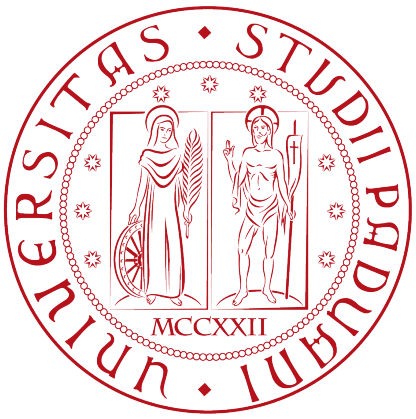
UNIVERSITA' DEGLI STUDI DI PADOVA

May 3, 2021



PREDICTING CANCERLECTINS

Ceccon Gioele, Pozzer Matteo, Toccane Alessandro

1 Introduzione

In questo documento si propone un metodo di codifica delle proteine, intese e trattate secondo la loro rappresentazione amminoacidica:

$$P = R_1 R_2 ... R_n$$

con P generica proteina e R_i i-esimo amminoacido con (i=1,..,n)
Il metodo ha come obbiettivo quello di trasformare una qualsiasi proteina in un array di lunghezza fissa. Gli array derivati dalle proteine, aventi tutti la stessa dimensione, rappresentano i pattern numerici forniti al classificatore SVM non-lineare della libreria LIBSVM, utilizzato nel nostro caso per individuare le "cancrolectine", proteine appartenenti alla famiglia delle lectine che svolgono un ruolo importante in determinati fenomeni tumorali.

Al fine di sviluppare un metodo di codifica che producesse in maniera efficiente pattern numerici sono stati sviluppati e analizzati diversi algoritmi. Per quanto riguarda la codifica, il problema principale incontrato è stato quello relativo alla dimensionalità dei pattern prodotti: trattandosi di un SVM non lineare, una dimensionalità alta comporta un calo delle prestazioni. Si è inoltre cercato di ottenere una codifica che limitasse la presenza di informazioni ridondanti: si è cercato di dare importanza non solo al valore del singolo amminoacido, ma anche, e soprattutto, alla sua posizione all'interno della catena aminoacidica.
Ad esempio, date due proteine:

$$P_1 = GM$$
$$P_1 = MG$$

Una codifica basata semplicemente sul tipo di aminoacidi presenti all'interno delle proteine produrrebbe due pattern identici. D'altra parte, una codifica basata oltre che sulla tipologia anche sulla posizione degli aminoacidi riesce a produrre due pattern diversi tra loro.

2 Descrizione Del metodo

L'idea alla base di questo metodo per la codifica delle proteine trae origine dall'approccio denominato "Amino Acid Composition". Questa tecnica effettua il conteggio delle occorrenze dei singoli aminoacidi presenti in una proteina, normalizzando le somme per la lunghezza della proteina stessa.

$$AS(i) = h(i)/N \quad i \in [1..20]$$

con *AS(i)* che rappresenta il numero di occorrenze di un dato amino acido in una proteina di lunghezza *N*.
Per la realizzazione di questo metodo, invece si è deciso di tener in considerazione anche la posizione in cui si trova ogni singolo amminoacido all'interno della proteina. Il risultato dell'operazione di codifica sarà sempre un vettore numerico di 20 elementi.
Ogni elemento del vettore fa riferimento solamente ad dato un amminoacido. Vi è una relazione lessicografica che lega una cella del vettore con il preciso amino-acido. Ovvero il primo elemento del vettore fa riferimento al amino-acido 'A', e l'ultimo elemento, di posizione 20, rimanda all'amino-acido 'Y'.
All'inizio ogni elemento del vettore viene inizializzato con un valore pari a 0. Dopodiché si scorre tutta la proteina, per ogni amminoacido il suo indice posizionale all'interno della Proteina viene diviso con la lunghezza della proteina stessa. Il risultato di tale operazione si va a sommare al valore già presente nella cella di riferimento del dato amino-acido all'interno del vettore di codifica.

$$Codifica(i) = Codifica(i) + j/N \quad i \in [1..20] \quad j \in [1...Size(Proteina)]$$

Il seguente metodo ha prestazioni temporali θ(n) con n che sta ad indicare la lunghezza della proteina.

Questo algoritmo ha ottenuto un'accuratezza del **75%**, riuscendo a distinguere correttamente **30** proteine su 40 del TestSet.

Metodo2 rappresenta il primo algoritmo inizialmente realizzato per la codifica delle proteine. Il seguente metodo prevede di codificare la proteina in un vettore di lunghezza fissa pari a 25 elementi. I primi 20 elementi, rappresentano

il numero di occorrenze di ogni singolo amminoacido all'interno della proteina. Nel ventunesimo elemento dell'array di codifica è presenta la lunghezza della proteina. Successivamente l'intera sequenza amminoacidica viene suddivisa in quattro sottogruppi distinti, di pari dimensione. Il valore di ogni gruppo è dato dalla somma della codifica ASCII degli amminoacidi presenti. Il risultato di quest'ultima operazione viene infine assegnato alle restanti quattro celle del vettore.

Nella seguente tabella vengono riportate le prestazioni di altri metodi di codifica, testati con le stesse impostazioni di SVM e lo stesso DataSet.

metodo	**accuratezza**	**Pattern classificati correttamente**
2-Gram	62.5%	25
Metodo2	67.5%	27
Global Encoding	70%	28
Amino Acid Composition	75%	30

3 Pseudo Codice

Algorithm 1 Descrittore

```
1:  function DESCRITTORE (proteina)
2:      keyset ← Lista aminoacidi
3:      codifica ← Integer Array(20)
4:      value ← Integer Array(20)
5:      i ← 0
6:      while i < size(Value) do
7:          value[i] ← i
8:          i ← i+1
9:          M ← Map(keyset,value)
10:     j ← 0
11:     while j < size(Proteina) do
12:         index ← codifica[M[Proteina[j]]]
13:         codifica[index] ← codifica[index] + j/size(Proteina)
14:         j ← j+1
15:     return codifica
```