Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione
Corso di Laurea in Bioingegneria

Analisi di terne contigue di residui per il riconoscimento di interfacce.

Laureando:
Giacomo Cecchinato

Relatore:
Ch.mo Prof. Carlo Ferrari

Anno accademico 2013/2014

Indice

Introduzione

Le interazioni proteina-proteina, e proteina-ligando hanno un ruolo cruciale nel controllo metabolico, nella trasduzione dei segnali, nella regolazione genica e in altri aspetti dell'organizzazione strutturale e funzionale della cellula. Il legame tra un enzima ed un ligando può, ad esempio, determinare un'attivazione o un'inibizione dello stesso. Se la proteina è un recettore il legame può risultare in un agonismo o in un antagonismo. Per questo uno degli ambiti più importanti della ricerca bioinformatica è il "docking proteico", definito come la predizione, attraverso metodi computazionali, della struttura quaternaria che un complesso proteico assumerebbe in un organismo vivente sotto determinate condizioni. Nell'uomo si stima vi siano tra le 100.000 e i 600.000 interazioni tra proteine ma sperimentalmente si è arrivati a descriverne circa 50.000. Le tecniche di predizione si rendono ancor più necessarie nello studio di specie per cui si hanno pochi dati sperimentali. Il futuro obbiettivo è quello di poter sintetizzare in vitro proteine a seconda della loro funzione biologica. Le tecniche di docking sono infatti comunemente sfruttate per:

- identificare in un database molecole che potrebbero legarsi ad una proteina di interesse;
- predire configurazione finale di un complesso proteina-proteina o proteina-ligando;
- predire ligandi che potrebbero attivare enzimi in grado di degradare molecole tossiche;

Gli strumenti computazionali odierni possono fornire utili informazioni a differenti livelli di risoluzione. Il presente lavoro si concentra su un aspetto di grande importanza che è la predizione dei siti di legame, ovvero, data una struttura proteica di cui si conosca la configurazione terziaria, determinare quali siano i residui facenti parte di un sito di interazione. Questo permetterebbe di restringere la ricerca della configurazione del complesso ad una porzione precisa della superficie delle proteine interagenti, di cui si potranno analizzare altre caratteristiche come, ad esempio la forma. In questo modo la ricerca sarebbe ristretta ad una zona particolare e non estesa a tutta la proteina.

Nonostante la potenza di calcolo abbia permesso di aumentare i gradi di libertà degli algoritmi, come si può vedere dal numero di pubblicazioni sul "flexible docking", essa rimane comunque uno dei limiti di qualsiasi approccio.

La tesi effettua un'analisi di strutture proteiche di cui è nota la possibilità di interazione e si effettua una predizione delle interfacce analizzando gruppi di tre residui consecutivi. Questo viene fatto per ridurre la risoluzione rispetto al tipo di approccio più frequente

che analizza il singolo residuo, e per considerare l'influenza reciproca di residui adiacenti. Figura 1.

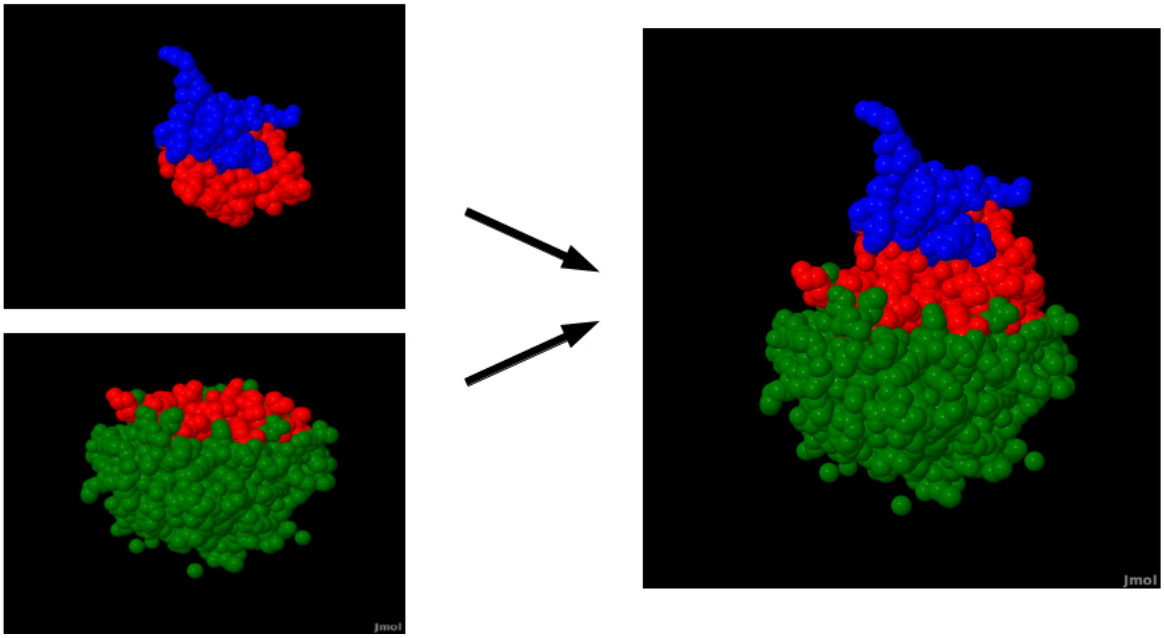

Figura 1: Due proteine che formano un complesso. L'immagine rappresenta, tramite JMol[18], a sx i raggi di Van der Waals degli atomi di due proteine, dove in rosso abbiamo gli atomi di interfaccia, e a dx il complesso da esse formato.

Parte I

Identificazione di interfacce: stato dell'arte

Capitolo 1

Predizione dell'interazione tra proteine

Vengono qui descritte le tecniche che tentano di determinare a grandi linee se due proteine possano interagire.

Per predire una possibile interazione si possono utilizzare dei criteri di vicinanza. La prossimità tra proteine può essere intesa sotto diversi aspetti: si può parlare di proteine spazialmente vicine tra loro, oppure si analizza la distanza tra i geni che le codificano. Tale distanza è da intendersi o come distanza spaziale nel genoma o come distanza nella scala evolutiva.

Di seguito vengono elencati i criteri di prossimità che possono essere indice di un'affinità funzionale.

1.1 Comparti cellulari

Proteine che si trovano in comparti cellulari differenti (nucleo, mitocondri, Golgi..) saranno considerate, in generale, non in condizione di interagire per il semplice fatto che non avranno la possibilità di incontrarsi. Questa osservazione è sfruttata più per la validazione di altri metodi o per fissare degli standard piuttosto che caratterizzare degli algoritmi di predizione a se stanti. Infatti alcuni esperimenti in vitro portano a non escludere a priori questa possibilità.

1.2 Alberi filogenetici

La similarità tra alberi filogenetici da una idea della misura della co-evoluzione dei geni codificanti proteine (Figura 1.1). Infatti le proteine interagenti solitamente evolvono di pari passo poichè mutazioni di una portano alla perdita o alla compensazione dell'altra al fine di preservare le capacità di interazione. Gli alberi filogenetici non rappresentano necessariamente l'esatta storia evolutiva di un gene, o di un dato organismo, ed in effetti nella maggior parte dei casi non lo fanno. Analisi basate solo su questo tipo di dati possono essere alterate da diversi fattori, tra cui il trasferimento genico orizzontale, l'ibridizzazione

tra specie diverse situate a grande distanza sull'albero prima che l'ibridizzazione stessa avvenisse, l'evoluzione convergente e la conservazione delle sequenze geniche.

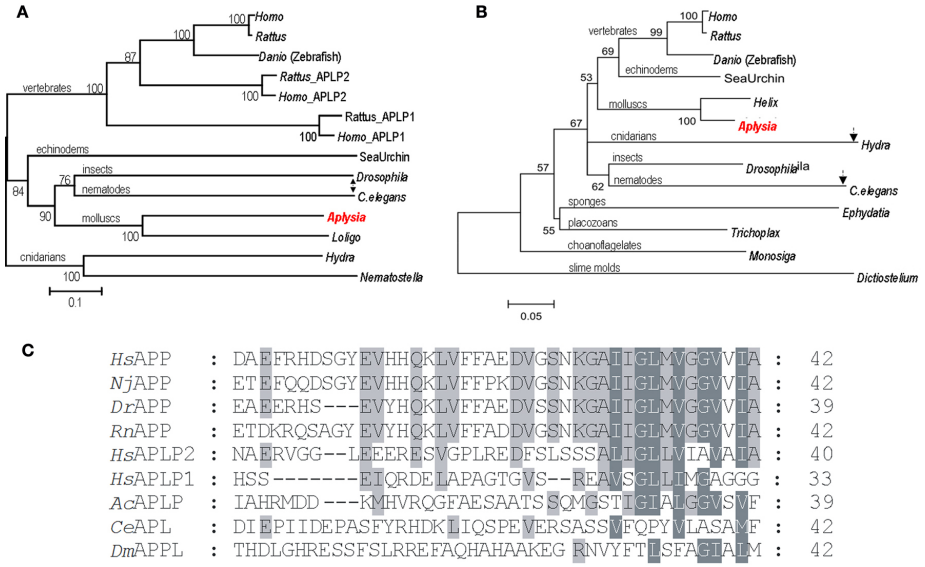

Figura 1.1: Filogenia delle proteine associate al morbo di Alzheimer. (A) Albero filogenetico del precursore della proteina amiloide (B) Albero filogenetico delle Presenilins. (C) Conservazione di un sito di una proteina amiloide.

1.3 Contesto genomico

L'analisi del contesto genomico si basa sull'osservazione che le proteine funzionali all'interazione tendono ad essere codificate da geni spazialmente vicini nel genoma. Ciò avviene probabilmente per motivi di efficienza di trascrizione. La prossimità è intesa in termini di vicinanza genomica, fusione genica e co-occorrenza di geni nel genoma [2].

1.3.1 Vicinanza genomica

Alcuni geni adiacenti nel genoma saranno probabilmente dediti alla codifica di proteine simili. Esistono dei metodi dedicati, appunto, alla misura della distanza tra geni. A livello quantitativo la vicinanza genomica sembra essere il tipo di informazione più indicativo, ma ha il limite di poter essere valutato solo per proteine che hanno ortologhi in batteri e archeobatteri. I geni ortologhi sono geni ibridi creati per unire porzioni di geni differenti o per unire un gene ad un promotore per regolare la trascrizione.

1.3.2 Fusione genica

Durante l'evoluzione i geni possono fondersi in geni più grandi o dividersi in più piccoli secondo i meccanismi rispettivamente di fusione e fissione genica schematizzati in Figura 1.2. Questa organizzazione genomica si verifica per favorire l'efficienza della trascrizione di geni correlati. L'osservazione che due geni possono comparire anche in un gene più grande, detto "sequenza Rosetta stone", è indicativo del fatto che questi sono funzionalmente legati.

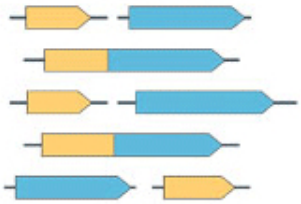

Figura 1.2: Schematizzazione del processo di fusione genica: due geni codificanti proteine interagenti vanno a fondersi assieme.

1.3.3 Co-occorrenza

La ricerca di una co-occorrenza è basata sull'idea che geni funzionalmente correlati tendano a comparire nello stesso genoma indipendentemente dalla loro posizione. Questo è uno dei metodi più potenti in quanto non è detto che tutte le proteine che interagiscono abbiano i propri geni fusi o vicini tra loro. Metodologicamente, tuttavia, è il metodo più rozzo poichè il rapporto segnale/rumore è molto piccolo ed è necessario un confronto accurato tra molti genomi per migliorare l'accuratezza.

1.4 Letteratura

Il numero di pubblicazioni biomediche è in continua crescita. Vi sono numerose ricerche focalizzate all'estrazione automatica di informazioni dalle pubblicazioni scientifiche. Queste includono ricerca di co-occorrenza di termini, o la presenza di sinonimi in articoli scientifici. Ad esempio due proteine tra di loro funzionali hanno una certa probabilità di comparire in uno stesso articolo. [10]

Capitolo 2

Predizione di interfacce di interazione

Vengono qui descritte i metodi principali per determinare le zone di interazione delle proteine.

La maggior parte delle tecniche di predizione di interfacce utilizzano vari parametri relativi ai residui e sfruttano metodi di machine learning (ML). Questi costituiscono una struttura che permette di utilizzare in modo coerente informazioni eterogenee. Di seguito vengono esposte le categorie di parametri dei residui più usate e gli algoritmi di ML più comuni per la soluzione di questo problema.

2.1 Informazioni sui residui

Le informazioni possono essere raggruppate in:

- caratteristiche strutturali: proprietà fisico-chimiche proprie dei residui e proprietà relative al loro collocamento nella struttura terziaria;
- caratteristiche sequenziali: basate sulla codifica, tramite matrici di "score" posizione specifici, dei residui e dei relativi vicini, generati allineando la sequenza con sequenze omologhe;
- caratteristiche energetiche: analizzano le interazioni molecolari presenti nella struttura terziaria;
- altro: molte caratteristiche, anche apparentemente prive di significato biologico, possono essere utilizzate;
- combinazioni delle precedenti.

2.1.1 Informazioni strutturali

Analisi su larga scala di complessi proteici hanno mostrato che i residui di interfaccia presentano differenti caratteristiche strutturali e fisico-chimiche. Si possono fare delle con-

siderazioni di carattere generale riguardo a quali di queste siano maggiormente informative [7].

- idrofobicità: residui idrofobici sono più frequenti sulle interfacce dei complessi;
- carica elettrica: residui carichi, in particolare l'arginina, sono frequenti sulle interfacce;
- superficie accessibile al solvente (ASA): è più alto nelle interfacce è una delle caratteristiche più efficaci nella predizione di interfacce;
- fattore cristallografico: è più basso nei residui di interfaccia;
- catene laterali: sono meno predisposte ad assumere configurazioni alternative se facenti parte di interfacce; più rigide sono più possono diminuire il costo entropico della formazione dei complessi;
- conservazione: non è determinabile la sua efficacia come parametro per la predizione in quanto è stato dimostrato, in alcuni lavori che le interfacce non sono molto più conservate di altre parti della proteina;
- β sheets e α elica: sono strutture che si trovano sulle interfacce, più facilmente le prime più raramente le seconde.

Oltre a quelli precedentemente descritti è possibile valutare anche parametri da loro derivati: ad esempio l'accessibilità al solvente relativa alla catena laterale o principale oppure "scores" basati sul fattore cristallografico oppure indici di protrusione.

2.1.2 Informazioni sulla sequenza: frequenze amminoacidiche

La struttura primaria è una delle caratteristiche maggiormente utilizzate per la codifica delle proteine. Infatti, si ha la possibilità di effettuare una predizione basandosi sulla sola sequenza proteica, anche se la struttura della proteina non è disponibile. I predittori basati su questo tipo di informazioni sono chiamati predittori di interfacce "de novo" o "ab initio" perché non sono derivati da altri parametri. You et al. [3], ad esempio, utilizzano vettori che contengono le frequenze di terne di residui. Vengono utilizzati 7 gruppi di residui, classificati per momento di dipolo e volume della catena laterale. Si considerano quindi terne ottenute dalle permutazioni di questi 7 gruppi ottenendo così vettori da 343 parametri (Figura 2.1), dove ogni valore corrisponde alla frequenza di una determinata terna nella proteina di interesse. Non vengono valutate tutte le permutazioni dei 20 residui in quanto si avrebbe un vettore di dimensione 20^3, troppo esteso per le possibilità di calcolo attuali. Questi metodi potrebbero sembrare simili a quelli menzionati nella sezione precedente, tuttavia, in questo caso, non sono caratteristiche che determineranno un'identificazione, ma saranno solo uno dei parametri valutati in un algoritmo di ML.

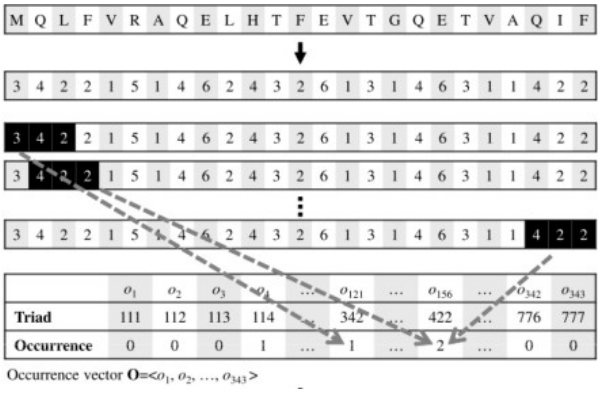

Figura 2.1: Analisi della frequenza delle terne di residui: tramite una "sliding window" si analizza la sequenza consentendo la valutazione di quante volte una sequenza compare, e costruendo quindi il vettore di occorrenza.

2.1.3 Informazioni evoluzionistiche

Sembra che i residui conservati tra proteine omologhe siano più frequenti nelle zone di interazione. Infatti se l'evoluzione tende alla conservazione in una determinata posizione questo potrebbe essere un indice di importanza funzionale del residuo. In biologia il termine omologia ha il significato particolare di indicare che due strutture, ad esempio due organi, hanno un'origine evolutiva comune. Il termine omologia si applica anche a sequenze di acidi nucleici e proteine. Invece, quando non si è certi di un'origine evolutiva comune si dovrebbe parlare di similarità.

Un metodo diffuso per la valutazione della conservazione dei residui utilizza matrici di scores posizione-specifici (PSSM). Solitamente viene effettuato un allineamento con BLAST che confronta la proteina con un database. Gli scores sono valori compresi tra 0 e 1, attribuiti secondo la funzione $x' = \frac{1}{1-e^{-x}}$ dove x è il valore assegnato dalla matrice e x' è il valore della x dopo essere stata scalata. La matrice è costruita assegnando ad ogni posizione di una sequenza proteica un vettore di 21 valori dove i primi venti rappresentano quanto verosimilmente ogni amminoacido sarà presente in tale posizione. L'ultimo valore è un discriminante. La matrice è rappresentata in Figura 2.2.

2.1.4 Informazioni energetiche

Un approccio differente per lo studio della struttura delle proteine si basa su analisi di energia. Solitamente un valore energetico elevato implica un'instabilità. In pratica si descrive l'energia della molecola tramite una funzione costo che tiene conto della posizione degli atomi, della distanza tra loro, della lunghezza e della torsione dei legami. Esempi in Figura 2.3.

Di solito questa viene espressa come una somma di parametri pesati di diversi tipi di interazioni. Questi possono essere determinati teoricamente o sperimentalmente. La funzione energia deve essere indipendente dalla particolare struttura analizzata, altrimenti sarebbe necessario costruire una funzione differente per ogni molecola. I valori energetici

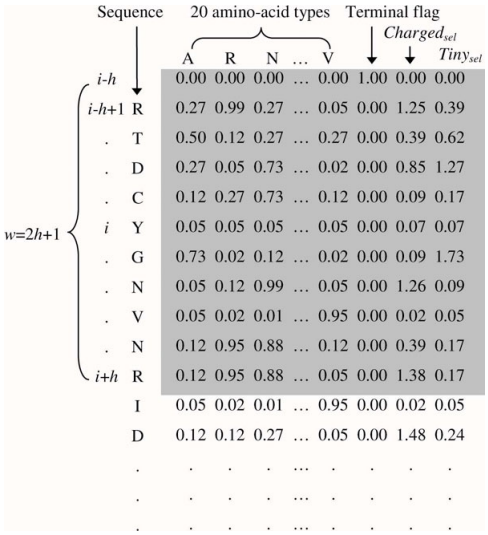

Figura 2.2: Matrice degli scores. Per ogni posizione nella sequenza è indicata la probabilità con cui un dato residuo sarà presente in tale posizione.

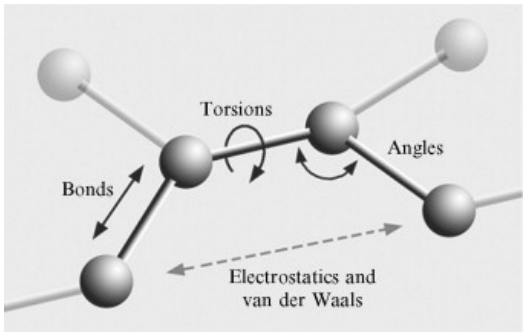

Figura 2.3: Le interazioni principali che vengono valutate: distanze, torsioni e angolazioni dei legami.

possono essere ricavati per una molecola intera: nel caso delle proteine per singoli residui o separatamente per catene principali e laterali. In generale è stato rilevato che i residui di interfaccia hanno un'elevata energia elettrostatica e un'alta energia nelle catene laterali [9].

2.1.5 Osservazione sui parametri

La potenza degli algoritmi ML permette di utilizzare parametri di ogni natura, che abbiano o meno evidenti legami biologici con l'interazione proteica, con un approccio quindi a "black box". Per rendere l'idea si cita il lavoro di Donaldson et al.[10] ove si cita una procedura di identificazione che prevede l'utilizzo di un analizzatore di nomi e della codifica delle proteine. Questi vengono usati per cercare nel titolo e nell'abstract degli articoli in PubMed. Ogni articolo viene codificato in base al numero di termini contenuti. Il peso di ognuno di questi termini di interesse viene rappresentato dalla frequenza del termine, ovvero dal numero di occorrenze di uno di questi nel documento, e dalla frequenza inversa del documento, che è l'inverso del numero dei documenti che contengono il termine. Un termine è considerato una parola o due parole adiacenti che compaiono in almeno 3 documenti.

2.2 Algoritmi Machine Learning

Poiché nessuna delle proprietà menzionate è in grado di determinare un'identificazione non ambigua delle regioni di interfaccia, è necessario aumentare l'accuratezza della predizione combinandole assieme tramite un algoritmo di machine learning. È necessario dunque scegliere l'algoritmo più opportuno. La maggior parte delle applicazioni ML fornisce un'interfaccia "user-friendly", dove è necessario solamente inserire i dati in modo che il ricercatore possa concentrarsi maggiormente su questi che non sull'implementazione dell'algoritmo.

Di seguito saranno elencati i due algoritmi maggiormente utilizzati negli studi recenti. Tali predittori differiscono tra loro per il particolare algoritmo usato, per la scelta dei parametri e per il metodo usato per selezionare sottoinsiemi di parametri dall'insieme di partenza.

2.2.1 Alberi decisionali

Gli alberi decisionali (Figura 2.4) vengono implementati ricorsivamente in questo modo: per un assegnato insieme di dati e relativi parametri:

1. considero il parametro che meglio divide il mio insieme di dati in n sottoinsiemi;
2. applico il criterio di decisione e trovo gli n sottoinsiemi;
3. ripeto i due passi precedenti per ogni sottoinsieme fino a che non viene rispettata una condizione che può essere:

a. tutti i dati appartengono ad una sola classe, ovvero trovo un parametro che mi individua un solo sottoinsieme non nullo;

b. non ci sono più parametri da utilizzare.

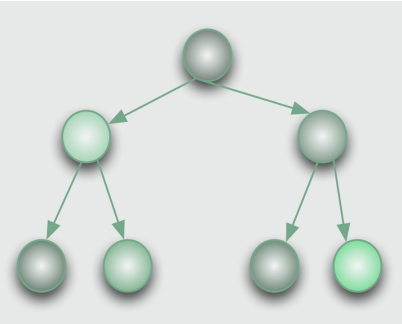

Figura 2.4: Albero di decisione.

Il problema è dunque quelli di valutare quali siano i parametri che determinano i migliori split. L'idea è quella di ricercare una grandezza che dia un'indicazione della "purezza" dei sottoinsiemi ottenuti dalla partizione. Questa grandezza dovrà soddisfare dei requisiti:

- sarà nulla quando un sottoinsieme è puro, ovvero tutti i campioni appartengono alla stessa classe;
- sarà massimale quando tutte le classi possibili sono equivalenti;
- dovrà essere indipendenti dall'ordine delle diramazioni.

L'entropia è una misura molto utilizzata che soddisfa questi requisiti. Si definisce come: $entropia(p_1, p_2, ..., p_n) = -p_1 logpp_1 - p_2 logp_2, ..., p_n logp_n$, dove:

- p_i è la frazione della classe campione i-esima nel sottoinsieme;
- n è il numero totale delle classi.

Esempio. Si consideri un dataset di 14 campioni di cui 9 positivi e 5 negativi. L'entropia del dataset di partenza sarà 0.940. Si effettua uno split in 3 rami che avranno 2-3,4-0,3-2 campioni positivi e negativi rispettivamente. Calcolando le entropie dei 3 nodi sono 0.971, 0 e 0.971 per un totale di 0.693, rivelando che l'operazione di split aumenta la "purezza" del dataset [10].

Concludendo, gli alberi decisionali hanno i seguenti vantaggi:

- sono facili da interpretare poiché sfruttano una semplice logica booleana;
- non richiedono una grande preelaborazione dei dati (es. normalizzazione);
- sono in grado di accettare dati eterogenei e non solamente numerici;

- sono robusti in quanto facilmente validabili;
- sono efficienti su grandi dataset.

I limiti messi in evidenza da questo algoritmo sono invece:

- l'approccio "greedy" in quanto cerca una soluzione ottima da un punto di vista globale attraverso una soluzione ottima ad ogni passo, al contrario delle soluzioni polinomiali. Non è quindi garantito che l'albero ottenuto sia ottimale;
- possono portare all'overfitting se gli alberi creati sono troppo complessi. Bisognerà quindi intervenire con il "pruning";
- la distorsione determinata dagli attributi che hanno un maggior numero di classi.

[19]

2.2.2 Support Vector Machine (SVM)

La SVM è uno strumento molto utilizzato per risolvere problemi biomedicali per via della sua accuratezza e cerca di separare i campioni in classi diverse. Di solito i dati vengono portati dallo spazio originale allo spazio trasformato tramite una trasformazione non lineare (Figura 2.5). La separazione dei parametri può quindi essere eseguita individuando una retta, un piano o un iperpiano come in Figura 2.5, a seconda della dimensione dello spazio dei parametri, scegliendo poi quello con il margine massimo.

Questo medoto ha due vantaggi:

- può generare un modello non lineare;
- previene l'overfitting finchè il criterio di decisione è lineare nello spazio trasformato.

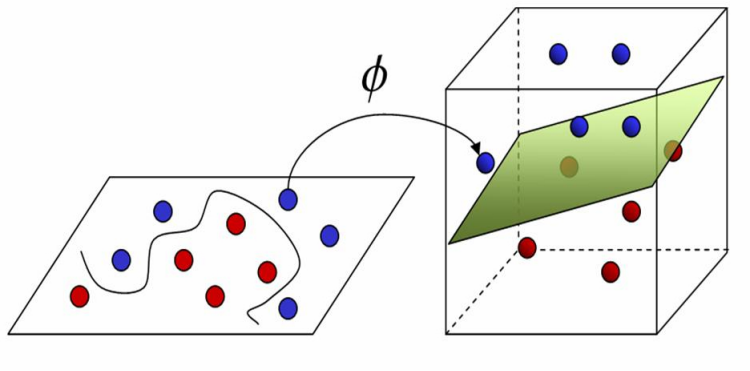

Figura 2.5: Trasformazione dallo spazio originale allo spazio trasformato e individuazione iperpiano.

L'overfitting è infatti uno dei problemi principali per la predizione nei metodi ML, ed è tanto più rilevante quanto più complesso è il modello.

Negli approcci recenti, molto spesso, la complessità dell'algoritmo è notevole e l'overfitting viene trattato a posteriori. L'SVM, invece, è un buon compromesso in quanto può generare modelli molto complessi, a seconda della trasformazione adottata, ma utilizza per la classificazione un iperpiano che è un criterio di decisione molto semplice. L'SVM è chiamata così in quanto usa vettori di supporto per modellare sia la trasformazione che l'iperpiano. Trasformare i dati originali dallo spazio campionato con una funzione non lineare ad un nuovo spazio significa che il modello lineare nel nuovo spazio diventa non lineare nello spazio originale. Per esempio, in due dimensioni $x = (a, b)$, una trasformnazione non lineare in uno spazio a tre dimensioni potrebbe essere $x' = (a^2, ab, b^2)$. Se uno strumento ML trova una suddivisione nel nuovo spazio, potrebbe non essere una linea retta nello spazio originale.

Si fa qui notare che, in principio, ogni strumento, come un albero decisionale, potrebbe essere usato per fare la decisione nello spazio trasformato. La ricerca in questo ambito, si sta dedicando alla definizione di iperpiani sempre più efficienti. Il limite principale di questo algoritmo è la complessità computazionale. Eseguire un'interazione completa del genoma umano tramite la SVM richiederebbe anni. È necessario quindi trovare dei metodi ML di complessità inferiore. Per ottenere un compromesso tra complessità ed accuratezza si può utilizzare, ad esempio, la tecnica detta Relaxed Variable Kernel Density Estimation (RVKDE). Il kernel del RVKDE costituisce un metodo non parametrico per stimare la densità di probabilità di una funzione di una variabile casuale. Inoltre è un ordine di grandezza più veloce del SVM ed ha una capacità descrittiva comparabile [10].

2.2.3 Osservazioni sui metodi ML

Ciò che questi metodi di ML visti in questo capitolo hanno in comune sono:

- l'utilizzo di molte e varie sorgenti di dati;
- le prestazioni sono influenzate dalla qualità dei dati sperimentali;
- sono quasi sempre finalizzati alla determinazione dell'interazione e non si dedicano allo studio della non interazione. Infatti un dataset standard richiede anche campioni negativi, ovvero proteine non interagenti. Tuttavia non c'è molta letteratura riguardante metodi per la determinazione della non interazione tra proteine, osservazione, questa, ricorrente nelle recenti pubblicazioni [10].

Parte II

Predizione di interfacce tramite analisi di terne di residui

Nel lavoro di tesi è stato implementato un modello per la predizione delle interfacce proteiche utilizzando un algoritmo di machine learning istruito con le caratteristiche di gruppi di tre residui contigui.

Il motivo di questa scelta è stato quello di:

1. effettuare una caratterizzazione a risoluzione inferiore di quella che si ottiene analizzando il singolo residuo;

2. tenere conto che le proprietà di un residuo sono influenzate dalle caratteristiche dei residui adiacenti, in quanto i gradi di libertà della catena principale sono limitati.

Le caratteristiche di queste terne sono state determinate tramite l'analisi e l'elaborazione dei parametri dei residui che le compongono, non essendo presenti software che estraggono caratteristiche di gruppi di più residui. L'identificazione finale evidenzierà quali di queste terne saranno o meno presenti nelle interfacce. Questa identificazione potrà essere poi sfruttata per elaborazioni successive.

Alcuni approcci recenti, ad esempio, considerano come parametri di un residuo anche i parametri dei residui vicini, pesati secondo qualche criterio. La vicinanza tra residui può essere determinata tramite:

1. l'applicazione di una distanza di cut-off ovvero due residui devono stare in un certo intorno misurato da tutti gli atomi o dal carbonio α o altro, a seconda delle scelte (Figura 2.6);

2. la costruzione di un diagramma di Voronoi (Figura 2.6).

Figura 2.6: Sx: schematizzazione dell'applicazione di una distanza Euclidea ad ogni residuo. Dx: di diagramma di Voronoi in 2 dimensioni.

Entrambe le tecniche richiedono un'analisi di ogni punto, ovvero ogni atomo, della proteina. La distanza di cut-off è inoltre frutto di una scelta a priori, scelta che ha un effetto sui risultati. Il diagramma di Voronoi, invece, non richiede distanze di cut-off, ma potrebbe identificare come adiacenti residui "lontani" [4]. Tramite l'approccio scelto non

si esclude comunque l'utilizzo di queste tecniche importanti, ma si effettua un'analisi che tiene conto della vicinanza dei residui in una maniera non costosa computazionalmente, e che comunque permette che la vicinanza nello spazio sia valutata in seguito su di un minor numero di punti.

In sintesi, come schematizzato in Figura 2.7, l'elaborazione si sviluppa in questo modo: da un dataset di complessi proteici [6]

1. si considerano le catene delle proteine singolarmente ovvero senza considerare il complesso di cui fanno parte;
2. si estraggono i parametri dalla struttura primaria e terziaria dei singoli residui;
3. si considerano terne contigue (non tramite finestra mobile) di residui caratterizzandole con i parametri elaborati dei residui che le compongono;
4. si utilizza un algoritmo a Random Forest il cui training viene effettuato utilizzando i parametri delle terne di cui si conosce l'appartenenza o meno ad un interfaccia;
5. identificate le terne di interfaccia, si effettuano delle prove su proteine test.

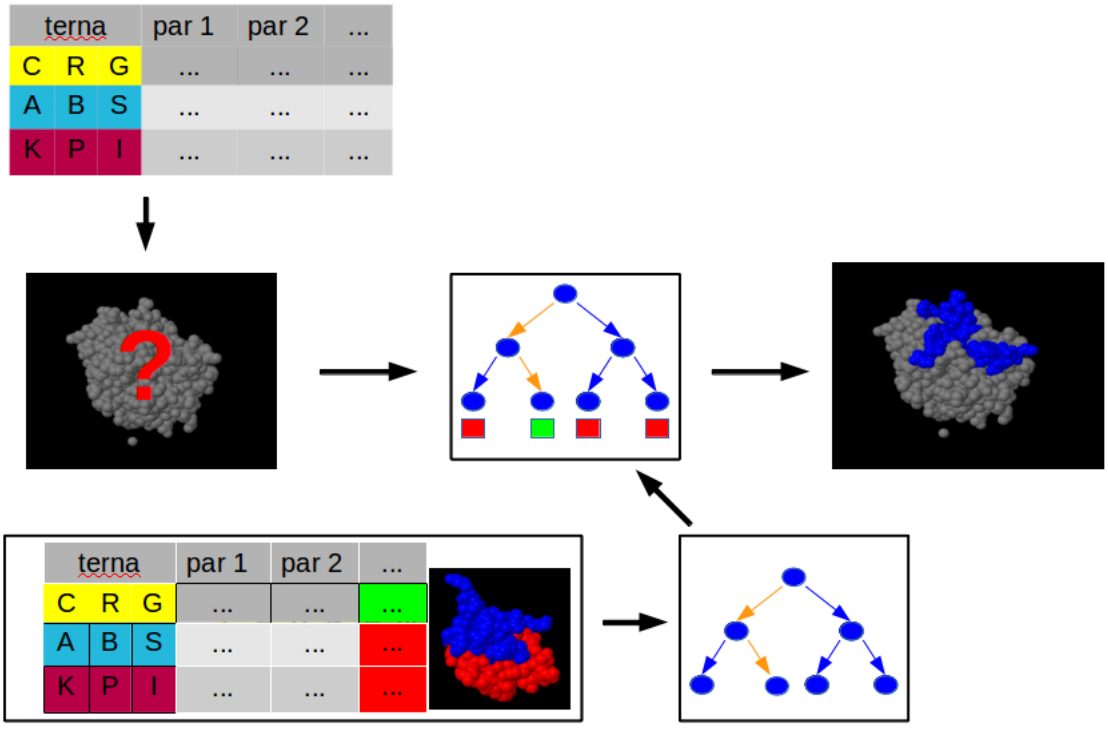

Figura 2.7: Riassunto dell'algoritmo utilizzato. L'algoritmo ML viene addestrato con terne note e poi utilizzato per predire nuove terne di interfaccia.

Capitolo 3

Estrazione dei parametri

Sono stati analizzati dei complessi proteici appartenenti al dataset B100 [6] che rappresenta uno dei benchmark più recenti per la validazione di algoritmi di docking. Da questi sono stati ricavati i parametri di interesse, classificabili tra quelli descritti nella prima parte. I valori sono stati determinati da ogni singolo residuo, e poi rielaborati per rappresentare le terne di appartenenza in quanto non vi sono software progettati per l'analisi di terne.

Di seguito vengono descritti in dettaglio i parametri utilizzati.

3.1 Parametri strutturali

Le caratteristiche della struttura sono molto importanti per la caratterizzazione dei residui di interfaccia. È facilmente intuibile che residui idrofilici tendano a disporsi sulla superficie [7], come è probabile che residui con un'estesa superficie esterna abbiano più probabilità di trovarsi su di un'interfaccia rispetto a residui situati all'interno della proteina. Questi parametri sono stati ricavati da PSAIA [5] impostando come raggio del solvente 1.4$\mathring{A}$. Questa scelta verrà giustificata in seguito.

Un esempio dei valori estratti si trova in Figura 3.1.

```
chain residue_id residue total_asa backbone_asa sidechain_asa
polar_asa nonpolar_asa total_rasa backbone_rasa sidechain_rasa
polar_rasa nonpolar_rasa avg_PDX maximum_DPX minimum_DPX
maximum_CX minimum_CX hydrophobicity
A 4 LYS 173.612 58.6288 114.983 81.1964 92.4157 88.5144
164.919 71.6005 111.856 74.8002 0 0 0 3.00764 2.02025 -3.9
A 5 PRO 51.881 12.398 39.483 0.5702 51.3108 38.7808 37.6839
39.1386 2.72042 45.4802 0.509467 2.23504 0 1.74207 1.12651 1.6
A 6 ILE 114.078 6.2883 107.79 6.2883 107.79 65.789 20.2326
75.7323 24.1486 73.1473 0.919658 3.39586 0 1.97711 0.653949
4.5
A 7 TRP 84.3017 0 84.3017 10.4966 73.8051 35.1082 0 40.6058
19.4742 39.6333 1.07555 3.02876 0 1.81618 0.211614 -0.9
```

Figura 3.1: Estrazione dei parametri strutturali di residui dal complesso 1A2K[17] dall'output di PSAIA.

3.1.1 Idrofobicità

Ogni residuo ha il suo valore di idrofobicità, indipendente dalla struttura di appartenenza. I valori utilizzati come standard sono riportati in Figura 3.1. La scala è basata su un insieme di osservazioni sperimentali ottenute dalla letteratura. Una sliding window di lunghezza determinata calcola l'idrofobicità media in un segmento, quindi il valore finale per ogni amminoacido sarà la media dei valori assunti nella traslazione della finestra. Si è visto che le zone più esterne delle proteine hanno una composizione rilevante di residui idrofobici.

3.1.2 Superficie Accessibile al Solvente (ASA) e derivati

La superficie accessibile al solvente (ASA) è solitamente espressa in $\mathring{A}^2$ ed è uno delle caratteristiche più utilizzate ed informative. Viene calcolata usando l'algoritmo della "sfera rotolante" rappresentato in Figura 3.2 ovvero è il luogo dei punti individuati dal centro di una sfera di raggio $1.4\mathring{A}$ (approssimazione del raggio della molecola dell'acqua) attorno alla superficie di Van der Waals degli atomi della struttura. L'ASA relativa (RASA) è il rapporto tra l'ASA del residuo e l'ASA standard del residuo. Quest'ultima viene determinata calcolando il valore medio che ogni residuo assume nel momento in cui si trova al centro di una tripletta, es. ALA-X-ALA; le triplette considerate sono 1000, selezionate in maniera casuale, ma sempre con il residuo di interesse al centro. Sia che per l'ASA che per la RASA vengono calcolati valori:

1. total: somma del valore di tutti gli atomi;

2. backbone: somma del valore degli atomi della catena principale;

3. side chain: somma del valore degli atomi della catena laterale;

4. polar: somma del valore degli atomi N e O;

5. non polar: somma del valore degli atomi di carbonio.

Amino	Hydro
ALA	1.8
ARG	-4.5
ASN	-3.5
ASP	-3.5
CYS	2.5
GLN	-3.5
GLU	-3.5
GLY	-0.4
HIS	-3.2
ILE	4.5
LEU	3.8
LYS	-3.9
MET	1.9
PHE	2.8
PRO	1.6
SER	-0.8
THR	-0.7
TRP	-0.9
TYR	-1.3
VAL	4.2

Elemento	raggio ($\mathring{A}$)
Idrogeno	1.20
Carbonio	1.70
Azoto	1.55
Ossigeno	1.52

Tabella 3.1: Sx: Scala standard di idrofobicità degli amminoacidi [1]. Dx: Una delle stime dei raggi di Van der Waals di alcuni elementi.

3.1.3 Indici di protrusione DPX e CX

Il DPX (depth index) di un atomo è definito come la sua distanza in $\mathring{A}$ dal più vicino atomo accessibile al solvente, ovvero avente ASA>0. Sarà nullo per gli atomi accessibili e positivo e crescente per gli atomi più interni. Il CX, invece, calcola il numero di atomi pesanti che stanno entro una certa distanza ($10\mathring{A}$) da ogni atomo che non sia di idrogeno. Questo numero è moltiplicato per il volume atomico medio trovato nelle proteine, da cui si ottiene il volume occupato dalla proteina entro la sfera (V_{int}). Il volume V_{ext} rimanente è la differenza tra il volume della sfera e V_{int}. Da cui si ha CX$= \frac{V_{ext}}{V_{int}}$. Sia per il DPX che per il CX se ne calcola il valore:

1. mean total: somma del valore di tutti gli atomi;
2. maximum: il più alto dei valori calcolati per tutti gli atomi;
3. minimum: il più basso dei valori calcolati per tutti gli atomi.

3.2 Parametri energetici

Sono stati inseriti anche parametri relativi al contributo energetico di ogni residuo nella struttura. Questi sono stati ricavati da FoldX [9], un algoritmo che utilizza un campo

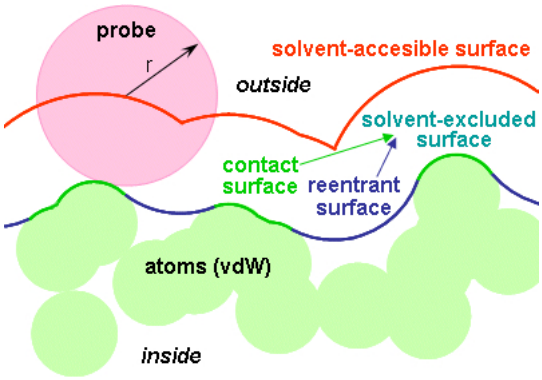

Figura 3.2: Schematizzazione dell'algoritmo di determinazione della superficie accessibile al solvente (ASA)

di forze empirico per il calcolo dell'effetto di mutazioni puntuali e energia di interazione nelle proteine. Il calcolo è basato sull'energia libera di folding di Gibbs, e decomposta in diversi termini energetici. I parametri utilizzati sono in Figura 3.3.

```
chain residue_id residue BackHbond SideHbond Electro
entrop_sc entrop_mc energy_VdW energy_SolvH energy_SolvP
sidechain_Occ mainchain_Occ
A 4 LYS 0 0 -0.141271 0.348292 0.181585 -0.418445 -0.542769
1.11737 864.22 930.907
A 5 PRO -0.485331 0 0 0.25628 1.11568 -0.830805 -1.22693 1.36
783.621 1342.07
A 6 ILE -0.666958 0 0 0.230365 1.43488 -0.869981 -1.03652
1.30228 794.806 1475.42
A 7 TRP -0.6975 0 0 0.82863 1.76168 -1.95274 -2.47514 2.14423
2941.16 1634.04
A 8 GLU -0.517739 0 -0.144986 0.655252 1.35128 -1.1801
-1.44359 2.0508 1128.04 1493.74
```

Figura 3.3: Estrazione dei parametri di residui energetici dal complesso 1A2K[17] dall'output di FoldX.

3.2.1 Legame a Idrogeno

Backbone e Sidechain Hydrogen Bond rappresentano il contributo della formazione dei legami a idrogeno tra gli atomi della catena principale, il primo, e tra gli atomi della catena laterale, il secondo.

3.2.2 Interazioni di Van der Waals

Le interazioni di Van der Waals rappresentano una penalizzazione energetica qualora due atomi si venissero a trovare entro una distanza inferiore alla somma dei rispettivi raggi di VdW, i cui valori sono riportati in Figura 3.1. Atomi troppo vicini tenderebbero a respingersi, quindi aumenterebbero l'instabilità. Pur avendo considerato solo la somma

complessiva il software permette anche di rilevare le energie di interazione di dipolo e torsionale.

3.2.3 Entropia

L'entropia è un costo energetico che cresce quando due molecole hanno libertà rotazionale e transazionale ridotte a causa del loro legame. Questo costo è valutato separatamente per la catena principale e laterale.

3.2.4 Interazione con il solvente

Sono parametri sperimentali ricavati da prove in cui gli amminoacidi vengono trasferiti dall'acqua ad un solvente organico. Questo processo simula ciò che avviene in un amminoacido durante il ripiegamento: la proteina si ripiega in un certo modo nel solvente e successivamente si riconfigura in ambiente idrofobico.

3.2.5 Occupazione

L'occupazione di un residuo, o della sua catena principale/laterale, è il volume occupato da questi nella proteina di appartenenza. È misurato tramite una funzione che elabora il volume degli atomi in questione.

3.2.6 Contributo elettrostatico

È calcolato da una semplice implementazione della legge di Coulomb

Capitolo 4

Analisi di terne

4.1 Costruzione delle terne

Per ogni residuo delle proteine contenute nel dataset si ottiene dagli output di PSAIA e FoldX un vettore contenente i parametri descritti. Si precisa che un (esiguo) numero di proteine del dataset sono state scartate poiché i software di elaborazione non erano in grado di processarne la struttura. Inoltre alcuni residui possono subire una mutazione per motivi di efficienza energetica durante l'elaborazione di FoldX, aspetto di cui è necessario tenere conto nel momento in cui si integrano risultati provenienti da software differenti.

Ogni catena proteica è stata suddivisa in gruppi di tre residui consecutivi a partire dal primo residuo della catena. Sono stati scartati al più due residui finali qualora avanzassero (ossia nel caso in cui il numero di residui della catena non fosse divisibile per 3). Si può eventualmente valutare se possa essere significativo fare in modo che i residui da scartare debbano essere quelli situati ai due estremi o i due ad un solo estremo come in questo caso. I parametri dei singoli residui vengono quindi elaborati per rappresentare la terna di cui fanno parte (Figura 4.1). Differenti elaborazioni sono state effettuate ma i risultati che hanno determinato una predizione migliore sono stati ottenuti effettuando una media algebrica dei valori dei tre residui. Questo non significa che questa sia la scelta migliore in generale ma andrà valutata assieme ad altre scelte fatte nell'implementazione.

4.2 Determinazione delle terne di interfaccia per il training

Dovendo istruire un algoritmo di ML è necessario utilizzare parametri di terne di cui si conosca l'appartenenza o meno ad un'interfaccia. Le terne di interfaccia dei complessi del dataset utilizzato [6] non sono note ma possono essere determinate, dato che se ne conosce la struttura quaternaria. La determinazione di queste non può essere effettuata direttamente, sempre per il motivo per cui non si hanno a disposizione software che effettuino l'analisi a terne.

```
1 chain chain chain residue_id residue_id residue_id residue
  residue residue BackHbond SideHbond Electro entrop_sc
  entrop_mc energy_VdW energy_SolvH energy_SolvP sidechain_Occ
  mainchain_Occ chain chain chain residue_id residue_id
  residue_id residue residue residue total_asa backbone_asa
  sidechain_asa polar_asa nonpolar_asa total_rasa backbone_rasa
  sidechain_rasa polar_rasa nonpolar_rasa avg_PDX maximum_DPX
  minimum_DPX maximum_CX minimum_CX hydrophobicity chain chain
  chain residue_id residue_id residue_id residue residue
  residue isInterface
2 A A A 4 5 6 LYS PRO ILE -1.1522890000000001 0.0
  -0.04709033333333334 0.834937 2.732145 -2.119231
  -2.8062190000000005 3.77965 2442.647 3748.397 A A A 4 5 6 LYS
  PRO ILE 339.571 77.3151 262.25600000000003 88.0549 251.5165
  193.0842 222.8355 186.4714 138.72502 193.42770000000002
  0.476375 3.39586 0.0 3.00764 0.653949 0.7333333333333334 A A
  A 4 5 6 LYS PRO ILE true
3 A A A 7 8 9 TRP GLU GLN -2.529849 0.0 -0.048328666666666666
  2.1077339999999998 4.532640000000001 -4.19292 -5.11067
  5.68138 5213.79 4450.12 A A A 7 8 9 TRP GLU GLN
  228.94259999999997 9.7821 219.1605 111.8685 117.0741
  117.75189999999998 27.724210000000003 137.3931
  113.21419999999999 105.1849 0.6126563333333334 3.02876 0.0
  2.25621 0.211614 -2.6333333333333333 A A A 7 8 9 TRP GLU GLN
  true
```

Figura 4.1: Parametri per terne di residui.

Si è scelto quindi di procedere nel modo seguente:

1. si analizza ogni catena con un software che determini i residui di interazione. Per questo è stato usato PSAIA, individuando i residui che avessero almeno una coppia di atomi (esclusi gli atomi di idrogeno) a distanza inferiore a $6\mathring{A}$ dai residui dell'altra proteina (Figura 4.2): il valore scelto sta nel range [4,6]$\mathring{A}$ che solitamente viene utilizzato per determinare l'interazione tra residui [20] [5].

2. si considera una terna di interfaccia una terna di cui almeno uno dei suoi 3 residui faccia parte di un'interfaccia: questo è stato fatto per non perdere informazione sui pochi, in proporzione, residui di interfaccia. È possibile vedere l'identificazione della terna in Figura 4.2, in corrispondenza del parametro isInterface.

L'algoritmo ML è stato istruito con 8000 terne appartenenti a 60 complessi proteici.

chain	residue_id	residue	isInterface
A	4	LYS	0
A	5	PRO	1
A	6	ILE	0
A	7	TRP	1
A	8	GLU	1
A	9	GLN	0
A	10	ILE	0
A	11	GLY	0
A	12	SER	0
A	13	SER	0
A	14	PHE	0
A	15	ILE	0
A	16	GLN	0
A	17	HIS	0
A	18	TYR	0
A	19	TYR	0
A	20	GLN	0
A	21	LEU	0
A	22	PHE	1
A	23	ASP	0
A	24	ASN	0
A	25	ASP	0

Figura 4.2: Sx: concetto di distanza tra residui di due proteine formanti un complesso. Dx: individuazione dei residui di interfaccia del complesso 1A2K[17] tramite PSAIA: 1=interfaccia, 0=non interfaccia

Capitolo 5

Predizione di terne di interfaccia tramite Random Forest

La scelta dell'algoritmo di ML da utilizzare è ricaduta sul Random Forest in quanto:

- affronta bene problemi con elevato numero di parametri;
- conferisce alti livelli di accuratezza;
- è resistente all'overfitting in quanto si ha introduzione di casualità a diversi livelli;
- la fase di training è veloce anche con migliaia di predittori;
- non è necessario selezionare le variabili da utilizzare;
- permette di determinare i parametri più informativi tra quelli utilizzati tramite una misura di "importanza";
- i risultati sono invarianti a trasformazioni monotone delle variabili.

Inoltre è un algoritmo di facile comprensione e implementazione [12] [11].

5.1 Teoria del Random Forest

La foresta può essere creata secondo differenti modalità; nella pubblicazione originale[12] viene costruita come segue. Dato un insieme di n dati con relativi m parametri:

1. considero un numero $m << M$ di parametri
2. considero un training set campionando con reinserimento (metodo detto di "bootstrapping") n volte dall'insieme di n dati originale, vedi Figura 5.1;
3. i dati rimanenti vengono usati per determinare l'errore dell'albero;
4. per ogni nodo dell'albero vengono scelti casualmente (eventualmente con reinserimento) $m << M$ parametri, di solito $m = \sqrt{M}$;

5. di questi m viene scelto il nodo che determina lo split migliore nell'insieme degli m;

6. ogni albero viene fatto crescere fino alla sua massima estensione, senza utilizzare tecniche di pruning;

7. Ogni albero si sviluppa su circa il 63% dei dati di training originali: il 37% rimanente (Out Of Bag (OOB)) viene usato per testare l'albero stesso; le statistiche sul RF sono tutte basate su questo OOB.

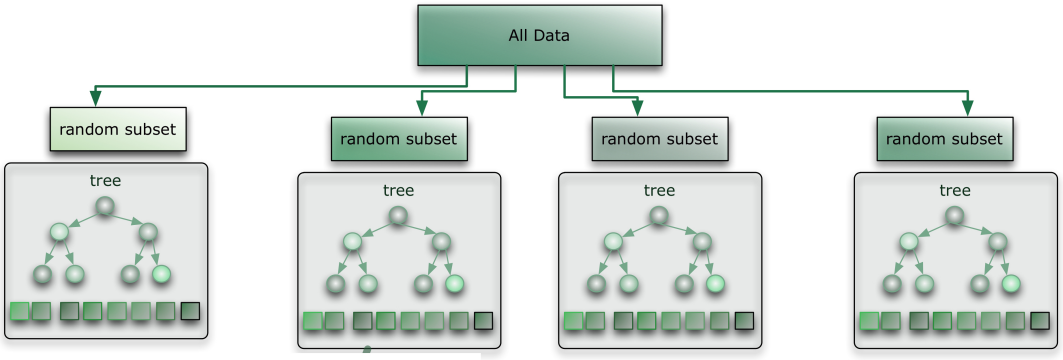

Figura 5.1: Schema Random Forest: gli alberi vengono istruiti con un sottoinsieme casuale dei dati.

Un caratteristica particolarmente attraente delle Random Forests è che esse generano una stima di quali variabili siano importanti per la classificazione, offrendo la possibilità di selezionare solo un sottoinsieme che risulti ottimale dal punto di vista statistico. La procedura che permette di fare questa valutazione è la seguente. Per ogni albero si classificano i casi OOB per quel particolare albero. Per il j-esimo albero si abbiano C_j casi classificati correttamente. Si permutano quindi casualmente i valori della i-esima variabile per i casi OOB e si riclassificano i soggetti. Siano CP_{ij} i casi classificati correttamente dopo la permutazione. Si calcola il valore di importanza della i-esima variabile I_{ij}: $I_{ij} = C_j - CP_{ij}$. Si ripete il procedimento su tutti gli alberi della foresta e si fa la media dei valori ottenuti: $I_i = \text{mean}(I_{ij})$; l'indice I_i stima l'importanza della i-esima variabile. L'errore standard di tale stima viene valutato come se tali valori fossero indipendenti. Dividendo la stima I_{ij} per il suo errore standard si ha la stima normalizzata dell'importanza della i-esima variabile. Un altro indice che permette di valutare l'importanza di una variabile è basato sulla somma su tutti gli alberi della foresta dell'indice di decrescita dell'impurità di Gini dovuto all'uso di ciascuna variabile. Questa stima di importanza è solitamente consistente con la precedente [21].

5.2 Predizione di terne di inerfaccia tramite RF

Il training della foresta è stato eseguito su 8000 terne. La foresta è stata costituita da 200 alberi[4] e i parametri tra cui scegliere ad ogni split è stato 5 su 27 disponibili. Nella

pubblicazione originale infatti viene consigliato un numero di parametri per ogni split pari a $\sqrt{n}$ con n numero dei parametri.

La decisione finale viene determinata guardando la moda dei risultati (Figura 5.2).

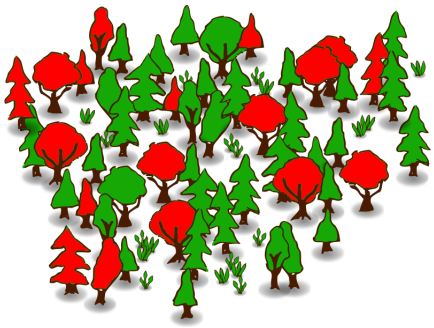

Figura 5.2: Una terna viene classificata da ogni albero della foresta e la decisione finale sarà determinata dalla maggioranza dei risultati.

Un valore utilizzato per stimare la bontà della predizione è Out Of Bags (OOB) estimate error rate. Questo viene calcolato, utilizzando il 37% dei dati esclusi dalla costruzione dell'n-esimo albero, nel modo seguente:

1. prendo uno dei dati esclusi nella costruzione di un albero;
2. classifico questo dato utilizzando tale albero;
3. calcolo quante il dato viene classificato erroneamente ogni volta che si trova OOB;
4. faccio la media di questo valore per ogni dato.

Nel nostro caso il valore è stato del 18%.

I parametri risultati più informativi, ovvero quelli che hanno riportato il valore di "importance" più elevato sono risultati essere il "minimum DPX" e il "main chain occupation". A seguire i parametri relativi alla superficie accessibile.

Di seguito è indicato il codice R[13] utilizzato per il Random Forest.

La prima parte segue per inizializzare il pacchetto "randomForest" e per caricare il file in cui si deve riportare un'intestazione contenente i nomi dei parametri separati da uno spazio.

```
library(randomForest)
nomeFile="file.txt"
proteine<-read.table(nomeFile, sep=" ", header=TRUE)
```

Quindi si scelgono le terne su cui eseguire il training. Inoltre sono da indicare le colonne corrispondenti ai parametri utilizzati.

```
train = proteine[ c(0:8000),(0:27) ]
```

Tra questi poi va indicato il parametro che determina la divisione nelle due classi, in questo caso "isInterface". Inoltre si indica che si vuole avere la misura di "importance" dei parametri, si imposta il numero degli alberi e il numero di parametri da valutare ad ogni split. Il comando successivi servono solo per visualizzare i risultati e i valori di importanza.

```
RF = randomForest(isInterface ~ ., data=train, importance=TRUE,ntree=200,ntry=5)
print(RF)
importance(RF)
```

Volendo predire delle terne incognite è sufficiente caricarle e effettuare la classificazione tramite il modello appena ottenuto. Il comando table serve a visualizzare la matrice di confusione.

```
terneIncognite=proteine[ c(8001:9000),(0:27) ]
protein.predict=predict(RF,terneIncognite)
print(protein.predict)
t = table(observed=test[,'isInterface'], predict=proteine.predict)
```

Veri Negativi: 4185	Falsi Positivi: 423
Falsi Negativi: 1045	Veri Positivi: 2347

Tabella 5.1: Matrice di confusione.

La matrice di confusione ottenuta con il campione di 8000 terne è illustrata nella Tabella 5.1. Inoltre l'errore di classificazione è di circa il 10% per le terne non di interfaccia e del 30% per le terne di interfaccia. In Figura 5.3 si vede una proteina per la quale vengono predette le terne di interfaccia, rappresentate in funzione dei propri atomi.

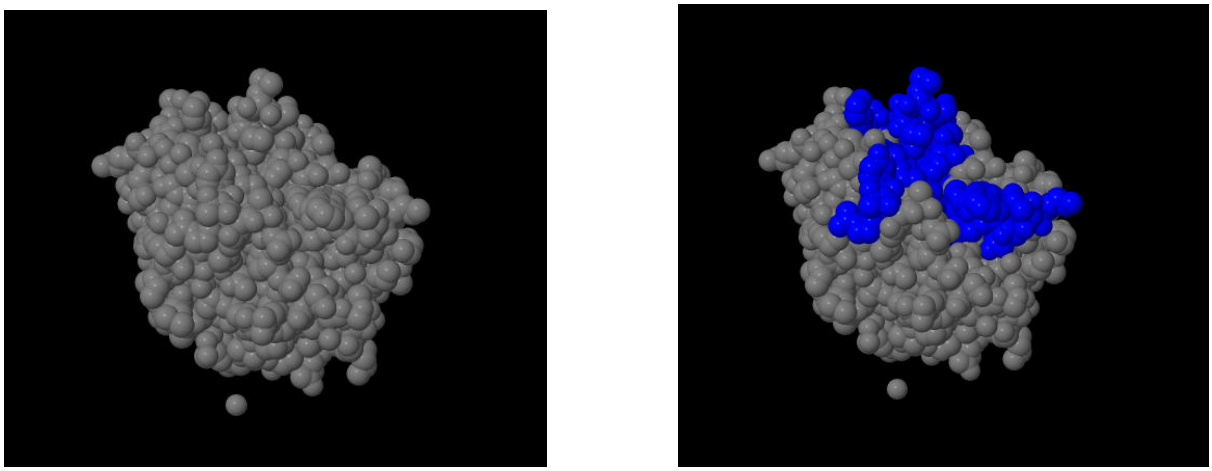

Figura 5.3: Atomi appartenenti alle terne di interfaccia (in blu) predette per la proteina 1CGI[17]

Parte III

Conclusioni e sviluppi futuri

I risultati ottenuti sono solo in parte soddisfacenti. Le terne individuate come di interfaccia seguendo il metodo descritto sono sottostimate. L'errore nella classificazione è molto alto soprattutto per quanto riguarda l'individuazione delle effettive terne di interfaccia. Il motivo è da ricercarsi in diversi aspetti dell'algoritmo:

- molti parametri non sono sufficientemente informativi, ovvero non caratterizzano a sufficienza i residui di interfaccia, e ciò si ripercuote sulla caratterizzazione delle terne;
- riguardo la determinazione dell'interfaccia di complessi noti per il training si sottolinea che:
    a. è necessario decidere una soglia, la cui scelta influenzerà la predizione. Gong et al.[20] utilizzano una soglia di $5\mathring{A}$. Vi sono anche altri criteri per scegliere residui di interfaccia, ad esempio, Aloy and Russell[10] definiscono interagenti i residui che abbiano legami a idrogeno N-O$\leq 3.5\mathring{A}$, ponti salini N-O$\leq 5.5\mathring{A}$ o interazioni di Van der Waals C-C$\leq 5\mathring{A}$. Altri misurano la variazione della ASA prima e dopo la formazione del legame[5]. Altri ancora visualizzano adiacenze tra residui di proteine differenti tramite il diagramma di Voronoi.
    b. è da verificare quale sia il metodo più conveniente per passare dal singolo residuo alla terna; esiste il rischio infatti che considerando di interfaccia solo terne di cui tutti i residui risultano di interfaccia si possano rilevare un numero di terne di interfaccia molto limitato.
- il dataset influisce molto la bontà della predizione: infatti il random forest tende a mantenere nella predizione il rapporto tra il numero di dati appartenenti alle due classi: imporre un numero più o meno uguale di "true" e "false" porta ad una sovrastima del numero di terne di interfaccia, e un peggioramento dell'errore associato però ad una crescita dell'indice di importanza dei parametri;
- la determinazione dei parametri energetici si basa su un modello empirico.

Possibili miglioramenti potrebbero essere:

- utilizzare di criteri più precisi per la determinazione delle interfacce, cosa che dovrebbe creare meno ambiguità;
- è necessario studiare a fondo come debba essere impostato il training set per produrre i risultati più verosimili;
- determinare parametri più informativi dei residui;
- considerare la possibilità di individuare parametri specifici per le terne; esempi potrebbero essere:

a. un coefficiente che tenga conto del grado ripiegamento/torsione della catena principale della terna;

b. differenza massima/minima, tra parametri dei residui componenti, quali DPX o ASA.

Oltre a quelli elencati vi sono anche problemi con i quali dovrà avere a che fare chiunque si occupi di problemi di individuazione di interfacce. Molti software non sono opensource e ad esempio non è disponibile un'implementazione dell'algoritmo di Connoly per la determinazione della superficie accessibile. Per questo è necessario prendere per buoni i risultati che vengono dati.

Se vi è la necessità di analizzare dataset proteici si ha una disponibilità limitata o nulla di software che soddisfano requisiti essenziali per un'analisi efficiente ovvero:

- siano stand alone, in quanto molti sono utilizzabili solo da server;
- permettano l'analisi di più strutture alla volta, con tempi di elaborazione ragionevoli, cosa che, per altro, si ha solitamente per i software stand alone;
- consentano un parsing agevole dei file di output;
- siano ottimizzati per l'analisi di file del Protein Data Bank (PDB) che sta diventando il database di riferimento;
- permettano di analizzare più di una caratteristica per volta.

Sarebbe di fondamentale importanza riuscire ad integrare tra di loro software di analisi di parametri, sfruttando anche le potenzialità delle librerie BioJava, BioPython e BioPerl, che consentono una robusta analisi della struttura a partire da file del PDB, ma sono carenti dal punto di vista dell'estrazione dei parametri dalle sue componenti.

Recentemente è stato introdotto Taverna[22], un sistema che permette di gestire facilmente una serie di servizi, "web servers" compresi, per l'analisi di dati bioinformatici. Questi non devono essere installati e possono essere, ad esempio, utilizzati in cascata. Gli schemi di lavoro possono essere salvati e riutilizzati risparmiando tempo considerevole al ricercatore. Sarebbe quindi molto importante integrare in questo sistema il maggior numero di strumenti.

Bibliografia

[1] Jack Kyte, Russell F. Doolittle: *A simple method for displaying the hydropathic character of a protein.* Department of Chemistry University of California 1984.

[2] Martijn Huynen, Berend Snel, Warren Lathe and Peer Bork *Exploitation of gene context* 2000 Elsevier Science

[3] Chi-Yuan Yu, Lih-Ching Chou and Darby Tien-Hao Chang: *Predicting protein-protein interactions in unbalanced data using the primary structure of proteins.* BMC Bioinformatics 2010

[4] Segura et al.: *Improving the prediction of protein binding sites by combining heterogeneous data and Voronoi diagrams.* BMC Bioinformatics 2011 12:352.

[5] Josip Mihel, Mile Šikić, Sanja Tomić, Branko Jeren and Kristian Vlahoviček: *PSAIA – Protein Structure and Interaction Analyzer.* BMC Structural Biology 2008, 8:21.

[6] Howook et al.: *Protein–protein docking benchmark version 3.0.* Proteins 2008; 73:705–709.

[7] Shide Liang, Chi Zhang, Song Liu and Yaoqi Zhou: *Protein binding site prediction using an empirical scoring function.* Nucleic Acids Research, 2006, Vol. 34, No. 13.

[8] Raphael Guerois, Jens Erik Nielsen and Luis Serrano: *Predicting Changes in the Stability of Proteins and Protein Complexes: A Study of More Than 1000 Mutations.* J. Mol. Biol. (2002) 320, 369–387

[9] Joost Schymkowitz, Jesper Borg, Francois Stricher, Robby Nys Frederic Rousseau and Luis Serrano *The FoldX web server: an online force field.* Nucleic Acids Research, 2005, Vol. 33,

[10] Darby Tien-Hao Chang *Computational Approaches to Predict Protein Interaction.* Computational and Experimental Tools, Dr. Weibo Cai (Ed.), ISBN: 978-953-51-0397-4, 2012.

[11] Sikic et al.: *Prediction of Protein–Protein Interaction Sites in Sequences and 3D Structures by Random Forests.* Biol 5(1): e1000278. doi:10.1371/journal.pcbi.1000278

[12] Leo Breiman *RANDOM FORESTS*Statistics Department University of California Berkeley, CA 94720 September 1999.

[13] R Development Core Team (2008). *R: A language and environment for statistical computing. R Foundation for Statistical Computing.* Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.

[14] A. Liaw and M. Wiener *Classification and Regression by randomForest.* R News 2(3), 18-22.(2002)

[15] Andreas Prlic; Andrew Yates; Spencer E. Bliven; Peter W. Rose; Julius Jacobsen; Peter V. Troshin; Mark Chapman; Jianjiong Gao; Chuan Hock Koh; Sylvain Foisy; Richard Holland; Gediminas Rimsa; Michael L. Heuer; H. Brandstatter-Muller; Philip E. Bourne; Scooter Willis *BioJava: an open-source framework for bioinformatics in 2012* Bioinformatics 2012.

[16] Jenny Gu (Editor), Philip E. Bourne *Structural Bioinformatics, 2nd Edition* ISBN: 978-0-470-18105-8 1067 pages April 2009, Wiley-Blackwell

[17] F.C.Bernstein, T.F.Koetzle, G.J.Williams, E.E.Meyer Jr., M.D.Brice, J.R.Rodgers, O.Kennard, T.Shimanouchi, M.Tasumi *The Protein Data Bank: A Computer-based Archival File For Macromolecular Structures.* J. of. Mol. Biol., 112 (1977): 535.

[18] *Jmol: an open-source Java viewer for chemical structures in 3D. http://www.jmol.org/*

[19] Wikipedia *Decision tree* Dec. 2013

[20] Sungsam Gong, Changbum Park, Hansol Choi, Junsu Ko, Insoo Jang, Jungsul Lee, Dan M Bolser, Donghoon Oh, Deok-Soo Kim and Jong Bhak *A protein domain interaction interface database: InterPare* BMC Bioinformatics

[21] Matteo Dell'Omodarme *ESERCITAZIONI DI STATISTICA BIOMEDICA* agosto 2012

[22] Katherine Wolstencroft, Robert Haines, Donal Fellows, Alan Williams, David Withers, Stuart Owen, Stian Soiland-Reyes, Ian Dunlop, Aleksandra Nenadic, Paul Fisher, Jiten Bhagat, Khalid Belhajjame, Finn Bacall, Alex Hardisty, Abraham Nieva de la Hidalga, Maria P. Balcazar Vargas, Shoaib Sufi, and Carole Goble: *The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud*Nucleic Acids Research, First published online May 2, 2013. doi:10.1093/nar/gkt328