Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

OXFORD

Predicting secondary structure of proteins: a comparison between GOR method and Support Vector Machines

Matteo Bolner

Abstract

Motivation: The determination of protein structure is one of the most important open problems of bioinformatics. In this project, two methods for the determination of the secondary structure of proteins, GOR and SVM, are implemented on a set of protein sequences and compared in performance. The potential of such methods is discussed in depth.
Results: Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text
Availability: Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text
Contact: name@bio.com
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The determination of protein function is a key passage in the understanding of biological systems, and is necessary for medical research and for the development of pharmaceutical drugs or for the improvement of other biotechnological applications of proteins (e.g design of novel enzymes). In order to determine the function of a protein it is necessary to know its three-dimensional structure: nowadays, the most reliable way to obtain it is experimentally, usually by determining its electron density through X-ray cristallography. However, experimental methods are expensive and time consuming, which is why on the Protein Data Bank (PDB) CITAZIONE there are only circa 147000 deposited protein structures, compared to the circa 561000 manually annotated and reviewed protein sequences on Swiss-Prot CITAZIONE. One of the core open problems of bioinformatics is the development of methods for the prediction of a protein's three-dimensional structure starting from its primary structure, the linear sequence of aminoacids. While the primary structure of a protein dictates its three-dimensional structure, when taken by itself as a string of characters it is not very informative: as of now, it is not possible to use it to accurately predict the tertiary structure. The secondary structure is much more informative: the folding of local segments of the protein in α-helices, β-sheets or coils can provide critical information for the identification of functional protein domains, in addition to constraints necessary for tertiary structure prediction techniques such as homology modelling, fold-recognition techniques, ab-initio and others CITAZIONE. Methods for the prediction of the secondary structure have been under development since the 1960's : the so-called "first generation" methods try to correlate the content of the single residues with the corresponding secondary structure; the most widely known is the Chou-Fasman method CITAZIONE, based on the relative frequency of each amino acid in each secondary structure derived from a set of solved protein structures. The parameters derived from these frequencies are used to predict local secondary structure motifs from aminoacidic sequences. The accuracy of these methods (around 50-60%) has been improved upon by "second generation" methods, which consider not only the single residue propensity but also the adjacent residues: for example, the GOR method CITAZIONE uses a sliding window of residues and implements information theory concepts and bayesian statistics. However, the accuracy of second generation methods is stalled at slightly more than 60%. The breakthrough came with the third generation of techniques, characterized by the usage of evolutionary information derived from the alignment of multiple homologous sequences: this way, the conservation of structure with variation in sequence is accounted for. Combined with the increasingly large datasets of protein structures and the advanced algorithms implemented, these techniques can reach more than 70% of accuracy. PSIPRED CITAZIONE, one of the most popular predictors available, implements two feed-forward neural networks that process psi-blast output; its multi-class accuracy can reach 80-85%. Jpred4 CITAZIONE is another well-known predictor, based on an ensemble of 100 Bidirectional Recursive Neural Networks (BRNNs); together with the prediction of secondary structure, it also predicts solvent accessibility and coiled-coil regions.

CITA ANCHE ARTICOLI IN CARTELLA DROPBOX SS-PREDICTION

Machine learning techniques such as neural networks and support vector machines CITAZIONE are taking an increasingly bigger role in the prediction of secondary structure: the aforementioned Jpred4 and PSIPRED implement neural networks, and SVMs are being used as

classifiers to assign secondary structures to residues using sequence profiles as input [https://www.ncbi.nlm.nih.gov/pubmed/11327775]. The goal of this project is to compare the performance of the GOR and SVM methods: to do so, after training them with sequence profiles obtained from multiple sequence alignments, they are used to to predict the secondary structure of a restricted set of sequences. The predictions are then compared with the original secondary structure of the sequences, which was already known, to obtain the model performance. PRELIMINARY RESULTS

2 Material and methods

2.1 Training dataset

The training of the models was performed on the dataset used in the training of the Jpred4 predictor, obtained from 1987 representative sequences from each superfamily in SCOP v2.04 CITAZIONE: the choice of this heterogeneous dataset assures that all possible three-dimensional folds are covered enough, and reduces the likelihood of trivially detectable similarities between sequences; with this data, a good snapshot of the real protein space is provided.
From the original set of sequences, additional filtering steps were performed and 489 sequences removed:

- 3D structure with resolution lower than 2.5Å [306]
- sequence length lower than 30 residues and higher than 800 [27]
- domains made up of multiple chains [20]
- missing or incomplete DSSP information [110]
- inconsistencies between PDB, DSSP and other file definitions [17]

The resulting dataset was split into training [1348] and blind testing sets [150]. While the training set was used for the training of the models in this project, the blind testing set obtained from the Jpred dataset was not used, and a new one was generated from scratch. For each sequence of the training set, the residue sequence in FASTA format and the secondary structure sequence in DSSP format were provided.

2.1.1 Statistical analysis of the dataset (and blindset???)

Some basic statistical analyses were performed on the dataset:

- distribution of the secondary structure
- residue composition of the entire dataset
- fraction of helix, strand and coil conformations
- composition of windows of 17 residues !!!da capire se anche coil e overall nelle heatmap!!!! (overall, helix, strand and coil residues)
- taxonomic classification (kingdom and species)
- structural classification (SCOP class)

2.2 Blind testing dataset

Since the training of the model is performed on the whole Jpred4 dataset, in order to evaluate the final performance of the models an additional dataset is required; the so-called "blind" set must be completely independent from the training set, in order to avoid biased predictions. The set should also be a good representative of the real protein space, in order to verify the generality of the models.

2.2.1 Selection of proteins and internal redundancy reduction

To obtain the set, a search on PDB was performed with the following parameters:

- deposit date after january 2015 (after the release of the Jpred4 dataset in 2014)
- x-ray resolution under 2.5Å
- chain length between 50 and 300 residues
- wild type protein (no engineered mutations)
- retrieve only representatives at 30% sequence identity (to avoid redundancy)

An additional step of internal redundancy reduction was performed with blastclust CITAZIONE: the sequences with a similarity higher than 30% over an area covering more than 30% of their length were clustered together, and a representative of each of the 3113 clusters obtained was selected.

2.2.2 External redundancy reduction

However, there may still be some similarity with elements of the training set: to avoid this redundancy, the cluster representatives were formatted into a BLAST database with makeblastdb CITAZIONE. The training set was then compared to the database with blastp: all sequences with similarity higher than 30% were removed from the list.
From the list, 150 sequences were randomly selected to build the final blind set.

2.2.3 Assigning secondary structure to the sequences

From the PDB files of the blind set, the secondary structure sequence was inferred with DSSP (Define Secondary Structure of Proteins) CITAZIONE. DSSP implements an algorithm which calculates the most probable secondary structure assignments given the geometry of the protein 3D structure. The main idea behind the program is that the different secondary structure types in the atomic coordinates of a protein can be discriminated by identifying the corresponding hydrogen bonding patterns. Instead of using backbone ϕ and ψ angles or Cα positions, which require the adjustment of many parameters, the presence or absence of an H-bond can be characterized by a single parameter, which is the cutoff in the bond energy. Structures obtained with X-ray cristallography do not contain most hydrogen atom coordinates, since the wavelength of X-rays (circa 1Å) is too large to detect them. DSSP discards any present hydrogen atom from the PDB file and places a hydrogen atom at 1Å from each backbone N atom, in the opposite direction from the backbone C=O bond. The electrostatic energy of each potential H-bond between all atoms is then calculated, and if it is lower than 0.5 kcal/mol then the bond is considered existent. DSSP defines eight types of secondary structure according to their H-bonding patterns; for the purpose of this project, they were grouped as following:

- 3,4, and 5-turn helix (G,H,I)=> **Helix (H)**
- residue in isolated β-bridge (B) and extended strand participating in β-ladder (E) => **Strand (E)**
- H-bonded turn (T), bend (S) and no assignment("", empty) => **Coil (C or -)**

Finally, the secondary structure sequences were extracted from the dssp files.

2.3 Obtaining the sequence profiles

As introduced earlier, using evolutionary information when training secondary structure predictors yields better results. Multiple sequence alignments of sequences belonging to proteins of the same family contain much more information than the single sequences, since conserved regions usually indicate structurally or functionally important residues. Sequence profiles are a way to represent in a more compact way multiple sequence alignments: a matrix of dimension L x 20, where L is the length of the target protein sequence and 20 the number of possible residues. Each element of the matrix represents the occurrence frequency of each residue on each aligned position. For the purpose of this project, the models were trained and tested with sequence profiles rather than single sequences; in order to

obtained the profiles, each sequence of both training set and blind testing set was searched by similarity agains a large sequence database. To obtain the database, all the sequences present on UniprotKB/SwissProt CITAZIONE were indexed with the makeblastdb tool, specifying the protein nature of the sequences. PSI-BLAST CITAZIONE was then chosen to search homologous sequences in the database. While BLAST runs on a single iteration, PSI-BLAST performs multiple search iterations using a Position Specific Substitution Matrix (PSSM) computed during the search. On the first iteration, a standard BLAST search is performed using a pre-defined substitution matrix, such as BLOSUM62; a multiple sequence alignment is computed by stacking the pairwise local alignments obtained; the PSSM is then computed from the MSA. In the following iterations, BLAST searches are performed using the PSSM as substitution matrix, and multiple sequence alignments are computed from the sequences found using the PSSM obtained in the first iteration;the aligned sequences are clustered by sequence identity. Finally, after a user-defined number of iterations, or if no more alignments can be found with e-value below the specified treshold, the algorithm stops and the MSA is returned along with the PSSM. To obtain the profiles needed, each sequence underwent three iterations of PSI-BLAST run against the database with an e-value threshold of 0.01 and number of alignments set to 1000, due to the large size of the database.

2.4 The GOR method

2.4.1 Description

The Garnier-Osguthorpe-Robson (GOR) method was originally developed in 1978 CITAZIONE, and implements information theory and bayesian statistics for protein secondary structure prediction. Over the years, it underwent several improvements and refinements, with the addition of larger databases, the detailing of statistics accounting for amino acid pairs and triplets, and most importantly the inclusion of evolutionary information in the form of sequence profiles. CITAZIONE/I The underlying idea is similar to some previous methods, such as Chou-Fasman CITAZIONE?: the observed frequency of residues in a particular secondary structure conformation can help predict the secondary structure starting only from the primary structure. The extent to which the presence of a particular residue R influences the probability of having the conformation S is determined with the following information function:

$$I(S;R) = log\frac{P(S|R)}{P(S)} = logP(S|R) - logP(S) \tag{1}$$

where P(S|R) is the conditional probability of observing the conformation S when the residue is R, and P(S) is the marginal probability of observing the conformation S. Using the probability chain rule, the equation can be rewritten as

$$I(S;R) = log\frac{P(R,S)}{P(S)P(R)} \tag{2}$$

Where P(R,S) is the joint probability of observing residue R in conformation S, and P(R) is the marginal probability of observing the residue R. The novelty of the GOR method lies in the consideration of the influence of neighbouring residues on the conformation of the given residue. This is reflected in the information function, which is extended over a d number of residues preceding and following the central one:

$$I(S;R_{-d},...R_0,...R_d) = log\frac{P(S|R_{-d},...R_0,...R_d)}{P(S)} = \tag{3}$$

$$= log\frac{P(S,R_{-d},...R_0,...R_d)}{P(R_{-d},...R_0,...R_d)P(S)} \tag{4}$$

The value of d is usually 8; this limit is not arbitrary but was based on studies of information content at increasing separations. (CITAzIONE(The GOR method)) However, calculating the joint probabilities of S and all the possible residues in all the possible positions in the window isn't computationally feasible, and would require databases much larger than currently available to obtain reliable probabilities. To avoid this problem, the residues in the window are considered to be statistically independent. The formula then becomes

$$log\frac{P(S)\prod_{k=-d}^{d}P(R_k|S)}{P(R_{-d},...R_0,...R_d)P(S)} = log\prod_{k=-d}^{d}\frac{P(R_k,S)}{P(S)P(R_k)} \tag{5}$$

$$= \sum_{k=-d}^{d} I(S;R_k) \tag{6}$$

Each $I(S;R_k)$ of the summatory can be computed as already seen in equation 2. Each parameter in the mentioned equations is derived from the data used in the training of the model:

- $P(R_k,S)$: frequency of residues of type R observed at position k in windows where the central residue R_0 is in conformation S
- $P(R_k)$: frequency of observed residues at position k
- $P(S)$: frequency of each conformation S

DA SPOSTARE the highest information function of the possible three secondary structures (H,E,C) determines the most likely conformation.

In the more recent iterations of the GOR method, the input is in the form of sequence profiles: the information function is then represented as

$$I(S;PW) = \sum_{k=-d}^{d}\sum_{R_k} PW[R_k] * I(S;R_k) \tag{7}$$

For each element of the window, $I(S;R_k)$ must be calculated for each residue frequency of the corresponding line in the profile. The highest value of the three possible information functions (H,E,C) represents the most likely conformation for the central position of the considered window in the sequence profile. IMPLEMENTATION???

2.5 Support Vector Machines

Support Vector Machines (SVM) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis CIT WIKIPEDIA. The first implementation of SVMs for secondary structure prediction goes back to 2001, when S.Hua and Z.Sun assembled a tertiary classifier for the three conformations H,E, and C from the combination of several binary classifiers CITAZIONE. The method achieved notable results, with a three-state accuracy of 73.5% SPOSTARE IN INTRODUZIONE?? Compared to other machine learning techniques such as neural networks, SVMs have the advantage of being less prone to overfitting and being able to handle large feature spaces; the implementation of the kernel function allows the implicit mapping of the data to a high-dimensional space, which facilitates the separation of the data in classes. In this project, SVMs were used only for classification, and not for regression.

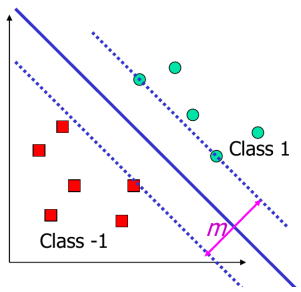

Fig. 1. Hyperplane separating the feature space in two classes

The way SVMs work is by calculating the hyperplane which divides the feature space in two classes, while maximizing the distance between the support vectors (margin, m in fig. 1), which are defined as the closest points for each class to the hyperplane.
Given a training set with n elements, each having a value $x_i \in \mathbb{R}^d$ and belonging to a class $y_i \in \{-1, +1\}$, we define the hyperplane with the equation $w^T x + b = 0$. We can then say that $y_i(w^T x_i + b) \geq \frac{\rho}{2}$, with ρ being the margin. The support vectors are the points where $y_s(w^T x_s + b) = \frac{\rho}{2}$; after rescaling w and b by $\frac{\rho}{2}$ the distance from them to the hyperplane is:

$$r = \frac{y_s(w^T x_s + b}{\|w\|} = \frac{1}{\|w\|} \tag{8}$$

And therefore ρ can be expressed as $2r = \frac{2}{\|w\|}$

The maximization of ρ then becomes the following optimization problem:

$$minimize \quad \frac{1}{2}\left\|w^2\right\|, \ subject\,to \quad y_i(w^T x_i + b) \geq 1 \, \forall i \tag{9}$$

To solve it, a Lagrange multiplier αi is introduced to construct the dual problem:

$$maximize \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \left\langle x_i, x_j \right\rangle \tag{10}$$

$$subject\,to \quad \alpha_i \geq 0 \ \forall i \quad and \ \textstyle\sum_{i=1}^{n} \alpha_i y_i = 0$$

A global maximum of αi can always be found; w can be recovered with $w = \sum_{i=1}^{n} \alpha_i y_i x_i$, even thought it is not explicitly needed for the classification of new points. All x_i with non-zero αi are support vectors. b can be obtained with $b = y_k - \sum_s \alpha_s y_s x_s^T x_k$. The classifying function for new points becomes:

$$y(x) = \sum_s \alpha_s y_s (x_s^T x) + b \tag{11}$$

With x being the new point. This formulation however admits only training sets with all points grouped in a way in which they are linearly separable: in the case of even a single outsider, the model must be able to consider it as an error up to a certain extent. To do so, the so called "soft margin" is used: a slack variable ξi, being the upper bound of allowed errors, is introduced in the optimization problem (equation 9) in the following way:

$$minimize \quad \frac{1}{2}\left\|w^2\right\| + C \sum_{i=1}^{n} \xi_i \tag{12}$$

$$subject\,to \quad y_i(w^T x_i + b) \geq 1 - \xi_i \quad with\, \xi_i \geq 0, \ \forall i$$

C being the hyperparameter that weighs the maximization of the margin, even if it includes errors, against the minimization of errors on the training points.
In some cases the training set is not linearly separable, even with the implementation of soft margins. The original input space however can always be transformed to a new feature space (usually higher dimensional), where linear operations correspond to non-linear operations in the original space; a good transformation can render linearly separable problems which are not linearly separable in their original dimension. As seen in equation 11, the classification relies on the scalar product between support vectors and the unlabeled point, which after a transformation ϕ becomes $K(x_s, x') = \phi(x_s^T \phi(x')$. This is the kernel function, which implicitly remaps the data without explicitly carrying out the transformation of all the input data. The classifying function can be updated by substituting the scalar product with the kernel function. There are several kernel functions that can be used with support vector machines; the one utilized in this project is the gaussian/radial basis function (RBF):

$$K(x_s, x') exp(-\gamma \left\|x_s - x'\right\|^2) \tag{13}$$

With $\gamma = \frac{1}{2\sigma^2}$; the input space is transformed to an infinite-dimensional feature space, where every point is mapped to a gaussian function. The combination of the functions for the support vectors determines the separator.

IMPLEMENTATION???

3 Discussion

4 Conclusion

Acknowledgements

References