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

Katarina Elez [1,*]

[1] International Master in Bioinformatics, University of Bologna, Bologna, 40126, Italy

[*] To whom correspondence should be addressed.

Submitted for revision on November 15, 2018; Final version submitted on November 20, 2018

Motivation: Next generation sequencing technologies are continually increasing the sequence-structure gap. Methods for predicting protein tertiary structure often rely on secondary structure prediction, which is an important open problem in the field of bioinformatics. The main objective of this study was to compare a method based on information theory and statistics with an SVM-based method in order to access the importance of using machine learning algorithms.
Results: Evolutionary information provided by multiple sequence alignment proved very important for an accurate secondary structure prediction. The GOR model reaches a three-class accuracy of 62% on the blind test set compared to 75% of the SVM-based model on the same dataset. The results showcase the significant gain in performance when using machine learning for predicting protein secondary structure. The final model, however, performs worse than current state-of-the-art methods. Possible improvements are discussed.
Availability: The method is available on https://github.com/evgenije/protein-ss-pred.
Contact: katarina.elez1@gmail.com
Supplementary information: Supplementary data is included as a separate file.

1 Introduction

High throughput sequencing techniques have produced huge amounts of data in the recent years and as a direct consequence of this enormous inflow, the number of protein sequences in the UniProt database has exceeded 127 million, as of October 2018 (The UniProt Consortium, 2017). Only around 45 thousand of those sequences have at least one three-dimensional structure in the Protein Data Bank (PDB) while for the remaining vast majority it has to be predicted. The most reliable method for protein tertiary structure prediction is comparative modeling (e.g. Sali and Blundell (1993)) but it requires a template structure with a high degree of sequence similarity to the target protein (>35%). When a suitable template is not available, fold recognition and ab initio methods are used. Both of these approaches can benefit greatly from constraints provided by the secondary structure.

The earliest methods developed for secondary structure prediction were based on single amino acid propensities (Chou and Fasman, 1974) or propensities of amino acids in a window around the central residue (Garnier *et al.*, 1978). The latter method, known as GOR, reaches an accuracy of ∼60% and has been particularly popular due to its simplicity. Soon it became clear that evolutionary information contained in multiple sequence alignments can be exploited to improve the prediction. Nevertheless, it was not before larger databases and machine learning algorithms were used that the accuracy surpassed 70% (Rost and Sander, 1994). Some of the most popular predictors today are based on neural networks and include: PSIPRED 3.2 (81.6% accuracy, Jones (1999)), JPred 4 (82% accuracy, Drozdetskiy *et al.* (2015)) and SSpro 5.2 (92.9$ accuracy, Magnan and Baldi (2014)).

The aim of this study was to develop a method for protein secondary structure prediction. Specifically, a comparison was made between GOR and support vector machines (SVMs) in order to better understand the benefits of the machine learning approach. The SVM model performed significantly better, reaching an accuracy of 75% on the blind test set.

2 Datasets

2.1 Training set

Training set was obtained from a previous work by Drozdetskiy *et al.* (2015). The starting set from which it was built contained 1987 single representative sequences from each superfamily in SCOPe v2.04 (Fox *et al.*, 2014). Subsequently, sequences having the following characteristics were removed: belonging to proteins whose structures have a resolution of >2.5Å (1681 retained), having <30 or >800 residues (1654 retained), containing partial domains (1634 retained), having missing DSSP information for >9 consecutive residues (1524 retained), belonging to proteins with inconsistencies between PDB, DSSP and ASTRAL file definitions (1507 retained). The set was split into a training set with 1357 sequences and a blind test set with 150 sequences. PSI-BLAST analysis of the training set failed to produce hits for 9 sequences, which were then removed, leaving the final training set of 1348 sequences. FASTA and DSSP files for all sequences were downloaded from http://www.compbio.dundee.ac.uk/jpred4/downloads/retr231.tar.gz. Sequence profiles were successfully generated for 1248 of those sequences,

1200 of which had non-null profiles and were effectively used for training the predictor.

2.2 Blind test set

An independent dataset was constructed in order to objectively evaluate the performance of the predictors. Sequences were selected from the PDB (rcsb.org, Berman *et al.* (2000)) in September 2018, according to the following criteria:

- Experimental Method - X-RAY
- X-ray Resolution - 2.0 or less
- Deposit Date - 1 January 2015 or after
- Chain Length - between 50 and 300
- Macromolecular Type - Protein but not DNA or RNA or Hybrid
- Wild Type Protein (Include Expression Tags - Yes, Percent coverage of UniProt sequence - >=70%).

Only representatives at 30% sequence identity were retrieved resulting in 985 sequences. They were downloaded as a custom tabular report which caused other chains from the chosen structures to be included in the dataset. The undesired chains were removed by reducing the internal redundancy to 30%, once again. The external redundancy with respect to the training set was also reduced to 30% which retained 804 sequences. The most probable secondary structure for each residue of the blind test set was assigned using DSSP 2.2.1 (Kabsch and Sander, 1983). The original output was parsed and sequences of residues and secondary structures were extracted into FASTA and DSSP files, respectively. The eight DSSP classes were reduced to three using the following mapping: [HGI] -> H, [EB] -> E, [ST' '] -> C. Additional filters were applied: sequences belonging to proteins having breaks in their DSSP files were removed (547 retained), as well as sequences with missing non-terminal residues in their PDB files (380 retained). Finally, 343 sequence profiles were generated and 328 non-null profiles were used in the testing procedure.

2.3 Statistical analysis of datasets

The two datasets used in this study (training set and blind test set) were statistically analyzed and compared. For the training set coil was the most common conformation (42.4%), followed by helix (35.5%) and strand (22.1%) (see Supplementary Fig. S1). In the blind test set helix was slightly more abundant (38.86%) than coil (37.02%) and strand was, once again, the least common conformation (24.12%) (see Supplementary Fig. S2).

Propensity of the amino acids to be in a certain conformation showed exactly the same trend for both datasets and is in accordance with previous observations (Koehl and Levitt, 1999). Specifically: methionine, alanine, leucine, glutamate, lysine, glutamine and arginine were found to prefer helices, while large aromatic residues (tryptophan, tyrosine and phenylalanine) and Cβ-branched amino acids (isoleucine, valine, and threonine) tended to adopt strand conformations (see Supplementary Fig. S3 and S4). Propensities in a 17-residue window with its central residue in a helix/strand conformation were also highly similar (see Supplementary Fig. S5 and S6).

The majority of sequences in both datasets come from Bacteria, followed by Eukaryota, Archea and Viruses (see Supplementary Fig. S7 and S8). The difference between the first two most abundant kingdoms is much more prominent in the training set (48.33% vs 39.57%) than in the blind test set (62.85% vs 26.01%). Grouped by species, in both training and blind test sets (see Supplementary Fig. S9 and S10) the majority of sequences belong to Homo sapiens (19.36% and 8.72%, respectively), followed by Escherichia coli (11.27% and 6.54%, respectively).

3 Methods

3.1 Redundancy reduction

Internal redundancy of the blind test set was reduced using BLASTCLUST 2.2.26 (Altschul *et al.*, 1990), with the sequence identity threshold set to 30%. The first member of each cluster was retained. In order to reduce external redundancy of the blind test set with respect to the training set BLASTP version 2.7.1+ from the BLAST package was used, with an E-value threshold of 0.01 (Altschul *et al.*, 1990). All sequences from the blind test set having at least 30% sequence identity with respect to at least one sequence from the training set were removed.

3.2 Sequence profile generation

Structure is significantly better conserved than sequence. Specifically, proteins with more than 35% sequence identity over more than 100 aligned residues have similar structures (Rost, 1999). This means that a multiple sequence alignment (MSA) of homologous proteins contains much more information about structure than single sequences alone. Specifically, a sequence profile derived from an MSA defines, for each position, which residues can be substituted by which others, reflecting important constraints posed by evolution. Secondary structure prediction methods benefit greatly from this evolutionary information and it is, indeed, at the basis of all more recently developed predictors.

PSSM files were generated using PSI-BLAST 2.7.1+ from the BLAST package for both training and blind test sets (Altschul *et al.*, 1990). For convenience, the search was performed against the SwissProt database, with the E-value threshold set to 0.01 and the maximum number of iterations set to 3. Sequence profiles were extracted from the PSSM files and the frequencies were normalized in the range 0-1 by simply dividing all values by 100.

3.3 GOR method

Garnier-Osguthorpe-Robson (GOR) is a second-generation method for protein secondary structure prediction. It assumes that the conformation of a given residue depends on amino acid propensities of that residue and of those in its sequence context. Specifically, in order to predict the secondary structure, it considers an n-residue window centered on a given residue, therefore analyzing its (n-1)/2 nearest neighbors on each side.

GOR is based on information theory and Bayesian statistics. The information function, in the case of secondary structure, can be defined as:

$$I(S;R) = log\left[\frac{P(S|R)}{P(S)}\right] = log\left[\frac{P(R,S)}{P(S)*P(R)}\right] \quad (1)$$

where P(S|R) is the conditional probability of observing conformation S given residue R, P(S) is the probability of observing conformation S, P(R,S) is the joint probability of observing residue R and conformation S and P(R) is the probability of observing residue R. S can be any of the three possible conformations (helix, strand or coil) while R can be any of the twenty possible amino acids.

If an n-residue window is considered, (1) becomes:

$$I(S;W) = log\left[\frac{P(S|W)}{P(S)}\right] = log\left[\frac{P(W,S)}{P(S)*P(W)}\right] \quad (2)$$

with $W = R_1, R_2, ..., R_n$, where P(S|W) is the conditional probability of observing a central residue in conformation S given window W, P(W,S) is the joint probability of observing window W with its central residue in conformation S, P(W) is the probability of observing window W and R_j is the residue in the j-th position of the window.

The method makes a simplifying assumption that the residues $R_1 ... R_n$ in a window are statistically independent, therefore:

$$P(W) = \prod_{j=1}^{n} P(R_j) \tag{3}$$

and (2) becomes:

$$\begin{aligned} I(S;W) &= log\left[\frac{P(W,S)}{P(S)*P(W)}\right] = log\left[\frac{P(S)*P(W|S)}{P(S)*P(W)}\right] \\ &= log\left[\frac{P(S)*\prod_{j=1}^{n} P(R_j|S)}{P(S)*\prod_{j=1}^{n} P(R_j)}\right] \\ &= log\left[\frac{\prod_{j=1}^{n} P(S)*P(R_j|S)}{\prod_{j=1}^{n} P(S)*P(R_j)}\right] \\ &= log \prod_{j=1}^{n} \frac{P(R_j,S)}{P(S)*P(R_j)} = \sum_{j=1}^{n} I(S;R_j) \end{aligned} \tag{4}$$

where $P(R_j)$ is the probability of observing residue R in the j-th position of the window, P(W|S) is the conditional probability of observing window W given that its central residue is in conformation S, $P(R_j|S)$ is the conditional probability of observing residue R in the j-th position of the window given that its central residue is in conformation S and $P(R_j,S)$ is the joint probability of observing residue R in the j-th position of the window with its central residue in conformation S.

Given a window W, the conformation S* predicted by GOR for its central residue is the one having the highest value of the information function:

$$S^* = argmax_S I(S;W) = argmax_S \sum_{j=1}^{n} I(S;R_j). \tag{5}$$

Since P(W) in (4) is the same for all three I(S;W), the conformation having the highest value of the information function is the one having the highest sum of $P(R_j,S)/P(S)$ for j=1...n. Each element can be calculated as:

$$\frac{P(R_j,S)}{P(S)} = \frac{\frac{\#R_j,S}{N}}{\frac{\#S}{N}} = \frac{\#R_j,S}{\#S} = P(R_j|S) \tag{6}$$

where $\#R_j,S$ is the number of times residue R is observed in the j-th position of the window with its central residue in conformation S, #S is the number of times conformation S is observed for the central residue of the window and N is the total number of residues.

The original method was slightly modified in this study. The GOR model was trained on sequence profiles and not on individual sequences in order to take advantage of the evolutionary information. During the training procedure $\#R_j,S$ was calculated for j=1...17 by increasing the individual counts by the residue frequencies found in the sequence profile. The counts were then divided by the corresponding #S corrected for the missing window positions. Therefore, from the profiles of the training examples three different propensity matrices (for each of the three possible conformations) were obtained. The matrices contained 20x17 elements and each element indicated the propensity of one of the twenty possible amino acids to be in the j-th position of the window given that its central residue is in conformation S. A total of 1020 parameters were estimated.

In the prediction phase, if a sequence is provided it is first transformed into a profile by setting the frequency of all residues in the i-th line of the profile to 0 except for the frequency of the residue in the i-th position of the sequence which is set to 1. When the sequence has been transformed or a profile is directly provided, the algorithm predicts the secondary structure by calculating:

$$S^* = argmax_S \sum_{j=1}^{n} \sum_{R} PW[R_j] * I(S;R_j) \tag{7}$$

where PW is a window of the sequence profile, $PW[R_j]$ is the frequency of residue R in the j-th position of the window and $I(S;R_j)$ is the corresponding element of the propensity matrix.

3.4 Support Vector Machines (SVMs)

Support Vector Machine (SVM) is a powerful and broadly used machine learning algorithm. Its objective is to maximize the margin which is the distance between the decision boundary and the training examples that are closest to this boundary, the so-called support vectors. A large margin is preferable because it tends to have a lower generalization error (it is less prone to overftting) (Raschka and Mirjalili, 2015).

The decision boundary is a separating hyperplane defined as:

$$\mathbf{w}^T\mathbf{x_i} + b = 0 \tag{8}$$

and for every example the following holds:

$$\mathbf{w}^T\mathbf{x_i} + b \geq 1 - \xi_i \text{ if } y_i = 1 \tag{9}$$

$$\mathbf{w}^T\mathbf{x_i} + b \leq -1 + \xi_i \text{ if } y_i = -1 \tag{10}$$

or equivalently:

$$y_i(\mathbf{w}^T\mathbf{x_i} + b) \geq 1 - \xi_i \text{ with } \xi_i \geq 0, \forall i \tag{11}$$

where $\mathbf{w}$ is a weight vector, $\mathbf{x_i}$ is an example vector, $\mathbf{w}^T\mathbf{x_i}$ is the scalar product between them, b is a bias term, ξ is a slack variable and y_i is the class of the example vector $\mathbf{x_i}$.

Maximizing the margin is the same as minimizing:

$$\frac{1}{2}||\mathbf{w}||^2 + C\Big(\sum_i \xi_i\Big) \tag{12}$$

where $||\mathbf{w}||$ is the length of vector $\mathbf{w}$ and C is a penalty parameter, under the constraint that the examples are classified correctly ((11) holds for i=1...n).

The problem can be solved by maximizing the dual Lagrangian:

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j \mathbf{x_i}^T\mathbf{x_j} \tag{13}$$

subject to $0 \leq \alpha_i \leq C$ and $\sum_{i=1}^{n} \alpha_i y_i = 0$ where α_i is a Lagrange multiplier associated with the i-th constraint. Given a solution, the support vectors are vectors $\mathbf{x_i}$ with non-zero α_i and the value of the classification function for a new example can be calculated as:

$$f(x) = \mathbf{w}^T\mathbf{x} + b = \sum_s \alpha_s y_s \mathbf{x_s}^T\mathbf{x} + b \tag{14}$$

where $\mathbf{x_s}$ is a support vector.

Non-linearly separable problems can always be mapped into a higher-dimensional space where they become linearly separable. This is done be substituting $\mathbf{x_i}^T\mathbf{x_j}$ in (13) with a kernel function $K(\mathbf{x_i},\mathbf{x_j})$.

The SVM model was implemented in Python using scikit-learn library (Pedregosa *et al.*, 2011). Each window of the dataset was linearized into a vector of 340 components and used as an example together with the secondary structure of its central residue. A radial basis function (RBF) kernel was used. The penalty parameter C of the error term (which controls the trade-off between training classification accuracy and margin size) and the RBF kernel coefficient γ (which defines how far the influence of a single training example reaches) were both optimized using a grid-search procedure. The optimal C and γ were selected from the ranges $\{10^0, 10^1\}$ and $\{2^{-1}, 2^{-2}, 2^{-3}\}$, respectively.

3.5 Scoring indexes

Performance of the predictor was evaluated at the residue level by calculating sensitivity (SEN), positive predictive value (PPV) and Matthews correlation coefficient (MCC), for each of the three possible classes, as defined:

$$SEN = \frac{TP}{TP + FN} \tag{15}$$

$$PPV = \frac{TP}{TP + FP} \tag{16}$$

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{17}$$

where TP, TN, FP and FN are true positives, true negatives, false positives and false negatives, respectively, obtained by reducing the 3-class confusion matrix to a binary confusion matrix.

Predictor's three-class accuracy (Q_3) was also measured at the residue level in the following way:

$$Q_3 = \frac{TP_1 + TP_2 + TP_3}{N} \tag{18}$$

where TP_1, TP_2 and TP_3 are true positives for each of the three possible conformations, respectively.

In order to discriminate between similar and dissimilar segment distributions, segment overlap (SOV) was determined at the sequence-level, for each of the three classes, as described in Zemla *et al.* (1999):

$$SOV(S) = 100 * \frac{1}{N_S} * \sum_{\{(s_o,s_p)|s_o \cap s_p \neq \emptyset\}} \left[\frac{minov(s_o, s_p) + \delta(s_o, s_p)}{maxov(s_o, s_p)} * len(s_o) \right] \tag{19}$$

with

$$N_S = \sum_{\{s_o|s_o \cap s_p \neq \emptyset\}} len(s_o) + \sum_{\{s_o|s_o \cap s_p = \emptyset\}} len(s_o) \tag{20}$$

where len(s_o) is the number of residues in segment s_o, minov(s_o,s_p) is the length of the actual overlap of s_o and s_p, maxov(s_o,s_p) is the total extent for which either of the segments s_o, and s_p has a residue in state S, while

$$\delta(s_o, s_p) = min \begin{cases} maxov(s_o, s_p) - minov(s_o, s_p) \\ minov(s_o, s_p) \\ int(len(s_o)/2) \\ int(len(s_p)/2) \end{cases} \tag{21}$$

where len(s_p) is the number of residues in segment s_p.

3.6 Evaluation procedure

The original training set (1348 sequences) was randomly split into five equally-sized subsets. The split was performed at the level of sequences so all subsets contained (approximately) the same number of sequences but the total number of residues differed between subsets. After profile generation the number of non-null profiles in the five subsets was: 244, 243, 233, 242 and 238, respectively. For each run of the cross-validation procedure the predictor was trained on a different combination of four subsets and scoring indexes were calculated during testing on the remaining subset. Mean and standard deviation over the five cross-validation runs were obtained for each index.

The predictor was then trained on the entire training set and evaluated on the blind test set in order to obtain a more reliable estimate of the performance.

Table 1. GOR cross-validation and blind test scoring indexes

SEN_H	0.86±0.01	0.83
SEN_E	0.62±0.01	0.60
SEN_C	0.42±0.01	0.42
PPV_H	0.58±0.01	0.60
PPV_E	0.54±0.02	0.58
PPV_C	0.80±0.01	0.73
MCC_H	0.50±0.01	0.46
MCC_E	0.45±0.01	0.46
MCC_C	0.40±0.01	0.39
Q_3	0.62±0.01	0.62
SOV_H	65.51±1.67	62.70
SOV_E	58.82±1.98	63.18
SOV_C	43.05±0.95	45.57

SEN is sensitivity, PPV is positive predictive value, MCC is Matthews correlation coefficient, Q_3 is three-class accuracy and SOV is segment overlap. The first values are averages across five folds with their standard deviations while the second values are those obtained for testing on the blind test set.

3.7 Comparison with other predictors

Predictions for all the sequences of the blind test set were obtained through the REST API of the JPred 4 server (Drozdetskiy *et al.*, 2015) and through the command line version of SSpro 5.2 (Magnan and Baldi, 2014). Scoring indexes were calculated for both of these predictors.

4 Results and Discussion

4.1 GOR performance

Scoring indexes from cross-validation and blind testing for the GOR method are reported in Table 1. The highest sensitivity of prediction is observed for helices, while the highest positive predictive value can be detected for coil. Matthews correlation coefficient is rather balanced for all three classes and in the range of 40-50%. With a three-class accuracy of 62% on both datasets the method performs as expected (Q_3 of 60-65%). Finally, segment overlap is much higher for helix and strand with respect to coil.

All indexes show low standard deviation for cross-validation and comparable results on the blind test set which means that the model is stable. A somewhat larger difference in the positive predictive value for coil can be explained by the larger abundance of coil in the training set (42.4%) with respect to that of the blind test set (37.02%).

4.2 SVM performance

Table 2 reports scoring indexes from cross-validation and blind testing for the SVM method with different sets of parameters. The sensitivity of prediction for helix and strand increases with the increase of C and with the decrease of γ. For coil, the opposite is true: it increases with the decrease of C and with the increase of γ, but is always the highest of the three sensitivities. Positive predictive value increases with the increase of C for all conformations, while it is proportional to γ in case of helix and strand but inversely proportional to γ in case of coil. The highest values of PPV and MCC can be observed for helix. Matthews correlation coefficient increases with the increase of C and the decrease of γ for all secondary structures. The same happens with Q_3 and SOV. The three-class

Table 2. SVM cross-validation and blind test scoring indexes

$C=10^{-1}$	$\gamma=2^{-1}$		$\gamma=2^{-2}$		$\gamma=2^{-3}$	
SEN_H	0.42±0.03	0.50	0.70±0.03	0.66	0.77±0.02	0.69
SEN_E	0.18±0.02	0.28	0.38±0.02	0.46	0.49±0.02	0.54
SEN_C	**0.93±0.00**	**0.92**	0.86±0.01	0.86	0.82±0.00	0.84
PPV_H	0.84±0.01	0.86	0.80±0.02	0.81	0.77±0.02	0.79
PPV_E	0.79±0.03	0.82	0.76±0.02	0.80	0.73±0.02	0.77
PPV_C	0.51±0.01	0.49	0.63±0.01	0.59	0.69±0.01	0.63
MCC_H	0.47±0.02	0.52	0.62±0.02	0.59	0.64±0.01	0.60
MCC_E	0.32±0.02	0.40	0.46±0.02	0.52	0.51±0.01	0.56
MCC_C	0.33±0.02	0.38	0.49±0.01	0.49	0.54±0.01	0.52
Q_3	0.59±0.01	0.60	0.70±0.01	0.69	0.73±0.01	0.71
SOV_H	39.63±2.33	46.15	67.62±2.24	64.59	74.01±0.93	67.55
SOV_E	18.22±1.71	33.52	40.68±3.06	52.33	52.53±3.1	61.64
SOV_C	39.00±1.55	45.08	62.03±0.87	64.03	67.98±0.9	69.09
$C=10^{0}$	$\gamma=2^{-1}$		$\gamma=2^{-2}$		$\gamma=2^{-3}$	
SEN_H	0.68±0.02	0.67	**0.80±0.01**	**0.73**	**0.80±0.01**	0.72
SEN_E	0.39±0.03	0.49	0.54±0.02	0.60	**0.59±0.01**	**0.62**
SEN_C	0.88±0.01	0.89	0.83±0.00	0.85	0.82±0.00	0.85
PPV_H	**0.85±0.01**	**0.87**	0.82±0.02	0.85	0.82±0.01	0.85
PPV_E	**0.79±0.02**	**0.83**	0.77±0.02	0.82	0.75±0.02	0.80
PPV_C	0.62±0.01	0.58	0.71±0.01	**0.65**	**0.72±0.01**	**0.65**
MCC_H	0.65±0.02	0.65	**0.71±0.01**	**0.67**	**0.71±0.01**	**0.67**
MCC_E	0.48±0.02	0.56	0.56±0.01	**0.63**	**0.58±0.01**	**0.63**
MCC_C	0.49±0.01	0.50	0.57±0.01	**0.56**	**0.58±0.01**	**0.56**
Q_3	0.70±0.01	0.70	**0.76±0.01**	**0.75**	**0.76±0.01**	**0.75**
SOV_H	63.72±1.64	63.09	74.97±1.13	**69.16**	**76.08±1.25**	68.64
SOV_E	39.48±3.56	52.42	54.64±2.68	64.18	**59.21±2.78**	**67.18**
SOV_C	58.99±1.51	61.97	69.56±0.88	**71.00**	**70.12±1.06**	70.93

SEN is sensitivity, PPV is positive predictive value, MCC is Matthews correlation coefficient, Q_3 is three-class accuracy and SOV is segment overlap. For each set of parameters, the first values are averages across five folds with their standard deviations while the second values are those obtained for testing on the blind test set. The highest value for each index is highlighted in bold.

accuracy reaches 76% for cross-validation (75% for blind testing), while the segment overlap is the highest for helix, followed by coil and strand.

The standard deviation for cross-validation is low for all indexes and the results are comparable with those obtained on the blind test set. The biggest difference between the two is in the indexes for strand which are noticeably higher on the former, especially SOV.

Considering all the metrics, the model obtained with parameters $C=10^0$ and $\gamma=2^{-3}$ has been chosen as the final one.

4.3 Comparison between GOR, SVM, JPred and SSpro

Scoring indexes of GOR and of the best SVM model ($C=10^0$ and $\gamma=2^{-3}$) on the blind test set are, once again, reported in Table 3, together with those obtained for testing with single sequences and not with sequence profiles. Performance of JPred and SSpro is provided for comparison.

Sensitivity of prediction for helices and positive predictive value for coil is high for the GOR model, higher than that of both SVM and JPred. It is also noticeable that the model significantly underpredicts coil (low SEN_C, MCC_C and SOV_C, but high PPV_C). This can be explained by the fact that these regions are less regular and appear more variable in the MSA so the propensities are less informative.

The best SVM model with respect to GOR is characterized by much higher values for almost all indexes, especially the sensitivity of prediction for coil (>40% better). In the case of SVM, however, the coil might be slightly overpredicted (high SEN_C and low PPV_C). The method is much more sensitive in predicting helices (72%) and coil (86%) with

Table 3. GOR, SVM, JPred and SSpro blind test scoring indexes

	GOR	SVM	JPred	SSpro
SEN_H	0.83 (0.72)	0.72 (0.68)	0.79	0.90
SEN_E	0.60 (0.56)	0.62 (0.41)	0.72	0.86
SEN_C	0.42 (0.42)	0.85 (0.77)	0.85	0.89
PPV_H	0.60 (0.57)	0.85 (0.70)	0.90	0.93
PPV_E	0.58 (0.48)	0.80 (0.68)	0.85	0.91
PPV_C	0.73 (0.68)	0.65 (0.59)	0.69	0.83
MCC_H	0.46 (0.36)	0.67 (0.50)	0.75	0.86
MCC_E	0.46 (0.35)	0.63 (0.42)	0.72	0.85
MCC_C	0.39 (0.35)	0.56 (0.44)	0.61	0.77
Q_3	0.62 (0.57)	0.75 (0.65)	0.80	0.88
SOV_H	62.70 (56.24)	68.64 (57.50)	78.81	91.63
SOV_E	63.18 (56.65)	67.18 (43.92)	78.04	89.83
SOV_C	45.57 (44.59)	70.93 (61.90)	78.28	89.74

SEN is sensitivity, PPV is positive predictive value, MCC is Matthews correlation coefficient, Q_3 is three-class accuracy and SOV is segment overlap. Values inside the parenthesis are those obtained for testing with single sequences.

respect to strands (62%). This is a commonly encountered problem and the explanation might be in the fact that hydrogen bonds of helices are between residues in close proximity to each other (3-5 residues apart depending on the type of helix) while strands are more dependent on long-range interactions. In addition, strand segments are usually shorter and are therefore harder to predict accurately probably because their signal is not strong enough. A possible solution would be to search for an appropriate complementary strand and then refine the original prediction or to consider the structure of homologous proteins.

JPred is, in turn, consistently better than the SVM model, particularly in predicting strands (SEN_E, MCC_E and SOV_E are all much higher). It suffers, though, from the same drawback of overpredicting coil as the SVM model. Finally, SSpro outperforms all the other predictors with an impressive three-class accuracy of 88% and indexes that are highlly balanced between classes.

The use of sequence profiles gives a significant advantage to the methods developed in this study. When tested on single sequences the GOR model is only 57% accurate (5% loss) while the SVM's Q_3 drops down to 65% (10% loss) which shows that it gains much more by using evolutionary information than GOR.

Another obvious advantage is given by the adoption of the machine learning approach. This is due to the fact that methods based on machine learning discover hidden patterns in the data which are not detectable by statistical methods. The latter are straightforward to understand but have to be defined explicitly, while the former do not require any prior assumption about the relationships between variables. For high dimensional datasets the statistical approach is no longer feasible because there are too many factors to take into consideration and that is where machine learning algorithms, such as SVM, come into play.

Distributions of segment lengths for each of the three conformations and for each of the four predictors are visible in Figure 1. As a comparison, distributions from the DSSP assignments are also shown (black line). For helices, GOR and SVM overpredict short segments, while JPred underpredicts them. The distribution of helical segment lengths predicted by SSpro is the closest one to the observed. In the case of strands, JPred underpredicts short segments (1-3 residues long), while GOR overpredicts segments of medium length (3-6 residues long). Other predictors show distributions similar to the observed with a slight underprediction of longer segments. The opposite is true for coil, for which all predictors except GOR overpredict longer segments. On the other side, the underprediction

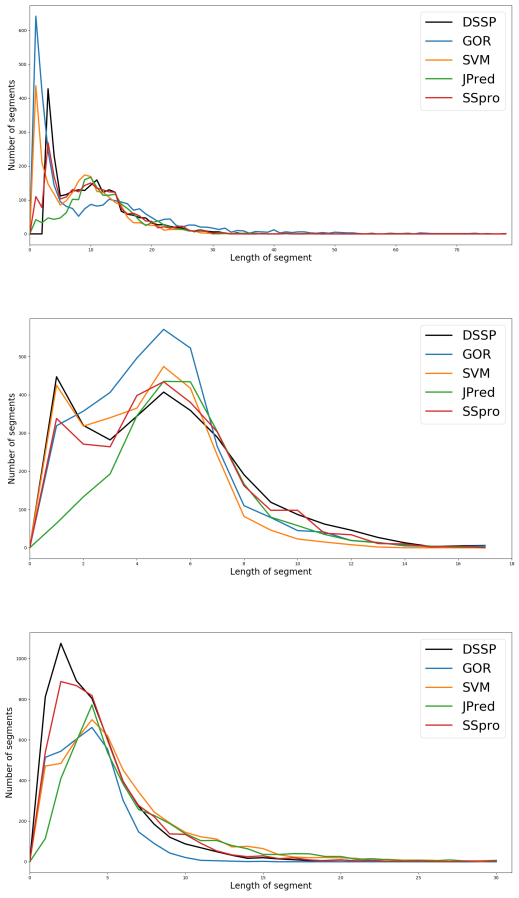

Fig. 1. Distributions of lengths of predicted segments for helix, strand and coil conformations, respectively.

of shorter segments is very common, with SSpro, once again, being the most similar to the DSSP.

5 Conclusion

Evolutionary information is very important for protein secondary structure prediction. The GOR model tested on sequence profiles shows a three-class accuracy of 62%. Despite it performing much worse than other models, its reasoning for giving a particular prediction is clearly evident which is its main advantage with respect to machine learning methods that are black boxes in terms of principles governing their predictions. An example of such a method is the SVM model developed in this study. With a Q_3 index of 75% it shows that machine learning can go well beyond statistical methods and identify hidden patterns in the data. It is, however, outperformed by more sophisticated methods such as JPred and SSpro.

The most immediate improvement for both methods developed here would be to run the PSI-BLAST search against the UniRef50 database instead of the SwissProt database, in order to obtain more divergent profiles. The SVM model might be further improved by including additional features, especially those based on structural similarity to homologous sequences which seems to work very well for SSpro. Finally, a reliability index can be attributed to each residue which would allow the user to filter unreliable predictions.

References

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic local alignment search tool. *Journal of Molecular Biology*, **215**(3), 403–410.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The Protein Data Bank. *Nucleic Acids Research*, **28**(1), 235–242.

Chou, P. Y. and Fasman, G. D. (1974). Conformational parameters for amino acids in helical, Î²-sheet, and random coil regions calculated from proteins. *Biochemistry*, **13**(2), 211–222.

Drozdetskiy, A., Cole, C., Procter, J., and Barton, G. J. (2015). JPred4: a protein secondary structure prediction server. *Nucleic Acids Research*, **43**(W1), W389–W394.

Fox, N. K., Brenner, S. E., and Chandonia, J.-M. (2014). SCOPe: Structural Classification of Proteins–extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic Acids Research*, **42**(D1), D304–D309.

Garnier, J., Osguthorpe, D., and Robson, B. (1978). Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *Journal of Molecular Biology*, **120**(1), 97–120.

Jones, D. T. (1999). Protein secondary structure prediction based on position-specific scoring matrices. *Journal of Molecular Biology*, **292**(2), 195–202.

Kabsch, W. and Sander, C. (1983). Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**(12), 2577–2637.

Koehl, P. and Levitt, M. (1999). Structure-based conformational preferences of amino acids. *Proceedings of the National Academy of Sciences of the United States of America*, **96**(22), 12524–12529.

Magnan, C. N. and Baldi, P. (2014). SSpro/ACCpro 5: almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics*, **30**(18), 2592–2597.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

Raschka, S. and Mirjalili, V. (2015). *Python Machine Learning*. Packt Publishing, 2nd edition.

Rost, B. (1999). Twilight zone of protein sequence alignments. *Protein Engineering, Design and Selection*, **12**(2), 85–94.

Rost, B. and Sander, C. (1994). Combining evolutionary information and neural networks to predict protein secondary structure. *Proteins: Structure, Function, and Bioinformatics*, **19**(1), 55–72.

Sali, A. and Blundell, T. L. (1993). Comparative Protein Modelling by Satisfaction of Spatial Restraints. *Journal of Molecular Biology*, **234**(3), 779 – 815.

The UniProt Consortium (2017). Uniprot: the universal protein knowledgebase. *Nucleic Acids Research*, **45**(D1), D158–D169.

Zemla, A., Venclovas, Č., Fidelis, K., and Rost, B. (1999). A modified definition of Sov, a segment-based measure for protein secondary structure prediction assessment. *Proteins: Structure, Function, and Bioinformatics*, **34**(2), 220–223.