Alma Mater Studiorum · Università di Bologna

FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E NATURALI

Corso di Laurea in Scienze di Internet

PREDIZIONE DELL' EFFETTO DELLE MUTAZIONI SULLA VARIAZIONE DELLA STABILITÀ PROTEICA CON METODI DI "MACHINE-LEARNING"

Tesi di Laurea in Algoritmi e Strutture Dati

Relatore:
Chiar.mo Prof.
Margara Luciano

Presentata da:
Berovalis Christos

Correlatori:

Chiar.mo Prof.

Di Lena Pietro

Chiar.mo Prof.
Fariselli Piero

Sessione I
Anno Accademico 2009/2010

. . . per i sbagli che facciamo
e le qualità che abbiamo.
Per tutti quelli che
hanno creduto in me.

Premessa

Questa relazione è una descrizione dettagliata di tutto il lavoro fatto durante la tesi. Tale lavoro, consiste nell'implementare un software in grado a predire la variazione della stabilità di una proteina sottoposta ad una mutazione. Il predittore implementato fa utilizzo di tecniche di Machine-Learning ed, in particolare, di SVM.

Nel dettaglio, il lavoro di questa tesi consiste nell'analisi delle prestazioni di un predittore di stabilità precedentemente implementato [1]. In particolare, il lavoro svolto riguarda l'analisi delle prestazioni del predittore sotto opportune variazioni dei parametri di input e relativamente all'utilizzo di nuova informazione rispetto a quella utilizzata dal predittore basilare.

Lo scopo finale era quello di aggiungere nuove informazioni, come i profili e le PSSM relative alle proteine, e testare che la prestazione del predittore sia positiva.

Indice

Elenco delle figure

Elenco delle tabelle

Capitolo 1

Introduzione

1.1 Introduzione

Un requisito importante per la progettazione delle proteine è l'abilità di poter predire i cambiamenti della stabilità di una proteina, a causa di una mutazione puntiforme (*point mutation*). Sono stati descritti diversi metodi che trattano questa operazione e le loro prestazioni sono state esaminate tenendo conto la correlazione lineare globale fra i dati predetti e quelli sperimentali.[1, 2] Tali metodi sono principalmente basati sullo sviluppo delle funzioni con ene-rgia differenti, adatte per computare la variazione della stabilità dell'energia libera, dovuta alla sostituzione di un residuo alla volta nella sequenza proteica (*mutazione*).[1, 2]

Recentemente, è stata creata una base di dati con dei dati termodinamici sulla variazione della stabilità proteica sulla singola point mutation. Ciò permette l'applicazione di tecniche di machine-learning a predire la stabilità dei cambiamenti dell'energia libera sulla mutazione a partire dalla sequenza della proteina. [1, 2] I metodi basati sul energia computano ordinariamente i cambiamenti della stabilità ($\Delta\Delta G$) relativi alla mutazione di una proteina. La loro accuratezza verso la database sperimentale è valutata considerando la correlazione fra i dati originali contro quelli ottenuti sperimentalmente. La

correlazione globale può arrivare fino al 95% secondo la scelta del database di mutazioni. [1, 2]

In molti casi, per la modellazione della stabilità proteica, sarebbe conveniente conoscere l'affidabilità di un metodo e le valutazioni statistiche associate a tutti i valori previsti del cambiamento della stabilità dell'energia libera su una singola mutazione puntiforme in una data catena proteica. Effettivamente qualsiasi $\Delta\Delta G$ previsto consiste di un valore assoluto numerico (la quantità di cambiamento di stabilità) e di un segno. Un valore positivo del $\Delta\Delta G$ corrisponde ad un aumento della stabilità, mentre un negativo ad una diminuzione.[2]

La previsione corretta del senso di $\Delta\Delta G$ del cambiamento di stabilità è quindi più importante, per il problema attuale, del relativo valore assoluto. Una correlazione lineare tra dati previsti e sperimentali non fornisce un'indicazione diretta della correttezza del segno del $\Delta\Delta G$. Esistono dei parametri termodinamici relativi alla mutagenesi, come il pH e la temperatura, che sono condizioni sperimentali. A tale riguardo, i metodi basati sull'energia, per presumere che le mutazioni vengano effettuate in circostanze fisiologiche, devono tener conto di questi parametri. Questa limitazione può essere sorpassata usando le machine-learning. [2]

Grazie alla disponibilità di grandi database di dati termodinamici sulle proteine mutate, ora è possibile applicare le tecniche di Machine-Learning al problema relativo alla predizione della variazione della stabilità della proteina basata sulle singole mutazioni puntiformi.

Capitolo 2

Problema Biologico

2.1 Proteine

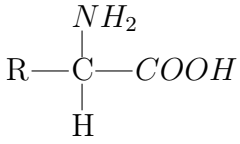
Le proteine sono delle sostanze altamente complesse che sono presenti in tutti gli organismi viventi. Hanno un valore nutritivo molto alto e sono direttamente convolte in processi chimici essenziali per la vita. L'importanza delle proteine è stata riconosciuta dai chimici nel 19° secolo che ha coniato il nome di tali sostanze dal greco "*proteios*", che significa "*colui che arriva primo*". Le proteine sono specie-specifiche, cioè quelle di una specie sono diverse da quelle di un'altra. Esse sono anche organo-specifiche, per esempio, all'interno di un singolo organismo, le proteine muscolari differiscono da quelle del cervello e del fegato. Ogni proprietà che caratteriza un organismo vivente, dipende dalle proteine. Tante sono le loro funzionalità. Alcune, per esempio, controllano il flusso degli elettroni, un compito di massima importanza per la fotosintesi. Altre, che vengono chiamate ormoni, sono in grado di trasportare tutte le informazione importanti tramite le cellule o gli organi dei diversi o-rganismi. Ci sono, inoltre, delle proteine, che controllano il trasporto delle molecole tramite le membrane che dividono le cellule tra di loro, oppure che aiutano il sistema immunitario di un'orgnanismo e queste ultime sono chiamati "*anticorpi*". Un altro gruppo di proteine sono quelle

che fanno parte delle componenti principali del sistema muscoloso, le quali permettono la trasformazione dell'energia chimica in quella meccanica ma sono anche necessarie per la visione e l'udito. In fine, tante sono le proteine che servono per la creazione dell'architettura interna delle cellule. Nonostante le differenze tra le loro funzionalità, tutte le proteine sono dei polimeri di residui amminoacidici che sono uniti tra di loro in maniera lineare. La base delle loro diversitá si trova nel legame peptidico, responsabile dell'unione degli amminoacidi e della creazione delle proteine, ma anche nelle loro strutture tridimensionali, che sono tutte diverse tra di loro. Soltando tramite queste strutture siamo in grado di capire le funzioni delle proteine.

2.2 Amminoacidi

Come abbiamo detto il ruolo biologico delle proteine viene definito dalla loro struttura tridimensionale, chiamata anche struttura terziaria, la quale sarebbe in ogni caso la conseguenza della sequenza amminoacidica, chiamata anche struttura primaria. Gli aminoacidi proteinogenici sono venti, detti anche essenziali (o *standard*). Questi sono gli elementi costitutivi (*monomeri*) delle proteine.

Come aminoacido si definisce *qualsiasi molecola organica che consiste di almeno uno gruppo funzionale dell'ammina (-NH2), almeno uno di acido carbossilico (-COOH), e un gruppo organico R (detto* lato catena*) il quale è unico per ogni amminoacido.* Il termine amminoacido è l'abbreviazione di α-amino [*α-amino acido*] carbossilico. Ogni molecola contiene un'atomo di carbonio centrale (C) , denominato α-carbonio, al quale sia un aminoacido e un gruppo carbossilico sono legati. Gli altri due vincoli dell' α-atomo di carbonio sono generalmente associati ad un atomo di idrogeno (H) e al gruppo R. Il residuo laterale (gruppo R), è di natura eterogenea. Questo significa che se R = H, si ha la glicina e di conseguenza l'amminoacido più semplice che esista. La formula generale di un amminoacido è la seguente:



Ogni amminoacido differisce da tutti gli altri nella particolare struttura chimica del loro gruppo R (Figura 2.1).

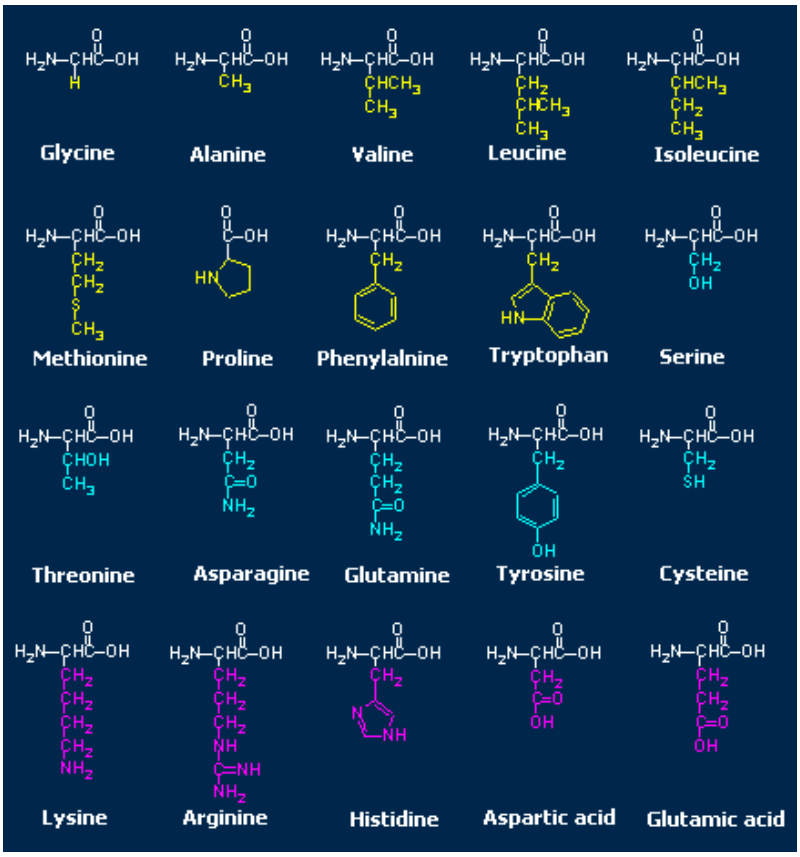


Figura 2.1: *Amminoacidi con i Loro Gruppi R*

Nella figura sotto indicata (Figura 2.2), ci sono tutti e venti gli amminoacidi con le rispettive strutture, i nome e le abbreviazioni con le quali sono conosciuti. Le piccole sfere rappresentano gli atomi di idrogeno, mentre quelle più grandi gli atomi di carbonio. I restanti atomi sono rappresentati dal loro simbolo.

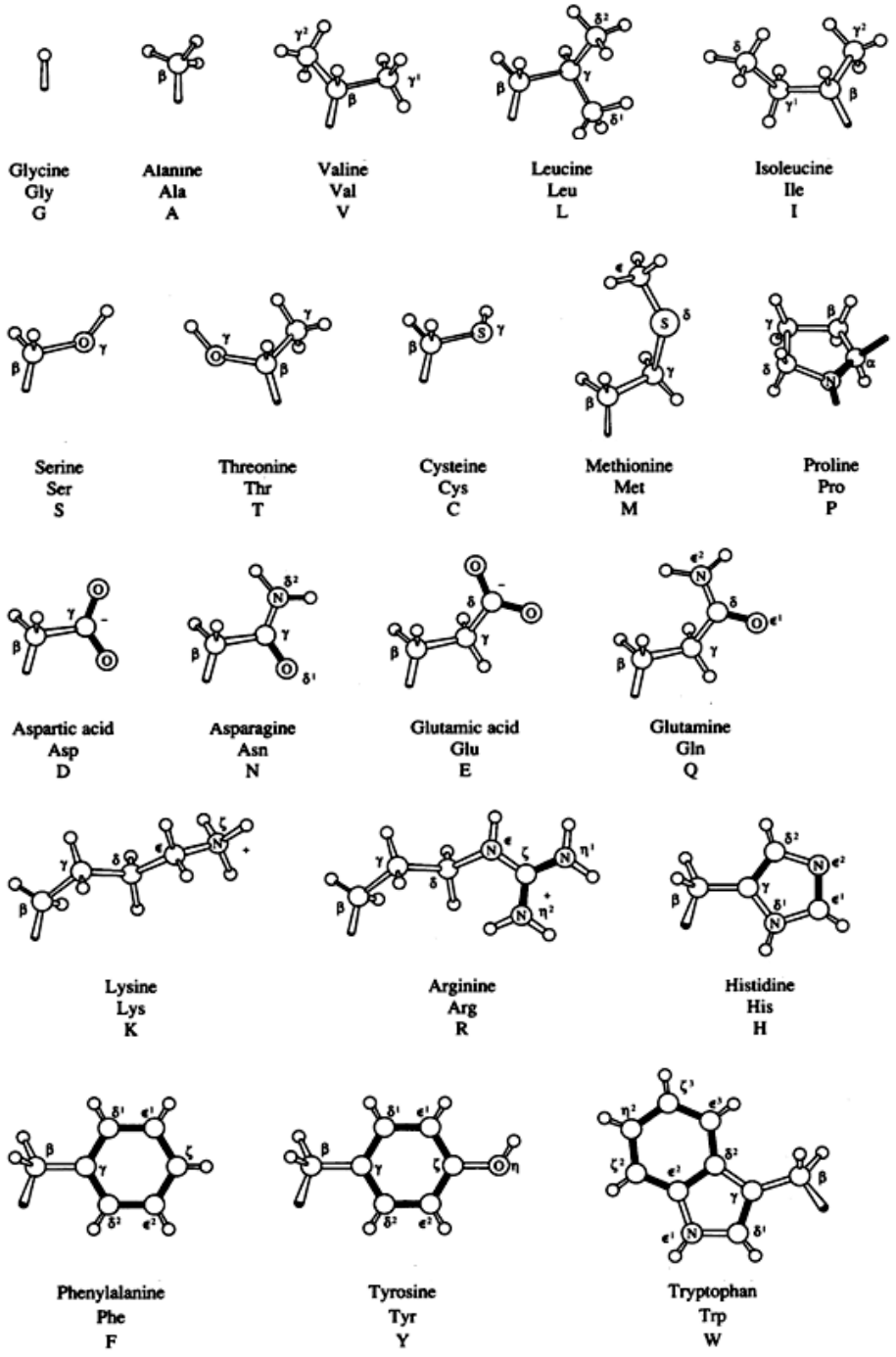


Figura 2.2: *Amminoacidi e le Loro Strutture*

Gli amminoacidi vengono legati tra di loro da un legame che in biochimica prende il nome di legame o giunto peptidico. L'unione di due o più amminoacidi lascia alle due estremità della loro catena due gruppi liberi, che possono ulteriormente reagire legandosi ad altri amminoacidi. Una catena di più amminoacidi, che prende il nome generico di polipeptide oppure di oligopeptide in caso che il numero di amminoacidi coinvolti è limitato, costituisce una proteina. Tale catena che rappresenta la sequenza unica di aminoacidi, viene chiamata catena peptidica o amino-acid sequence, e definisce la struttura tridimensionale di ogni proteina. Le proteine vengono chimicamente definite dalla loro struttura primaria (Figura 2.3).

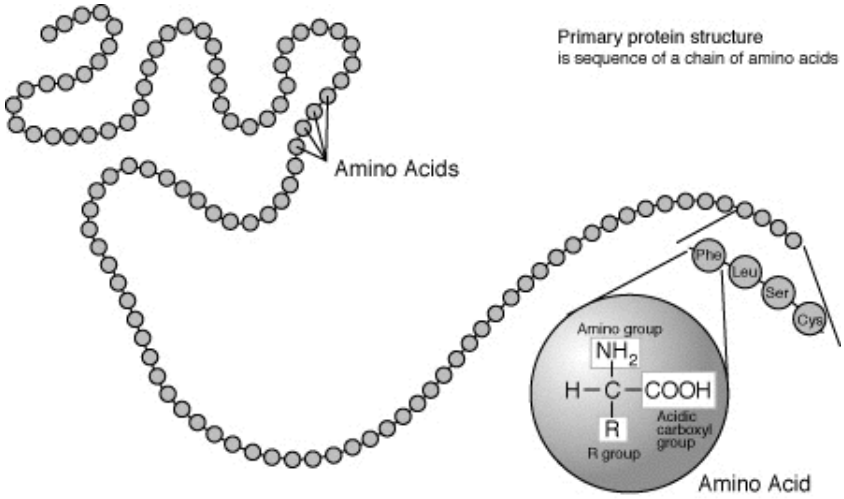


Figura 2.3: *Struttura Primaria*

Ognuno degli ammminoacidi che formano una catena polipeptidica, può essere sostituito da un altro come risultato di una mutazione, formando cosi innumerevoli alterazioni strutturali della proteina. Le proprietà di queste proteine possono differire l'una dall'altra in funzione del particolare amminoacido che è stato sostituito.

2.3 Mutazioni

A livello molecolare, un'alterata informazione genetica può condurre ad un errore nella sintesi delle proteine causando una modificazione della loro struttura primaria, producendo cosi proteine strutturalmente anomale. Queste alterazioni vengono chiamate mutazioni.

Come *mutazione* si deffinisce una trasformazione del materiale genetico (*genoma*) di una cellula di un organismo vivente o di un virus che può essere permanente e che può trasmettersi ai discendenti.[5, 6, 7]

Il genoma degli organismi contiene l'informazione genetica necessaria perche sia formata una proteina. La maggior parte dei genomi degli organismi sono costituiti da DNA, mentre quelli virali possono essere sia di DNA che di RNA. Le unita funzionali del DNA, sono i geni, formati da sequenze nucleotidiche multiple di tre (*codone*)[3](Figura 2.4).

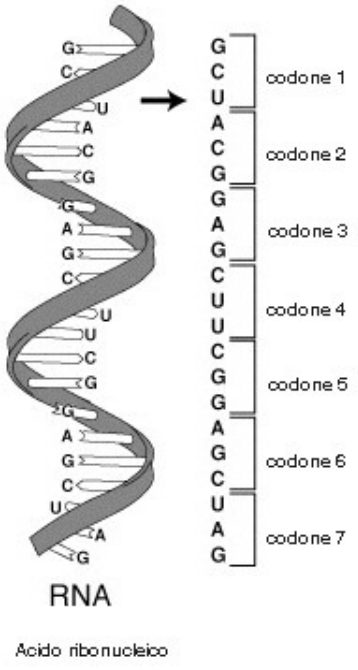


Figura 2.4: *Codone*

Un *gene* è normalmente composto da due regioni, una regolatrice e una codificante. La prima, è responsabile per la trascrizione del gene al momento opportuno durante lo sviluppo. Mentre la seconda, codificante, porta il codice genetico per la struttura di una molecola funzionale, generalmente una proteina.

I cambiamenti più gravi avvengono nei geni. Quindi, una mutazione che cambia la sequenza del DNA può cambiare durante la trascrizione la sequenza aminoacidica e potenzialmente ridurre o disattivare la funzionalità di una proteina.

Un cambiamento nella sequenza della regione regolatrice di un gene, può influenzare negativamente le proprietà della proteina e creare gravi disfunzioni cellulari. D'altra parte, molte mutazioni sono silenziose, senza mostrare alcun effetto evidente a livello funzionale.[5, 6, 7]

Tante sono le cause che possono provocare una mutazione. Possono essere il risultato di incidenti durante le operazioni normali sulle transazioni chimiche del DNA, ma anche durante una replicazione. Un'altra causa, può essere la esposizione a radiazione ad alta energia elettromagnetica (ad esempio, la luce ultravioletta o i raggi X), oppure ad ambienti con sostanze chimiche altamente reattive. Le mutazioni sono gli elementi di base grazie ai quali possono svolgersi i processi evolutivi. Determinano infatti la cosiddetta va-riabilità genetica, ovvero la condizione per cui gli organismi differiscono tra di loro per uno o più caratteri. La selezione naturale opera tramite la ricombinazione genetica, la quale promuove le mutazioni favorevoli a scapito di quelle sfavorevoli o addirittura letali. Essendo la maggior parte delle mutazioni non favorevoli, gli organismi hanno sviluppato diversi meccanismi per la riparazione del DNA dai vari danni che può subire, riducendo in questo modo il tasso di mutazione.

Ce ne sono diversi tipi. Il più semplice riguarda le variazioni delle singole coppie di basi, chiamata *base-pair substitutions*. In molti di questi casi, vengono sostituiti gli amminoacidi di una specifica posizione della proteina

codificata con degli altri non corretti, causando nella maggior parte delle volte, l'alterazione delle funzioni della proteina. Alcune sostituzioni di tipo base-pair producono un codone(codon) di stop. Normalmente, quando si verifica un codone di stop, alla fine di un gene, si arresta la sintesi proteica, ma quando si verifica in una posizione anomala, si ottiene una proteina troncata e non funzionale.

Un altro tipo di mutazione, è il *frameshift* che si verifica quando un numero dei nucleotidi, non un multiplo di tre, vengono inseriti o eliminati dalla sintesi proteica. Il risultato è che durante la trascrizione dal punto del gene mutato in poi, tutti gli amminoacidi vengono traduti in maniera sbagliata. Combinazioni più complesse che possono consistere in sostituzioni di base, inserimenti ed eliminazioni, si osservano anche in casi di geni già mutanti.[5, 6, 7]

I cambiamenti all'interno dei geni vengono chiamate *mutazioni puntiformi (**point mutations**)*[8](Figura 2.5). Le point mutations possono derivare da mutazioni spontanee che si verificano durante la replicazione del DNA.

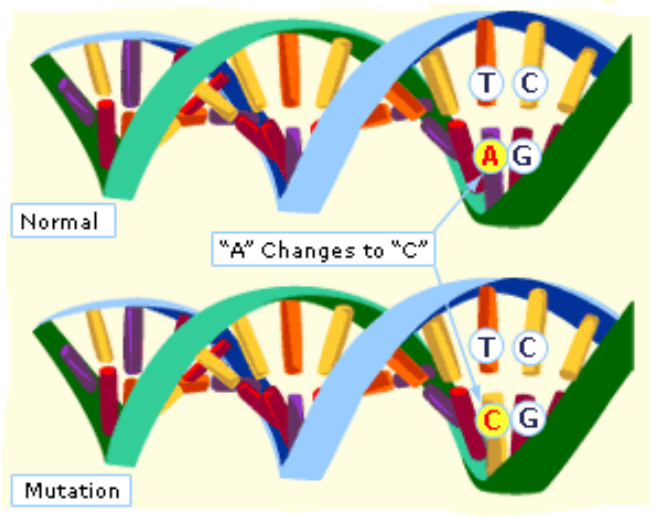


Figura 2.5: *Point Mutation*

Capitolo 3

Problema Bioinformatico

3.1 Energia Libera

È noto che tutte le proteine sono costituite da catene polipeptidiche differenti, le quali per svolgere correttamente le loro funzioni devono essere strutturate nella cosiddetta conformazione nativa. Tale conformazione è quella struttura tridimensionale stabile e funzionale, che viene caratterizzata da un minimo di energia potenziale e che consente alla proteina di svolgere adeguatamente la funzione a cui è deputata. Il processo che porta il peptide *unfolded* alla proteina strutturata nella forma *nativa*, biologicamente attiva, prende il nome di "folding"[9] (Figura 3.1).

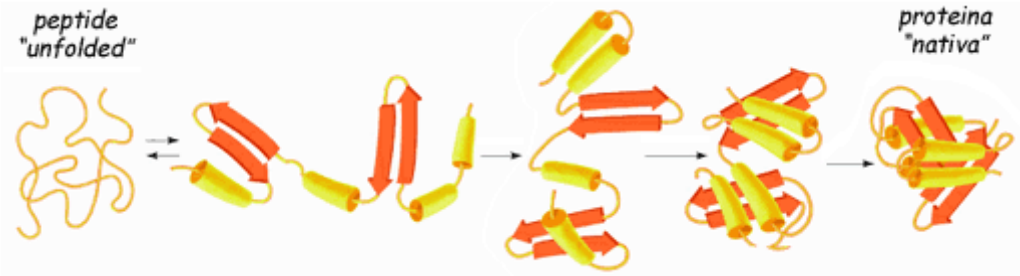


Figura 3.1: *Folding*

Il processo di folding, come tutti i processi spontanei, è una reazione chi-

mica in cui il sistema rilascia energia libera (***Gibbs free energy***)(Figura 3.2), spesso sotto forma di calore, e si sposta ad uno stato energetico più basso, termodinamicamente più stabile. Le reazioni di questo tipo sono, appunto, spontanee, ossia avvengono naturalmente con il corso del tempo e in precise condizioni di pressione e temperatura. Una proteina si denatura spontaneamente quando l'energia libera dello stato unfolded è minore dell'energia libera dello stato nativo. L'energia libera costante ha due co-mponenti, l'*entalpia "H"*, ovvero l'energia del sistema, e l'*entropia "S"*, una funzione di stato legata al livello di disordine nel sistema.[10]

$$\mathbf{G = H - T*S} \tag{3.1}$$

dove T è la temperatura espressa in gradi kelvin.

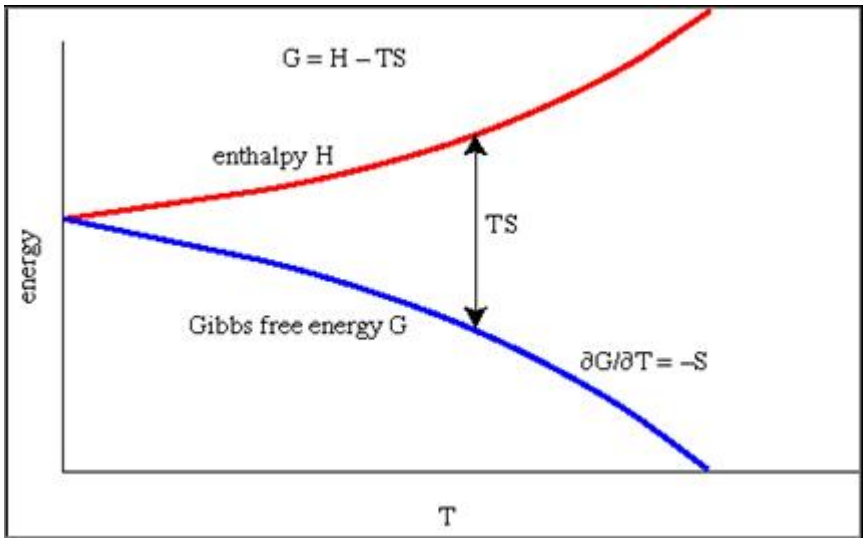


Figura 3.2: *Gibbs Free Energy*

In un processo spontaneo l'energia libera G assume il valore minimo. Nel caso delle proteine potremmo schematicamente e semplicisticamente assumere che esistano due soli stati: lo stato nativo e lo stato unfolded (la proteina non ha struttura ben definita). L'energia libera di questi due stati

dipende dalle grandezze termodinamiche, per esempio dalla temperatura. A bassa tempe-ratura è minore l'energia libera dello stato nativo, mentre ad alta temperatura è minore l'energia libera dello stato unfolded. In generale, note le energie libere, per poter capire quale dei due stati è quello prevalente in condizioni termodinamiche date, basta calcolare la differenza di energia libera tra i due stati:

$$\mathbf{G}_{nativo} - \mathbf{G}_{unfolded} = \mathbf{H}_{nativo} - \mathbf{H}_{unfolded} + \mathbf{T}(\mathbf{S}_{nativo} - \mathbf{S}_{unfolded}) \quad (3.2)$$

$$\mathbf{\Delta G} = \mathbf{\Delta H} \text{ - } \mathbf{T}^*\mathbf{\Delta S} \quad (3.3)$$

Se ΔG è maggiore di zero vuol dire che $G_{unfolded}$ è minore di G_{nativo} e dunque lo stato *Unfolded* è quello favorito. Se invece ΔG è minore di zero, allora $G_{nativo} < G_{unfolded}$ e dunque lo stato favorito è quello *nativo*.[9, 10]

3.2 Metodi Utilizzati

Nella progettazione delle proteine un problema importante da considerare è quello di capire se e in quale grado, una mutazione influenzerà la stabilità della nuova proteina rispetto al *wild type*. Diversi metodi sono stati implementati al fine di affrontare questo compito. La maggior parte dei quali, si basa sullo sviluppo di funzioni di energia diverse, adatte per calcolare la variazione della stabilità dell'energia libera.[1] Un altro metodo è quello che prevede la direzione verso la quale viene spostata la stabilità della proteina a causa della mutazione (vale a dire il segno $\Delta\Delta$G), anziché stimare direttamente i cambiamenti relativi alla stabilità sulla mutazione proteica, cioè il valore di $\Delta\Delta$G.

$$\mathbf{\Delta\Delta G} = \mathbf{\Delta G}_{mutato} - \mathbf{\Delta G}_{nonmutato} \quad (3.4)$$

Si può quindi predire la variazione della stabilità della struttura della proteina mutata, e, per la prima volta, prevedere in quale grado una mutazione

in una sequenza proteica influenzerà o meno la stabilità della proteina nativa. [1]

Tali predittori si basano su tecniche di machine learning. In particolare, nel presente lavoro mostreremo l'implementazione di un predittore basato su **Support Vector Machines (*SVM*)**.

3.2.1 Machine-Learning

Machine-Learning è una disciplina scientifica, strettamente connessa con il settore dell'intelligenza artificiale, che si occupa della progettazione e dello sviluppo di algoritmi che consentono ai sistemi informatici di evolvere comportamenti basati su dati empirici, come dati del sensore(sensor data) o database. Tale procedura viene chiamata *"data mining"*.

Tramite la tecnica di Machine-Learning si cerca di comprendere a livello matematico quali sono le capacità, le informazioni e i principi algoritmi-ci necessari per avere dei computer in grado di imparare dai dati e migliorare con l'apprendimento le loro prestazioni. Gli obiettivi di questa scienza sono sia la progettazione di migliori metodi di apprendimento automatici sia la comprensione delle questioni fondamentali del processo di apprendimento.

La disciplina di Machine-Learning trova applicazione in diversi settori come per esempio la visione artificiale o la robotica, la bioinformatica o anchel'ingegneria del software. I problemi affrontati dai metodi Machine-Learning riguardano *pattern recognition*, la *classificazione*, la *regressione* o la *density estimation*.

3.2.1.1 Algoritmi e Aprocci

L' apprendimento automatico (*Machine-Learning*) per poter trarre conclusioni da un campione, usa la teoria della statistica insieme a modelli matematici. C'è ,quindi, un duplice ruolo dell'informatica; in primo luogo, nell'addestramento, servono algoritmi efficienti per risolvere il problema,

ma anche per memorizzare e elaborare i dati. Mentre, in secondo luogo, una volta che un modello è stato appreso, deve essere un'efficiente rappresentazione e soluzione algoritmica. Quindi, l'efficienza del apprendimento o di inferenza dell' algoritmo scelto, è tanto importante quanto la sua accuratezza predittiva.

Esistono vari approcci e algoritmi di Machine Learning. Alcuni di questi vengono descritti sotto.

Artificial Neural Network: Una rete neurale artificiale (*Artificial Neural Network (ANN)*, o rete neurale (*NN Neural Network*)), è un modello matematico/informatico di calcolo basato sulle reti neurali biologiche. Tale modello è costituito da un gruppo di interconnessioni di informazioni costituite da neuroni artificiali, ossia costrutti matematici che imitano le proprietà dei neuroni viventi. Questi modelli matematici possono essere utilizzati sia per ottenere una comprensione delle reti neurali biologiche, ma ancor di più per risolvere problemi ingegneristici di intelligenza artificiale come quelli che si pongono in diversi ambiti tecnologici (in elettronica, informatica, simulazione, e altre discipline). In termini pratici le reti neurali sono strutture non-lineari di dati statistici organizzate come strumenti di modellazione. Esse possono essere utilizzate per simulare relazioni complesse tra ingressi e uscite che altre funzioni analitiche non riescono a rappresentare. Una rete neurale artificiale è costituita da numerosi nodi (neuroni) collegati tra di loro. Ci sono due tipi di neuroni, quelli dell'ingresso e quelli di elaborazione. Esiste un valore detto "*peso*" che indica l'efficacia sinaptica della linea di ingresso e serve a quantificarne l'importanza. Un ingresso molto importante ha un peso elevato, mentre un ingresso poco utile all'elaborazione ha un peso inferiore. Una rete neurale artificiale riceve segnali esterni tramite i neuroni d'ingresso, ciascuno dei quali è collegato con numerosi nodi interni, organizzati in più livelli. Ogni nodo elabora i segnali ricevuti e trasmette il risultato a nodi successivi. Si continua cosi fino ad arrivare al livello di uscita.

Bayesian Network: Un Bayesian network (*Belief Network, BN*) è un modello probabilistico di grafica che rappresenta un insieme di variabili casuali e le loro indipendenze condizionali tramite un *grafo diretto aciclico (DAG)*. Le reti Bayesiane sono grafi diretti e aciclici i cui nodi rappresentano le variabili casuali in senso Bayesiano. Possono, quindi, essere quantità osservabili, le variabili latenti, i parametri sconosciuti o ipotesi. Gli archi, invece, rappresentano le dipendenze condizionali; i nodi che non sono collegati rappresentano variabili che sono condizionalmente indipendenti l'uno dall'altro. Ogni nodo è associato ad una funzione di probabilità che prende in input un particolare insieme di valori dal suo nodo padre (parent node) e fornisce la probabilità della variabile rappresentata da se stesso. Per esempio, se i nodi genitori sono m variabili booleane, la funzione di probabilità potrebbe essere rappresentata da un tavolo di 2^m voci, una per ciascuna delle 2^m combinazioni possibili dei suoi genitori di essere vero o falso. Le reti Bayesiane che modellano delle sequenze di variabili (ad esempio i segnali vocali o sequenze di proteine) sono chiamati reti Bayesiane dinamiche (*dynamic Bayesian network*). Una Bayesian network potrebbe essere applicata per rappresentare le relazioni probabilistiche tra malattie e sintomi. Tenuto conto dei sintomi, la rete può essere utilizzata per calcolare le probabilità della presenza di varie malattie.

Hidden Markov Model: Un hidden Markov model *(HMM)* è un modello statistico in cui il sistema modellato viene considerato un processo di Markov con lo stato inosservato. Un HMM può essere considerato come la più semplice rete dinamica bayesiana *(dynamic Bayesian network)*. In un classico modello di Markov, lo stato è direttamente visibile all'osservatore, e quindi le sue probabilità di transizione sono gli unici parametri. Dall'altra parte, in un HMM, lo stato non è direttamente visibile, a differenza del output dipendente che lo è. L'aggettivo 'hidden' si riferisce alla sequenza di stati attraverso i quali passa il modello, e non ai parametri. Anche se

i parametri del modello sono noti, il modello rimane sempre 'hidden'. Gli Hidden Markov Models sono particolarmente noti per la loro applicazione nel *temporal pattern recognition* (riconoscimento di forme temporali), come per esempio nella scrittura a mano, nel *riconoscimento dei gesti*, nelle *partiture musicali* e la *bioinformatica*.

Cluster Analysis: Il Clustering (o *Cluster Analysis*) è un insieme di tecniche di analisi multivariata dei dati, volte alla selezione e raggruppamento di elementi omogenei in un insieme di dati. Tutte le tecniche di clustering si basano sul concetto di distanza tra due elementi. Infatti la bontà delle analisi ottenute dagli algoritmi di clustering dipende molto dalla scelta della metrica, e quindi da come è calcolata la distanza. Le funzioni di distanza più comuni sono:

- *La distanza euclidea.*
- *La distanza di Manhattan (aka norma taxi o 1-norma)*
- *La norma del massimo (aka norma infinito)*
- *La distanza di Mahalanobis* corregge i dati per diverse scale e le correlazioni delle variabili
- *L'angolo tra due vettori* può essere utilizzato come una misura di distanza quando si usa il clustering con dati ad alte dimensionali.
- *La distanza di Hamming* misura il numero minimo di sostituzioni necessarie per cambiare un membro in un altro.

Un'altra distinzione importante è se il clustering utilizza distanze simmetriche o asimmetriche. Molte delle funzioni di distanza sopra indicate hanno la proprietà che le distanze sono simmetriche (la distanza tra l'oggetto A a B è uguale alla distanza da B ad A). Questa proprietà non e' sempre verificata. Gli algoritmi di clustering raggruppano gli elementi sulla base

della loro distanza reciproca, e quindi l'appartenenza o meno ad un insieme dipende da quanto l'elemento preso in esame è distante dall'insieme stesso. Le tecniche di clustering si possono basare principalmente su due filosofie:

- Dal basso verso l'alto *(Bottom-Up)*: Questa filosofia prevede che inizialmente tutti gli elementi siano considerati cluster a sé, e poi l'algoritmo provvede ad unire i cluster più vicini. L'algoritmo continua ad unire elementi al cluster fino ad ottenere un numero prefissato di cluster, oppure fino a che la distanza minima tra i cluster non supera un certo valore.

- Dall'alto verso il basso *(Top-Down)*: All'inizio tutti gli elementi sono un unico cluster, e poi l'algoritmo inizia a dividere il cluster in tanti cluster di dimensioni inferiori. Il criterio che guida la divisione è sempre quello di cercare di ottenere elementi omogenei. L'algoritmo procede fino a che non ha raggiunto un numero prefissato di cluster.

Support Vector Machine: **SVM** è l'acronimo per ***Support Vector Machine*** ed è un insieme di metodi di apprendimento automatico per la regressione e la classificazione di dati.[12]Questa tecnica che mira a istruire un sistema informatico in modo da consentirgli di risolvere dei compiti in automatico viene chiamato Apprendimento Supervisionato. Lo scopo delle SVM è quello di classificare gli esempi, di un insieme di addestramento, a classi diverse, per costruire in seguito un modello che sarà in grado di predire in quale classe ricadono i nuovi esempi. Per fare questo una SVM rappresenta gli esempi come punti in uno spazio in modo che le diverse classi vengano separate maggiormente possible. Siccome la dimensione dello spazio è variabile, esistono tanti gap che separano gli esempi, che vengono chiamati ***hyperplanes***[11] *(iperpiani)*. Tra tutti i hyperplanes possibili viene scelto quello che massimizza la distanza dagli esempi di ogni classe, e viene detto ***maximum-margin hyperplane*** *(iperpiano di separazione)*. L'apprendimento consiste nel trovare appunto l'iperpiano di separazione migliore per gli

esempi, e usato in seguito per predire in quale classe appartengono i nuovi esempi, basandosi sulla loro posizione nello spazio.

Gli esempi più vicini all'iperpiano di separazione vengono detti vettori di supporto ***(support vector)*** ed è da essi che le SVM prendono nome.

Inizialmente le SVM si usavano per risolvere problemi in cui c'era un vettore di p-dimensioni (una lista di 'p' numeri) e si doveva separare con un iperpiano di p-1 dimensioni. Si parlava, quindi, di un classificatore lineare(Figura 3.3). Successivamente si e' verificato che si possono usare anche con classificazioni non lineare applicando il kernel trick. Alcuni dei kernel piu' communi sono indicati sotto:

- *Polynomial (homogeneous)*: $k(x_i, x_j) = (x_i * x_j)^d$
- *Polynomial (inhomogeneous)*: $k(x_i, x_j) = (x_i * x_j + 1)^d$
- *Radial Basis Function*: $k(x_i, x_j) = exp(-\gamma||x_i - x_j||^2)$ per $\gamma > 0$
- *Gaussian Radial basis function*: $k(x_i, x_j) = exp(-\frac{||x_i - x_j||^2}{2\sigma^2})$
- *Hyperbolic tangent*:$k(x_i, x_j) = tanh(\kappa x_i * x_j + c)$ per alcuni k>0 e c<0

Dati alcuni dati di "training" D, un insieme di n punti della forma:

$$D = \{(X_i, c_i)|X_i \in (-1, 1)\}_{i=1}^{n}$$

dove c_i può essere o 1 o -1, indicando la classe in cui fa parte il punti X_i.

Ogni X_i è un vettore di p dimensioni. Si vuole trovare l'iperpiano di separazione "*maximum-margin hyperplane*" che divide i punti con $c_i = 1$ da quelli con $c_i = -1$. Ogni iperpiano può essere scritto come un insieme di punti X che soddisfano l'equazione:

$$W * X - b = 0$$

dove W è un vettore normale, perpendicolare all'iperpiano. Il parametro $\frac{b}{||W||}$ determina l'offset del iperpiano dall' origine del vettore W.

Si vogliono trovare i W e b che possono massimizzare il margine o la distanza tra gli iperpiani paralleli che sono distanti il più possibile, pur separando i dati. Tali iperpiani, sono descritti dalle seguenti equazioni:

$$W * X - b = 1 \text{ e } W * X - b = -1$$

Se i dati di training sono linearmente separabili, si possono selezionare i due iperpiani di margine in un modo che non ci siano punti tra loro and poi cercare di massimizzare la loro distanza (Figura 3.4).

Tramite la geomtria, si può trovare che la distanza tra i due iperpiani è $\frac{2}{||W||}$, quindi, si deve minimizzare il $||W||$. lnoltre si deve evitare che i punti dati rientrino nel margine, aggiungendo il vincolo seguente:

Per ogni i:

- Se X_i fa parte della prima classe, $W * X_i - b \geq 1$
- Se X_i fa parte della seconda classe, $W * X_i - b \leq -1$

Questo può essere scritto anche cosi :

$$c_i(W * X_i - b) \geq 1, \forall(-1 \leq i \leq n)$$

Mettendo tutto insieme, si ottene il problema di ottimizzazione:

Minimizzare in (W,b) ||W|| ($\forall$ i = 1,...,n)), $c_i(W * X_i - b) \geq 1$

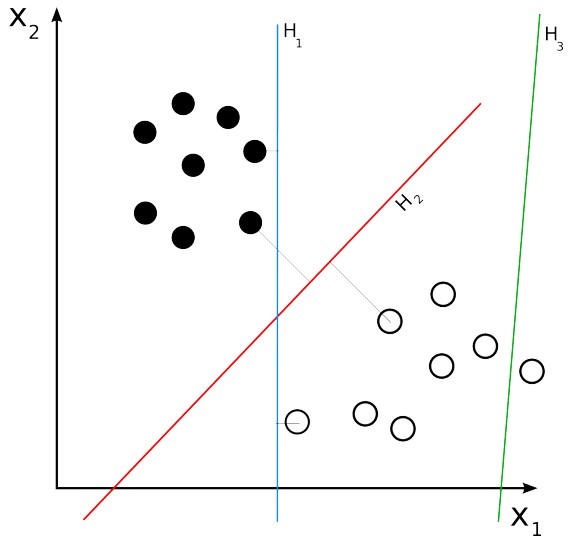


Figura 3.3: *Separating Hyperplanes*. H3 (verde) non separa le 2 classi. H1 (blu) e H2 (rosso) lo fanno. H1 (blu) con un piccolo margine e H2 (rosso) con il margine massimo.

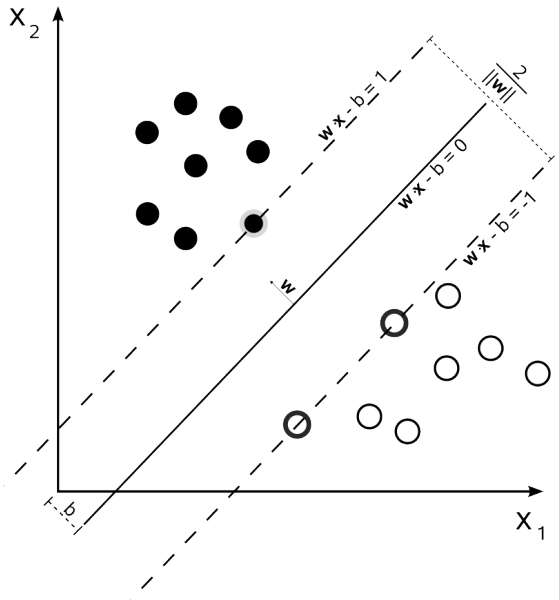


Figura 3.4: *Maximum-Margine Hyperplane*. Maximum-Margine Hyperplane e margini ottenuti tramite SVM.

3.3 Predittore

In questo capitolo vengono esposte tutte le scelte e le informazioni ralative all'implementazione del predittore di stabilità. Innanzitutto, vengono spiegati i metodi usati e le modalità scelte tramite le quali si valutano i risultati ottenuti dal predittore. Inoltre, vengono descritti i parametri contenenti tutte le informazioni, relative alle proteine, neccessarie per effetuare i diversi test ma anche le diverse codifiche usate per creare gli input da passare al predittore.

3.3.1 Scelte Implementative

Lo scopo finale è quello di predire se una determinata mutazione aumenta o diminuisce la stabilità proteica, senza calcolare il valore esatto di $\Delta\Delta G$. Una volta fatta la predizione, si devono valutare i risultati ottenuti e determinare alla fine l'accuratezza del predittore. La tecnica che si usa per valutare i risultati di un'analisi statistica si chiama *Cross-Validation* e consiste nella suddivisione di un dataset in sottoinsiemi complementari.[1, 2] Ad ogni passo di cross validation si esegue l'addestramento della rete su un se di sottoinsiemi (training set) mentre la qualita' e l'accuratezza del predittore addestrato si valida sul sottoinsieme complementare (test set). Per ridurre la variabilità, si effettuano tanti passi di Cross-validation utilizzando sempre partizioni diverse. Alla fine il risultato si ottiene calcolando la media di tutti i passi. Alcuni dei metodi di Cross-validation piu' communi sono:

- **Repeated Random Sub-Sampling Validation**
- **K-fold Cross-Validation**
- **k x 2 Cross-Validation**
- **Leave-One-Out**

Nel nostro caso, si usa l'ultimo tipo, il ***"Leave-One-Out"***. Tale metodo consiste nell'estrarre dall'intero set uno o più elementi, addestrare il predittore sul resto degli elementi e valutare, in seguito, l'accuratezza della predizione sul set di elementi rimosso. Lo stesso processo si effettua per ogni singolo elemento di ogni sottoinsieme. Al termine del procedimento si ottengono tante valutazioni della bontà del predittore.

I due tipi di dataset che vengono usati nel caso delle proteine, sono stati scelti da una database di dati termodinamici basati sulle mutazioni delle proteine che si chiama "ProTherm".[13] Sono stati scelti 206 sets (103 per ogni tipo, training e test set) che contengono informazioni su 3224 singole mutazioni su 142 diverse proteine.

3.3.1.1 Parametri

A fine di evidenziare le caratteristiche principali, che sono responsabili delle variazioni della stabilità proteica, dovute ad una mutazione, vengono usati molti parametri sui quali si basano le diversi codifiche di input. Di seguito, vengono descritti uno per uno tutti i parametri, relativi alla proteine.

Classe: Le classi rappresentano la variazione dell'energia della struttura della proteina sottoposta ad una mutazione. Ci sono tre diverse classi, indicate tramite i numeri "-1", "0" e "1" che rappresentano rispettivamente la variazione negativa, la neutra e quella positiva. L'accuratezza dei risultati ottenuti dai test principali, si basa su questo parametro.

Accessibilità: La *superficie accessibile al solvente* o *"accessibilità"*, di un atomo è la superficie dell'atomo che è esposta al solvente. Per calcolare l'accessibilità, si immagina di ruotare una molecola d'acqua sulla superficie della proteina, considerando l'area di contatto dalla molecola. Su ciascun residuo i valori di accessibilità vanno da 0 a 300 $Å^2$. Per permettere un confronto tra diversi aminoacidi, si considera la percentuale di area accessibile rispetto

a quella totale *(accessibilità relativa)*. I residui con un'accessibilità relativa minore del 9% vengono chiamati *"idrofobi"* e tendono ad essere scarsamente solubili in acqua e si ritrovano solitamente immersi *(buried)* nella parte interna delle proteine. Mentre, quelli con una accessibilità relativa maggiore del 36%, gli *"idrofili"*, sono dotati di catena laterale "polare" o "carica" che interagisce favorevolmente con l'acqua. Tali residui, sono considerati esposti *(exposed)*. In conclusione, si può dire che L'accessibilità al solvente è un valore strettamente collegato all'organizzazione spaziale delle strutture secondarie delle proteine, utile per indicare le posizioni di ogni residuo, sulla superficie o nel nucleo.

Distance Threshold: La soglia sulla distanza *(Distance Threshold)* è, innanzitutto, un valore variabile perché può essere scelto da chi effettua i test ma e anche uno dei parametri più importanti perche su di quello si basano le modalità usate per i test.

La distanza a cui si riferisce la soglia è la distanza tra due residui nella struttura tridimensionale della proteina. Per calcolare il valore della distanza ci sono diversi metodi come per esempio: *Manhattan distance (o taxicap)*, *maximum norm (o infinity norm)*, *Mahalanobis distance*, *Hamming distance* o *Euclidean distance*). In questo caso la funzione utilizzata è quella che calcola la ***Distanza Euclidea***.

Profili-PSSM: In base a questi due parametri si calcola l'accuratezza del predittore e si stabilisce se le sue prestazioni hanno avuto un miglioramento significativo o meno. Per ogni proteina c'è uno profilo e una PSSM *(Position Specific Scoring Matrix)* che vengono rappresentate tramite una matrice "P" Lx20, dove L è la lunghezza della sequenza della proteina in questione e 20 sono i diversi amminoacidi.

Per calcolare i profili, delle proteine, si usa uno strumento, chiamato **Blast** *(Basic Local Alignment Search Tool)* che tramite un algoritmo confronta una sequenza di interesse con un database di sequenze, già conosciute,

a scopo di trovare delle somiglianze (allineamento). La matrice 'P' del profilo rappresenta la ***Multiple Sequence Alignment (MSA)*** (*sequenza di multi-allineamento*). Ogni elemento P_{ai} della "P", indica la frequenza normalizzata di un residuo di tipo 'a' nella i-esima posizione.

Una PSSM *(Position Specific Scoring Matrix)*, invece, può prendere dei valori sia positivi che negativi. Tali valori indicano la frequenza con la quale avviene la sostituzione di un specifico amminoacido. Se la sostituzione si verifica meno frequentemente del previsto i valori sono negativi, nel caso contrario, invece, sono positivi. Le PSSM vengono costruite tramite **PSI-Blast***(Position Specific Iterative BLAST)* e si basano sui profili ottenuti inizialmente tramite BLAST.

3.3.2 Input

Tutti i parametri discritti sopra, vengono utilizzati per creare vettori di elementi che costituiscono gli input da passare nella SMV, per poter addestrare prima e testare dopo i diversi set di proteine. In totale ci sono 6 diversi formati di input. Due principali e altri quattro che si ottengono aggiugendo ai primi le matrici di profilo e PSSM.

Le due principali modalità sono costituite da un vettore di 42 valori. In entrambi i casi, il primo numero rappresenta la classe appartiene ogni vettore di input e può prendere i valori "-1", "0" e "1". I prossimi 20 valori (uno per ognuno dei 20 amminoacidi 'V', 'L', 'I', 'M', 'F', 'W', 'Y', 'G', 'A', 'P', 'S', 'T', 'C', 'H', 'R', 'K', 'Q', 'E', 'N', 'D') servono per definire in modo esplicito la mutazione. Con " -1" viene indicato l'elemento che è stato mutato, con "1" il nuovo residuo che va a sostituire quello originale, mentre tutti quelli rimasti vengono indicati con 0. In seguito c'è un valore che rappresenta l' accessibilità dell' aminoacido relativo alla mutazione. Per ogni residuo, tale valore, viene normalizzato secondo la sua accessibilità massima. Nella tabella sotto indicata vengono raggruppate le accessibilità massime dei 20 tipi di amminoacidi.

Accessibilità Massime degli Amminoacidi									
A	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**K**	**L**
106.0	135.0	163.0	194.0	197.0	84.0	184.0	169.0	205.0	164.0
M	**N**	**P**	**Q**	**R**	**S**	**T**	**V**	**W**	**Y**
188.0	157.0	136.0	198.0	248.0	130.0	142.0	142.0	227.0	222.0

Tabella 3.1: Accessibilità Massime degli Amminoacidi

La formula corretta per calcolare il valore di accessibilità è la seguente:

$$Min\{100, Round[100 * (Acc_{pos}/Acc_{max})]\} \tag{3.5}$$

Infine, negli ultimi 20 valori si può trovare l'unica differenza che distingue i due formati di input. Nella prima modalità, i valori assegnati ad ogni posizione, rappresentano il numero degli aminoacidi che si trovano in una distanza minore o uguale ad una soglia, dall'amminoacido mutato. Mentre, nella seconda modalità, non ha importanza il numero totale di ogni aminoacido ma se esista o meno, almeno un residuo di ogni tipo di aminoacidi. Questo significa che ogni posizione può prendere il valore 1 se esiste almeno un residuo entro la soglia scelta e 0 in caso contrario. La distanza in questione, si riferisce alla distanza euclidea tra il residuo mutato e tutti gli altri che fanno parte della proteina e viene misurata in Å(*Angstrom*).

Come abbiamo già detto, a partire dalle prime due codifiche di input, vengono create altre quattro. Due per ognuna delle prime modalità. Per la

creazione delle nuove codifiche, vengono usate le 2 matrici, chiamate profilo e PSSM *(position-specific scoring matrix)*, in aggiunta ai 42 valori iniziali.

La riga i-esima di un profilo o di una PSSM corrisponde al residuo della i-esima posizione della proteina. Se il residuo mutato è l' i-esimo residuo,vengono aggiunte nella codifica dell' input le righe del profilo che vanno da "i-k" a "i+k". Il "k" è una soglia che può variare.

Nei casi in cui "i-k" diventa negativo oppure "i+k" diventa più lungo della sequenza, nell'input vengono aggiunti, rispettivamente al inizio o alla fine, tanti vettori di zeri quanti ne servono per occupare tutte le posizioni.

3.3.3 Calcolo dell'Accuratezza

Una volta effettuati tutti i test e ottenuti tutti i risultati si deve calcolare l'accuratezza. A proposito di questo si usano due valori: "*l'accuratezza media* ***(Mean Accuracy)***" e "*l'accuratezza totale* ***(Total Accuracy)***".

Il calcolo dell'accuratezza totale, si basa sul calcolo di una matrice chiamata **"Confusion Matrix"**. Ogni riga della matrice rappresenta le istanze di una classe come sono state previste, dal predittore, mentre ogni colonna rappresenta le istanze di una classe reale. In una *Confusion Matrix*, 3x3, ogni valore ha un significato specifico:

		Actual		
		-1	**0**	**1**
Predicted	**-1**	a	b	c
	0	d	e	f
	1	g	h	i

Tabella 3.2: Confusion Matrix

- a è il numero di previsioni corrette che un'istanza appartiene alla classe **-1**

- b è il numero di previsioni errate che un'istanza appartiene alla classe **-1** mentre in realtà appartiene alla classe **0**
- c è il numero di previsioni errate che un'istanza appartiene alla classe **-1** mentre in realtà appartiene alla classe **1**
- d è il numero di previsioni errate che un'istanza appartiene alla classe **0** mentre in realtà appartiene alla classe **-1**
- e è il numero di previsioni corrette che un'istanza appartiene alla classe **0**
- f è il numero di previsioni errate che un'istanza appartiene alla classe **0** mentre in realtà appartiene alla classe **1**
- g è il numero di previsioni errate che un'istanza appartiene alla classe **1** mentre in realtà appartiene alla classe **-1**
- h è il numero di previsioni errate che un'istanza appartiene alla classe **1** mentre in realtà appartiene alla classe **0**
- i è il numero di previsioni corrette che un'istanza appartiene alla classe **1**

L'accuratezza totale si calcola tramite la seguente equazione:

$$T.A. = \frac{a+e+i}{a+b+c+d+e+f+g+h+i} \tag{3.6}$$

Per ottenere, invece, *l'accuratezza media*, prima di tutto, si calcola, per ogni round di cross-validation, l'accuratezza tramite la matrice e poi si calcola la media e la deviazione standard di tutti i singoli risultati . La *deviazione standard* (***Standard Deviation***) rappresenta la differenza tra la i-esima misura e la media. Se tutte le deviazioni sono molto piccole, le nostre misure saranno tutte vicine e quindi, si presume che siano molto precise.

Capitolo 4

Risultati e Conclusioni

4.1 Risultati

4.1.1 Risultati dei Test in Assenza di Profili e PSSM

Inizialmente vengono effetuati dei test sulle prime due modalità, usando come soglia i numeri da 8 fino a 11, a scopo di trovare la soglia che permette l'accuratezza migliore. La tabella indicata sotto indica i risultati ottenuti. I valori "M.A." (Mean Accuracy), "S.D."(Standard Deviation), "T.A." (Total Accuracy) rappresentano rispettivamente l'accuratezza media, la deviazione standard e l'accuratezza totale basata sulla matrice di classificazione.

Predictor Model: Modalità 1		
Distance Threshold	**M.A.±S.D.**	**T.A.**
D.T. 8	***48.82±25.27***	55.09
D.T. 9	48.69±26.05	55.80
D.T. 10	46.89±26.94	***56.14***
D.T. 11	48.42±25.56	55.30

Tabella 4.1: Risutalti per Modalità 1, Soglia: (8, 9, 10, 11)

Predictor Model: Modalità 2		
Distance Threshold	**M.A.±S.D.**	**T.A.**
D.T. 8	50.28±24.79	56.30
D.T. 9	51.83±26.05	56.67
D.T. 10	50.94±25.96	56.80
D.T. 11	***52.45±25.83***	***58.31***

Tabella 4.2: Risutalti per Modalità 2, Soglia: (8, 9, 10, 11)

Da questi risultati possiamo concludere che la predizione migliore osservando l'accuratezza totale per la prima modalità si ottiene usando la *soglia 8*. Mentre il calcolo della accuratezza tramite la matrice ci indica come soglia con il miglior risultato, la *soglia 10*. Invece per la modalità 2 sia tramite la accuratezza totale, ottenuta facendo la media di tutte accuratezze individuali, sia calcolandola tramite la matrice finale la predizione migliore si ottiene per la *soglia 11*. La deviazione standard ci dice che la qualità della predizione varia di molto da proteina a proteina.

4.1.2 Risultati dei Test con Profili e PSSM

Una volta effettuati i primi test, vengono identificate le distance threshold che riportano i risultati migliori, per entrambe le modalità e si effettuano i nuovi test aggiungendo come parametri i profili e le PSSM. Per la modalità 1 usiamo le soglie 8 e 10, tenendo conto dei risultati tramite l'accuratezza media e accuratezza totale, rispettivamente. Mentre per la seconda modalità i test si effettuano per la soglia 11. In entrambi i casi la soglia "k" prende i valori 0, 6, 7 o 8.

Predictor Model: Modalità 1, Distance Threshold: 8				
	PSSM		**Profilo**	
Threshold K	*M.A.±S.D.*	*T.A.*	*M.A.±S.D.*	*T.A.*
K 0	***54.37±26.89***	***59.08***	***49.05±27.95***	***53.79***
K 6	46.43±29.69	48.29	45.93±30.63	47.15
K 7	46.03±29.91	47.86	45.93±30.63	47.15
K 8	46.23±30.23	47.07	45.94±30.63	47.15

Tabella 4.3: Risutalti per Modalità 1, Soglia: 8 con PSSM e Profilo

Predictor Model: Modalità 1, Distance Threshold: 10				
	PSSM		**Profilo**	
Threshold K	*M.A.±S.D.*	*T.A.*	*M.A.±S.D.*	*T.A.*
K 0	***53.81±26.91***	***59.70***	***49.30±28.38***	***53.58***
K 6	45.89±30.13	48.17	45.93±30.63	47.15
K 7	46.24±30.22	47.83	45.93±30.63	47.15
K 8	46.20±30.24	47.67	45.94±30.63	47.18

Tabella 4.4: Risutalti per Modalità 1, Soglia: 10 con PSSM e Profilo

Predictor Model: Modalità 2, Distance Threshold: 11				
	PSSM		**Profilo**	
Threshold K	*M.A.±S.D.*	*T.A.*	*M.A.±S.D.*	*T.A.*
K 0	***52.24±27.05***	***59.05***	***49.27±27.40***	***54.26***
K 6	46.57±29.75	48.42	45.93±30.63	47.12
K 7	46.03±29.91	47.86	45.93±30.63	47.22
K 8	45.99±29.90	47.70	45.94±30.63	47.25

Tabella 4.5: Risutalti per Modalità 2, Soglia: 11 con PSSM e Profilo

Come si può vedere dalle tabelle sopra indicate, usando entrambe le modalità del predittore, in aggiunta dei profili e delle PSSM con threshold che varia tra 6, 7 e 8 , i risultati indicano un peggioramento dell'accuratezza. Invece quando si usa la soglia k uguale a 0, non solo si ottengono valori migliori da quelli ottenuti dalle altre soglie (6, 7, 8) ma in alcuni casi anche da quelli ottenuti dal predittore senza l'uso dei profilli e le PSSM.

Più nello specifico, dai due test effettuati è chiaro che il vicinato dei residui mutati (k=6, 7, 8) non fornisce nessuna informazione aggiuntiva al predittore sia con l'utilizzo di PSSM sia con l'utilizzo di profili. I dati contenuti nella PSSM relativi al solo residuo mutato (k=0) ci permettono di aumentare l'accuratezza della predizione rispetto al modello base. Al contrario, l'informazione contenuta nel profilo relativamente alla posizione mutata non sembra essere invece rilevante.

4.2 Conclusioni

La creazione di una base di dati contenente delle informazioni sulla stabilita delle proteine, ha aumentato le possibilità di utilizzo dei metodi di machine-learning per fare delle predizioni sulla variazione della stabilita proteica. Lo scopo di questa tesi era quello di implementare una versione semplificata di un software già esistente, tramite l'applicazione delle tecniche di machine-learning e più nello specifico dell' approccio di SVM. Tale software ha il compito di effettuare delle predizioni considerando nuove parametri che vengono passati come input. In seguito, viene calcolata l'accuratezza dei risultati per testare le prestazioni del predittore e poter decidere se le informazioni aggiungente servono al miglioramento dei risultati ottenuti o meno.

I parametri aggiuntivi sono i profili e le PSSM delle proteine e i test sono stati effettuati per 'k' uguale a 0, 6, 7 e 8. Dai risultati ottenuti si può concludere che in generale i parametri non aggiungono molte informazioni

al predittore. Mentre soltanto nel caso di 'k' uguale a 0 c'e' un leggero miglioramento sulle predizioni.

Bibliografia

[1] Capriotti E., Fariselli P., Casadio R., I-mutant2.0: predicting stability changes upon mutation from the protein sequence or structure, In: Nucleic Acids Research Journal vol 33, Oxford Journals, 2005.

[2] Capriotti E., Fariselli P., Casadio R., A Neural-Network-Based method for predicting protein stability changes upon a single point mutation, In: Bioinformatics vol 20, Oxford University Press, 2004.

[3] Creighton, Thomas H. (1993). Chapter 1. Proteins: structures and molecular properties. San Francisco: W. H. Freeman.

[4] Turanov AA, Lobanov AV, Fomenko DE, Morrison HG, Sogin ML, Klobutcher LA, Hatfield DL, Gladyshev VN (January 2009). Genetic code supports targeted insertion of two amino acids by one codon. Science 323

[5] Bertram J., The molecular biology of cancer, Mol. Aspects Med. 21, 2000

[6] Aminetzach YT, Macpherson JM, Petrov DA., Pesticide resistance via transposition-mediated adaptive gene truncation in Drosophila, 2005.

[7] Burrus V, Waldor M, Shaping bacterial genomes with integrative and conjugative elements, Res. Microbiol, 2004.

[8] Freese E., The Difference between Spontaneous and Base-Analogue Induced Mutations of Phage T4, 1959.

[9] Wetlaufer D.B., Nucleation, rapid folding, and globular intrachain regions in proteins, 1973.

[10] Perrot, Pierre, A to Z of Thermodynamics, Oxford University Press, 1998.

[11] Boser B.E., Guyon T.M.,Vapnik V.N., Training Algorithm for Optimal Margin Classifiers, 1992

[12] RussellS., Norvig P., Intelligenza artificiale: un approccio moderno, Prentice Hall, 2003.

[13] Bava, K. A., Gromiha, M. M. , Uedaira, H. , Kitajima, K. , Sarai, A. ProTherm, versione 4.0: termodinamici database per le proteine e mutanti acidi nucleici Res., 32, D120-D121, 2004.

Ringraziamenti