%, il metodo basato sull'albero decisionale migliora di 10 punti percentuali le prestazioni della sola SVM-Sequence. Successivamente, per valutare meglio la bontà del nostro approccio, si sono confrontate le prestazioni predittive del metodo ibrido rispetto ad altri sistemi disponibili online.

I "tools" presi in esame sono Polyphen (Ramensky et al., 2002) e SIFT (Ng and Henikoff, 2003). Il primo è un metodo basato anche su un approccio ad albero decisionale e prende in considerazione molte informazioni derivanti da: parametri di tipo strutturale, annotazioni funzionali ed informazioni evolutive. Il secondo si basa solamente sulla similarità in sequenza, considerando la conservazione dei residui nelle famiglie proteiche. A differenza del nostro metodo ibrido questi predittori non sono sempre in grado di fornire una risposta classificativa per una certa mutazione. Questo è dovuto alla mancanza di parametri critici come l'informazione funzionale o quella evolutiva per una certa famiglia proteica. Il metodo ibrido fornisce sempre una predizione in quanto sfrutta una delle due SVM descritte in precedenza. Nella tabella 6.2 sono riportati i risultati del confronto deri vari metodi presi in esame testati sull' HumVar data set. È da tenere in considerazione che solo le prestazioni dell'HybridMeth sono valutate secondo una procedura di cross-validation.

Tabella 6.2 Confronto tra l'HybridMeth ed altri metodi disponibili sul web

Method	Q2	P[D]	Q[D]	P[N]	Q[N]	C	PM%
PolyPhen[1]	0.72	0.62	0.72	0.80	0.73	0.44	93
SIFT[2]	0.67	0.76	0.67	0.56	0.66	0.33	94
HybridMeth*	0.74	0.80	0.76	0.65	0.70	0.46	100

Risultati ottenuti dal web server: (1) http://www.bork.embl-heidelberg.de/PolyPhen/; (2) scaricato da http://blocks.fhcrc.org/sift/SIFT.html e fatto girare in locale. I dati si riferiscono al data set HumVar.*Solo le prestazioni di HybridMeth derivano da una procedura di cross-validation 21185 mutazioni. PM è la percentuale di predizioni ottenute sul data set

Quello che si vede è che Polyphen ed HybridMeth son più accurati di SIFT. HybridMeth raggiunge il valore massimo di accuratezza generale e di correlazione dei tre metodi messi a confronto e per di più copre il 100% del data set HumVar. Polyphen e SIFT non riescono a predire circa 1500 mutazioni del data set per mancanza delle informazioni necessarie per portare a termine le loro predizioni.

Come ultimo test per avvalorare la robustezza del metodo, rispetto anche a Polyphen e SIFT , abbiamo selezionato un insieme di mutazioni proteiche puntiformi umane che non sono presenti nel set di training/testing HumVar e sulle quali il metodo pertanto non è stato addestrato. Tale set è quello indicato come NewHumVar determinato come descritto nella sezione "Data Sets". I risultati di tale analisi sono riportati nella tabella 6.3.

Tabella 6.3 Confronto dei vari metodi sul nuovo set di mutazioni (NewHumVar)

Method	Q2	P[D]	Q[D]	P[N]	Q[N]	C	PM%
PolyPhen[1]	0.72	0.30	0.63	0.92	0.73	0.28	79
SIFT[2]	0.69	0.32	0.55	0.87	0.72	0.22	88
HybridMeth	0.73	0.34	0.74	0.94	0.73	0.36	100

I dati riportati riguardano 935 mutazioni proteiche. PM è la percentuale di predizioni ottenute sul data set

Come si vede ancora una volta il metodo HybridMeth ha delle prestazioni migliori rispetto agli altri metodi soprattutto per quello che riguarda il coefficiente di correlazione, ed ha delle prestazioni paragonabili a quelle trovate per il set di training/testing. Possiamo dunque concludere che un sistema ad albero decisionale, come quello descritto, che sfrutta dei sistemi di apprendimento automatico come le SVM è in grado di discriminare mutazioni proteiche puntiformi con delle prestazioni almeno paragonabili se non addirittura superiori, ad altri metodi ben referenziati in letteratura.

Conclusioni

La ricerca svolta presso il laboratorio dell'unità di Biocomputing, ha riguardato lo sviluppo di metodi computazionali atti a fornire soluzioni predittive al problema della stabilità proteica in seguito a mutazioni proteiche puntiformi e la possibile relazione tra queste e l'insorgenza di malattie genetiche umane.

L'argomento in esame è stato studiato sotto due diversi aspetti:

i) l'utilizzo di algoritmi di apprendimento automatico per determinare la variazione del ΔG di stabilità per i mutanti delle proteine a partire solo da informazioni sulla sequenza;

ii) l'utilizzo di algoritmi di apprendimento automatico per determinare la relazione tra mutazioni puntiformi e malattie a partire da informazioni sulla sequenza e di carattere evolutivo.

Per quanto concerne il primo punto è stato implementato un modello basato su Support Vector Machines, che tiene conto dell'amminoacido sostituito, del suo intorno in sequenza e delle condizioni sperimentali con cui sono stati condotti gli esperimenti di mutagenesi. Questo metodo potrebbe risultare molto utile nelle applicazioni di protein engeneering, per la sintesi di proteine più o meno stabili e dunque per l'applicazione in campo medico-farmacologico. Il metodo si è dimostrato anche in grado di predire correttamente un set, seppur piccolo, di mutazioni per cui sono noti dati di stabilità in proteine che causano malattie legate al folding. I risultati ottenuti sono stati presentati alla IV edizione del European Conference on Computational Biology (ECCB) e pubblicato sulla rivista internazionale Bioinformatics (Capriotti et al., 2005). È disponibile una versione online del metodo alla pagina http://gpcr2.biocomp.unibo.it/cgi/predictors/I-Mutant2.0/I-Mutant2.0.cgi

Questo modello di stabilità a causa dei pochi dati disponibili

in merito alle malattie legate al folding non permette di costruire un sistema di utilizzo su larga scala per l'analisi degli effetti funzionali delle mutazioni puntiformi.

Spinti da questo problema, per quanto riguarda il secondo punto, è stato progettato un altro sistema basato su SVM per la predizione della correlazione tra le mutazioni proteiche puntiformi e le malattie genetiche, a partire da esempi noti annotati nella banca dati SWISS-PROT. In particolare, utilizzando informazioni di tipo evolutivo e di sequenza è stato realizzato un sistema ibrido che sfrutta un albero decisionale per classificare le mutazioni come polimorfismi neutri o come polimorfismi che influenzano negativamente la funzionalità proteica. Questo approccio si è dimostrato avere delle prestazioni migliori rispetto ad altri due sistemi disponibili online, richiede un numero di informazioni minore per la classificazione ed è sempre in grado di fornire una predizione per qualsivoglia mutazione. I risultati di tale lavoro sono stati pubblicati sulla rivista Bioinformatics (Capriotti et al., 2005). È disponibile una versione online del metodo alla pagina http://gpcr2.biocomp.unibo.it/cgi/predictors/PhD-SNP/PhD-SNP.cgi .

Attualmente si sta implementando il metodo in modo da incorporare informazioni di tipo evolutivo più sofisticate rispetto al semplice rapporto tra le sequenze del residuo wild-type e mutato utilizzato ed i risultati preliminari sono molto incoraggianti.

Pubblicazioni e Partecipazione a Congressi

Partecipazione a Scuole, Congressi e Workshop

Remo Calabrese ha partecipato:

- Bologna Winter School *Predicting 3D Structure of Difficult Proteins*, Bologna 3-9 Febbraio 2002;
- Bologna Winter School *Hot Topic in Structural Genomics*, Bologna 9-15 Febbraio 2003;
- Bologna Winter School *The State of the Art of Protein-Protein Interaction Networks*, Bologna 8-14 Febbraio 2004.
- Bologna Winter School *HOW COMPLEX IS FUNCTIONAL GENOMICS?*, Bologna 13-19 Febbraio 2005
- Bologna Winter School *APPLIED BIOINFORMATICS The test case of the Human Genome*, Bologna 13-17 Febbraio 2006
- Bologna Winter School *BIOINFORMATICS FOR SYSTEMS AND SYNTHETIC BIOLOGY*, Bologna 18-23 Febbraio 2007

Le scuole avanzate di Bologna costituiscono un forum internazionale per la discussione di tematiche rilevanti nel settore Bioinformatico/Biologico Computazionale e sono finanziate anche dalla European Science Foundation.

RC ha inoltre partecipato facendo comunicazioni del proprio lavoro e/o presentando posters a:

- XII Scuola Nazionale di Biofisica *e Proteomica e Biofisica*, Bressanone (BZ) 13-15 Settembre 2004;
- XIII Scuola Nazionale di Biofisica *Biofisica della Cellula*, Bressanone (BZ) 7-9 Settembre 2005;
- IV edizione dell' dell' *European Conference on Computational Biology (ECCB)* Madrid dal 28-09-2005 al 01-10-2005
- BITS06 meeting, Bologna (BO) 28-29 Aprile 2006

Abstract di presentazioni a congressi

Protein-Protein interactions with neural networks, detecting the interaction patches (Bressanone 2004) R.Calabrese, P. Fariselli and R. Casadio

The melting pot of tools for function prediction (Abstract Book CASP6) R.Calabrese, P. Fariselli, I. Rossi and R. Casadio

Predizione di malattie genetiche umane dovute a mutazioni proteiche puntiformi (Bressanone 2005). R. Calabrese, E. Capriotti, P. Fariselli, P. Marani and R. Casadio

Predicting the insurgence of human genetic diseases due to single point protein mutation using machine learning approach (ECCB05) R. Calabrese, E. Capriotti and R. Casadio

A computational approach for detecting peptidases and their specific inhibitors at the genome level (BITS06) P. Fariselli, L. Bartoli, R. Calabrese, D. G Mita, R. Casadio

Pubblicazioni in riviste internazionali con referee ed impact factor

Capriotti E, Fariselli P, Calabrese R, Casadio R (2005) Predicting protein stability changes from sequences using support vector machines. Bioinformatics 21 Suppl 2 (ii1-115)

Capriotti E, Calabrese R, Casadio R. (2006) Predicting the insurgence of human genetic diseases associated to single point protein mutations with support vector machines and evolutionary information.Bioinformatics

Bartoli L., Calabrese R., Fariselli P.,Mita D.G. and Casadio R. (2007) A computational approach for detecting peptidases and their specific inhibitors at the genome level. BMC Bioinformatics 8 (Suppl 1):S3

Capitoli di libri

Casadio R, Calabrese R, Capriotti E, Compiani M, Fariselli P, Marani P, Montanucci L, Martelli PL, Rossi I, Tasco G - Machine learning and the prediction of protein structure: the state of the art- 10th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2004)- Perugia, 4-9/7/2004, Casa Editrice La Sapienza, Roma, pagg 933-940

Casadio R, Calabrese R, Tasco G, Capriotti E, Compiani M, Marani P, Montanucci L, Rossi I, Martelli PL, Fariselli P- Metodi di Machine Learning per la predizione di strutture proteiche e della loro interazione- Convegno Bioinformatica: sfide e prospettive. Università del Sannio, 17-18/12/2003. F.Angeli Editore 51.

Attività Didattica

RC ha svolto attività di supporto alla didattica per i seguenti anni:

ATTIVITA' DI TUTORATO:

2003 - Laboratorio di Biologia Computazionale (CDL BIOTECNOLOGIE)

2004 - Laboratorio di Biologia Computazionale (CDL BIOTECNOLOGIE)

2005 - Laboratorio di Biologia Computazionale (CDL BIOLOGIA)

2006 - Laboratorio di Biologia Computazionale (CDL BIOLOGIA)

BIOINFORMATICS

Vol. 21 Suppl. 2 2005, pages ii54–ii58
doi:10.1093/bioinformatics/bti1109

Protein Structure and Function

Predicting protein stability changes from sequences using support vector machines

Emidio Capriotti, Piero Fariselli, Remo Calabrese and Rita Casadio*

Laboratory of Biocomputing, CIRB/Department of Biology, University of Bologna, Bologna, Italy

ABSTRACT

Motivation: The prediction of protein stability change upon mutations is key to understanding protein folding and misfolding. At present, methods are available to predict stability changes only when the atomic structure of the protein is available. Methods addressing the same task starting from the protein sequence are, however, necessary in order to complete genome annotation, especially in relation to single nucleotide polymorphisms (SNPs) and related diseases.

Results: We develop a method based on support vector machines that, starting from the protein sequence, predicts the sign and the value of free energy stability change upon single point mutation. We show that the accuracy of our predictor is as high as 77% in the specific task of predicting the $\Delta\Delta G$ sign related to the corresponding protein stability. When predicting the $\Delta\Delta G$ values, a satisfactory correlation agreement with the experimental data is also found. As a final blind benchmark, the predictor is applied to proteins with a set of disease-related SNPs, for which thermodynamic data are also known. We found that our predictions corroborate the view that disease-related mutations correspond to a decrease in protein stability.

Availability: http://gpcr2.biocomp.unibo.it/cgi/predictors/I-Mutant2.0/I-Mutant2.0.cgi

Contact: casadio@alma.unibo.it

1 INTRODUCTION

Protein stability change upon site-specific mutations is a relevant problem both for protein design and for the comprehension of protein function (Daggett and Fersht, 2003). For this reason, different methods have been described to predict stability changes observed upon residue substitutions in the original protein sequence. They are based mainly on the development of different energy functions and are suited to computing the stability free energy changes in protein structures when mutating one residue at a time in the sequence (Prevost *et al*., 1991; Topham *et al*., 1997; Pitera and Kollman, 2000; Gilis and Rooman, 1997; Kwasigroch *et al*., 2002; Funahashi *et al*., 2001; Guerois *et al*., 2002; Zhou and Zhou, 2002). An alternative approach based on a neural network (NN) system was recently proposed (Capriotti *et al*., 2004). In this application, instead of directly estimating the relative stability changes upon protein mutation (the $\Delta\Delta G$ value), an NN predicts the direction towards which the mutation shifts the stability of the protein (namely the sign of $\Delta\Delta G$). It could be towards a positive or negative $\Delta\Delta G$ value, corresponding to an increase or decrease of stability, respectively. This prediction is sufficient to evaluate the overall effect of the mutation on the protein stability.

*To whom correspondence should be addressed.

Other relevant thermodynamic parameters in mutagenesis are experimental conditions such as pH and temperature (Bava *et al*., 2004). In this respect, energy-based methods need to fit these parameters assuming that the mutations are carried out at physiological conditions. This problem was also overpassed by the machine learning approach (Capriotti *et al*., 2004), which takes these variables as input.

All the methods mentioned above are, however, limited in that prediction can be carried out only when the protein 3D structure is available. For wide-scale genome analysis, it is necessary to develop applications that can predict stability variation upon mutation starting from the protein sequence. This is particularly relevant to assessing whether a given mutation may or may not lead to protein misfolding and diseases (Dobson, 2003).

In this paper we develop a method based on support vector machines (SVMs) that predicts protein stability changes due to single point mutation starting from the sequence. Owing to the availability of a large database of thermodynamic data for mutated proteins (Bava *et al*., 2004) we are able to show that for the specific task of predicting the $\Delta\Delta G$ sign, our method reaches an accuracy value as high as 77% and a satisfactory correlation agreement when assigning the $\Delta\Delta G$ values. Furthermore, we show that the prediction of protein stability decrease correlates well with a blind set of thermodynamic measurements performed with disease-related mutated chains of the prion and transthyretin proteins.

2 METHODS

2.1 The protein database

Our data set is derived from the current release (December 2004) of the Thermodynamic Database for Proteins and Mutants (ProTherm by Bava *et al*., 2004). The data set of proteins was extracted from ProTherm with the following constraints:

(1) the $\Delta\Delta G$ value has been experimentally detected and is reported in the database;

(2) the data are relative to single mutations (no multiple mutations have been taken into account).

After this filtering procedure, we ended up with a data set consisting of 2048 different single mutations obtained from 64 different protein sequences. The final set is available at http://gpcr2.biocomp.unibo.it/∼emidio/I-Mutant2.0/dbMutSeq.html.

2.2 The data set of disease-related mutations

In order to test our predictor on the task of predicting whether diseases induced by single point mutations can destabilize the protein folding, we collected mutations for two experimentally well characterized proteins: the prion protein (PRIO_HUMAN) and transthyretin (TTHY_HUMAN). We collected

all the disease-related mutations known to destabilize the protein folding for which thermodynamic data could also be found in the literature. We included those disease-related mutations that have been reported as having promoted conformational changes and whose 3D structure has been deposited in the Protein Data Bank (PDB). We ended up with 20 mutations for the two proteins, among which some are associated with diseases such as Gerstmann–Strussler and Creutzfeldt–Jakob syndromes and some with amyloidosis for prion and transthyretin, respectively. These data were used as a blind test for our predictor.

2.3 The predictor

We address two different tasks: (1) the prediction of the sign of the protein stability change upon single point mutation and (2) the prediction of the $\Delta\Delta G$ value. The former case is a classification task, discriminating two classes as described before (Capriotti *et al*., 2004). In the latter case we deal with a regression-fitting problem. When developing methods addressing both tasks, we adopted the same type of input. Thus, and for the user, the only difference between tools predicting the $\Delta\Delta G$ sign and those predicting the $\Delta\Delta G$ values is the output type.

Two machine learning algorithms were implemented: (1) a standard feed-forward NN, with the back-propagation algorithm as a learning procedure, and (2) an SVM with several kernels.

For the classification task, the NN architecture consists of a one-layer perceptron with two hidden nodes and one output node that codifies for the increased protein stability ($\Delta\Delta G \geq 0$, desired output set to 1) or for the destabilizing mutation ($\Delta\Delta G < 0$, desired output set to 0). The decision threshold is set equal to 0.5. The same classification labeling and decision threshold are used for the SVMs. Similar to the previous method for predicting stability changes starting from the protein structure (Capriotti *et al*., 2004), the input vectors (the same for NN and SVM) consist of 42 values. The first two input values account for the temperature and the pH at which the stability of the mutated protein was measured. The next 20 (the 20 residue types) explicitly define the mutation: we set to -1 the element corresponding to the deleted residue and to 1 the new introduced residue (all the remaining elements are kept equal to 0). The final 20 input values encode the sequence residue environment (again the 20 neurons represent the 20 residue types). Each of these input neurons is provided with the number of the encoded residue type, to be found inside a window centered at the residue that undergoes the mutation and that symmetrically spans the sequence to the left (N-terminus) and to the right (C-terminus) with variable lengths from 7 to 23 residues.

The NNs are our own implemented software. For the SVM implementation we use LIBSVM (http://www.csie.ntu.edu.tw/∼cjlin/). We tested the following available kernels:

Linear $K(x_i, x_j) = x_i T x_j$;
Polynomial $K(x_i, x_j) = (G x_i T x_j + r)^d$;
Sigmoid $K(x_i, x_j) = \tan \mathrm{h}(G x_i T x_j + r)$;
RBF $K(x_i, x_j) = \exp(-G||x_i - x_j||^2)$.

When assigning the $\Delta\Delta G$ values, only the SVM with the RBF kernel is considered. The same input of the classification task is adopted. In this case the SVMs directly compute the regression and the output is the predicted $\Delta\Delta G$ value for a given mutation.

2.4 Scoring the performance

Results obtained with NNs and SVMs are evaluated using a cross-validation procedure on the data set. The reported data for the classification and regression tasks are obtained adopting a 20-fold cross-validation procedure; we also adopted larger and smaller divisions (from 10- to 30-fold cross-validation) in order to asses the stability of the methods and found no difference. Grouping of the data into sets for cross-validation was performed in such a way that the positive and the negative examples respected the original distribution of the whole set. Furthermore, we kept the same mutations (when reported at different experimental conditions) in the same set to prevent an overestimation of the results. For each tested method we adopted the same cross-validation sets; thus, results obtained with different methods can be directly compared since testing was done under the same conditions.

Several measures of accuracy are routinely used. For sake of completeness, here we review the ones adopted in this paper. The efficiency of the predictor is scored using the statistical indexes defined below.

The overall accuracy is

$$\mathrm{Q2} = p/N \tag{1}$$

where p is the total number of correctly predicted residues and N is the total number of residues.

The correlation coefficient C is defined as

$$C(s) = [p(s)n(s) - u(s)o(s)]/D \tag{2}$$

where D is the normalization factor

$$D = [(p(s) + u(s))(p(s) + o(s))(n(s) + u(s))(n(s) + o(s))]^{1/2} \tag{3}$$

for each class s (+ and − for positive and negative $\Delta\Delta G$ values, respectively); $p(s)$ and $n(s)$ are the total number of correct predictions and correctly rejected assignments, respectively, and $u(s)$ and $o(s)$ are the numbers of under- and overpredictions.

The coverage for each discriminated structure s is evaluated as

$$Q(s) = p(s)/[p(s) + u(s)] \tag{4}$$

where $p(s)$ and $u(s)$ are the same as in Equation (2).

The probability of correct predictions $P(s)$ (or accuracy for s) is computed as

$$P(s) = p(s)/[p(s) + o(s)] \tag{5}$$

where $p(s)$ and $o(s)$ are the same as in Equation (2) (ranging from 1 to 0).

The reliability score for each network prediction is also assigned. With one output NN this is obtained by computing

$$\mathrm{Rel}(i) = 20 * \mathrm{abs}(O(i) - 0.5) \tag{6}$$

For computing regression we use the standard correlation (R) and root mean squared standard error (RMSE) values.

3 RESULTS AND DISCUSSION

3.1 Predicting the sign of the protein stability change from sequence

We have previously shown that with an NN-based method over 80% of the mutations in a data set containing 1615 examples were correctly assigned provided that the protein 3D structure was known (Capriotti *et al*., 2004). In this paper we focus on the protein sequence and predict whether a mutation along the sequence increases or decreases the corresponding protein stability without referring to the 3D structure. The results obtained with the different machine learning predictors specifically developed for this task are reported and compared in Table 1. It is interesting to notice that even though the information is only relative to the sequence, an SVM endowed with an RBF kernel reaches an accuracy of 0.77, with a correlation coefficient of 0.42. This finding indicates that a piece of information relevant to the protein folding stability can be traced back to the sequence nearest neighbors of the residue that undergoes mutation. Apparently the RBF kernel is better suited to this task than others. This may indicate that this kernel type properly captures the underlying properties in the residue local environment conducive to the protein stability/instability related also to temperature and pH (routinely physiological) at which mutation occurs.

In Table 2 we show that the best accuracy is reached when the sequence window is 19 residues long. In Table 2 we also test the information pertaining to an infinite window by including the effect

Table 1. Cross-validation performance of the NN and SVM

Method	Q2	$P(+)$	$Q(+)$	$P(-)$	$Q(-)$	C
NeuralNet	0.73	0.39	0.56	0.77	0.87	0.30
SVM-linear	0.67	0.41	0.28	0.73	0.84	0.13
SVM-polynomial	0.73	0.58	0.38	0.77	0.88	0.30
SVM-sigmoid	0.68	0.44	0.27	0.73	0.85	0.15
SVM-RBF	0.77	0.69	0.46	0.79	0.91	0.42

$+$ and $-$: the index is evaluated for positive and negative signs of the protein free energy stability change; for the definition of the different indexes see Section 2.3. The window length for both methods included 19 residues.

Table 2. Cross-validation performance of different window lengths using a RBF kernel

Window	Q2	$P(+)$	$Q(+)$	$P(-)$	$Q(-)$	C
7	0.74	0.58	0.36	0.77	0.89	0.30
11	0.73	0.85	0.12	0.73	0.99	0.25
15	0.76	0.64	0.38	0.78	0.91	0.35
19	0.77	0.69	0.46	0.79	0.91	0.42
23	0.76	0.64	0.44	0.79	0.90	0.38
Whole sequence	0.73	0.59	0.32	0.76	0.90	0.28

For notation see Table 1.

Table 3. Q2 accuracy as a function of the mutated residue type

Native\new	Charged	Polar	Apolar
Charged	0.65 (4%)	0.72 (7%)	0.69 (12%)
Polar	0.57 (5%)	0.76 (5%)	0.77 (13%)
Apolar	0.80 (5%)	0.88 (9%)	0.80 (40%)

Each cell represents a particular type of mutation classified according to chemico-physical properties. Rows account for the wild-type residue (native) and the column positions define the new residues in the mutant proteins (new). In brackets the relative fraction in the protein set (2048) of a given residue type is shown.

of the whole sequence. It is evident that the accuracy diminishes, and this indicates that the whole sequence composition is not as specific as the local sequence environment in terms of determining the sign of the stability change. Also in this case the correlation coefficient is different from random.

The analysis of the SVM accuracy as a function of the chemico-physical properties of the mutations indicates that the protein stability changes involving charged/charged, polar/charged and charged/apolar mutations score lower than those involving apolar/apolar swaps (Table 3), and this suggests that for charged and polar residues at the surface or for charged residues involved in salt-bridges, more information than the local sequence environment is necessary for a high predictive score.

The overall Q2 accuracy is computed as a function of the reliability index (Rel). This identifies a relationship between the reliability value and the predictor accuracy, as shown in Figure 1. The value of the reliability index and its relationship to the prediction accuracy

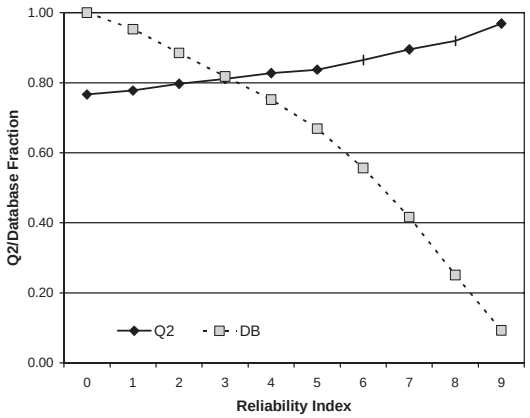


Fig. 1. Q2 accuracy of SVM-RBF as a function of the reliability index (Rel) of the prediction [Equation (6)]. DB is the fraction of the data set with Rel values higher or equal to a given threshold.

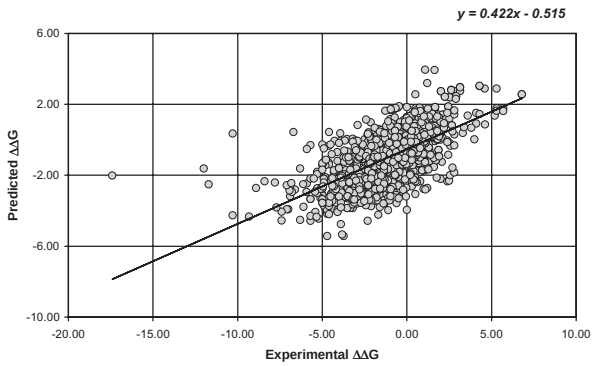


Fig. 2. Regression between predicted and expected values of free energy change upon mutation starting from the protein sequence [$R = 0.62$, RMSE $= 1.45$ $(0.422x - 0.515)$ Kcal/mol].

may help in selecting which mutations are more suited to increasing or decreasing protein stability in a rational computer-aided protein design even at the genomic level.

3.2 Predicting the free energy values of protein stability change from sequence

In specific cases, not only the sign of the mutation but also the exact value of the free energy stability change may be necessary for selecting the mutation type. We have previously shown that coupling machine learning with energy-based methods could provide an excellent solution to this problem (Capriotti *et al*., 2004). However, this is restricted to the small subset of proteins for which a 3D structure is available. Since the aim of this paper is to extend the prediction of stability changes upon mutation to the sequence space, we also implement a SVM that predicts the exact $\Delta\Delta G$ values. This is done using the ν-regression SVM with RBF kernel (libSVM).

In Figure 2 we show the regression between the predicted and the expected $\Delta\Delta G$ values. Predictions are obtained using a 20-fold cross-validation. The R (regression) value is equal to 0.62 with a RMSE of 1.45 Kcal/mol. It should be stressed that this correlation is obtained by starting from the protein sequence and that to our knowledge this is the first method capable of performing the task at

Table 4. Prediction of disease-related mutations

Protein	Mutation	Effect	Predicted stability change	Rel	Experimental $\Delta\Delta G$ (Kcal/mol)	Ref.
Human prion (PRIO_HUMAN)						
	P102L	GSD	Increase	2	0.2 ± 0.6	Apetri *et al*. (2004)
	M129V	Polymorphism	Decrease	6	-0.3 ± 0.5	Liemann and Glockshuber (1999)
	V180I	**GSD**	**Decrease**	**2**	$\mathbf{-0.5 \pm 0.4}$	**Liemann and Glockshuber (1999)**
	T183A	**CJD**	**Decrease**	**6**	$\mathbf{-4.6 \pm 0.7}$	**Liemann and Glockshuber (1999)**
	T190V	Polymorphism	Decrease	2	0.2 ± 0.6	Liemann and Glockshuber (1999)
	F198S	**GSD**	**Decrease**	**7**	$\mathbf{-2.5 \pm 0.4}$	**Liemann and Glockshuber (1999)**
	E200K	CJD	Decrease	5	-0.1 ± 0.6	Liemann and Glockshuber (1999)
	R208H	**CJD**	**Decrease**	**7**	$\mathbf{-1.4 \pm 0.6}$	**Liemann and Glockshuber (1999)**
	V210I	CJD	Decrease	2	-0.3 ± 0.6	Liemann and Glockshuber (1999)
	Q217R	**GSD**	**Increase**	**1**	$\mathbf{-2.1 \pm 0.4}$	**Liemann and Glockshuber (1999)**
	M166V	Polymorphism	Decrease	6	SC(1E1J)	Calzolai *et al*. (2000)
	S170N	Polymorphism	Increase	1	SC(1E1P)	Calzolai *et al*. (2000)
	R220K	Polymorphism	Decrease	7	SC(1FKC)	Calzolai *et al*. (2000)
Transthyretin (TTHY_HUMAN)						
	V50M	**Amyloidosis**	**Decrease**	**6**	$\mathbf{-2.2 \pm 2.4}$	**Shnyrov *et al*. (2000)**
	L75P	**Amyloidosis**	**Decrease**	**5**	$\mathbf{-1.5 \pm 2.3}$	**Shnyrov *et al*. (2000)**
	T139M	Unclassified	Decrease	0	-0.1 ± 2.8	Shnyrov *et al*. (2000)
	T80A	Amyloidosis	Decrease	6	SC(1TSH)	a
	S97Y	Amyloidosis	Increase	2	SC(2TRY)	a
	Y134C	Amyloidosis	Increase	0	SC(1IIK)	a
	V142I	Unclassified	Decrease	2	SC(1TTR)	a

GSD, Gerstmann–Straussler disease; CJD, Creutzfeldt–Jakob disease; Rel, reliability index (see Measure of Accuracy); SC, structural conformational changes determined by comparing the native (1QLX, human prion protein; 1BM7, human transthyretin) with the mutated 3D structures (PDB codes are reported within parentheses); a, derived by comparison between the native structure and the mutated as reported in the PDB files through the SWISSPROT links. Bold lettering indicate the subset of mutations in which the $\Delta\Delta G$ values is $\geq$0.5 Kcal/mol.

hand at this level of efficiency. For this reason we suggest that our approach can be successfully applied when protein structures are not available and thermodynamic data on protein stability need to be analyzed in terms of molecular properties.

3.3 Disease-related single nucleotide polymorphisms and the prediction of protein stability changes

Evidence is accumulating that many disease-causing mutations exert their effects by altering protein folding (Wang and Moult, 2001, 2003; Dobson, 2003; Selkoe, 2003). An interesting application of our method is therefore the prediction of protein stability changes when mutations are known to correlate to diseases.

In Table 4 the predicted thermodynamic data for 20 mutations of the human prion protein and human transthyretin are shown and either compared with the experimental $\Delta\Delta G$ values, when available, or related to conformational changes, when known with atomic resolution. The sign of the stability change is correctly predicted in all cases but two, with a correlation coefficient of 0.42. On this blind test the performance is similar to that on the training/testing set.

It is also interesting to note that the protein stability decrease upon mutation correlates with maladies in 77% of the experimental data. Moreover, if we focus only on the subset in which the $\Delta\Delta G$ changes are $\geq$0.5 Kcal/mol, all the mutations correspond to diseases. On this subset of experimental data, our predictor fails only in one case to assign the correct $\Delta\Delta G$ sign. However, if we sort the predictions by the reliability index value, all the predictions made with reliability index >2 agree with the experimental data. The results of this test are therefore in agreement with the general idea that defective protein folding is one of the causes of human diseases and suggest also a possible application of this predictor to correlate single nucleotide polymorphisms and diseases related to protein instability.

ACKNOWLEDGEMENTS

This work was supported by the following grants: “Hydrolases from Thermophiles: Structure, Function and Homologous and Heterologous Expression” of the Ministero della Istruzione dell’Università e della Ricerca (MIUR); a PNR 2001–2003 (FIRB art 8) project on postgenomics to R.C. E.C. is supported by a grant from the European Union’s VI Framework Programme for the BioSapiens Network of Excellence project. P.F. acknowledges an MIUR grant on proteases.

Conflict of Interest: none declared.

REFERENCES

Apetri,A.C. *et al*. (2004) The effect of disease-associated mutations on the folding pathway of human prion protein. *J. Biol. Chem.*, **279**, 31048–31052.

Bava,K.A. *et al*. (2004) ProTherm, version 4.0: thermodynamic database for proteins and mutants. *Nucleic Acids Res.*, **32**, D120–D121.

Calzolai,L. *et al*. (2000) NMR structures of three single-residue variants of the human prion protein. *Proc. Natl Acad. Sci.* USA, **97**, 8340–8345.

Capriotti,E. *et al*. (2004) A neural-network-based method for predicting protein stability changes upon single point mutations. *Bioinformatics*, **20** (suppl. 1), I63–I68.

Daggett,V. and Fersht A.R. (2003) Is there a unifying mechanism for protein folding? *Trends Biochem. Sci*., **28**, 18–25.

Dobson,C.M. (2003) Protein folding and misfolding. *Nature*, **426**, 884–890.

Funahashi,J. *et al*. (2001) Are the parameters of various stabilization factors estimated from mutant human lysozymes compatible with other proteins? *Protein Eng*., **14**, 127–134.

Gilis,D. and Rooman,M. (1997) Predicting protein stability changes upon mutation using database-derived potentials: solvent accessibility determines the importance of local versus non-local interactions along the sequence. *J. Mol. Biol.*, **272**, 276–290.

Guerois,R. *et al*. (2002) Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *J. Mol. Biol.*, **320**, 369–387.

Kwasigroch,J.M. *et al*. (2002) PoPMuSiC, rationally designing point mutations in protein structures. *Bioinformatics*, **18**, 1701–1702.

Liemann,S. and Glockshuber,R. (1999) Influence of amino acid substitutions related to inherited human prion diseases on the thermodynamic stability of the cellular prion protein. *Biochemistry*, **38**, 3258–3267.

Pitera,J.W. and Kollman,P.A. (2000) Exhaustive mutagenesis in silico: multicoordinate free energy calculations on proteins and peptides. *Proteins*, **41**, 385–397.

Prevost,M. *et al*. (1991) Contribution of the hydrophobic effect to protein stability: analysis based on simulations of the Ile-96-Ala mutation in barnase. *Proc. Natl Acad. Sci.* USA, **88**, 10880–10884.

Shnyrov,V.L. *et al*. (2000) Comparative calorimetric study of non-amyloidogenic and amyloidogenic variants of the homotetrameric protein transthyretin. *Biophys. Chem.*, **88**, 61–67.

Selkoe,D.J. (2003) Folding proteins in fatal ways. *Nature*, **426**, 900–904.

Topham,C.M. *et al*. (1997) Prediction of the stability of protein mutants based on structural environment-dependent amino acid substitution and propensity tables. *Protein Eng.*, **10**, 7–21.

Wang,Z. and Moult,J. (2001) SNPs, protein structure, and disease. *Hum. Mutat.*, **17**, 263–270.

Wang,Z. and Moult,J. (2003) Three-dimensional structural location and molecular functional effects of missense SNPs in the T cell receptor Vbeta domain. *Proteins*, **53**, 748–757.

Zhou,H. and Zhou,Y. (2002) Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. *Protein Sci.*, **11**, 2714–2726.

Bioinformatics Advance Access published January 19, 2007

System biology

Protein-protein interaction site prediction based on conditional random fields

Ming-Hui Li*, Lei Lin, Xiao-Long Wang and Tao Liu

Bioinformatics Research Group, ITNLP Lab, Department of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

Associate Editor: Trey Ideker

ABSTRACT

Motivation: We are motivated by the fast-growing number of protein structures in the Protein Data Bank with necessary information for prediction of protein-protein interaction sites to develop methods for identification of residues participating in protein-protein interactions. We would like to compare conditional random fields (CRFs)-based method with conventional classification-based methods which omit the relation between two labels of neighboring residues to show the advantages of CRFs-based method in predicting protein-protein interaction sites.

Results: The prediction of protein-protein interaction sites is solved as a sequential labeling problem by applying CRFs with features including protein sequence profile and residue accessible surface area. The CRFs-based method can achieve a comparable performance with state-of-the-art methods, when 1276 nonredundant hetero-complex protein chains are used as training and test set. Experimental result shows that CRFs-based method is a powerful and robust protein-protein interaction site prediction method and can be used to guide biologists to make specific experiments on proteins.

Availability: http://www.insun.hit.edu.cn/~mhli/site_CRFs/index.html

Contact: mhli@insun.hit.edu.cn

1 INTRODUCTION

Biological functions and processes are performed through the interactions among proteins, RNA or DNA. It is of great significance for protein mimetic engineering, elucidation of molecular pathways and drug design to understand characteristics of protein interfaces (Lichtarge, *et al.*, 2002; Sowa, *et al.*, 2001; Zhou, 2004). Protein-protein interaction is an important factor for determining protein function (Letovsky and Kasif, 2003; Nabieva, *et al.*, 2005). Furthermore, identification of interface residues can help the construction of a structural model for a protein complex(Cyril Dominguez, 2003).

The availability of more and more protein structures in the Protein Data Bank (PDB) (Berman, *et al.*, 2000) makes prediction of protein-protein interaction sites possible. Machine learning methods, such as neural networks (ANN) (Chen and Zhou, 2005; Fariselli, *et al.*, 2002; Zhou and Shan, 2001) and support vector machines (SVM) (Bradford and Westhead, 2005; Chung, *et al.*, 2006; Koike and Takagi, 2004; Res, *et al.*, 2005) have been successfully applied in this field. These studies consider sequential, structural or evolutionary features such as amino acid residue composition (Chen and Zhou, 2005; Chung, *et al.*, 2006; Koike and Takagi, 2004; Res, *et al.*, 2005; Zhou and Shan, 2001), spatial neighboring residues (Wanga, *et al.*, 2006; Zhou and Shan, 2001), accessible surface area (Koike and Takagi, 2004), structural conservation score (Chung, *et al.*, 2006) and residue evolutionary information (Res, *et al.*, 2005; Wanga, *et al.*, 2006). Most of these methods focus on prediction of protein-protein interaction sites on surface of proteins with known structures (Koike and Takagi, 2004; Zhou and Shan, 2001). However, only protein local sequential information is used in study of Ofran and Rost (2003). Res, *et al.* (2005) use protein sequential and evolutionary information to predict proteins interaction sites without structural information. Recently, Liang, *et al.* (2006) present an empirical score function, which is a linear combination of energy score, interface propensity and residue conservation score for prediction of protein binding sites.

These traditional methods take protein-protein interaction prediction as a classification task and separately study each residue, so one interface residue is identified at a time. One drawback of these methods is the relation between two labels (interface or noninterface) of neighboring residues is not taken into consideration. However, as a matter of fact, sequentially or spatially neighboring residues should have similar characters in forming interface. Chung, *et al.* (2006) noticed this relation and used the clustering as a post-processing strategy to remove the isolated interface residues predicted by SVMs and include the noninterface residues surrounded by several predicted interface residues.

In order to acquire the inter-relation information between neighboring residues, prediction of protein interaction sites was formalized as a sequence labeling task in our study. Sequence labeling tasks are very common tasks in natural language processing such as part-of-speech tagging (Lafferty, *et al.*, 2001; Ratnaparkhi, 1996), named-entity recognition (Chinchor, 1998), and information extraction (Freitag and McCallum, 2000). Recently, conditional random fields (CRFs)(Lafferty, *et al.*, 2001; Sutton and McCallum, 2006) are successfully applied to solve sequence labeling problems and are also proved their effectiveness in solving problems in bioinformatics such as protein secondary structure prediction and protein fold recognition (Liu, *et al.*, 2004; Liu, *et al.*, 2005). The advantage of CRFs is that it can integrate both rich state features and transition features between label states. Furthermore, CRFs have advantages over traditional graphical models such as hidden Markov models (HMMs)(Rabiner, 1989) and maximum entropy Markov models (MEMMs) (Mccallum, *et al.*, 2000). It is one of the outstanding methods used for labeling sequence data. In this

*To whom correspondence should be addressed.

study, given a protein sequence with structural information, each residue needs to be labeled as an interface residue or noninterface residue.

CRFs are efficient methods for labeling sequence data, and different from the classification methods such as SVMs and maximum entropy method(ME) (Rosenfeld, 1996).In this paper, we compared the performance of CRFs in predicting protein interaction site with state-of-the-art methods, such as SVMs and ANN. CRFs can be used to label residues of the whole protein sequence, but only the residues on surface were chosen to compare with other methods. Basic features including sequence profile and accessible surface area of spatially neighboring residues were used for comparison of CRFs with other methods for performance. Experimental result shows that CRFs-based method is comparable with the conventional classification methods on 1276 nonredundant chains of hetero complexes selected from the PDB.

2 MATERIAL AND METHODS

2.1 Data set

All x-ray diffraction protein structures which have multiple chains and resolution of less than 3.5 Å were extracted from the PDB (July, 2005) (Berman*, et al.*, 2000). Protein chains shorter than 40 residues were removed. For each structure, we selected chain pairs with more than 20 interfacial residues on each chain. A residue is considered to be an interface residue if the distance between any of its heavy atom and any heavy atom of its interacting chains is less than 5 Å (Chen and Zhou, 2005; Koike and Takagi, 2004; Zhou and Shan, 2001). For PDB structure with more than two chains, each chain was selected for at most one time. For protein chain that interacts with multiple partners, only one partner with the most interfacial residues was selected as its partner. Finally, a total of 15264 chain pairs were selected.

In order to get nonredundant protein chains of hetero complexes, we adopted the method of Chung *et al.*(2006). All these selected chains were compared using BLAST(McGinnis and Madden, 2004). Two chains were assigned with the same cluster if (1) over 90% of their sequences were aligned and (2) the sequence identity was equal or greater than 30%. All above chains were clustered in this way. One representative chain of each cluster was selected. Hetero complexes with longer chains were selected in this study. Two interacting protein chains were defined as homo complex if over 90% of them were aligned and the sequence identity over the aligned region was more than 95% (Chen and Zhou, 2005). Thus 1276 chains (312858 residues) were selected as nonredundant protein chains of hetero complexes.

The surface residues were defined using the criterion of at least 15% solvent accessible surface area exposure to solvent (Chung*, et al.*, 2006; Rost and Sander, 1994). The solvent accessible surface area (ASA) of each residue was calculated using the DSSP program (Kabsch and Sander, 1983). A total of 200,482 residues (about 64.1%) were collected as surface residues from all these chains. Since a protein chain within a complex with more than one chain may form more than one interface. Within these interfaces, there is generally a main large interface while residues in other minor interfaces can be treated as interface or noninterface residues, or even excluded from data set. In our experiment, we consider all these three cases and generated three types of data set (Type I, II, and III). Their statistical information is tabulated in Table 1.

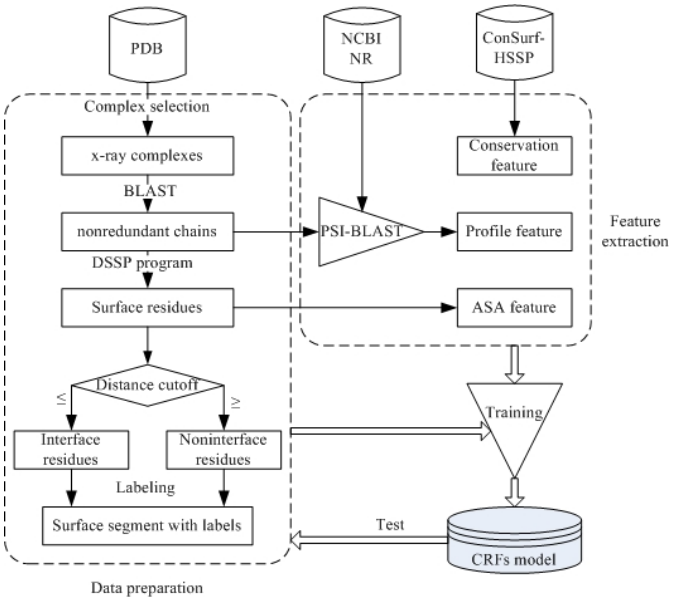


Fig. 1. Overview of CRFs-based protein interaction site prediction system

Table 1. Summary of three types of data sets

Data type	Chains	Res.	Surface res.	Interface res.
Type I[a]	1276	312858	200482	56831 (28.3%)
Type II[b]	1276	312858	200482	74455 (37.1%)
Type III[c]	1276	312858	183326	56831 (31.0%)

[a]Minor interface as negetive examples.
[b]Minor interface as positive examples.
[c]Exclude minor interface from training set.

Surface residue sequence segments were collected. The surface residue sequence segment is sequential continuous residue segment which are all surface residues. Each residue within the segment was labeled as interface or noninterface residue. These segments were used to train and test CRFs.

The fact that there are more noninterface residues than interface residues in the training set leads to higher precision and lower recall for many classifiers such as SVMs and ANN (Chen and Zhou, 2005; Chung*, et al.*, 2006; Koike and Takagi, 2004). These researchers used trimmed data set, the ratio of positive and negative examples are set to about 1:1. To evaluate the robustness and performance of different methods, we conduct experiments on both complete and trimmed data sets of all above three data types. The left dashed-line rounded rectangle in Figure 1 illustrates the process of data preparation.

2.2 Conditional random fields used for labeling sequence data

In order to predict protein-protein interaction sites, we address this problem as a sequence labeling task. Protein surface residues were extracted and the surface residue segments were treated as sequence data. Residues on surface segments were labeled as interface or noninterface residues using CRFs.

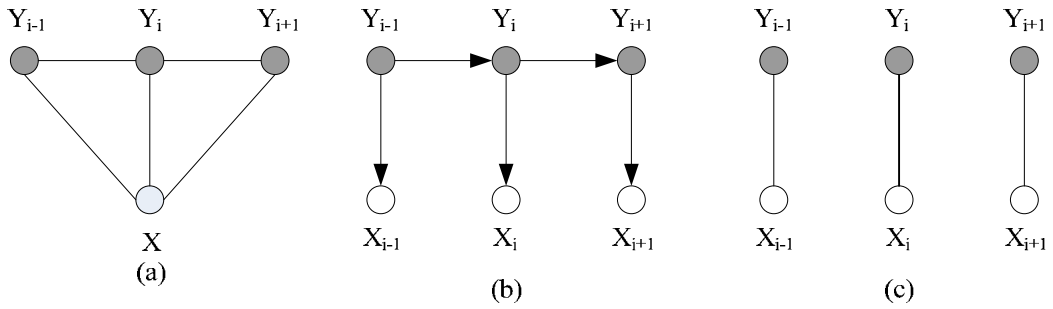


Fig. 2. Structure of chain-structured CRFs, simple HMMs and ME

Conditional random fields (CRFs) were proposed by Lafferty, *et al.* for labeling sequence data (Lafferty*, et al.*, 2001). Given a sequence of observations $X = (x_1, x_2, \ldots, x_n)$, we want to get the most probable label sequence $Y = (y_1, y_2, \ldots, y_n)$, i.e. $Y^* = \arg\max_Y P(Y \mid X)$. CRFs are undirected graphical models (as opposed to directed graphical models such as HMMs) and the conditional probability P(Y|X) is computed directly. Figure 2 shows the structures of CRFs, HMMs and ME. Both CRFs and HMMs suit to label sequence, differing from the probability solution formulation. HMMs obtain the target label sequence Y by maximizing the joint probability of X and Y (Rabiner, 1989), but HMM can not use long distance features which limits the broad application of this method. CRFs are exponential or log-linear models which can use any kind of features. By the fundamental theorem of random field(Lafferty*, et al.*, 2001), the joint distribution over label sequence Y given X can be given by the following conditional probability:

$$P(Y \mid X) \quad \frac{1}{Z(X)} \exp(\quad _j t_j(y_{i\ 1}, y_i, x, i) \quad \quad _j s_j(y_i, x, i)) \quad (1)$$

Where, $t_j(y_{i-1}, y_i, x, i)$ is a transition feature function of the entire observation sequence and the labels at position i and i-1 in the label sequence; $s_j(y_i, x, i)$ is a state feature function of the label at position i and the observation sequence. The index j in t_j and s_j is feature serial number to represent different features. Parameters λ_j and μ_j correspond with feature t_j and s_j, respectively, and they are learned via maximizing the conditional likelihood of the training data. Z(X) is a normalization factor. More details about CRFs can be referred from Lafferty(2001).

2.3 Prediction of protein-protein interaction sites based on CRFs

Here, sequence segments on protein surface are labeled by CRFs. The label set for residues is L = {I, N}, where I represents the interface residue and N represents the noninterface residue. Given a segment $X = (x_1, x_2, \ldots, x_n)$, the most possible label sequence $Y = (y_1, y_2, \ldots, y_n)$ ($y_i \in$ L) is obtained using CRFs.

2.4 Definition of features

The features for CRFs include transition and state features. We define several types of state features based on common features most used by other researches. Two kinds of state features, spatially neighboring residues profile and accessible surface area are taken as basic features for CRFs. Residue conservation is taken as an extended feature to test its effectiveness in CRFs.

2.4.1. Transition feature Transition feature is defined for each label pair (y and y' ∈ L) as follows:

$$t_{y,y'}(y_{i\ 1}, y_i, x, i) \quad \begin{matrix} 1 & if\ y_{i\ 1} \quad y\ and\ y_i \quad y' \\ 0 & otherwise \end{matrix} \quad (2)$$

Where, y_{i-1} and y_i are labels of residues at positions i-1 and i in the protein sequence x, respectively.

2.4.2. Profile feature of spatially neighboring residues Spatially neighboring residues profile feature was taken from multiple sequence alignment obtained from three iterations of PSI-BLAST searching against NCBI nonredundant database (NR, April 2006 release) under conditions E-value=0.001 and h=0.001 (Altschul*, et al.*, 1997). For each labeled residue, its profile features were taken from profiles of 15 nearest spatially neighbor residues (including the labeled residue). The profile value x was scaled to the [0,1] range by using the following function (Kim and Park, 2003):

$$scale(x) = \begin{cases} 0.0 & if\ x \le -5 \\ 0.5 + 0.1x & if -5 \le x \le 5 \\ 1.0 & if\ x \ge 5 \end{cases} \quad (3)$$

The spatially neighboring residue profile feature is defined for each label-amino pair (y ∈ L and aa ∈ amino acid alphabet) as:

$$s^{pro}_{y,aa}(y_i, x_k, i) \quad \begin{matrix} scale(PSSM(x_k, aa)) & if\ y_i \quad y \\ 0 & otherwise \end{matrix} \quad (4)$$

Where, $PSSM(x_k, aa)$ is the element of position-specific scoring matrix for amino acid *aa* at position k in protein sequence. x_k is from the spatially neighboring residues list of x_i.

2.4.3. ASA feature Accessible surface area (ASA) feature represents the relative accessible surface area (scaled by the nominal maximum area of each residue). For convenience, we use ASA to represent the relative accessible surface area of residues.

$$s^{ASA}_{y}(y_i, x_k, i) \quad \begin{matrix} ASA(x_k) & if\ y_i \quad y \\ 0 & otherwise \end{matrix} \quad (5)$$

Where, ASA of each residue is calculated using DSSP program (Kabsch and Sander, 1983). x_k is from the spatially neighboring residues list of residue x_i.

2.4.4. Residue conservation feature Residue conservation feature represents the degree of evolutionary conservation at each residue position and was obtained from the conservation score in the ConSurf-HSSP database (Glaser, *et al.*, 2005). This score is based on the relative entropy and correlates with the functional importance of position. According to the conservation score, the residues were classified into nine categories of conservation (from grade 1 to grade 9). Residue conservation feature is expressed by the conservation grade divided by 10:

$$s_y^{con}(y_i, x_k, i) \quad \begin{matrix} grade(x_k)/10 & if\ y_i \quad y \\ 0 & otherwise \end{matrix} \tag{6}$$

2.4.5. Summary of state feature set The right dashed-line rounded rectangle in Figure 1 illustrates the process of feature extraction. Table 2 gives the feature type and corresponding dimensions.

2.5 Implementation of conditional random fields

FlexCRFs is a conditional random field toolkit for segmenting and labeling sequence data (Phan and Nguyen, 2005). The current version of FlexCRFs can not be used to deal with continuous real value features, so we modified it to solve this problem. In this study, we adopted the first-order Markov CRFs. The parameter init_lambda_val was set to 0.05 and other parameters were set by default. Figure 1 illustrates the whole implementation of our protein interaction labeling system based on CRFs.

3 RESULTS AND DISCUSSION

3.1 Cross-validation and scoring

The performance of each method is measured using three-fold cross-validation. The whole data set (hetero-complex chains) was randomly divided into three subsets with equal number of chains. Each method was trained and tested three times with three different training and test sets. For each time, two subsets were used as training data and the remaining subset was used as test data.

All methods are measured according to the evaluation of residue labeling (or classification) based on the following quantities:

- TP is the number of true positives which are residues correctly classified as interface residues;
- TN is the number of true negatives which are residues correctly classified as noninterface residues;
- FP is the number of false positives which are noninterface residues incorrectly classified as interface residues;
- FN is the number of false negatives which are interface residues incorrectly classified as noninterface residues.

Then we used the following measures to evaluate the labeling (and classification) performance:

$$\text{Precision}=\frac{\text{TP}}{\text{TP+FP}} \tag{7}$$

$$\text{Recall}=\frac{\text{TP}}{\text{TP+FN}} \tag{8}$$

$$\text{F1}=\frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision+Recall}} \tag{9}$$

$$\text{Accuracy}=\frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{10}$$

$$\text{Correlation coefficient} = \frac{\text{TP}\times\text{TN}-\text{FP}\times\text{FN}}{\sqrt{(\text{TP}+\text{FN})(\text{TP}+\text{FP})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}} \tag{11}$$

Table 2. Summary of state feature set

Feature type	Dimension
Profile	1~300 (20*15)
ASA	301~315
Conservation	316~330

Precision, recall F1 are all used to measure the performance for labeling or classifying interface residues, while accuracy is to measure the performance for labeling or classifying the whole test data set. Correlation coefficient (CC) is to measure the correlation between predictions and actual test data.

3.2 Performance of CRFs versus other classification methods

Support vector machines (SVMs), neural network (ANN) and maximum entropy model (ME) are selected to compare with our method. All of them are discriminative classification methods. SVMs and ANN are state-of-the-art methods for predicting protein-protein interaction sites (Chen and Zhou, 2005; Chung, *et al.*, 2006; Fariselli, *et al.*, 2002; Koike and Takagi, 2004; Res, *et al.*, 2005; Zhou and Shan, 2001) and CRFs are extension of ME (Lafferty, *et al.*, 2001; Sutton and McCallum, 2006). LIBSVM (Chang and Lin, 2001) was used as the SVM implementation with radial basis function as kernel. The values of γ and regularization parameter C were set to be 0.1 and 10, respectively. Neural Network Toolbox in Matlab was used as ANN implementation and a feed-forward, back-propagation neural networks was used (Chen and Zhou, 2005). The neural network contained an input layer with 21×15 nodes, a hidden layer with 20 nodes, and an output layer with two nodes. ME implementation of Zhang was used and can be downloaded freely from http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html.

First, we tested these methods on basic feature set: profile of spatially neighboring residues and ASA feature. We tested them on six data sets, and the evaluation results are tabulated in Table 3. Among three complete data sets, CRFs perform best according to F1-measure, which shows that CRFs can obtain better trade-off between precision and recall automatically. Other methods suffer from the unbalanced training data greatly and they get higher precision and lower recall on complete data sets, which agrees with result of Chen and Zhou (2005). CRFs-based method is more robust with respect to different ratio between positive and negative examples of training set.

Table 3. Performance of CRFs versus other classification methods on all data sets using basic features[a]

Data set	Method	Precision (random)[g]	Recall (random)[h]	F1-measure	Accuracy	CC
Complete[b] Type I[d]	SVMs	-[i]	-	-	-	-
	ANN	0.590 (0.257)	0.061 (0.016)	0.110	0.750	0.126
	ME	0.522 (0.257)	0.257 (0.065)	0.344	0.751	0.232
	CRFs	0.471 (0.257)	0.403 (0.103)	0.434	0.733	0.262
Trimed[c] Type I	SVMs	0.412 (0.255)	0.596 (0.152)	0.487	0.680	0.275
	ANN	0.350 (0.257)	0.566 (0.144)	0.432	0.621	0.182
	ME	0.363 (0.257)	0.622 (0.158)	0.459	0.626	0.219
	CRFs	0.364 (0.257)	0.566 (0.144)	0.443	0.637	0.203
Complete Type II[e]	SVMs	0.698 (0.335)	0.309 (0.104)	0.429	0.724	0.321
	ANN	0.618 (0.335)	0.259 (0.087)	0.365	0.698	0.242
	ME	0.599 (0.335)	0.363 (0.122)	0.452	0.705	0.283
	CRFs	0.594 (0.335)	0.415 (0.139)	0.488	0.709	0.303
Trimed Type II	SVMs	0.538 (0.335)	0.627 (0.210)	0.579	0.695	0.344
	ANN	0.439 (0.335)	0.638 (0.214)	0.520	0.606	0.215
	ME	0.475 (0.335)	0.651 (0.218)	0.550	0.643	0.274
	CRFs	0.536 (0.335)	0.595 (0.199)	0.564	0.692	0.328
Complete Type III[f]	SVMs	0.577 (0.277)	0.312 (0.086)	0.405	0.746	0.282
	ANN	0.631 (0.277)	0.136 (0.038)	0.224	0.739	0.200
	ME	0.577 (0.277)	0.312 (0.086)	0.405	0.746	0.282
	CRFs	0.578 (0.277)	0.377 (0.105)	0.457	0.751	0.316
Trimed Type III	SVMs	0.488 (0.277)	0.615 (0.170)	0.544	0.714	0.345
	ANN	0.412 (0.277)	0.610 (0.169)	0.492	0.651	0.252
	ME	0.416 (0.277)	0.641 (0.177)	0.504	0.651	0.267
	CRFs	0.435 (0.277)	0.627 (0.174)	0.513	0.671	0.287

[a]Basic features including spatial neighboring residue profiles and ASA.
[b]All data in training set are used to train these methods.
[c]Training set obtained by randomly removing some noninterface residues are used to train these methods. There are about equal amount of positive and negative examples in trim data set.
[d]Minor interface as negetive examples (Type I).
[e]Minor interface as positive examples (Type II).
[f]Exclude minor interface from training set (Type III).
[g]Values in parentheses are randomly predicted values. The precision of random prediction is calculated as: the total number of interaction sites residues/the total number of residues.
[h]Values in parentheses are randomly predicted values. The recall of random prediction is calculated as: the total number of predicted residues as interaction sites by each method/the total number of residues.
[i]SVMs can't predict any interaction site.

Among three trimed data sets, the performance of CRFs is next to the best performance obtained by SVMs method according to F1-measure and CC. Removing some non-interfacial residues from training set (in trimed data set) reduces the performance of CRFs, since these removed residues still contain useful information for predicting interaction sites. We will discuss this phenomenon in the following section.

Both CRFs and ME are exponential models based on maximum entropy principle. From the result, we can notice that the CRFs outperform ME greatly in most data sets, which shows that CRFs method are more suitable for labeling protein interaction sites than ME method. The performance of ANN is worst according to our experiment.

3.3 The effect of different ratio of positive and negative examples for CRFs and SVMs

We generated a series of training sets by randomly removing different number of negative examples from the original Complete Type I data set. The evaluation result of F1-measure and CC changing with the ratio of positive and negative examples is shown in Figure 3. We can see that the performance of CRFs is stable when the ratio of Pos/Neg is between 0.3 and 0.7 and the CRFs achieve the best performance when Pos/Neg is about 0.4. It means that CRFs can obtain the best performance when only very few negative examples are removed. When the ratio of Pos/Neg is above 0.7, the (CC) performance of CRFs will decline. SVMs can not obtain any interaction sites when the Pos/Neg ratio is below 0.4. So the effect of the Pos/Neg ratio for SVMs is more serious than it is for CRFs. This experiment has been done only on Type I data set, while results on other two data sets may be different.

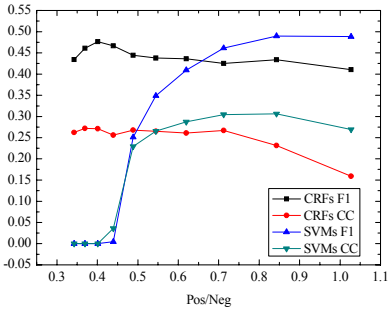
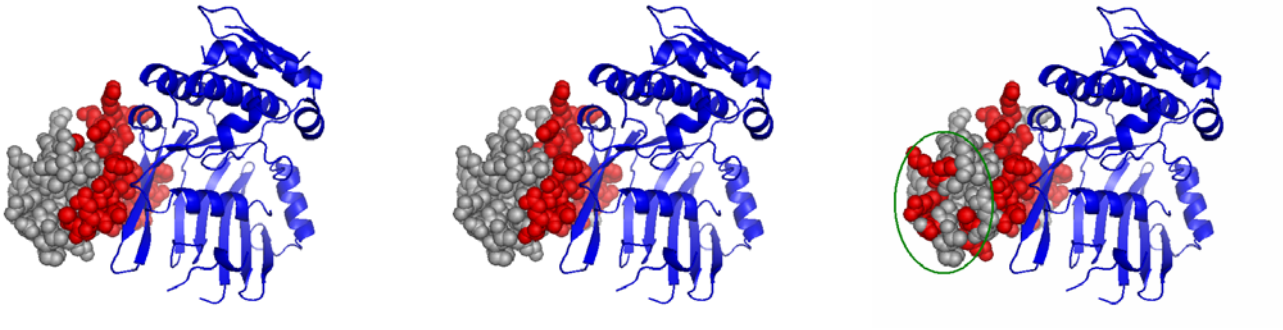


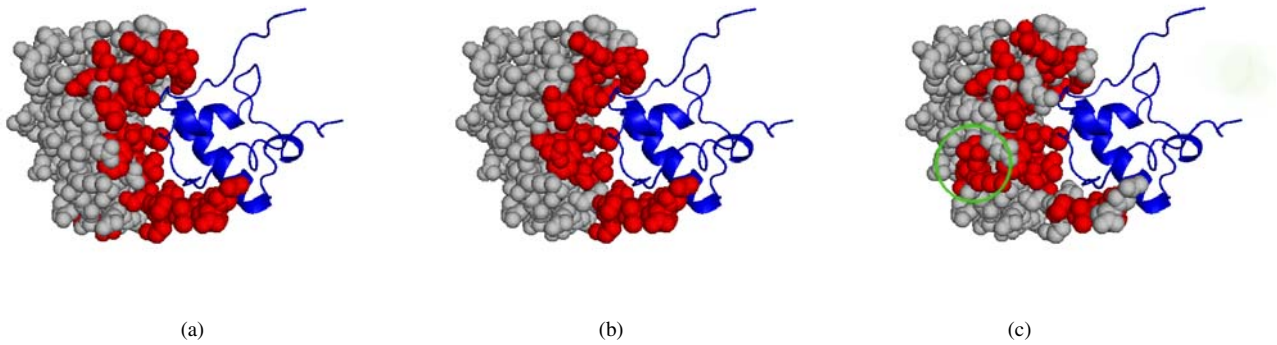
Fig. 3. CRFs and SVMs performance changing curves with different ratio of Pos/Neg. Pos/Neg is the ratio between positive and negative examples in training set.

3.4 Some predicted examples by CRFs and SVMs



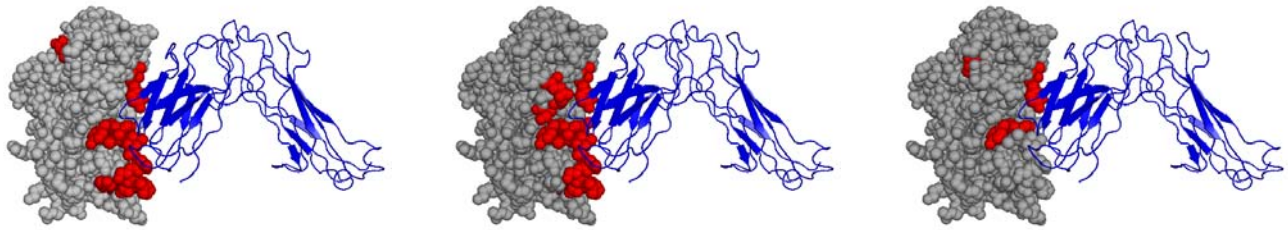
(a) (b) (c)

Fig. 4. Predicted interface residues (red color) for deleted in conserved region of Rad21/Rec8 like protein (Kleisin; PDB code 1W1W:E) identified by (a) CRFs and (c) SVMs. (b) The actual interface residues. The binding partner is the RecF/RecN/SMC N terminal domain (blue).



(a) (b) (c)

Fig. 5. Predicted interface residues (red color) for 30S ribosomal subunit S6 (PDB code 1FJG:F) identified by (a) CRFs and (c) SVMs. (b) The actual interface residues. The binding partner is 30S ribosomal subunit S18 (blue). The residues within the green circle in (c) are far away from the binding site.



(a) (b) (c)

Fig. 6. Predicted interface residues (red color) for Streptococcal pyrogenic enterotoxin C (SpeC) (PDB code 1KTK:A) identified by (a) CRFs and (c) SVMs. (b) The actual interface residues. The binding partner is Human T cell receptor beta chain (blue).

We give some examples that are predicted by SVMs and CRFs trained on trimed Type I data set. The first example is the SC SMC1HD:SCC1-C complex (Haering C.H., *et al.* 2004). The Kleisin is the conserved region of Rad21/Rec8 like protein which has 22 residues located on the interface with its partner according to the above definition of interaction residue [Figure 4(b)]. The CRFs predict 27 residues to be interface which covers 20 interfacial residues (recall: 91%, precision: 74%) [Figure 4(a)]. The SVMs predict 21 residues to be interface which covers 13 interfacial residues (recall: 59%, precision: 62%) [Figure 4(c)]. We can see that most of the false positives from SVMs locate on outside of the actual

interface, i.e. the green cycle in Figure 4(c). CRFs can successfully distinguish interface and noninterface residues for this protein.

The second example is complex of the ribosomal subunit 30S, a complex of 20 polypeptide chains with a 1522 nucleotide long 16S RNA (Carter*, et al.*, 2000). The S6 chain is in our data set and the interface between S6 and S18 was studied by us. The prediction results are shown in Figure 5. The interface residues of S6 (binding with S18) centralize in its hollow [Figure 5(b)]. This interface region is accurately identified by CRFs covering about 86% of the actual binding site with a precision of 73% [Figure 5(a)]. The prediction result by SVMs covers only 68% of the actual binding sites with a precision of 56%, including a error region far away from the binding site i.e. residues within the green circle of Figure 5(c).

The last example given by us is complex of sreptococcal pyrogenic enterotoxin C (SpeC) with a human T cell receptor beta chain (Sundberg, E.J., *et al*., 2002). There are 17 residues located on the interface[Figure 6(b)]. CRFs can label the majority these residues with coverage of 65% [Figure 6(a)], while SVMs only correctly label 4 interface residues with coverage of only 23.5%[Figure 6(c)]. Clearly, it is difficult to characterize the interfacial feature by SVMs.

3.5 Test CRFs on extended feature

We add residue conservation features to CRFs method which is also trained on Type I data set. These features are obtained from conservation score in the ConSurf-HSSP database (Glaser*, et al.*, 2005), which are different from that of Chung (2006) and Res (2005). Experimental result is tabulated in Table 4, from which we can see that the value of CC of CRFs-2 on two data types all descend. According to our experimental result, better performance can not be obtained by adding these features to CRFs.

Table 4. Performance of CRFs using basic and extended features

Type	Precision	Recall	F1	Accuracy	CC
Complete	0.516	0.304	0.383	0.750	0.252
Trim	0.376	0.510	0.433	0.659	0.202

4 CONCLUSION AND FUTURE WORK

Protein-protein interaction sites prediction is tackled as a sequence labeling problem using conditional random fields which is different from conventional classification based methods. Features used for conditional random fields include sequence profile and residue accessible surface area of spatially neighboring residues. Comparative experiments of CRFs-based method and other classification-based methods including SVMs, ANN, and ME on 1276 nonredundant chains of hetero complexes show that CRFs-based method achieves the best performance on complete data sets. On the trimmed data sets, the performance of CRFs is comparable with state-of-the-art methods, such as ANN and SVMs. CRFs method is more robust than conventional classification methods when using data sets with different ratio of positive and negative examples. Our study indicates the feasibility of using CRFs to predict protein-protein interaction sites and guides specific experiments for biologists.

In our experiment, the residue conservation feature did not contribute to the performance of CRFs. It shows that simply adding this feature to CRFs is not suitable for this task. Choosing proper features is a challenging work and we will investigate more effective features in the future. Information of binding protein chains will also be considered in our future work.

ACKNOWLEDGMENTS

This research work is funded by National Natural Science Foundation of China (60673019). The authors would like to thank the reviewers for their valuable comments. Thanks go to Xuan-Hieu Phan from Japan Advanced Institute of Science and Technology for providing the original version of FlexCRFs source code, Dr. Chih-Jen Lin from National Taiwan University for providing the LIBSVM tool, and Le Zhang from University of Edinburgh for providing the Maximum Entropy Modeling Toolkit.

REFERENCES

Altschul, S.F., et al. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, Nucleic Acids Research, 25, 3389-3402.

Berman, H.M., et al. (2000) The Protein Data Bank, Nucleic Acids Research, 28 235-242

Bradford, J.R. and Westhead, D.R. (2005) Improved prediction of protein–protein binding sites using a support vector machines approach, 21, 1487-1494.

Carter, A.P., et al. (2000) Functional insights from the structure of the 30S ribosomal subunit and its interactions with antibiotics, Nature, 407, 340-348.

Chang, C.-C. and Lin, C.-J. (2001) LIBSVM : a library for support vector machines, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

Chen, H. and Zhou, H.-X. (2005) Prediction of interface residues in protein-protein complexes by a consensus neural network method: Test against NMR data, Proteins: Structure, Function, and Bioinformatics, 61, 21-35.

Chinchor, N. (1998) MUC-7 Named Entity Task Definition. Proceedings of The Seventh Message Understanding Conference.

Chung, J.-L., et al. (2006) Exploiting sequence and structure homologs to identify protein-protein binding sites, Proteins: Structure, Function, and Bioinformatics, 63, 630-640.

Cyril Dominguez, R.B., and Alexandre M. J. J. Bonvin (2003) HADDOCK: A Protein-Protein Docking Approach Based on Biochemical or Biophysical Information, J. Am. Chem. Soc., 125, 1731 -1737.

Fariselli, P., et al. (2002) Prediction of protein-protein interaction sites in heterocomplexes with neural networks, Eur. J. Biochem., 269, 1356-1361.

Freitag, D. and McCallum, A. (2000) Information Extraction with HMM Structures Learned by Stochastic Optimization. Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence. 584-589.

Glaser, F., et al. (2005) The ConSurf-HSSP Database: The Mapping of Evolutionary Conservation Among Homologs Onto PDB Structures, PROTEINS: Structure, Function, and Bioinformatics, 58, 610-617.

Haering C.H., et al. (2004) Structure and stability of cohesin's Smc1-kleisin interaction, Mol Cell., 15, 951-964.

Kabsch, W. and Sander, C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen bonded and geometrical features, Biopolymers, 22, 235-242.

Kim, H. and Park, H. (2003) Protein secondary structure prediction based on an improved support vector machines approach, Protein Engineering Design and Selection, 16, 553-560.

Koike, A. and Takagi, T. (2004) Prediction of protein-protein interaction sites using support vector machines, Protein Engineering Design and Selection, 17, 165-173.

Kojic, M. and Holloman, W.K. (2004) BRCA2-RAD51-DSS1 Interplay Examined from a Microbial Perspective, Cell Cycle, 3, 247-248.

Lafferty, J., et al. (2001) Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. 18th International Conference on Machine Learning (ICML). 282-289.

Letovsky, S. and Kasif, S. (2003) Predicting protein function from protein/protein interaction data: a probabilistic approach, Bioinformatics, 19, i197-i204.

Liang, S., et al. (2006) Protein binding site prediction using an empirical scoring function, Nucleic Acids Research, 34, 3698-3707.

Lichtarge, O., et al. (2002) Evolutionary traces of functional surfaces along G protein signaling pathway, Methods Enzymol, 344, 536-556.

Liu, Y., et al. (2004) Comparison of probabilistic combination methods for protein secondary structure prediction, Bioinformatics, 20, 3099-3107.

Liu, Y., et al. (2005) Segmentation Conditional Random Fields (SCRFs): A New Approach for Protein Fold Recognition. ACM International conference on Research in Computational Molecular Biology (RECOMB05). 408-422.

Mccallum, A., et al. (2000) Maximum Entropy Markov Models for Information Extraction and Segmentation. Proceedings of the Seventeenth International Conference on Machine Learning. 591-598.

McGinnis, S. and Madden, T.L. (2004) BLAST: at the core of a powerful and diverse set of sequence analysis tools, Nucleic Acids Research, 32, W20-W25.

Nabieva, E., et al. (2005) Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps Bioinformatics, 21, i302-i310.

Ofrana, Y. and Rosta, B. (2003) Predicted protein-protein interaction sites from local sequence information, FEBS Letters, 544, 236-239.

Phan, X.-H. and Nguyen, L.-M. (2005) FlexCRFs: Flexible Conditional Random Field Toolkit, http://www.jaist.ac.jp/~hieuxuan/flexcrfs/flexcrfs.html.

Rabiner, L.R. (1989) A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proceedings of the IEEE, 77, 257-286.

Ratnaparkhi, A. (1996) A Maximum Entropy Model for Part-Of-Speech Tagging. Proceedings of the Conference on Empirical Methods in Natural Language Processing.

Res, I., et al. (2005) An evolution based classifier for prediction of protein interfaces without using protein structures, Bioinformatics, 21, 2496-2501.

Rosenfeld, R. (1996) A maximum entropy approach to adaptive statistical language modeling, Computer, Speech and Language, 10, 187-228.

Rost, B. and Sander, C. (1994) Conservation and Prediction of Solvent Accessibility in Protein Families, PROTEINS: Structure, Function, and Genetics, 20, 216-226.

Sowa, M.E., et al. (2001) Prediction and confirmation of a site critical for effector regulation of RGS domain activity, Nat Struct Biol, 8, 234-237.

Sutton, C. and McCallum, A. (2006) An Introduction to Conditional Random Fields for Relational Learning. In Getoor, L. and Taskar, B. (eds), Introduction to Statistical Relational Learning. MIT Press.

Sundberg, E.J., et al. (2002) Structures of two streptococcal superantigens bound to TCR beta chains reveal diversity in the architecture of T cell signaling complexes, Structure 10, 687-699.

Wanga, B., et al. (2006) Predicting protein interaction sites from residue spatial sequence profile and evolution rate, FEBS Letters, 580, 380-384.

Zhou, H.-X. (2004) Improving the understanding of human genetic diseases through predictions of protein structures and protein-protein interaction sites, Curr Med Chem, 11, 539-549.

Zhou, H.-X. and Shan, Y. (2001) Prediction of protein interaction sites from sequence profile and residue neighbor list, Proteins: Structure, Function, and Genetics, 44, 336-343.

BMC Bioinformatics

BioMed Central

Research

A computational approach for detecting peptidases and their specific inhibitors at the genome level

Lisa Bartoli†[1], Remo Calabrese†[1], Piero Fariselli*[1], Damiano G Mita[2] and Rita Casadio[1]

Address: [1]Laboratory of Biocomputing, CIRB/Department of Biology, University of Bologna, Bologna, Italy and [2]Department of Experimental Medicine, Biotechnology and Molecular Biology Section, Second University of Naples, Naples, Italy

Email: Lisa Bartoli - lisa@biocomp.unibo.it; Remo Calabrese - remo@biocomp.unibo.it; Piero Fariselli* - piero@biocomp.unibo.it; Damiano G Mita - mita@igb.cnr.it; Rita Casadio - casadio@alma.unibo.it

* Corresponding author †Equal contributors

from Italian Society of Bioinformatics (BITS): Annual Meeting 2006
Bologna, Italy. 28–29 April, 2006

Published: 8 March 2007

BMC Bioinformatics 2007, **8**(Suppl 1):S3 doi:10.1186/1471-2105-8-S1-S3

This article is available from: http://www.biomedcentral.com/1471-2105/8/S1/S3

Abstract

Background: Peptidases are proteolytic enzymes responsible for fundamental cellular activities in all organisms. Apparently about 2–5% of the genes encode for peptidases, irrespectively of the organism source. The basic peptidase function is "protein digestion" and this can be potentially dangerous in living organisms when it is not strictly controlled by specific inhibitors. In genome annotation a basic question is to predict gene function. Here we describe a computational approach that can filter peptidases and their inhibitors out of a given proteome. Furthermore and as an added value to MEROPS, a specific database for peptidases already available in the public domain, our method can predict whether a pair of peptidase/inhibitor can interact, eventually listing all possible predicted ligands (peptidases and/or inhibitors).

Results: We show that by adopting a decision-tree approach the accuracy of PROSITE and HMMER in detecting separately the four major peptidase types (Serine, Aspartic, Cysteine and Metallo- Peptidase) and their inhibitors among a non redundant set of globular proteins can be improved by some percentage points with respect to that obtained with each method separately. More importantly, our method can then predict pairs of peptidases and interacting inhibitors, scoring a joint global accuracy of 99% with coverage for the positive cases (peptidase/inhibitor) close to 100% and a correlation coefficient of 0.91%. In this task the decision-tree approach outperforms the single methods.

Conclusion: The decision-tree can reliably classify protein sequences as peptidases or inhibitors, belonging to a certain class, and can provide a comprehensive list of possible interacting pairs of peptidase/inhibitor. This information can help the design of experiments to detect interacting peptidase/inhibitor complexes and can speed up the selection of possible interacting candidates, without searching for them separately and manually combining the obtained results. A web server specifically developed for annotating peptidases and their inhibitors (HIPPIE) is available at http://gpcr.biocomp.unibo.it/cgi/predictors/hippie/pred_hippie.cgi

Background

Peptidases (proteases) are proteolytic enzymes essential for the life of all organisms. The relevance of peptidases is proved by the fact that 2–5% of all genes encode for peptidases and/or their homologs irrespectively of the organism source [1]. In the SwissProt database [2] about 18% of sequences are annotated as "undergoing proteolytic processing", and there are over 550 known and putative peptidases in the human genome. It is also worth noticing that more than 10% of the human peptidases are under investigation as drug targets [3]. Proteases are responsible for a number of fundamental cellular activities, such as protein turnover and defense against pathogenic organisms. Since the basic protease function is "protein digestion", these proteins would be potentially dangerous in living organisms, if not fully controlled. This is one of the major reasons for the presence of their natural inhibitors inside the cell. All peptidases catalyze the same reaction, namely the hydrolysis of a peptide bond, but they are selective for the position of the substrate and also for the amino acid residues close to the bond that undergoes hydrolysis [4,5]. There are different classes of peptidases identified by the catalytic group involved in the hydrolysis of the peptide bond. However the majority of the peptidases can be assigned to one of the following four functional classes:

- Serine Peptidase
- Aspartic Peptidase
- Cysteine Peptidase
- Metallopeptidase

In the serine and cysteine types the catalytic nucleophile can be the reactive group of the amino acid side chain, a hydroxyl group (serine peptidase) or a sulfhydryl group (cysteine peptidase). In aspartic and metallopeptidases the nucleophile is commonly "an activated water molecule". In aspartic peptidases the side chains of aspartic residues directly bind the water molecule. In metallopeptidases one or two metal ions hold the water molecule in place and charged amino acid side chains are ligands for the metal ions. The metal may be zinc, cobalt or manganese, and a single metal ion is usually bound by three amino acid ligands [3]. Among the different ways to control their activity, the most important is through the interactions of the protein with other proteins, namely naturally occurring peptidase inhibitors. Peptidase inhibitors can or cannot be specific for a certain group of catalytic reactions. In general there are two kinds of interactions between peptidases and their inhibitors: the first one is an irreversible process of "trapping", leading to a stable peptidase-inhibitor complex; the second one is a reversible process in which there is a tight binding reaction without any chemical bond formation [4,6-8]. A shift of interest towards the mode of interaction of protein inhibitors with their targets is due to the possibility of designing new synthetic inhibitors. The research is driven by the many potential applications in medicine, agriculture and biotechnology.

In the last years, an invaluable source of information about proteases and their inhibitors has been made available through the MEROPS database [9], so that it is possible to search for known peptidase sequences (or structures) or peptidase-inhibitor sequences (or structures). Exploiting this source, in this paper we address the problem of relating a peptidase sequence (or inhibitor) with sequences that can putatively but reliably inhibit it (or proteases that can be inhibited by it). To this aim we implemented a method that first and reliably discriminates whether a given sequence is a peptidase or a peptidase-inhibitor, and afterwards gives a list of its putative interacting ligands (proteases/inhibitors). Our method provides answers to the following questions:

1) Given a pair of sequences, are they a pair of protease and inhibitor that can interact?

2) Given a protease (or inhibitor), can we predict the list of the proteins in a defined database that can inhibit (or be inhibited by) the query protein?

3) Given a proteome, can we compute the list of peptidases and their relative inhibitors for each protease class?

Results and discussion

Testing PROSITE and HMMER-Pfam capability of detecting MEROPS peptidases and inhibitors

The first step of our analysis is to evaluate the performance of PROSITE [10] on data sets of proteases and inhibitors, as derived from MEROPS [1,3,4,9]. Our method focuses on the four major classes of peptidases and their inhibitors as identified by the catalytic group involved in the hydrolysis of the peptide bond: Serine, Aspartic, Cysteine and Metallo- peptidases. In MEROPS there are annotations for 38 peptidase patterns and 20 inhibitor patterns. We adopted peptidases and inhibitors as annotated in MEROPS as the positive class (2793 peptidases and 1209 inhibitors). The negative counterpart was taken from PAPIA [11], and comprises non-inhibitor and non-peptidase non homologue sequences (2091 sequences) (see "Data sets" section). We start by running PROSITE on the PAPIA+MEROPS data sets. PROSITE can or cannot find a correct match. If a known inhibitor (peptidase) sequence is matched by a PROSITE inhibitor (peptidase) pattern we count it as a True Positive (TP), otherwise it is labeled as a False Negative (FN). Conversely, PAPIA sequences having

Table 1: PROSITE discriminating capability towards MEROPS proteases and inhibitors.

Data sets	*Q2*	*Q [pos]*	*Q [neg]*	*P [pos]*	*P [neg]*	*C*
MEROPS (proteases)/PAPIA(sequences)	0.78	0.61	1	1	0.66	0.63
MEROPS (inhibitors)/PAPIA (sequences)	0.90	0.73	1	1	0.86	0.79

For definition see Scoring indexes

a match with a PROSITE inhibitor (peptidase) pattern are False Positives (FP); otherwise they are True Negatives (TN).

In Table 1 the results obtained by filtering the PROSITE and the PAPIA+MEROPS data sets are listed. It is worth noticing that the PROSITE pattern search produces almost zero False Positives on the MEROPS+PAPIA data set, although with a significant number of False Negatives. This indicates that the method has a quite high specificity, but low coverage. In other words, a match has a high likelihood to be a true positive (high specificity); however due to the low coverage (61%, Table 1), still a non-match label may indicate a false negative (with a likelihood of 14% and 34% for inhibitors and peptidases, respectively).

In Table 2 we report the same type of analysis using HMMER-Pfam [12]. From the results it is evident that on average this method outperforms PROSITE. Our finding is in agreement with early observations indicating that Pfam is a better detection method than PROSITE [13]. We find that Pfam is more balanced than PROSITE, although with a slightly lower specificity (Table 1, 2).

The decision-tree method

The high level of PROSITE specificity prompted us to combine this pattern matching procedure with HMMER-Pfam by adopting a decision-tree method in order to take advantage of the features of both approaches (as described in Methods and shown in Figure 1). The results of the combined approach (as depicted into the flow chart of Figure 1) are then listed in Table 3. It appears that the overall performance is slightly improved over HMMER-Pfam alone. This is so particularly when the coverage of the positive class (Q [pos]) is considered.

Detection of possible protease-inhibitor interacting pairs

The most relevant issue addressed by this paper is the measure of the detection accuracy of possible peptidase-inhibitor interacting pairs. The idea is to address questions related to the putative peptidase/inhibitor interaction (or combined discriminative efficacy). In order to test the combined accuracy of our decision-tree with respect to the PROSITE and HMMER-Pfam methods, we have taken all the possible sequence combinations of our selected data set, namely peptidase/inhibitor, peptidase/PAPIA, inhibitor/PAPIA, peptidase/peptidase, inhibitor/inhibitor, PAPIA/PAPIA, excluding the self-combinations (a sequence against itself). By adopting this procedure we ended up with 18,559,278 pairs that were scored as described below.

We divided MEROPS peptidase sequences in four classes according to their biological activity: Aspartic (A), Cysteine (C), Metallo (M) and Serine (S) peptidases. We labeled the inhibitors in the same way, with the exception that one more class is present for them, labeled as U; this set clusters all the inhibitors that are able to inhibit to some extent all types of peptidases (the so called Universal inhibitors).

Among the 18,559,278 possible pairs only those pairs pertaining to proteases and inhibitors of the same class are counted as members of the positive class (amounting only to 7 % of all possible pairs). All the remaining pairs are labeled as negative examples. On this data set we tested PROSITE, HMMER-Pfam and the combined decision-tree (Figure 2). We also tested the reverse decision-tree in which HMMER and PROSITE are swapped (alternative combinations are equivalent). In Table 4 it is shown that despite of the fact that the overall accuracy (Q2) is very high for all methods, the decision-tree outperforms all the others as the increased values of all scoring indexes indicate. Actually, the decision-tree approach

Table 2: HMMER-Pfam discriminating capability towards MEROPS proteases and inhibitors For definition see Scoring indexes.

Data sets	*Q2*	*Q [pos]*	*Q [neg]*	*P [pos]*	*P [neg]*	*C*
MEROPS (proteases)/PAPIA(sequences)	0.94	0.93	0.98	0.98	0.92	0.91
MEROPS (inhibitors)/PAPIA (sequences)	0.93	0.83	0.99	0.98	0.91	0.85

For definition see Scoring indexes

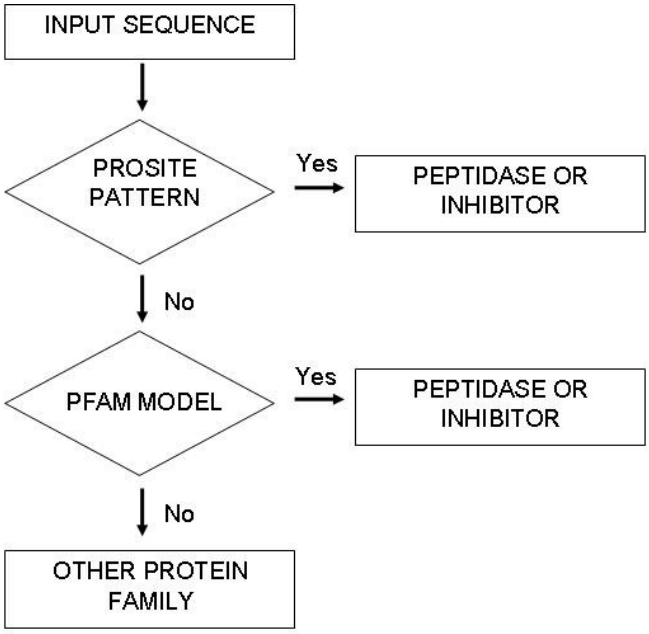


Figure 1
Flow-chart of the decision-tree method for the detection of peptidases and inhibitors.

shows the highest coverage and accuracy for both the peptidase-inhibitor interacting class and the negative set. It is also worth noticing that the correlation coefficient (C), that indicates the displacement from the random prediction, is very high for the decision-tree and it outperforms the second best method (HMMER) of 9 percentage points, with a false positive rate close to 0 (100-Q [neg]x100). This finding indicates that the decision-tree method can successfully be adopted to predict pairs of interacting peptidase/inhibitor, in order to sort out the subsets of possible interacting pairs of interest.

Annotating peptidases and their inhibitors in Human and Mouse genomes

We applied the decision-tree method scored above to perform a large-scale genome annotation of peptidases and corresponding inhibitors of the Human and Mouse proteomes. We retrieved all known coding sequences and novel peptides from Ensembl35 [November 2005] [14]. The Human proteome consists of 33,869 sequences; the Mouse proteome contains 36,471 sequences. The decision- tree method is compared with PROSITE and HMMER-Pfam in singling out peptidases and inhibitors (Table 5 and 6, respectively). The predictive performance of the decision-tree method in predicting putative pairs of peptidase/inhibitor for each major class of both proteomes is reported in Table 7. Our results corroborate the view that among peptidases, the Aspartic class is less populated than the other three and this is so in both proteomes. For inhibitors, the less populated classes are Aspartic, Cysteine and Universal.

Web server

In order to facilitate the user's search for protease/inhibitor interactions, we implemented a very simple web interface that exploits our developed decision-tree system. In practice it is possible to paste a sequence and the system checks whether that sequence is a protease or an inhibitor candidate. If the decision-tree returns a positive answer the server will provide the putative class among the four and the list of all possible known inhibitors (or proteases that might be inhibited by the query sequence). Furthermore, the web server furnishes also the corresponding lists of possible ENSEMBL protease-codes (or inhibitor-codes) of the Human and Mouse proteomes that belong to the predicted class of proteins and that can interact with the query sequence.

The server is available at [15].

Conclusion

In this paper we developed a decision-tree based method that exploits the features of PROSITE and HMMER-Pfam in annotating peptidases and inhibitors and that is capable of correctly and reliably predict whether a given peptidase can or cannot interact with an inhibitor. The decision-tree discriminates peptidases or inhibitors with a score as high as 96% (97%) of correct predictions, improving both the coverage and the specificity of the positive class (pairs peptidase/inhibitor of the same class and pairs peptidase/Universal inhibitor) over PROSITE and HMMER-Pfam. Furthermore the decision-tree method is capable of predicting if a given protein pair is a pair of protease and inhibitor that can interact. This task can help in sorting out and speeding up the selection of possible interacting partners. Given a protease or an inhibitor the decision-tree method computes the list of

Table 3: Decision-Tree discriminating capability towards MEROPS proteases and inhibitors.

Data sets	*Q2*	*Q [pos]*	*Q [neg]*	*P [pos]*	*P [neg]*	*C*
MEROPS (proteases)/PAPIA(sequences)	0.96	0.93	1	1	0.91	0.92
MEROPS (inhibitors)/PAPIA (sequences)	0.97	0.94	0.99	0.99	0.97	0.95

For definition see Scoring indexes

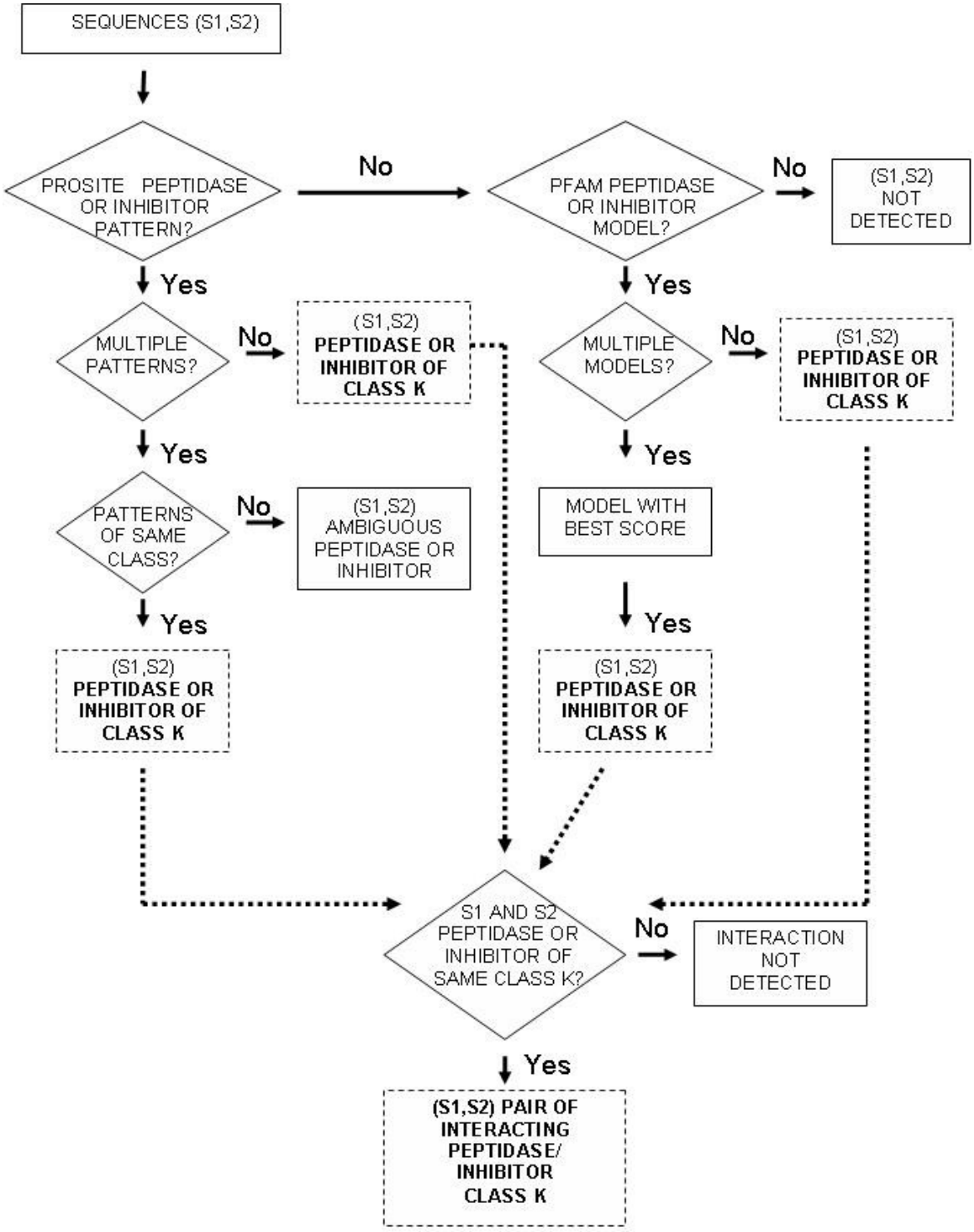


Figure 2
Flow-chart of the decision-tree method for the detection of possible peptidases/inhibitors interacting pairs. Each of the two input sequences is searched against Prosite and, in case of negative answer, against HMMER-Pfam. In both cases, when there is a match, the decision-tree method checks for the presence of multiple matches (patterns or models respectively). If there is a match, the method gives a positive answer for each sequence and only the peptidase and inhibitor sequences of the same class K (A, C, M, S, U) are classifed as possible interacting pairs.

Table 4: Scoring the detection of possible protease-inhibitor interactions with different methods.

Methods	*Q2*	*Q [pos]*	*Q [neg]*	*P [pos]*	*P [neg]*	*C*
Prosite	0.96	0.44	1	1	0.96	0.67
Hmm-Pfam	0.97	0.82	0.99	0.84	0.98	0.82
Decision-Tree	0.99	0.89	1	0.95	0.99	0.91
Reverse Decision-Tree	0.90	0.82	0.99	0.84	0.99	0.80

For definition of the statistical indexes see Scoring indexes

the proteins in a defined database that can inhibit or that can be inhibited by the query protein. Finally, given a proteome the system provides the lists of peptidases and their relative inhibitors for each discriminated class.

Methods

The data sets

MEROPS database, hosted at the Sanger Institute [1,3,4], is the main resource of information on peptidases and their natural and synthetic inhibitors [9]. In this paper we refer to the 7.10 Merops release (22/07/2005) that contains 30909 peptidase sequences (including homologs) and 3690 inhibitor sequences (including homologs). We downloaded all data with the exclusion of sequences unassigned to any family. We then ended up with a set that contains chains of 167 protease families and 52 inhibitors families. We retained only the most abundant MEROPS functional classes: Serine, Aspartic, Cysteine and Metallo- peptidases.

From the MEROPS database we removed all sequences belonging to Threonin and Glutamic classes and the sequences of unknown catalytic type because for these groups no natural inhibitors are known. Our final peptidase set contains 2793 protein sequences. We also filtered out the inhibitor data set removing the family sequences that have an auto-inhibitory peptide at the N-terminus. Actually, these are peptidases with self-inhibitory peptides (I09 and I29 families). The inhibitor data set contains 1209 protein sequences. These two data sets represent the positive examples class for our classification method.

As a negative data set we have taken a non-redundant set of representative protein structures, of known function and not including peptidases and their inhibitors. This set was extracted from PAPIA (PArallel Protein Information Analysis system) [11]. The final PAPIA-derived set consists of 2091 protein chains.

The decision-tree method

In order to predict if pairs of peptidase and inhibitor belong to the same class, we developed a system that performs two consecutive tasks: 1) extracts protease and inhibitor sequences from a given data set; 2) tests if they are compatible (if the inhibitor can interact with the protease). In order to solve this problem, we implemented a decision-tree method that processes the information obtained from PROSITE [10] and HMMER-Pfam [12,13] and detects if a query sequence could be annotated as peptidase or inhibitor. We selected PROSITE and Pfam since they are highly reliable methods for a classification task (see results).

PROSITE is a database of protein families and domains. It consists of biologically significant sites, patterns and profiles that help to reliably identify to which known protein family (if any) a new sequence belongs. We scanned all the data set against the PROSITE database (release 26/04/2005) with the "ps_scan" tool. Since we are interested in the detection of the presence/absence of patterns in the sequences, we used ps_scan for this task. We also set the options of skipping profiles and frequently matching patterns (unspecific) [10].

Table 5: Detection of proteases and inhibitors in the Human proteome.

	Peptidases					*Inhibitors*					
	A	*C*	*M*	*S*	*TOT*	*A*	*C*	*M*	*S*	*U*	*TOT*
Prosite	40	171	192	227	630	0	45	4	147	24	220
Pfam	164	575	626	698	2063	10	67	1099	446	52	1674
Decision-tree	183	600	654	735	2172	10	81	1099	501	68	1759

The different classes discriminated are: A = Aspartic-peptidase or inhibitor; C = Cysteine-peptidase or inhibitor; M = Metallo-peptidase or inhibitor; S = Serine-peptidase or inhibitor; U = Universal family of inhibitors.

Table 6: Detection of proteases and inhibitors in the Mouse proteome.

Method	*Peptidases*					*Inhibitors*					
	A	*C*	*M*	*S*	*TOT*	*A*	*C*	*M*	*S*	*U*	*TOT*
Prosite	96	181	234	242	753	0	59	4	171	21	255
Pfam	202	636	650	658	2146	16	84	1125	453	64	1742
Decision-tree	218	663	713	697	2291	16	91	1125	503	70	1805

For labels see Table 5.

Pfam is a large collection of multiple sequence alignments and hidden Markov models covering many common protein domains and families [12]. Pfam is a database consisting of two parts, the first is the curated part of Pfam-A containing over 7,973 protein families, and the second is Pfam-B automatically generated for a more comprehensive coverage of known proteins. We downloaded a copy of the Pfam database (22/08/2005) and we used the HMMER package to search our protein sequence data set against the Pfam-A models. The Pfam library contains all local Pfam-A HMMs in a HMMER searchable format. We run the "hmmpfam" program to search for matches to a query sequence and the Pfam model of interest. The Pfam models annotated in MEROPS specific for our classes are 145, and 36 for proteases and inhibitors, respectively. If a sequence matches more than one model we consider the model with highest score and lowest e-value as the best.

The basic engine is described in the flow-chart of Figure 1, where for a given input sequence, we first look for PROSITE matching, and then in case of negative answer, we proceed using a profile-HMM scanning (HMMER-Pfam). From Figure 1, it is clear that if a PROSITE match is found, no more search is carried out. This works only if the first method has a high specificity (even when the sensitivity is low).

In order to predict whether a pair of sequences can be a peptidase and an inhibitor of the same class we run the decision-tree twice: first with the PROSITE and Pfam parameters relative to the peptidase search, and second adopting the model and the regular expressions corresponding to the inhibitors.

Scoring indexes

All the results are evaluated using the following measures of efficiency. The fraction of correctly predicted residues is:

$$Q2 = (TP+TN)/(TP+TN+FP+FN)$$

where *TP* and *TN*, *FP* and *FN* are respectively: the number of true positives, true negatives, false positives and false negatives.

The correlation coefficient is defined as:

$$cor = [TP*TN - FP * FN]/D$$

where *D* is the normalization factor

$$D = [(TP+FP)(TP+FN)(TN+FP)(TN+FN)]^{1/2}$$

The coverage or the sensitivity for the positive and negative classes is defined as:

$$Q[pos] = TP/[TP+FN]$$

$$Q[neg] = TN/[TN+FP]$$

Table 7: Detection of peptidase/inhibitor pairs in the Human and Mouse proteomes.

Proteome	*AA*	*CC*	*MM*	*SS*	*AU*	*CU*	*MU*	*SU*	*TOTAL*
Human	1830	48600	718746	368235	12444	40800	44472	49980	1285107 (0.2 %)*
Mouse	3488	60333	802125	350591	15260	46410	49910	48790	1376907 (0.2%)*

AA = Aspartic peptidase/Aspartic peptidase inhibitor pairs; CC = Cysteine peptidase/Cysteine peptidase inhibitor pairs; MM = Metallo-peptidase/Metallo-peptidase inhibitor pairs; SS = Serine peptidase/Serine peptidase inhibitor pairs; AU = Aspartic peptidase/Universal peptidase inhibitor pairs; CU = Cysteine peptidase/Universal peptidase inhibitor pairs; MU = Metallo-peptidase/Universal peptidase inhibitor pairs; SU = Serine peptidase/Universal peptidase inhibitor pairs.

* percentage of all the possible sequence pairs (573.537.646 and 665.048.685, for Human and Mouse genomes, respectively)

The probability of correct predictions (accuracy or specificity) is computed as:

$P[pos] = TP/[TP+FP]$

$P[neg] = TN/[TN+FN]$

Authors' contributions

All the authors contributed to the ideas and planning of this project. RC and LB carried out the analysis and wrote the software. Rita Casadio and PF supervised the study. GDM contributed to the inhibitors/peptidases analysis. PF, Rita Casadio, RC and LB contributed to the writing of this manuscript. All authors read and approved the final manuscript.

Acknowledgements

We thank MIUR for the following grants: PNR-2003 grant delivered to PF, a PNR 2001–2003 (FIRB art.8) and PNR 2003 projects (FIRB art.8) on Bioinformatics for Genomics and Proteomics and LIBI-Laboratorio Internazionale di Bioinformatica, both delivered to RC. This work was also supported by the Biosapiens Network of Excellence project, which is funded by the European Commission within its FP6 Programme, under the thematic area "Life sciences, genomics and biotechnology for health", contract number LSHG-CT-2003-503265.

This article has been published as part of *BMC Bioinformatics* Volume 8, Supplement 1, 2007: Italian Society of Bioinformatics (BITS): Annual Meeting 2006. The full contents of the supplement are available online at http://www.biomedcentral.com/1471-2105/8?issue=S1.

References

1. Rawlings ND, O'Brien EA, Barrett AJ: ***MEROPS* : the protease database.** *Nucleic Acids Res* 2002, **30:**343-346.
2. Boeckmann B, Bairoch A, Apweiler R, Blatter MC, Estreicher A, Gasteiger E, Martin MJ, Michoud K, O'Donovan C, Phan I, Pilbout S, Schneider M: **The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003.** *Nucleic Acids Res* 2003, **31:**365-370.
3. Rawlings ND, Morton FR, Barrett AJ: ***MEROPS* : the peptidase database.** *Nucleic Acids Res* 2006, **34:**D270-D272.
4. Rawlings ND, Tolle DP, Barrett AJ: **Evolutionary families of peptidase inhibitors.** *Biochem J* 2004, **378:**705-716.
5. Tyndall JDA, Nall T, Fairlie DP: **Proteases universally recognize beta strands in their active sites.** *Chemical Reviews* 2005, **105(3):**973-999.
6. Gettins PGW: **Serpin structure, mechanism, and function.** *Chemical Reviews* 2002, **102:**4751-4803.
7. Krowarsch D, Cierpicki T, Jelen F, Otlewski J: **Canonical protein inhibitors of serine proteases.** *Cell Mol Life Sci* 2003, **60:**2427-2444.
8. Jackson RM, Russell RB: **The serine protease inhibitor canonical loop conformation: examples found in extracellular hydrolases, toxins, cytokines and viral proteins.** *J Mol Biol* 2000, **296:**325-334.
9. **MEROPS – the Peptidase database** [http://merops.sanger.ac.uk/]
10. Falquet L, Pagni M, Bucher P, Hulo N, Sigrist CJ, Hofmann K, Bairoch A: **The PROSITE database, its status in 2002.** *Nucleic Acids Res* 2002, **30:**235-238.
11. Akiyama Y, Onizuka K, Noguchi T, Ando M: **Parallel Protein Information Analysis (PAPIA) system running on a 64-node PC Cluster.** In *Proc the 9th Genome Informatics Workshop (GIW'98) Universal Academy Press*; 1998:131-140.
12. Bateman A, Coin L, Durbin R, Finn RD, Hollich V, Griffiths-Jones S, Khanna A, Marshall M, Moxon S, Sonnhammer EL, Studholme DJ, Yeats C, Eddy SR: **The Pfam protein families database.** *Nucleic Acids Res* 2004, **32(Database):**D138-D141.
13. Eddy SR: **Profile hidden Markov models.** *Bioinformatics* 1998, **14(9):**755-63.
14. Hubbard T, Andrews D, Caccamo M, Cameron G, Chen Y, Clamp M, Clarke L, Coates G, Cox T, Cunningham F, Curwen V, Cutts T, Down T, Durbin R, Fernandez-Suarez XM, Gilbert J, Hammond M, Herrero J, Hotz H, Howe K, Iyer V, Jekosch K, Kahari A, Kasprzyk A, Keefe D, Keenan S, Kokocinsci F, London D, Longden I, McVicker G, Melsopp C, Meidl P, Potter S, Proctor G, Rae M, Rios D, Schuster M, Searle S, Severin J, Slater G, Smedley D, Smith J, Spooner W, Stabenau A, Stalker J, Storey R, Trevanion S, Ureta-Vidal A, Vogel J, White S, Woodwark C, Birney E: **Ensembl 2005.** *Nucleic Acids Res* 2005, **33(Database):**D447-D453.
15. [http://gpcr.biocomp.unibo.it/cgi/predictors/hippie/pred_hippie.cgi].

Appendice A

Per poter valutare l'efficienza di un metodo predittivo, o confrontarlo ad altri esistenti, è necessario introdurre uno (o più) indici che forniscano una misura quantitativa delle capacità di generalizzazione. La più semplice misura della capacità predittiva è data dalla frazione delle predizioni corrette totali rispetto all'intero numero di risposte possibili. Se n è il numero di classi discriminate dal predittore (nei casi esaminati in questa tesi n=2), N è il numero totale di esempi osservati e p_i (veri positivi e veri negativi)è il numero di esempi correttamente predetti nella classe i si definisce:

$$Q_n = \sum_{i=1}^{N} \frac{p_i}{N} \qquad \text{(A.1)}$$

La A.1 per due sole classi (+ e -) la possiamo scrivere anche:

$$Q_2 = (TP + TN)/(TP + TN + FP + FN) \qquad \text{(A.2)}$$

Dove TP sono i veri positivi, TN i veri negativi, FP i falsi positivi ed FN i falsi negativi

Gli indici $Q_{i,}$ detti anche “coverage”, indicano la frazione di predizioni corrette per ciascun tipo di classe. Indicando con N il numero di esempi osservati e con u_i il numero di esempi sottopredetti nella classe i (ovvero i FN per la classe + ed i FP per la classe -):

$$Q_i = \frac{p_i}{N} = \frac{p_i}{p_i + u_i} \qquad \text{(A.3)}$$

Oppure:

Q[+]=TP/(TP+FN) e Q[-]=TN/(TN+FP)

Per quanto molto comunemente usati Q_n e Q_i , essi non tengono in considerazione le sovrappredizioni e possono essere affetti dall'abbondanza relativa delle classi nel data base. Perciò un indice più significativo è il coefficiente di correlazione (Matthews, 1975):

$$Corr_i = \frac{(p_i n_i - u_i o_i)}{[(p_i + u_i)(p_i + o_i)(n_i + u_i)(n_i + o_i)]^{1/2}} \qquad (A.4)$$

dove n_i è il numero di esempi che non appartenendo alla classe i sono stati correttamente assegnati ad altre classi ed o_i è il numero di esempi sovrapredetti nella classe i. Questo coefficiente assume i valori nell'intervallo [-1,1]; 1 indica la predizione completamente corretta, mentre 0 indica una predizione non migliore di quella casuale. Possiamo riscrivere il coefficiente di correlazione come:

$$Corr = \frac{(TP \cdot TN - FP \cdot FN)}{[(TP + FN) \cdot (TP + FP) \cdot (TN + FP) \cdot (TN + FN)]^{1/2}} \qquad (A.5)$$

Un altro indice utilizzato per tenere conto delle sovrapredizioni è la probabilità delle predizioni corrette (o accuratezza della classe i):

$$P_i = \frac{p_i}{p_i + o_i} \qquad (A.6)$$

Oppure :

P[+]=TP/(TP+FP), P[-]=TN/(TN+FN)

Un’altra misura standard è rappresentata dalla curva ROC (Receiver Operating Characteristic). La ROC è un grafico del tasso dei veri positivi (TPR= Q(i)) in funzione del tasso di falsi positivi (FPR= 1- P(i)).

Infine, è molto importante assegnare uno score di attendibilità ad ogni predizione, calcolato come:

$$RI(i)=10*abs(O(i)-t)*w(i) \qquad (A.7)$$

Dove O(i) è il valore dell’output della classe i, t è la soglia di decizione per l’assegnazione ad una della due classi (t=0.5) e w(i) è il peso della classe i (per ognuna delle due classi il peso è pari a 0.5)

Bibliografia

Ahmad F e Bigelow CC (1986). Estimation of the stability of globular proteins. *Biopolymers*. 25:1623-1633.

Akasako A, Haruki M, Oobatake M, Kanaya S (1997). Conformational stabilities of Escherichia coli RNase HI variants with a series of amino acid substitutions at a cavity within the hydrophobic core. *J Biol Chem.* 272(30):18686-18693.

Alber T (1989a). Mutational effects on protein stability. *Annu Rev Biochem.* 58:765-98.

Alber, T. (1989b). Prediction of Protein Structure and the Principles of Protein Conformation (Fasman, G. D., Ed.) :161, Plenum, New York.

Alber T, Sun DP, Wilson K, Wozniak JA, Cook SP, Matthews BW (1987). Contributions of hydrogen bonds of Thr 157 to the thermodynamic stability of phage T4 lysozyme. *Nature.* 330(6143):41-46.

Altschul SF Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W e Lipman DI (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucl Acids Res.* 25(17): 3389-3402.

Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990). Basic local alignment search tool. *J Mol Biol.* 215:403-310.

Altschul,S.F., Gish,W. (1996) Local alignment statistics. *Meth Enz.* 266:460-480.

Anfinsen CB (1973). Principles that govern the folding of protein chains. *Science*. 181(96):223-30.

Anfinsen CB, Haber E, Sela M, White FH (1961). The kinetics of formation of native ribonuclease during oxidation of the reduced polypeptide chain. *Proc Natl Acad Sci* USA. 47:1309-1314.

Anfinsen CB, Scheraga HA (1975). Experimental and theoretical aspects of protein folding. *Adv Protein Chem*. 29:205-300.

Bairoch A, Apweiler R (2000). The SWISS-PROT protein sequence database and its supplement TrEMBL. *Nucl Acids Res*. 28:45-48.

Baldi, P., Brunak, S., Chauvin, Y., Andersen, C. A. and Nielsen, H. (2000) Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16, 412-424.

Barbujani, G. and Goldstein, D. B. (2004) Africans and Asians abroad: genetic diversity in *Europe. Annu. Rev. Genomics Hum. Gene*.t, 5, 119-150.

Bash PA, Singh UC, Langridge R, Kollman PA (1987). Free energy calculations by computer simulation. *Science*. 236(4801):564-568.

Baum J, Dobson CM, Evans PA (1989). Characterization of a partly folded protein by NMR methods: studies on the molten globule state of guinea pig .alpha.-lactalbumin. *Biochemistry* 28:7-13.

Bava K. A., et al (2004) ProTherm, version 4.0: thermodynamic database for proteins and mutants *Nucleic Acids Res* 32 D120-D121

Bell, J. (2004) Predicting disease using genomics. *Nature*, 429, 453-456.

Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE (2000). The Protein Data Bank. *Nucleic Acids Res*. 28:235-242.

Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M. C., Estreicher, A., Gasteiger, E., Martin, M. J., Michoud, K., O'Donovan, C., Phan, I., Pilbout, S. and Schneider, M. (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003 *Nucleic Acids Res*., 31, 365-370.

Bryngelson JD, Onuchic JN, Socci ND, Wolynes PG (1995). Funnels, pathways, and the energy landscape of protein folding. A synthesis. *Proteins.* 21:167-195.

Bryngelson JD, Wolynes PG (1987). Spin glasses and the statistical mechanics of protein folding. *Proc. Natl. Acad. Sci.* USA. 84: 7524-7528.

Capriotti E, Fariselli P, Casadio R. (2004) A neural network-based method for predicting protein stability changes upon single point mutations. *Bioinformatics*, 20 Suppl 1:I63-I68

Capriotti, E., Fariselli, P., **Calabrese**, **R**. and Casadio, R. (2005a) Predicting protein stability changes from sequences using support vector machines. *Bioinformatics*, 21 (Suppl 2), ii54-ii58.

Capriotti, E., Fariselli, P. and Casadio, R. (2005b) I-Mutant2.0: predicting stability changes upon mutation from the protein sequence or structure. *Nucleic Acids Res*., 33 (Web server issue), W306-W310.

Capriotti, E., **Calabrese**, **R**. and Casadio, R. (2006) Predicting the insurgence of human genetic diseases associated to single point protein mutations with support vector machines and evolutionary information. *Bioinformatics*, 22(22):2729-34

Chen YW, Fersht AR, Henrick K (1993). Contribution of buried hydrogen bonds to protein stability. The crystal structures of two barnase mutants. *J Mol Biol.* 234(4):1158-1170.

Chih-Chung Chang and Chih-Jen Lin (2001) LIBSVM : a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

Collins, F. S., Brooks, L. D. and Chakravarti, A. (1998) A DNA polymorphism discovery resource for research on human genetic variation. *Genome Res.*, 8, 1229-1231.

Conde, L., Vaquerizas, J. M., Santoyo, J., Al-Shahrour, F., Ruiz-Llorente, S., Robledo, M. and Dopazo, J. (2004) PupaSNP Finder: a web tool for finding SNPs with putative effect at transcriptional level. *Nucleic Acids Res.*, 32, W242-248

Daggett,V. e Fersht A.R. (2003) Is there a unifying mechanism for protein folding? *Trends Biochem Sci.* 28:18-25.

Dayhoff,M.O., Scwartz,R.M., Orcutt,B.C. (1978) Atlas of protein sequence and structure. *National Biomedical Research Foundation Editions, Wasinghton* 5: 315-335.

De Gennes PG (1975). Collapse of a polymer chain in poor solvents. *J Phys.* (Paris) 36:L55-L57.

Dang LX, Merz KM, Kollman PA (1989). Free energy calculations on protein stability: Thr-157 .fwdarw. Val-157 mutation of T4 lysozyme. *J Am Chem Soc*. 111: 8505-8508.

Dill KA (1985). Theory for the folding and stability of globular proteins. *Biochemistry*. 24:1501-1509.

Dill KA, Alonso D0V e Hutchinson K (1989). Thermal stabilities of globular proteins *Biochemistry.* 28:5439-5449.

Edmonds, C. A., Lillie, A. S. and Cavalli-Sforza, L. L. (2004) Mutations arising in the wave front of an expanding population. *Proc. Natl. Acad. Sci. USA*, 101, 975-979.

Evans PA, Dobson CM, Kautz RA, Hatfull G, Fox RO (1987). Proline isomerism in staphylococcal nuclease characterized by NMR and site-directed mutagenesis. *Nature*. 329:266-268.

Frauenfelder H, Wolynes P G, (1994). Biomolecules: Where the Physics of Complexity and Simplicity Meet. *Phys. Today.* 47:58-64.

Funahashi J., Takano,K., and Yutani,K. (2001) Are the parameters of various stabilization factors estimated from mutant human lysozymes compatible with other proteins? *Protein Eng*. 14,127-134

Gills D, Rooman M (1996). Stability changes upon mutation of solvent-accessible residues in proteinsevaluated by database-derived potentials. *J Mol Biol.* 257(5):1112-1126.

Gills D, Rooman M. (1997) Predicting protein stability changes upon mutation using database-derived potentials: solvent accessibility determines the importance of local versus non-local interactions along the sequence. *J. Mol. Biol*. 272,276-290.

Goldstein, D. B. and Cavalleri, G. L. (2005) Genomics: understanding human diversity. *Nature*, 437, 1241-1242.

Goto Y e Fink AL (1989). Conformational states in .beta.-lactamase: molten-globule states at acidic and alkaline pH with high salt. .*Biochemistry* 28:945-952.

Goto Y, Calciano LJ e Fink AL (1990). Acid-induced folding of proteins. *Proc Natl Acad Sci* USA. 87:573-577.

Gotoh O (1982). An improved algorithm for matching biological sequences. *J Mol Biol.* 162:705-708.

Gromiha MM, An J, Kono H, Oobatake M, Uedaira H, Sarai A (1999). ProTherm: Thermodynamic Database for Proteins and Mutants. *Nucleic Acids Res.* 27(1):286-288.

Gromiha, MM, An J, Kono H, Oobatake M, Uedaira H, Prabakaran P, Sarai A (2000) ProTherm, version 2.0: thermodynamic database for proteins and mutants. *Nucleic Acids Res*. 28,283-285.

Guerois R, Nielsen JE, Serrano L (2002). Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *J Mol Biol*. 320(2):369-87.

Henikoff S , Henikoff JG (1992). Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci* USA. 89:10915-10919.

Karchin, R., Diekhans, M., Kelly, L., Thomas, D.J., Pieper, U., Eswar, N., Haussler, D. and Sali, A. (2005) LS-SNP: large-scale annotation of coding non-synonymous SNPs based on multiple information sources. *Bioinformatics*, 21, 2814-2820.

Lee C, Levitt M (1991). Accurate prediction of the stability and activity effects of site-directed mutagenesis on a protein core. *Nature*. 352(6334):448-451.

Levinthal C (1968). Are there pathways for protein folding? *J Chem Phys.* 85: 44-45.

Li, W., Jaroszewski, L. and Godzik, A. (2001) Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics*, 17, 282-283.

Lumry R, Biltonen R e Brandts JF (1966). Validity of the two-state hypothesis for conformational transitions of proteins. *Biopolymers* 4: 917-944.

Lumry R, e Eyring H (1954). *J . Phys. Chem*. 58, 110.

Matsumura M, Becktel WJ, Matthews BW (1988). Hydrophobic stabilization in T4 lysozyme determined directly by multiple substitutions of Ile 3. *Nature*. 334(6181):406-410.

Matthews, B. W. (1987a) Harvey Lect. 81, 33.

Matthews, B. W. (1987b) Genetic and structural analysis of the protein stability problem. *Biochemistry* 26, 6885-6888.

Miyazawa S, Jernigan RL (1994). Protein stability for single substitution mutants and the extent of local compactness in the denatured state. *Protein Eng.* 7(10):1209-1220.

Needleman SB, Wunsch CD (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol.* 48:443-453.

Ng, P. C. and Henikoff, S. (2002) Accounting for human polymorphisms predicted to affect protein function. *Genome Res*., 12, 436-446.

Ng, P. C. and Henikoff, S. (2003) SIFT: Predicting amino acid changes that affect protein function. *Nucleic Acids Res*., 31, 3812-3814.

Pace CN (1975). The stability of globular proteins. *CRC Crit Rev Biochem.* 3:1-43.
Post, CB e Zimm BH (1979). Internal condensation of a single DNA molecule. *Biopolymers* 18, 1487-1501.

Privalov PL (1979). Stability of proteins: small globular proteins. *Adv Protein Chem.* 33:167-241.

Privalov PL, Gill, SJ (1988). Stability of protein structure and hydrophobic interaction. *Adv Protein Chem.* 39:191-234,

Privalov PL, Kechinashvili NN (1974). A thermodynamic approach to the problem of stabilization of globular protein structure: a calorimetric study. *J .Mol Biol.* 86:665-684.

Privalov PL, Griko YuV, Venyaminov SYu, Kutyshenko VP (1986). Cold denaturation of myoglobin. *J Mol Biol.* 190:487-498.

Ramensky, V., Bork, P. and Sunyaev, S. (2002) Human non-synonymous SNPs: server and survey. *Nucleic Acids Res*., 30, 3894-3900.

Reumers, J., Schymkowitz, J., Ferkinghoff-Borg, J., Stricher, F., Serrano, L. and Rousseau, F. (2005) SNPeffect: a database mapping molecular phenotypic effects of human non-synonymous coding SNPs. *Nucleic Acids Res*., 33 (Database is-sue), D527-D532.

Riva, A. and Kohane, I. S. (2002) SNPper: retrieval and analysis of human SNPs. *Bioinformatics*, 18, 1681-1685.

Rost B (1999). The twilight zone of protein alignments. Protein Engineering 12, 85-94.

Sanchez IC (1979). Phase Transition Behavior of the Isolated Polymer Chain. *Macromolecules*. 12: 980-988.

Sander C, Schneider R (1991). Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins*. 9:56-68.

Schölkopf B. Smola A. J. (2002) Learning with kernels MIT Press

Schrier MY e Schrier EE (1976). Transfer free energies and average static accessibilities for ribonuclease A in guanidinium hydrochloride and urea solutions. *Biochemistry* 15:2607-2612.

Selkoe D. J., (2003) Folding proteins in fatal ways *Nature* 426, 900-904

Sherry, S. T., Ward, M. H., Kholodov, M., Baker, J., Phan, L., Smigielski, E. M. and Sirotkin, K. (2001) dbSNP: the NCBI database of genetic variation. *Nucleic Acids Res*., 29, 308-311.

Shortle D, Meeker A K (1989). Residual structure in large fragments of staphylococcal nuclease: effects of amino acid substitutions. *Biochemistry* 28:936-944.

Shortle D, Meeker AK, Freire E (1988). Stability mutants of staphylococcal nuclease: large compensating enthalpy-entropy changes for the reversible denaturation reaction. *Biochemistry* 27:4761-4768.

Shortle D, Stites WE, Meeker AK (1990). Contributions of the large hydrophobic amino acids to the stability of staphylococcal nuclease. *Biochemistry*. 29(35):8033-8041.

Simonson T, Brunger AT (1992). Thermodynamics of protein-peptide interactions in the ribonuclease-S system studied by molecular dynamics and free energy calculations. *Biochemistry*. 31(36):8661-8674.

Smith TF, Waterman MS (1981). Identification of common molecular subsequences. *J Mol Biol.* 147:195-197.

Stenson, P. D., Ball, E. V., Mort, M., Phillips, A. D., Shiel, J. A., Thomas, N. S., Abeysinghe, S., Krawczak, M. and Cooper, D. N. (2003) Human Gene Mutation Database (HGMD): 2003 update. *Hum. Mutat*., 21, 577-581.

Takano K, Ogasahara K, Kaneda H, Yamagata Y, Fujii S, Kanaya E, Kikuchi M, Oobatake M, Yutani K (1995). Contribution of hydrophobic residues to the stability of human lysozyme: calorimetric studies and X-ray structural analysis of the five isoleucine to
valine mutants. *J Mol Biol.* 254(1):62-76.

Takano K, Yamagata Y, Fujii S, Yutani K (1997). Contribution of the hydrophobic effect to the stability of human lysozyme: calorimetric studies and X-ray structural analyses of the nine valine to alanine mutants. *Biochemistry*. 36(4):688-698.

Tanford C (1968). Protein denaturation. *Adv Protein Chem.* 23:121-282.

Tanford C (1970). Protein denaturation. C. Theoretical models for the mechanism of denaturation. *Adv Protein Chem.* 24:1-95.

Tanford C (1980). The Hydrophobic Effect, 2nd ed., Wiley, New York.

Taylor,W.R. (1986) The classification of amino acid conservation. *J Theor Biol.* 119: 205-218.

Tidor B, Karplus M (1991). Simulation analysis of the stability mutant R96H of T4 lysozyme. *Biochemistry*. 30(13):3217-3228.

Tissot AC, Vuilleumier S, Fersht AR (1996). Importance of two buried salt bridges in the stability and folding pathway of barnase. *Biochemistry*. 35(21):6786-6794.

Topham CM, Srinivasan N, Blundell TL (1997). Prediction of the stability of protein mutants based on structural environment-dependent amino acid substitution and propensity tables. *Protein Eng.* 10(1):7-21.

van Gunsteren WF, Mark AE (1992). On the interpretation of biochemical data by molecular dynamics computer simulation. *Eur J Biochem*. 204:947-961.

Wang, D. G., Fan, J. B., Siao, C. J., Berno, A., Young, P., Sapolsky, R., Ghandour, G., Perkins, N., Winchester, E., Spencer, J., Kruglyak, L., Stein, L., Hsie, L., Topa-loglou, T., Hubbell, E., Robinson, E., Mittmann, M., Morris, M. S., Shen, N., Kil-burn, D., Rioux, J., Nusbaum, C., Rozen, S., Hudson, T. J., Lipshutz, R., Chee, M. and Lander, E. S. (1998) Large-scale identification, mapping, and genotyping of single-nucleotide polymorphisms in the human genome. *Science*, 280, 1077-1082.

Wang, Z. and Moult, J. (2001) SNPs, protein structure, and disease. *Hum. Mutat*., 17, 263-270.

Wetzel R, Perry LJ, Baase WA, Becktel WJ (1988). Disulfide bonds and thermal stability in T4 lysozyme. *Proc Natl Acad Sci* USA. 85(2):401-405.

Xu J, Baase WA, Baldwin E, Matthews BW (1998). The response of T4 lysozyme to large-to-small substitutions within the core and its relation to the hydrophobic effect. *Protein Sci*. 7(1):158-177.

Yu MH, King J (1984). Single amino acid substitutions influencing the folding pathway of the phage P22 tail spike endorhamnosidase. *Proc Natl Acad Sci* U S A. 81(21):6584-6588.

Yue, P. and Moult, J. (2006) Identification and analysis of deleterious human SNPs. *J. Mol. Biol*., 356, 1263-1274

Yutani K, Ogasahara K, Tsujita T, Sugino Y (1987). Dependence of conformational stability on hydrophobicity of the amino acid residue in a series of variant proteins substituted at a unique position of tryptophan synthase alpha subunit. *Proc Natl Acad Sci* USA. 84(13):4441-4444.

Vapnik V. Chervonenkis A. Y., (1971) On the uniform convergence of relative frequencies of events to their probabilities *Th. Prob. And its Applications*, 17 (2): 264-280

Vapnik V. Chervonenkis A. Y., (1991) The necessary and sufficient conditions for consistency in the empirical risk minimization method *Pattern Recognition and Image Analysis* 1(3):283-305

Zhou,H., Zhou,Y. (2002) Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. *Protein Sci.* 11:2714-2726.

Zhou H, Zhou Y (2004) Quantifying the effect of burial of amino acid residues on protein stability *Proteins* 54:315-322

Zimm BH, Bragg JK (1959). Theory of the phase transition between helix and random coil in polypeptidechains. *J. Chem. Phys.* 31:526-529.