Predicting Secretory Proteins with SignalP

Nielsen, Henrik

Published in:
Methods in Molecular Biology

Link to article, DOI:
10.1007/978-1-4939-7015-5_6

Publication date:
2017

Document Version
Peer reviewed version

Citation (APA):
Nielsen, H. (2017). Predicting Secretory Proteins with SignalP. *Methods in Molecular Biology*, *1611*, 59-73.
https://doi.org/10.1007/978-1-4939-7015-5_6

Predicting secretory proteins with SignalP

Henrik Nielsen

Abstract

SignalP is the currently most widely used program for prediction of signal peptides from amino acid sequences. Proteins with signal peptides are targeted to the secretory pathway, but are not necessarily secreted. After a brief introduction to the biology of signal peptides and the history of signal peptide prediction, this chapter will describe all the options of the current version of SignalP and the details of the output from the program. The chapter includes a case study where the scores of SignalP were used in a novel way to predict the functional effects of amino acid substitutions in signal peptides.

Key words

Signal peptides, prediction, secretion, protein sorting, protein subcellular location

1. Introduction

A signal peptide (SP) is the N-terminal part of a protein that is targeted to the secretory pathway in both pro- and eukaryotes [1] (see, however, Note 1). In eukaryotes, a protein with an SP will be targeted to the endoplasmic reticulum (ER) membrane and be co-translationally translocated across the membrane. In prokaryotes, translocation takes place across the cytoplasmic membrane (inner membrane in Gram-negative bacteria), and the process can happen during or after translation. The SP-carrying protein is threaded through a protein complex known as the translocon, comprising the subunits SecY, E, and G in bacteria and Sec61 α, β, and γ in eukaryotes [2]. During translocation, the SP is cleaved off by an enzyme known as signal peptidase I or leader peptidase (*Lep*) in bacteria or signal peptidase complex in eukaryotes [3]. See Notes 2-4 for exceptions to this general picture.

It is important to stress that the presence of an SP does *not* necessarily mean that the protein is secreted to the extracellular environment—it only means that it enters the secretory pathway. In all kinds of organisms, the protein could have one or more transmembrane helices downstream of the SP and therefore be retained in the membrane [4]. In eukaryotes, the protein could also be retained in one of the compartments that belong to the secretory pathway: the ER, the Golgi apparatus, or the lysosome/vacuole [5]; or it could be anchored to the outer face of the cytoplasmic membrane by a glycophosphatidylinositol (GPI) group [6]. In Gram-negative bacteria, the protein could be retained in the periplasm, or be inserted into the outer membrane as a β-barrel transmembrane protein [7]. In Gram-positive bacteria, the protein could be attached to the cell wall [8].

SPs are generally described as having three regions: an N-terminal n-region of variable length characterized by positive charge, a central h-region of at least 7 hydrophobic residues, and a C-terminal c-region of typically 3-7 polar residues. Positions –1 and –3 relative to the cleavage site are occupied by small uncharged residues; in bacteria predominantly Alanine. SPs of Gram-positive bacteria tend to be longer than those of Gram-negative bacteria, which in turn tend to be longer than eukaryotic SPs [1].

The SP is among the earliest prediction targets for bioinformatic algorithms, with the first simple prediction methods being published already in the 1980's [9–11]. In the early 1990's, a few machine learning methods were published [12, 13], but SignalP version 1.0 [14, 15] was in 1996 the first machine learning method for SP prediction to be made into a publically available web server. SignalP 1.0 and 1.1 were based on artificial neural networks (ANNs), while SignalP 2.0 from 1998 [16] added a hidden Markov model (HMM) prediction in order to better distinguish between SPs and signal anchors (transmembrane helices close to the N-terminus). SignalP 3.0 from 2004 [17] introduced the D-score for better discrimination between SPs and other sequences and retained the HMM option, while SignalP 4.0 from 2011 [18] is again purely ANN-based. While constructing SignalP 4.0, we did retrain the HMM part, but we found that it did not perform better than the ANNs in any of the performance parameters we tested. The most important new feature of SignalP 4.0 is the improved discrimination between signal peptides and transmembrane regions.

SignalP was updated to version 4.1 in 2012 with an option to set the D-score cutoff values so that the sensitivity is the same as that of SignalP 3.0, and an option to set the minimum cleavage site position in the sequence (the minimum SP length). More details about these options are given in Section 3.1. In addition, the documentation on the website was completely rewritten, and a FAQ was added.

Earlier versions of SignalP have repeatedly been reported as the best performing method in independent benchmarks [19–22]. SignalP 4 has not yet been independently evaluated, but in the SignalP 4.0 paper [18] we compared the performance to ten other methods and found that it was superior. The best competing methods were the combined SP and transmembrane helix predictors Phobius [23], Philius [24], and SPOCTOPUS [25]. Interestingly, the advantage of SignalP 4.0 over these three programs was larger for bacteria than for eukaryotes. This may be due to the fact that these three methods did not divide their training data into different organism groups but pooled them all together, resulting in methods that are optimized for the most abundant organism group in the data, the eukaryotes.

The performance values for SignalP 3.0 and 4.0 and the ten competing methods can be found in Table E of the supplementary materials of the SignalP 4.0 paper, which is available on the SignalP web site (click on “Article abstracts” and then “Update to SignalP v. 4.0”). It should be noted that those values are calculated by cross-validation on a homology-reduced data set, i.e. they are the performances you should expect when submitting proteins that are unrelated to anything in the SignalP 4.0 data set. When submitting close homologs to proteins in the SignalP 4.0 data set, a higher performance should be expected (compare the aforementioned Table E with the table on the “performance” page of the website documentation).

2. Materials

1. *Input data:* Amino acid sequences in FASTA format. Note that any letters not corresponding to the twenty standard amino acids, e.g. ‘U’, ‘B’, or ‘Z’, will be converted to ‘X’ and treated as unknown amino acids. See also Notes 5 and 6.
2. *Website:* SignalP 4.1 is available at http://www.cbs.dtu.dk/services/SignalP/, see Figure 1. The previous versions are also kept online; just click “version history” near the top of the page.
3. *Downloadable package:* For those who prefer running SignalP on their own computers, there is an option to download a software package for command line use. The package is free for academic institutions, while there is a license fee for commercial users. Academic users can go to the page http://www.cbs.dtu.dk/cgi-bin/nph-sw_request?signalp to fill out the details and accept the license, while commercial users are asked to contact software@cbs.dtu.dk. The package is available for

Linux, IRIX, Darwin (Mac OS X), and from March 2016 also for Windows computers via the free Unix-like environments provided by Cygwin [26] or MobaXterm [27].

3. Methods

Running SignalP with the default options is straightforward: On the website, you paste or upload the sequences and click "submit"; on the command line you write "`signalp` *`input.fasta`*". The output will tell you, for each sequence, whether there is an SP predicted, and if yes, where the cleavage site is predicted to be. However, as seen in Figure 1, there are a number of options, of which especially "*Organism group*" and "*Cutoff*" are important to know about, and there are details of the output format that will help interpret the predictions.

3.1. Options

Organism group: It is important to choose the correct organism group— Eukaryotes, Gram-negative bacteria, or Gram-positive bacteria—otherwise, the predictive performance will suffer. In this context, Gram-positive bacteria are defined as the phyla Actinobacteria (high G+C Gram-positive bacteria) and Firmicutes. Gram-negative bacteria are defined as all bacteria having both a plasma membrane and an outer membrane—basically all other bacteria except for the phylum Tenericutes (*Mycoplasma* and related genera). SignalP probably should not be used for Tenericutes at all, since they seem to lack a type I signal peptidase completely [28]. On the command line, organism group is chosen with one of the options "`-t euk`" (the default), "`-t gram-`", or "`-t gram+`". Concerning organism groups, see also Notes 7-11.

Output format: There are four levels of detail possible: "short", "standard", "long", and "all". The two first formats report scores and conclusion at the sequence level; "short" in a one-line format and "standard" in a more human-readable format. "Standard" is the default on the web server, and "short" on the command line. The "long" and "all" formats additionally report scores for each position in each sequence (for an explanation of the scores, see next section). The difference between "long" and "all" is that "long" reports scores for the chosen ANN method only, while "all" reports scores for both ANN methods (SignalP-noTM and SignalP-TM, see "*Method*" below for an explanation). On the command line, output format is chosen with the "`-f`" option; note that "standard" is chosen with "`-f summary`".

Graphics output: SignalP can make a plot of the scores for each position in each sequence in portable network graphics (PNG) format and optionally also in encapsulated postscript (EPS) format. The default on the web is to make PNG graphics, while the default on the command line is no graphics. If you want graphics from the command line, use the options "`-g png`" or "`-g png+eps`".

Method: SignalP 4 has two sets of ANNs: SignalP-noTM is trained with only cytosolic and nuclear proteins in the negative set, while SignalP-TM is trained with a negative set that also included transmembrane proteins. During training, we found that the two methods SignalP-TM and SignalP-noTM were to some extent complementary, i.e. SignalP-TM did not yield as good results as SignalP-noTM when there were no transmembrane sequences involved. As a compromise, SignalP 4 per default uses a heuristic to decide which of the two sets of networks is used for prediction of each sequence. If the user is positive that all proteins in the input are soluble, it is possible to override this heuristic and get a slightly better performance by using only the SignalP-noTM networks. This is done in the web interface by selecting "Input sequences do not include TM regions" and on the command line by including the option "`-s notm`".

Cutoff: The D-score (see next section) is used for determining whether each input sequence contains an SP or not. The user can set cutoff values (for SignalP-TM and SignalP-noTM separately) if a different balance between sensitivity and specificity is desired. The web interface offers two sets of predefined cutoff values, "Default" and "Sensitive". The "Default" cutoffs, corresponding to SignalP 4.0, are optimized to give the best Matthews correlation coefficient (see the "Performance" page on the website for definition), but they result in a quite conservative prediction with a sensitivity that is actually lower than that of SignalP 3.0. The "Sensitive" cutoffs, introduced in SignalP 4.1, are set to reproduce the sensitivity of SignalP 3.0. This of course results in a slightly higher false positive rate, but still significantly better than that of SignalP 3.0 when measured on the whole data set (with transmembrane proteins included in the negative set). Our recommendation is to use the "Sensitive" setting if it is important to avoid false negatives, but use the "Default" setting for estimating the proportion of SPs in an organism. The estimation by the "Default" cutoff was found to be in accordance with an estimate of the number of SPs in *Escherichia coli* by a recent proteogenomics study [29]. At the website, you can see the preset cutoff values change when you select "Default" or "Sensitive" or change the organism group. On the command line, the "Sensitive" cutoffs are selected by including the options "`-U 0.34 -u 0.34`" for organism group "`euk`" or "`-U 0.42 -u 0.42`" for organism groups "`gram+`" and "`gram-`".

Truncation of input sequence: By default, SignalP truncates every sequence to 70 amino acids before prediction. This gives enough included sequence after the cleavage site to give the optimal prediction for the vast majority of SPs. If you want to predict extremely long signal peptides, you can try a higher value, or disable truncation completely by entering 0 (zero). Note that the neural networks are trained with sequences with a maximal length of 70, and they include the relative position in the sequence in their input. Therefore, general performance may deteriorate if you change this setting. On the command line, truncation is changed with the "`-c`" option.

Minimal predicted signal peptide length: SignalP 4.0 could, in rare cases, erroneously predict extremely short signal peptides. These errors have in SignalP 4.1 been eliminated by imposing a lower limit on the cleavage site position (SP length). The minimum length is by default 10, but you can adjust it. Signal peptides shorter than 15 residues are very rare, at the time of writing there are 17 experimentally confirmed cases in UniProt that are not fragments. If you want to disable this length restriction completely, enter 0 (zero). On the command line, minimal SP length is changed with the "`-M`" option.

3.2. Output

The neural networks in SignalP produce three output scores for each position in the input sequence:

- *C-score* (raw cleavage site score): The output from the cleavage site networks, which are trained to distinguish SP cleavage sites from everything else. Note the position numbering of the cleavage site: The C-score is trained to be high at the position immediately *after* the cleavage site (the first residue in the mature protein).
- *S-score* (signal peptide score): The output from the signal peptide networks, which are trained to distinguish positions within SPs from positions in the mature part of the proteins and from proteins without SPs.
- *Y-score* (combined cleavage site score): A combination (geometric average) of the C-score and the slope of the S-score, resulting in a better cleavage site prediction than the raw C-score alone. This is due to the fact that multiple high-peaking C-scores can be found in one sequence, where only one is the true cleavage site. The Y-score distinguishes between C-score peaks by choosing the one where the slope of the S-score is steep.

The graphical output from SignalP (Figure 2 and Figure 3) shows the three different scores, *C*, *S* and *Y*, for each position in the sequence.

In the summary below the plot, the maximal values of the three scores are reported. In addition, the following two scores are shown:

- *mean S:* The average S-score of the possible SP (from position 1 to the position immediately before the maximal Y-score).
- *D-score* (discrimination score): A weighted average of the mean S and the maximal Y scores. This is the score that is used to discriminate signal peptides from non-signal peptides.

For a typical SP, the plot will resemble the one in Figure 2 with one peak in C- and Y-score and an S-score that is high in the beginning (close to the positive target value of 0.9) and then falls to a low value. For non-secretory proteins all the scores represented in the SignalP output should ideally be very low (close to the negative target value of 0.1).

The plot can give valuable information about the confidence of the prediction. For example, an intermediate S-score (close to 0.5) signifies that SignalP is unsure whether the sequence is a signal peptide, and two or more peaks in Y-score indicate that SignalP is unsure about the exact position of the cleavage site (see Figure 3). See also Notes 6 and 12.

Below the summary for each sequence, two files are provided via links: "data" and "gnuplot script". If you have the free graphics program gnuplot [30] on your computer, you can use these two files to customize your plot. If you want to keep these files when using the command line interface, you need to include the option "`-k`".

Below the output for all the sequences, two other files are provided via links, if at least one SP has been predicted. These are "processed fasta entries", a FASTA sequence file containing the sequences of those proteins that had predicted SPs, with the SP removed; and "gff file of processed entries", a file showing the signal peptides feature of those proteins that had predicted SPs in the format GFF (gene finding format). Note that these two files are not produced by default in the command line interface; if you want them, include the options "`-m` *`filename`*" for processed FASTA entries or "`-n` *`filename`*" for GFF entries. See, however, Note 13.

The file with processed FASTA entries can be very useful for downstream analysis of those proteins that were predicted to have SPs. For example, if the focus is on predicting secreted proteins, it should always be checked whether there are predicted transmembrane helices downstream of the SP. This can e.g. be done by submitting the processed FASTA entries to TMHMM [31, 32]. The advantage of using the processed FASTA entries instead of the entire sequences is that you get rid of the false positive transmembrane helix predictions that TMHMM often makes for SPs.

3.3. Case Study

Since 1996, SignalP has been used to predict countless SPs. The three most influential papers about SignalP [14, 17, 18] have been cited more than 11,000 times in total according to Web of Science [33]. It is difficult to single out one particular SP prediction study as more interesting than the others.

However, one study from a group at the company Genentech [34] deserves special mention, since they used the output scores of SignalP in a creative way we had not anticipated. The paper is from 2009 and is therefore based on SignalP 3.0, but the same approach should be applicable to version 4.1. The focus of the study is the prediction of the functional effects of amino acid substitutions in SPs. It is important to stress that SignalP was not designed for this purpose—SignalP has been trained on wild type sequences only, and mutated SPs which have lost their function partially or completely are more similar to wild type SPs than to wild type non-SPs. Mutated SPs can therefore be said to occupy a different part of sequence space than the wild type SPs and non-SPs comprising the SignalP training set, and the task of predicting consequences of amino acid substitutions in SPs is probably a harder problem than the one for which SignalP was designed.

Predicting whether amino acid substitutions (non-synonymous single nucleotide polymorphisms, nsSNPs) have functional consequences for proteins is an intensely studied problem in bioinformatics. In general, the problem is defined as the discrimination between a positive data set of known disease-causing mutations and a negative data set of presumed neutral nsSNPs based on the amino acid sequence and the pattern of conservation around each substituted amino acid. One of the first and best known predictors for this problem is SIFT (sorting intolerant from tolerant) [35–37].

The Genentech authors gathered data sets of disease-causing mutations and neutral nsSNPs which occurred within the signal peptide region of human secretory proteins, and they hypothesized that the disease-causing mutations interfered with signal peptide function and therefore should be predictable by SignalP. They then defined a novel score based on the "long" format output of SignalP. This was termed the "R-score" and defined thus:

$$R = \max(\Delta S_{1\ldots n}) - \min(\Delta S_{1\ldots n}) + \max(\Delta C_{1\ldots n}) - \min(\Delta C_{1\ldots n})$$

where ΔS_i is the difference in S-score between the mutant and the wild type at position i, and $\max(\Delta S_{1\ldots n})$ is the maximal value of that difference within the entire predicted SP (n is the cleavage site position predicted for the wild type). The same definitions apply to the terms with C-score.

The authors were able to show that the R-score was significantly better than the simple difference in D-score (ΔD) for discriminating between disease-causing mutations and neutral nsSNPs. The performance of the R-score was similar to that of the score from SIFT which does no SP prediction. Furthermore, they showed that R-score and SIFT-score were not correlated, suggesting that they contributed independent information about the discrimination. Accordingly, a simple combination of R- and SIFT-score gave a better discrimination than either score alone.

It could be interesting to see whether these results are still valid when using SignalP 4.1 and some of the newer alternatives to SIFT, such as PolyPhen-2 [38], PON-P2 [39], SNAP2 [40], etc. In addition, more advanced ways of combining SignalP output with the outputs of other programs may prove to perform even better. A Chinese group in 2012 used a Random Forest classifier coupled with a feature selection scheme to integrate SignalP 3.0 output with sequence profiles and physicochemical parameters [41]. Their approach is interesting, and they report a large increase in discrimination performance relative to the R-score; but their result may be marked by overfitting, since no effort has been done to avoid homology between training and test sets. Homology reduction has been a crucial element in the construction of the SignalP datasets ever since version 1.0 [42] and homology partitioning is used in recent nsSNP effect predictors [39, 40].

A closely related question is whether SignalP is able to predict functional consequences of artificially induced mutations in SPs. We have tried using both D-score and R-score on a set of mutated SPs from bacteria, but found the effects surprisingly difficult to predict (results not published). One possible future direction for SignalP could be to use such data in the training phase to improve the prediction of effects of mutations, whether naturally occurring or artificially induced. A method trained in this way might also yield improved predictions of secretion efficiency, cf. Note 12.

Notes

1. *SP definition:* Sometimes, especially in introductory textbooks, the term “signal peptide” has been used in a broader sense, meaning any (or any cleavable) sorting signal embedded in the amino acid sequence of a protein. However, the definition of SP given in Section 1 corresponds to the usage in most of the scientific literature, as well as in in UniProt [43], Wikipedia [44], and the Sequence feature ontology [45]. Signals for import into mitochondria and chloroplasts are properly termed *transit peptides* and can be predicted e.g. with the program TargetP [46, 47].
2. *Uncleaved SPs:* There are rare examples of SPs that are not cleaved (at the time of writing, there are 79 such cases annotated in UniProt). These should not be confused with *signal anchors*, which are transmembrane helices close to the N-terminus. Uncleaved SPs are very differently predicted by SignalP; some look like typical SPs, some look like typical non-SPs, and others have high S-scores but low C- and Y-scores.
3. *Bacterial lipoproteins:* Bacterial lipoproteins have special signal peptides which are cleaved by signal peptidase II, also known as Lipoprotein signal peptidase (*Lsp*). A diacylglyceryl group is attached to a conserved Cys residue in position +1 relative to the cleavage site, which bears no resemblance to the signal peptidase I cleavage site [48]. SignalP often predicts such sequences as SPs, but with a wrong cleavage site. For prediction of prokaryotic lipoproteins we recommend using the LipoP server [49, 50].
4. *Tat signal peptides:* The special SPs that direct some bacterial proteins through the Tat (Twin arginine translocation) pathway instead of the Sec pathway are not very well predicted by SignalP. These SPs have a special motif containing two Arginines in the N-terminal part, and they are in general longer and less hydrophobic than normal SPs [51, 52]. For prediction of Tat signal peptides we recommend using the TatP server [53, 54].
5. *Nucleotide sequences:* Note that SignalP will *not* produce sensible output from DNA sequences; it will treat such sequences as proteins exclusively consisting of Ala, Cys, Gly, and Thr.
6. *Start codon prediction:* Since SignalP predicts an N-terminal signal, it is dependent on a correct start codon assignment. A start codon assigned too far downstream will cut part of the true amino acid sequence, while a start codon assigned too far upstream will add arbitrary sequence to the N-terminus, both making it difficult to recognize a possible SP. If you get a prediction of an unusually long SP where the S-score is low in the beginning but then rises to a higher value, you should look for possible alternative start codons downstream of the annotated one. For eukaryotic sequences, you might want to check start codon predictions with the program NetStart [55, 56].
7. *Archaea:* There is no “Archaea” option among the organism groups. This is certainly not because Archaea do not have SPs, but because there are too few experimentally confirmed SPs from Archaea (at the time of writing, the number is 11 in UniProt).
8. *Viruses:* There is no “Viruses” option among the organism groups. Virus or phage SPs should be predicted according to their host organism group.
9. *Atypical Gram-positives:* Certain bacteria, notably *Deinococcus spp.*, have a thick cell wall and react positively to the Gram staining procedure, even though they also have an outer membrane [57]. Their SPs should probably be predicted with “Gram-negative bacteria” as organism group, although too few SPs from such organisms are experimentally known to answer this question confidently.

10. *SP diversity within eukaryotes:* It has long been known that some yeast signal peptides are not recognized by mammalian cells [58]. Therefore, it would be natural to assume that separate SignalP versions for yeast and Mammalia would provide better predictions than a common eukaryotic version. While developing SignalP 4.0 we tried dividing the eukaryotic data into animals, fungi, and plants and training separate methods for these three groups. However, this did not give any improvement, and performance for all three groups was better when using the method trained on all eukaryotic sequences together. This should be tested again for the next version of SignalP.
11. *SP diversity within bacteria:* The Gram-negative version of SignalP is probably biased towards *E. coli* and other γ-proteobacteria, since these constitute the bulk of the experimentally annotated Gram-negative SPs in UniProt. Newer results suggest that some bacteria have rather divergent cleavage site motifs [59]. Future versions of SignalP might therefore benefit from dividing the Gram-negative bacteria into several groups, if enough data are available.
12. *Secretion efficiency:* A frequently asked question is whether SignalP can predict secretion efficiency when attaching an SP to a heterologous protein—in other words, whether efficient SPs score higher than slowly secreting SPs. The answer is unfortunately not known. Intuitively, one would expect efficient secretors to have higher Y- and D-scores. However, SignalP is trained to recognize SPs against a background of non-SPs, regardless of secretion efficiency. This means that inefficient secretors are trained with the same target value as efficient secretors, as long as they are naturally occurring SPs. Therefore, the scores will not necessarily correlate with secretion efficiency.
13. *GFF version:* The gene finding format you get with the "`-n`" option is GFF version 2, which is actually deprecated [60]. This should be updated to GFF version 3 in the next SignalP version.

Acknowledgments

Heartfelt thanks go to all co-authors on the SignalP papers though the years: Søren Brunak, Jacob Engelbrecht, Gunnar von Heijne, Anders Krogh, Jannick Dyrløv Bendtsen, and Thomas Nordahl Petersen. In addition, I wish to thank the people who helped implementing the website and still work on keeping it up and running: Kristoffer Rapacki, Hans Henrik Stærfeldt, and Peter Wad Sackett.

References

1. von Heijne G (1990) The Signal Peptide. J Membr Biol 115:195–201. doi: 10.1007/BF01868635

2. Pohlschröder M, Prinz WA, Hartmann E, Beckwith J (1997) Protein translocation in the three domains of life: variations on a theme. Cell 91:563–566. doi: 10.1016/S0092-8674(00)80443-2

3. Dalbey RE, Lively MO, Bron S, Dijl JMV (1997) The chemistry and enzymology of the type I signal peptidases. Protein Sci 6:1129–1138. doi: 10.1002/pro.5560060601

4. von Heijne G (1988) Transcending the impenetrable: How proteins come to terms with membranes. Biochim Biophys Acta BBA - Rev Biomembr 947:307–333. doi: 10.1016/0304-4157(88)90013-5

5. Harter C, Wieland F (1996) The secretory pathway: mechanisms of protein sorting and transport. Biochim Biophys Acta BBA - Rev Biomembr 1286:75–93. doi: 10.1016/0304-4157(96)00003-2

6. Ferguson MAJ, Williams AF (1988) Cell-Surface Anchoring of Proteins via Glycosyl-Phosphatidylinositol Structures. Annu Rev Biochem 57:285–320. doi: 10.1146/annurev.bi.57.070188.001441

7. Duong F, Eichler J, Price A, et al (1997) Biogenesis of the Gram-negative bacterial envelope. Cell 91:567–573. doi: 10.1016/S0092-8674(00)80444-4

8. Mazmanian SK, Liu G, Ton-That H, Schneewind O (1999) Staphylococcus aureus Sortase, an Enzyme that Anchors Surface Proteins to the Cell Wall. Science 285:760–763. doi: 10.1126/science.285.5428.760

9. von Heijne G (1983) Patterns of Amino Acids near Signal-Sequence Cleavage Sites. Eur J Biochem 133:17–21. doi: 10.1111/j.1432-1033.1983.tb07424.x

10. McGeoch DJ (1985) On the predictive recognition of signal peptide sequences. Virus Res 3:271–286. doi: 10.1016/0168-1702(85)90051-6

11. von Heijne G (1986) A new method for predicting signal sequence cleavage sites. Nucleic Acids Res 14:4683–4690. doi: 10.1093/nar/14.11.4683

12. Ladunga I, Czakó F, Csabai I, Geszti T (1991) Improving signal peptide prediction accuracy by simulated neural network. Comput Appl Biosci CABIOS 7:485–487. doi: 10.1093/bioinformatics/7.4.485

13. Schneider G, Wrede P (1993) Development of artificial neural filters for pattern recognition in protein sequences. J Mol Evol 36:586–595. doi: 10.1007/BF00556363

14. Nielsen H, Brunak S, Engelbrecht J, von Heijne G (1997) Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. Protein Eng 10:1–6. doi: 10.1093/protein/10.1.1

15. Nielsen H, Engelbrecht J, Brunak S, Heijne GV (1997) A neural network method for identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. Int J Neural Syst 8:581–599. doi: 10.1142/S0129065797000537

16. Nielsen H, Krogh A (1998) Prediction of signal peptides and signal anchors by a hidden Markov model. Proc Int Conf Intell Syst Mol Biol 6:122–30.

17. Bendtsen JD, Nielsen H, von Heijne G, Brunak S (2004) Improved prediction of signal peptides: SignalP 3.0. J Mol Biol 340:783–95. doi: 10.1016/j.jmb.2004.05.028

18. Petersen TN, Brunak S, von Heijne G, Nielsen H (2011) SignalP 4.0: discriminating signal peptides from transmembrane regions. Nat Meth 8:785–786. doi: 10.1038/nmeth.1701

19. Menne KML, Hermjakob H, Apweiler R (2000) A comparison of signal sequence prediction methods using a test set of signal peptides. Bioinformatics 16:741–742. doi: 10.1093/bioinformatics/16.8.741

20. Klee E, Ellis L (2005) Evaluating eukaryotic secreted protein prediction. BMC Bioinformatics 6:1–7. doi: 10.1186/1471-2105-6-256

21. Choo K, Tan T, Ranganathan S (2009) A comprehensive assessment of N-terminal signal peptides prediction methods. BMC Bioinformatics 10:S2. doi: 10.1186/1471-2105-10-S15-S2

22. Zhang X, Li Y, Li Y (2009) Evaluating signal peptide prediction methods for Gram-positive bacteria. Biologia (Bratisl) 64:655–659. doi: 10.2478/s11756-009-0118-3

23. Käll L, Krogh A, Sonnhammer EL. (2004) A Combined Transmembrane Topology and Signal Peptide Prediction Method. J Mol Biol 338:1027–1036. doi: 10.1016/j.jmb.2004.03.016

24. Reynolds SM, Käll L, Riffle ME, et al (2008) Transmembrane Topology and Signal Peptide Prediction Using Dynamic Bayesian Networks. PLoS Comput Biol 4:e1000213. doi: 10.1371/journal.pcbi.1000213

25. Viklund H, Bernsel A, Skwark M, Elofsson A (2008) SPOCTOPUS: a combined predictor of signal peptides and membrane protein topology. Bioinformatics 24:2928–2929. doi: 10.1093/bioinformatics/btn550

26. Cygwin. https://cygwin.com/. Accessed 30 May 2016

27. MobaXterm free Xserver and tabbed SSH client for Windows. http://mobaxterm.mobatek.net/. Accessed 30 May 2016

28. Fraser CM, Gocayne JD, White O, et al (1995) The minimal gene complement of Mycoplasma genitalium. Science 270:397–404. doi: 10.1126/science.270.5235.397

29. Ivankov DN, Payne SH, Galperin MY, et al (2013) How many signal peptides are there in bacteria? Environ Microbiol 15:983–990. doi: 10.1111/1462-2920.12105

30. gnuplot homepage. http://www.gnuplot.info/. Accessed 30 May 2016

31. Krogh A, Larsson B, von Heijne G, Sonnhammer EL. (2001) Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. J Mol Biol 305:567–580. doi: 06/jmbi.2000.4315

32. TMHMM Server, v. 2.0. http://www.cbs.dtu.dk/services/TMHMM/. Accessed 30 May 2016

33. Henrik Nielsen D-4128-2011 - ResearcherID.com. http://www.researcherid.com/rid/D-4128-2011. Accessed 30 May 2016

34. Hon LS, Zhang Y, Kaminker JS, Zhang Z (2009) Computational prediction of the functional effects of amino acid substitutions in signal peptides using a model-based approach. Hum Mutat 30:99–106. doi: 10.1002/humu.20798

35. Ng PC, Henikoff S (2003) SIFT: predicting amino acid changes that affect protein function. Nucleic Acids Res 31:3812–3814. doi: 10.1093/nar/gkg509

36. Vaser R, Adusumalli S, Leng SN, et al (2016) SIFT missense predictions for genomes. Nat Protoc 11:1–9. doi: 10.1038/nprot.2015.123

37. SIFT - Predict effects of nonsynonmous / missense variants. http://sift.bii.a-star.edu.sg/. Accessed 30 May 2016

38. Adzhubei IA, Schmidt S, Peshkin L, et al (2010) A method and server for predicting damaging missense mutations. Nat Meth 7:248–249. doi: 10.1038/nmeth0410-248

39. Niroula A, Urolagin S, Vihinen M (2015) PON-P2: Prediction Method for Fast and Reliable Identification of Harmful Variants. PLOS ONE 10:e0117380. doi: 10.1371/journal.pone.0117380

40. Hecht M, Bromberg Y, Rost B (2015) Better prediction of functional effects for sequence variants. BMC Genomics 16:S1. doi: 10.1186/1471-2164-16-S8-S1

41. Qin W, Li Y, Li J, et al (2012) Predicting deleterious non-synonymous single nucleotide polymorphisms in signal peptides based on hybrid sequence attributes. Comput Biol Chem 36:31–35. doi: 10.1016/j.compbiolchem.2011.12.001

42. Nielsen H, Engelbrecht J, von Heijne G, Brunak S (1996) Defining a similarity threshold for a functional protein sequence pattern: The signal peptide cleavage site. Proteins Struct Funct Bioinforma 24:165–177. doi: 10.1002/(SICI)1097-0134(199602)24:2<165::AID-PROT4>3.0.CO;2-I

43. UniProt help: Signal peptide. http://www.uniprot.org/help/signal. Accessed 30 May 2016

44. Signal peptide - Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Signal_peptide. Accessed 30 May 2016

45. SO_0000418 < Ontology Lookup Service < EMBL-EBI. http://www.ebi.ac.uk/ols/ontologies/so/terms?iri=http%3A%2F%2Fpurl.obolibrary.org%2Fobo%2FSO_0000418. Accessed 30 May 2016

46. Emanuelsson O, Nielsen H, Brunak S, von Heijne G (2000) Predicting subcellular localization of proteins based on their N-terminal amino acid sequence. J Mol Biol 300:1005–1016. doi: 10.1006/jmbi.2000.3903

47. TargetP 1.1 Server. http://www.cbs.dtu.dk/services/TargetP/. Accessed 30 May 2016

48. von Heijne G (1989) The structure of signal peptides from bacterial lipoproteins. Protein Eng 2:531–534. doi: 10.1093/protein/2.7.531

49. Juncker AS, Willenbrock H, von Heijne G, et al (2003) Prediction of lipoprotein signal peptides in Gram-negative bacteria. Protein Sci 12:1652–62. doi: 10.1110/ps.0303703

50. LipoP 1.0 Server. http://www.cbs.dtu.dk/services/LipoP/. Accessed 30 May 2016

51. Cristóbal S, de Gier J-W, Nielsen H, von Heijne G (1999) Competition between Sec- and TAT-dependent protein translocation in Escherichia coli. EMBO J 18:2982–2990. doi: 10.1093/emboj/18.11.2982

52. Berks BC (2015) The Twin-Arginine Protein Translocation Pathway. Annu Rev Biochem 84:843–864. doi: 10.1146/annurev-biochem-060614-034251

53. Bendtsen JD, Nielsen H, Widdick D, et al (2005) Prediction of twin-arginine signal peptides. BMC Bioinformatics 6:167. doi: 10.1186/1471-2105-6-167

54. TatP 1.0 Server. http://www.cbs.dtu.dk/services/TatP/. Accessed 30 May 2016

55. Pedersen AG, Nielsen H (1997) Neural Network Prediction of Translation Initiation Sites in Eukaryotes: Perspectives for EST and Genome analysis. AAAI Press, Menlo Park, California, pp 226–233

56. NetStart 1.0 Prediction Server. http://www.cbs.dtu.dk/services/NetStart/. Accessed 30 May 2016

57. Thompson BG, Murray RGE (1981) Isolation and characterization of the plasma membrane and the outer membrane of Deinococcus radiodurans strain Sark. Can J Microbiol 27:729–734. doi: 10.1139/m81-111

58. Bird P, Gething MJ, Sambrook J (1987) Translocation in yeast and mammalian cells: not all signal sequences are functionally equivalent. J Cell Biol 105:2905–2914. doi: 10.1083/jcb.105.6.2905

59. Payne SH, Bonissone S, Wu S, et al (2012) Unexpected Diversity of Signal Peptides in Prokaryotes. mBio 3:e00339-12. doi: 10.1128/mBio.00339-12

60. GFF2 - GMOD. http://gmod.org/wiki/GFF2. Accessed 30 May 2016

Figure captions

Figure 1: The SignalP 4.1 web site, showing the input field and the available options.

Figure 2: SignalP output for a protein with a typical signal peptide, Human endoplasmic reticulum resident protein 44. Note that there is one conspicuous peak in Y-score at position 30, meaning that the signal peptide is predicted to be cleaved between amino acids 29 and 30. Please note that this protein is not secreted.

Figure 3: SignalP output for a protein with a less typical signal peptide, Mouse Trefoil factor 1. There is a five-fold “AQ” repeat around the cleavage site region, resulting in four peaks in Y-score. Although the first peak is the highest, corresponding to a predicted signal peptide length of 21, the prediction of the cleavage site position should be taken with caution in this case.