Build of a model for the detection of signal peptides in proteins

Paolo Mastrogiovanni[1]

[1] Master in Bioinformatics, University of Bologna, Bologna
E-mail: paolo.mastrogiovann2@studio.unibo.it

Abstract

Motivation: Developing a model to predict the presence of signal peptides in proteins might provide new insights about the function and interactions of proteins with low experimental data, and it may also reveal new potential drug targets. Here, two models with the same goal were built: one based on a position specific weight matrix, and the other on a support vector machine.

Results: The SVM model proved to be more efficient in the classification than the PSWM, as indicated by the performance metrics (i.e. mcc equal to 0.62 for the SVM and to 0.58 for the von Heijne method). The models were trained by taking in consideration the length of the signal peptides and the aminoacidic distribution as features. This proved to be insufficient in perfectly representing the motif, since the comparison between the false and correct predictions showed that these features can vary considerably.

Supplementary materials: https://github.com/pmastrogiovanni/signal_peptide_detection

Keywords: Signal Peptides, Support Vector Machines, Von Heijne

1.Introduction

Proteins with various functions are constantly being made within cells. These nascent proteins must be transported either out of the cell, or to the different organelles within the cell. To direct the transportation, newly synthesized proteins have an intrinsic signal peptide, functioning as "address tag"(Chou 2001). During or after translocation, a signal peptidase removes the SP cutting on the cleavage site(Almagro Armenteros et al. 2019). Three structurally and, possibly, functionally distinct regions have been identified as the basic building-blocks of a secretory signal sequence: a basic N-terminal region (n-region), a central hydrophobic region (h-region), and a more polar C-terminal region (c-region). The structural determinants for cleavage of the signal sequence seems to reside in the n- and h-regions, with positions -3 and -1 relative to the cleavage site (von Heijne 1986). But even if they share common features, the exact sequences are various from protein to protein. Due to the huge number of protein sequences in the database, and the number being still rapidly increased, it is highly desirable to develop a fast and accurate algorithm to identify the signal sequences and predict their cleavage sites (Cai, Lin, and Chou 2003). Cellular component, together with Biological Process and Molecular function, is one of the three aspects describing protein function in the Gene Ontology (GO)(Carbon et al. 2021; Ashburner et al. 2000). Hence, knowledge of protein localization allows the identification of potential protein interactors or surface exposed targets for drug discovery. For this purpose, many algorithms have been developed, with SignalP being the first publicly available method, now at the sixth release. The latest version is capable of detecting all types of signal peptides, by exploiting a protein language model to represent the motif(Teufel et al. 2022). Here, the same analysis was performed using two different machine learning algorithms, trained on the SignalP-5.0 dataset. The first method is based on the computation of a position specific weight matrix, as in (von Heijne, 1986), while the other is a support vector machine, as in (Cai, Lin, and Chou 2003).

2. Methods

2.1 Dataset

Training and Benchmark datasets were derived from the SignalP-5.0 dataset relying on the Uniprot Knowledgebase release 2018_04. It contains only reviewed entries (from SwissProt) for protein sequences longer than 30 residues. Only signal peptides with experimental evidence (ECO: 0000269) for the cleavage site were taken into consideration. All sequences were shortened to the first 50 N-terminal residues, based on the possible location of the signal peptide(Almagro Armenteros et al. 2019).

The training set is composed of 1723 eukaryotic sequences (258 positives, 1465 negatives), derived by random selection of a subset of the original SignalP-5.0 training dataset. For the cross-validation, the training set was randomly split into 5 equally- sized different subset of 345 sequences. The split was performed according to pairwise sequence similarity, to ensure no similarity between training and testing sequences. The Benchmark set is composed of 7456 sequences (209 positives, 7247 negatives), and it's the same used for benchmarking in the SignalP-5.0 dataset.

Residues in position [-13,+2] from the cleavage site where extracted for further analysis of the signal peptides. Signal peptides lengths have approximately the same distribution in the two sets, with mean/median ranging between 20 and 25 residues, as highlighted from the histograms (Supplementary figure 1). For a comparative analysis of the amino-acid composition of the signal peptides, a background distribution was downloaded from Swissprot(Bateman et al. 2021). As shown from the plots (Supplementary figure 2), signal peptides from both sets appear to be more hydrophobic than the background distribution, having an higher composition in Leucin and Alanine. The Hydrophobic composition of the signal peptides was confirmed by sequence logos (supplementary figure 3), where a common motif of two alanine residues can be identified (the AXA motif). Pie plots charts were produced to observe the distribution of the proteins among species, showing an overall majority of sequences belonging to the metazoan kingdom (supplementary figure 4.1), and an heterogeneous distributions among taxa (supplementary figure 4.2).

2.2 VonHeijne method

Von Heijne is a algorithm that was developed specifically to detect signal peptides in proteins(von Heijne 1986). The modelling is done on the region around the cleavage-site, down to the h-region. Starting from a training set of fragmented sequences (-13,+2) a Position Specific Probability Matrix (PSPM) is built, holding the frequency of each residue type at each position. To avoid zero probabilities in the PSPM, and hence the impossibility of computing the log-odds, pseudo-counts are added during the computation. In fact, the PSPM is used to build the Position-Specific Weight Matrix (PSWM), holding the log-odds between frequencies in the PSPM and the background distribution (From SwissProt). Given a sequence X of length L, the likelihood score of the presence of the motif given the PSWM can be computed as:

$$score_{(X|W)} = \sum_{i=1}^{L} Wx_i, i$$

For the VonHeijne method, 5-fold cross validation was applied to the training set. The dataset was split 5 times, leading to 5 runs: at each iteration one of the subsets was held out for the validation, while the others were used to compute the PSWM and the optimal threshold. The mean between the 5 obtained threshold was stored for testing. In each run MCC, accuracy, precision, recall and F1 score were computed; the mean of each score among the 5 runs highlighted the overall performance on the training set (Table 1). To test the performance of the model on the benchmark set, another PWSM was built, using the whole training dataset. For the detection of the signal peptide motif along the protein sequences, a sliding window of 15 residues was adopted to scan positions from 1 to 35 with the PSWM, obtaining a score for each position. The global score for the sequence was chosen to be the maximum positional score. The classification was done by evaluating the global score for each sequence with the threshold optimized during cross-validation.

2.3 Support Vector Machine

Support vector machines are supervised machine learning models, which can be used for classification. The algorithm has the goal to separate two or more classes of points in a feature space, by maximizing the margin between the separating hyperplane and the datapoints, identified as support vectors (points with non-zero Lagrangian multipliers). Slack variables can also be added to allow misclassification of difficult or noisy examples, leading to a soft margin classification. To control the behaviour of the soft margin, the hyper-parameter C is introduced: it trades off the relative importance of margin maximization and fitting the training data. High C values will lead to a narrow margin and low training error, and vice versa. In the event that the classes are not linearly separable, a kernel can be adopted: it maps the data to an higher-dimensional space in which the classes can be separated. Kernels can be adopted by exploiting the kernel trick: since in the dual formulation of the problem, and in its solution, training points appear only inside inner products, there is no need to compute the coordinates of each point in the transformed space, but only

the product between them. For the detection of the signal peptide motifs, the radial basis function kernel was chosen, since it is computes how close the points are to each other:

$$K(X_1, X_2) = exp(-\frac{||X_1 - X_2||^2}{2\sigma^2})$$

The hyper-parameter "σ" is the variance, and must be optimized during cross-validation(Schölkopf 1998).

For the application of the support vector machine, feature extraction was performed on the training set. Each sequence was encoded in a 20-dimensional vector corresponding to the normalized composition of the first K residues in the protein. K is here treated as an hyper-parameter, requiring optimization. 5-fold cross validation was adopted as previously explained. Hyper-parameters were optimized via grid search, by testing the following combinations:

- Values of C ranging between 1 and 20.
- Values of σ ranging between 0.3 and 2, or with the "scale" setting.
- Values of K ranging between 18 and 24. The range was decided based on the length distribution of the signal peptide sequences.

After the selection of the optimal hyperparameters based on the resulting mcc, a table storing the mean between the scoring metrics in the different folds was produced (Table 1). The testing was done on the Benchmarking set adopting the selected parameters.

2.4 Performance Metrics

The reference performance metric selected for the project was the Matthew Correlation Coefficient, based on the negatives/positives rate of proteins in the dataset. In fact, the two classes are highly unbalanced, having a larger amount of negatives. Because of how the mcc is computed, it is less sensible to these unbalances:

$$\mathrm{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

Accuracy, recall, precision and f1 score were evaluated as well, as shown in table 1 and 2. For the wrong prediction analysis, False positive rates were computed as:

$$FPR = \frac{FP}{Actual\ Negative} = \frac{FP}{TN + FP}$$

3. Results and discussion

Both methods were trained by five-fold cross validation. For the Von Heijne approach, a threshold for classification had to be selected, while for the support vector machines, grid search was exploited to select the best possible combination of Hyper parameters C, σ(for the rbf kernel) and K (the length of the signal peptides). Table 1 shows the averaged performance metrics for both methods.

	MCC	Accuracy	Precision	Recall	F1 score
VonHeijne	0.784 +/- 0.011	0.945 +/- 0.004	0.872 +/- 0.011	0.874 +/- 0.009	0.873 +/- 0.004
SVM	0.862 +/- 0.008	0.965 +/- 0.002	0.901 +/- 0.016	0.864 +/- 0.008	0.882 +/- 0.006

Table 1 – Averaged performance metrics computed during the 5-fold cross validation for both the Von Heijne method and the SVM training.

The optimal threshold for the von Heijne was obtained by averaging the selected ones for each fold, resulting in 8.2. SVM hyper parameters were chosen based on the maximization of the Matthew correlation coefficient: C = 3, σ = 0.4, K = 19. The chosen parameters were used to test the models on the benchmark dataset, results are shown in table 2. As shown from the statistics, SVM resulted to be more efficient in detecting signal peptides in proteins, but both methods led to many misclassifications: 53 FN and 128 FP for the SVM, 54 FN and 169 FP for the von Heijne (supplementary figure 5).

	MCC	Accuracy	Precision	Recall	F1 score
VonHeijne	0.581546	0.9700912	0.4783951	0.7416268	0.5816135
SVM	0.6284214	0.9757242	0.5492958	0.7464115	0.63286

Table 2 – Test results on the evaluation of the models on the Benchmark set.

3.1 - False Positives analysis

To analyse the false positives, the False Positive Rate was computed for both the Von Heijne and the Support vector Machine models.

To understand the reasons behind the misclassification, the presence of common features among the false positive entries was investigated. Because of the hydrophobic composition of signal peptides, other hydrophobic structures present in the N-terminus, such as transit peptides or hydrophobic alpha helices, might lead to an erroneous positive classification. To inspect this possibility, rest calls to the Uniprot database were made with the API, to count the number of negative results having a transmembrane alpha helix or a transit peptide (for the mitochondrion, the chloroplast, or the peroxisome) in the N-terminus. The obtained counts were used to compute

feature-related FPR(table 3). Besides the expected differences between the two methods on the general FPR, which are related to their performance, it can be seen from the table that the transmembrane FPR is notably higher compared to the other metrics. This result confirms that the model was mostly misled by the presence of alpha helices, which should be treated to improve the model performance.

	FPR	TM	TP	MTH	CHP	PRX
VonHejne	0.02332	0.2972973	0.0351759	0.0330688	0.0400641	0
SVM	0.0176625	0.3040541	0.0301508	0.0462963	0.0128205	0

Table 3 - False positive rates computed using the full benchmark set (FPR), proteins with a transmembrane element (TM), transit peptides (TP), mitochondrial transit peptides (MTH), chloroplast transit peptides (CHP), peroxisomal transit peptides (PRX).

3.2 - False Negatives analysis

For the false negatives, different analyses were done for the two methods since the errors might have been related to the assumptions on which the modelling was based. For the von Heijne method, the aminoacidic composition around the cleavage site was used as main feature; for this reason, we tried to spot differences in the composition between the false negatives and the true positives by comparing their sequence logos with the logo of the training set. As can be seen in supplementary figure 6, the training set is more similar to the true positives in respect to the false negatives: in the latter the AXA cleavage site is less represented. This explains the misclassification and shows that, in order to obtain a more precise model, different features besides the amino acid composition should be taken into consideration.

For the support vector machine, we inspected If the errors were related to the length of the signal peptides by plotting the length distribution for both false negatives and true positives (supplementary figure 7). The histograms show that many sequences among the false negatives have a length that differs from the selected k parameter. This means that for sequences shorter than k, we may have introduced some noise, while for sequences that are longer than k, we may have failed in representing the full signal peptide. Further analyses were done by comparing the aminoacidic composition of the false negatives with the true positive and the training set. The histogram (supplementary figure 8) highlights differences in the false negative's composition, characterized by a notably higher presence of arginine and a lower presence of leucine. To check if the difference in composition was related to a different distribution of species in the false negatives, a comparison with the other sets was made, but not any relevant difference was detected (supplementary figure 9).

4. Conclusions

In this project a model capable to predict the presence of signal peptides in protein sequences was built. Two machine learning algorithms were exploited for comparison: the von Heijne method and the support vector machine. Both methods were trained via 5-fold cross validation starting from the SignalP 5.0 training dataset, which provided a sufficient amount of sequences to properly optimize the related hyperparameters. The performance of the models was evaluated on the benchmark set with the chosen metrics (MCC, accuracy, precision, recall and F1 score). The support vector machine resulted more efficient, with an overall better performance compared to the von Heijne. Even so, both models committed many misclassifications in the testing phase. The dataset was then inspected to find the reasons behind the errors for both methods. From the analyses, appeared that the sequences lacking the signal peptide that were classified as positives (FP), presented transit peptides and, more notably, transmembrane alpha helices, which might have led to the misclassification since both structures are characterized by an hydrophobic composition, as the SP. For what regards instead the false negatives, a comparison based on the aminoacidic composition and the length distribution with the false positives and the training set was made to highlight any differences. False negatives are characterized by different SP lengths in respect to the optimal selected one, and a different AA composition (more valin and arginin instead of leucin and alanin). From the results, we can conclude that most of the errors were caused by assumptions on which the algorithms were trained, and that by exploiting different features, the overall performance of the model might improve.

References

Almagro Armenteros, José Juan, Konstantinos D. Tsirigos, Casper Kaae Sønderby, Thomas Nordahl Petersen, Ole Winther, Søren Brunak, Gunnar von Heijne, and Henrik Nielsen. 2019. "SignalP 5.0 Improves Signal Peptide Predictions Using Deep Neural Networks." *Nature Biotechnology 2019 37:4* 37 (4): 420–23

Ashburner, Michael, Catherine A. Ball, Judith A. Blake, David Botstein, Heather Butler, J. Michael Cherry, Allan P. Davis, et al. 2000. "Gene Ontology: Tool for the Unification of Biology." *Nature Genetics* 25 (1): 25

Bateman, Alex, Maria Jesus Martin, Sandra Orchard, Michele Magrane, Rahat Agivetova, Shadab Ahmad, Emanuele Alpi, et al. 2021. "UniProt: The Universal Protein Knowledgebase in 2021." *Nucleic Acids*

Research 49 (D1): D480–89.

Cai, Yu Dong, Shuo Liang Lin, and Kuo Chen Chou. 2003. "Support Vector Machines for Prediction of Protein Signal Sequences and Their Cleavage Sites." *Peptides* 24 (1): 159–61.

Carbon, Seth, Eric Douglass, Benjamin M. Good, Deepak R. Unni, Nomi L. Harris, Christopher J. Mungall, Siddartha Basu, et al. 2021. "The Gene Ontology Resource: Enriching a GOld Mine." *Nucleic Acids Research* 49 (D1): D325–34.

Chou, K. C. 2001. "Prediction of Signal Peptides Using Scaled Window." *Peptides* 22 (12): 1973–79. https://doi.org/10.1016/S0196-9781(01)00540-X.

Heijne, Gunnar von. 1986. "A New Method for Predicting Signal Sequence Cleavage Sites." *Nucleic Acids Research* 14 (11): 4683–90.

Schölkopf, Bernhard. 1998. "SVMs - A Practical Consequence of Learning Theory." *IEEE Intelligent Systems and Their Applications* 13 (4): 18–21.

Teufel, Felix, José Juan Almagro Armenteros, Alexander Rosenberg Johansen, Magnús Halldór Gíslason, Silas Irby Pihl, Konstantinos D. Tsirigos, Ole Winther, Søren Brunak, Gunnar von Heijne, and Henrik Nielsen. 2022. "SignalP 6.0 Predicts All Five Types of Signal Peptides Using Protein Language Models." *Nature Biotechnology* 40 (7): 1023.