From sequence to sorting: Prediction of signal peptides

Ph.D. thesis
Henrik Nielsen

Department of Biochemistry
Stockholm University
S-106 91 Stockholm
Sweden

and

Center for Biological Sequence Analysis
Department of Biotechnology
Technical University of Denmark
DK-2800 Lyngby
Denmark

Stockholm 1999

Cover illustration:
A SignalP prediction of the signal peptide cleavage site of Human cystatin C precursor. The true cleavage site is marked with an arrow. The C-score (the output from one neural network) predicts a false cleavage site, but the Y-score (a combination of outputs from two neural networks) is able to predict the cleavage site correctly.

Ph.D. thesis
Stockholm University

ISBN 91-7153-936-0
Kista Snabbtryck AB, 164 40 Kista, `http://www.kistasnabbtryck.se/`

Contents

Abstract

In the present age of genome sequencing, a vast number of predicted genes are initially known only by their putative nucleotide sequence. The newly established field of bioinformatics is concerned with the computational prediction of structural and functional properties of genes and the proteins they encode, based on their nucleotide and amino acid sequences.

Since one of the crucial properties of a protein is its subcellular location, prediction of protein sorting is an important question in bioinformatics. A fundamental distinction in protein sorting is that between secretory and non-secretory proteins, determined by a cleavable N-terminal sorting signal, the secretory signal peptide.

The main part of this thesis, including four of the six papers, concerns prediction of secretory signal peptides in both eukaryotic and bacterial data using two machine learning techniques: artificial neural networks and hidden Markov models. A central result is the SignalP prediction method, which has been made available as a World Wide Web server and is very widely used.

Two additional prediction methods are also included, with one paper each. ChloroP predicts chloroplast transit peptides, another cleavable N-terminal sorting signal; while NetStart predicts start codons in eukaryotic genes. For prediction of all N-terminal signals, the assignment of correct start codon can be critical, which is why prediction of translation initiation from the nucleotide sequence is also important for protein sorting prediction.

This thesis comprises a detailed review of the molecular biology of protein secretion, a short introduction to the most important machine learning algorithms in bioinformatics, and a critical review of existing methods for protein sorting prediction. In addition, it contains general treatment of the principles of data set construction and performance evaluation for prediction methods in bioinformatics.

Preface

This Ph.D. thesis is written for Department of Biochemistry, Stockholm University, under the Theoretical Chemistry programme.

The research project has been carried out as a collaboration between Center for Biological Sequence Analysis, Department of Biotechnology, Technical University of Denmark, and Department of Biochemistry, Stockholm University. Supervisor is professor Gunnar von Heijne, Department of Biochemistry, and co-supervisor is professor Søren Brunak, Center for Biological Sequence Analysis.

Physically, I have spent most of my time in Denmark, which is why my Danish address is used on the papers included in this thesis; but the bulk of the work actually took place in cyberspace, in the form of bits traveling along the Stockholm–Copenhagen digital cables.

The project has been financed by grants from the Danish National Research Foundation to Søren Brunak and from the Swedish Natural Research Council to Gunnar von Heijne. My own salary has been provided by the Danish National Research Foundation.

Henrik Nielsen, April 1999

Acknowledgments

I wish to thank everybody who made this project possible:

- My two supervisors, Gunnar von Heijne and Søren Brunak, for their inspiration and support throughout the whole process;
- Coauthors of the included papers: Jacob Engelbrecht (now at Novo Nordisk), Anders Gorm Pedersen, Anders Krogh, and Olof Emanuelsson—you have all been great to work with;
- Anders Gorm Pedersen, Søren Brunak, and Jan Gorodkin, for comments on earlier versions of this text;
- Stefan Nordlund for his willingness to accept me as a Ph.D. student of Department of Biochemistry although my physical presence has been less than the norm;
- Anders Gorm Pedersen (once again) for lots of chat and support, both scientific and personal;
- Kristoffer Rapacki for technical assistance, for help with data extraction, and for countless pleasant conversations;
- Jan Gorodkin for his endless supply of jokes—actually, a few of them were quite good;
- The whole group at CBS for hundreds of inspiring discussions and for making the center such a friendly place;
- Everybody in Gunnar von Heijne's and Arne Elofsson's groups for their friendly and warm welcome on all my Stockholm visits and for some good nights out in town—special thanks to Erik and Keng-Ling for your hospitality, and to Susanne for some unforgettable dinners;
- Eugene Koonin for his invitation to visit the NCBI;
- Erik Sonnhammer for discussions and several good ideas;
- Olaf Nielsen, Department of Genetics, University of Copenhagen, for instigating a collaboration project which has meant a lot to me, even though the tangible results have so far been few (I'll take a look at *S. pombe* signal peptides real soon now, I promise);
- Countless SignalP users, who at last succeded in convincing me that what I had done was practically useful after all;
- Course students in Copenhagen and Stockholm, who forced me to study and investigate the foundations of bioinformatics;
- The "Thursday Breakfast Club" at Pussy Galore's, for being such a pleasant lot;
- My mother, Elisabeth Westerstråhle, for many warm meals served at late hours during my thesis panic phase;
- and, of course, the Danish National Research Foundation for the funding.

Publications

Papers included in this thesis

I Henrik Nielsen, Jacob Engelbrecht, Gunnar von Heijne, and Søren Brunak.
Defining a similarity threshold for a functional protein sequence pattern: The signal peptide cleavage site.
PROTEINS: Structure, Function, and Genetics, **24**, 165–177, Feb. 1996.

II Henrik Nielsen, Søren Brunak, Jacob Engelbrecht, and Gunnar von Heijne.
Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites.
Protein Engineering, **10**, 1–6, Jan. 1997.

III Anders Gorm Pedersen and Henrik Nielsen.
Neural network prediction of translation initiation sites in eukaryotes: Perspectives for EST and genome analysis.
In *Proceedings of the Fifth International Conference on Intelligent Systems for Molecular Biology* (*ISMB* **5**), AAAI Press, Menlo Park, California, pp. 226–233, June 1997.

IV Henrik Nielsen, Søren Brunak, Jacob Engelbrecht, and Gunnar von Heijne.
A neural network method for identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites.
International Journal of Neural Systems, **8**, 581–599, Oct. 1997.

V Henrik Nielsen and Anders Krogh.
Prediction of signal peptides and signal anchors by a hidden Markov model.
In *Proceedings of the Sixth International Conference on Intelligent Systems for Molecular Biology* (*ISMB* **6**), AAAI Press, Menlo Park, California, pp. 122–130, June 1998.

VI Olof Emanuelsson, Henrik Nielsen, and Gunnar von Heijne.
ChloroP, a neural network-based method for predicting chloroplast transit peptides and their cleavage sites.
Protein Science, **8**, 978–984, May 1999.

In the text, the above papers will be referred to by their Roman numerals.

World Wide Web servers

Three methods developed during this project are made directly available to the scientific community as WWW prediction servers:

SignalP: Signal peptides and their cleavage sites in amino acid sequences from Gram-positive bacteria, Gram-negative bacteria, and eukaryotes.
`http://www.cbs.dtu.dk/services/SignalP/`

NetStart: Translation start in vertebrate and *Arabidopsis thaliana* DNA.
`http://www.cbs.dtu.dk/services/NetStart/`

ChloroP: Chloroplast transit peptides and their cleavage sites in plant proteins.
`http://www.cbs.dtu.dk/services/ChloroP/`

Additional papers

The papers below have been written or co-authored by me during the Ph.D. project, but are not considered part of the project:

- Per Klemm, Suxiang Tong, Henrik Nielsen, and Tyrrell Conway.
  The *gntP* gene of *Escherichia coli* involved in gluconate uptake.
  Journal of Bacteriology, **178**, 61–67, Jan. 1996.

- Henrik Nielsen, Søren Brunak, and Gunnar von Heijne.
  Machine learning approaches to the prediction of signal peptides and other protein sorting signals (REVIEW).
  Protein Engineering, **12**, 3–9, Jan. 1999.

- Rune B. Lyngsø, Christian N. S. Pedersen, and Henrik Nielsen.
  Metrics and similarity measures for hidden Markov models.
  Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology (*ISMB* **7**), to appear, Aug. 1999.

- Susana Cristóbal, Jan-Willem de Gier, Henrik Nielsen, and Gunnar von Heijne.
  Competition between the Sec and TAT protein translocation pathways in *Escherichia coli*.
  EMBO Journal, accepted for publication, 1999.

- Anders Gorm Pedersen, Henrik Nielsen, and Søren Brunak.
  Statistical analysis of translation initiation sites in eukaryotes: patterns are specific for systematic groups.
  Manuscript in preparation, 1999.

- Pierre Baldi, Søren Brunak, Yves Chauvin, and Henrik Nielsen.
  Assessing the accuracy of prediction algorithms for classification: an overview.
  Manuscript in preparation, 1999.

Abbreviations

3D	three-dimensional
aa	amino acids
cTP	chloroplast transit peptide
HMM	hidden Markov model
mTP	mitochondrial transit peptide
NN	neural network
SP	signal peptide
SPase	signal peptidase
SRP	signal recognition particle
TM	transmembrane
WWW	World Wide Web

Chapter 1

Introduction

1.1 Bioinformatics as a field

This project belongs to a field dedicated to the search for correlations between the primary structure (*i.e.*, the monomer sequence) of biological macromolecules and their three-dimensional structure and functional properties. During the latest years, common usage in the scientific world has dubbed this field “bioinformatics.” Earlier, the term “bioinformatics” has also been used as a synonym for “computational biology”, which actually covers a wider area including subjects as diverse as environmental models, population genetics, dynamic physiology models, and medical image processing. A more precise but slightly awkward term could be “computational biological sequence analysis.”

The “ultimate” goal of biological sequence analysis is the prediction of form and function of a whole organism from the nucleotide sequence of its genome. In practice, the problems addressed present only modest aspects of this, but they have nevertheless proved very difficult to solve. Two of the most well known problems within the field are the prediction of protein 3D structure from amino acid sequence, and gene finding, *i.e.*, the prediction of protein coding regions in genomic DNA.

Two factors contributing to the rapid growth of biological sequence analysis in the later years are the accumulation of biological data and the technical advances in computer hardware, software, and communication. As an example, the nucleotide database GenBank has been growing exponentially since 1983, with a current doubling period of approximately 15 months (Baldi & Brunak, 1998).

This project does not belong to protein structure prediction or gene finding, although an aspect of gene finding will be addressed (start codon prediction, see below). Instead, it concentrates on an aspect of protein sorting or prediction of protein location—a no less important but not quite as intensively studied area of biological sequence analysis. More specifically, it will concentrate on the prediction of one of the most fundamental questions in the determination of protein location: the presence or absence of a secretory signal peptide.

Machine-learning techniques such as hidden Markov models (HMMs) and neural networks (NNs) are ideally suited for pattern recognition tasks where relatively large amounts of data are present and where the patterns are “noisy” and not easily described by a compact set of rules. The fundamental idea behind these approaches is to learn to discriminate automatically from the data, using experimentally verified examples, which most often are extracted from the large public sequence and structure databases. In chapter 2, the most important methods used in bioinformatics will be reviewed, with an emphasis on NN and HMM methods. The chapter also contains a discussion of the “geometry” of the problems addressed in bioinformatics, and the nature of prediction or generalisation.

1.2 Prediction of protein sorting

Subcellular protein sorting, *i.e.*, the processes through which proteins are routed to their proper final destination within a cell, is a fundamental aspect of cellular life. In many cases, sorting depends on “signals” that can be identified already by looking at the primary structure of a protein. Thus, targeting to the secretory pathway, to mitochondria, and to chloroplasts normally depends on an N-terminal presequence or targeting peptide that can be recognised by receptors on the surface of the appropriate organelle. After targeting, membrane-embedded translocation machineries ensure the delivery of the protein to the interior of the organelle. In chapter 3, the targeting, translocation, and processing apparatus of the protein secretory pathway is reviewed.

By definition, the cell can recognise all kinds of protein sorting signals with almost 100% selectivity and specificity—the level of mis-sorting *in vivo* seems to be very low, although this aspect of the problem has not been much studied. Given that the sorting signals mentioned above seem to be, at least to a good approximation, defined by a linear, N-terminal stretch of the polypeptide, it would appear that we should be able to devise sequence-based methods that can recognise these signals with an efficiency approaching that of the cell itself. If such methods can be developed, they will clearly be of major use for, *e.g.*, genome analysis and automatic database annotation; at the same time, these massive data analysis tasks necessitate very accurate prediction methods.

Prediction of sorting signals has a long history starting by the early work on secretory signal peptides (von Heijne, 1983; McGeoch, 1985; von Heijne, 1986b), but it is only with the application of modern machine learning techniques that we seem to be approaching the necessary accuracy levels for actually using the predictions for annotation. Chapter 4 is a critical review of existing applications of bioinformatics methods to signal peptides and other problem within the protein sorting field.

1.3 History and scope of this project

For this project, two goals should be regarded as equally important: the characterisation and prediction of secretory signal peptides, and the investigation of the possibilities of biological sequence analysis by using different computational methods on the same biological problem. In order to obtain generally applicable results regarding the computational methods, I have attempted to define the biological problem as widely as practically feasible within the time limitations of a Ph.D. project. Thus, I have not limited the data to one group of organisms, but used sequences from both prokaryotes and eukaryotes. Although prediction of secretory signal peptides is the central problem, prediction of another protein sorting signal (the chloroplast transit peptide) and a nucleotide sequence pattern (the eukaryotic start codon) are included to show the generality of the methods.

For any bioinformatics project, the selection of the data set is critical. The positive and negative examples need to be defined, errors should be avoided as far as possible, and groups of too closely related homologues should be reduced, as described in chapter 5. While constructing the data set for signal peptide prediction, the choice of similarity measure and threshold for homology reduction turned out to be non-trivial, since earlier work in this field had concentrated on protein structure prediction, and it was not obvious that the same principles could be applied to a functional motif such as the signal peptide cleavage site. This prompted the investigation of similarity scores and alignment methods which is published in paper I and discussed in chapter 5.

The signal peptide prediction project resulted in the neural network-based method SignalP, which is described briefly in paper II and in more detail in paper IV. SignalP predicts presence of signal peptides and location of their cleavage site, and it exists in three versions, specific for eukaryotes, Gram-negative bacteria, and Gram-positive bacteria. In December 1996, SignalP was implemented as server, publicly accessible on the World Wide Web or via e-mail,[1] and it has been very heavily used. To date (April 1, 1999), the WWW server has received 70590 sequences, and the mail server has received 261021 sequences in (exactly) 2000 mails (in-house usage not included). Currently (January–March 1999), an average of 384 sequences are processed by the servers per day. Furthermore, SignalP has been licensed to approximately 20 academic and commercial sites worldwide.

The high usage of the server has resulted in a large number of citations to the accompanying publication, paper II, which to date has been cited 225 times (see table 4.1 on page 43). By December 1997, the paper had collected enough citations to make it to a list of "the red hot research papers of 1997," published in the March/April 1998 issue of the *ScienceWatch* newsletter from ISI (Institute for Scientific Information). This was acknowledged by *Protein Engineering*, the journal that published paper II, and they marked the event by printing a commissioned review on protein sorting prediction (Nielsen *et al.*, 1999) accompanied by an editorial comment.

In paper V, a hidden Markov model version of the signal peptide prediction (SignalP-HMM) is described. This comparative study of two machine learning technologies shows advantages and drawbacks of the HMM approach. First, it is able to improve the discrimination between signal peptides and signal anchors—uncleaved N-terminal transmembrane α-helices of type II membrane proteins—otherwise a weak point in neural network-based SignalP performance. On the other hand, prediction of the precise location of the cleavage site is not as good. Second, the HMM provides an assignment of the regions within the signal peptide. This assignment showed an unexpected two-peaked distribution of the length of the hydrophobic regions, discussed in detail in section 6.2.1.

ChloroP[2] is the equivalent of SignalP for predicting chloroplast transit peptides. It

[1] Addresses: `http://www.cbs.dtu.dk/services/SignalP/` and `signalp@cbs.dtu.dk`

[2] `http://www.cbs.dtu.dk/services/ChloroP/`

has been developed primarily by Olof Emanuelsson, the first author of paper VI, as an extension of a Master's degree project (Emanuelsson, 1998) under supervision by Gunnar von Heijne and myself. The construction of ChloroP has been inspired by SignalP, but at certain points, the nature of the problem and the availability of data has prompted different choices, both regarding data validation (see chapter 5) and post-processing of neural network output (see chapter 6).

A difficulty for prediction of signal peptides—or any other N-terminal sorting signals—is that the position of the N-terminus of the preprotein rarely is known experimentally. This is particularly troublesome when using genomic data, where protein coding regions are predicted by gene finding algorithms containing numerous potential sources of error. Wrong start codon assignments can produce false negatives, since the resulting sequence may either contain only a partial signal peptide sequence, or a signal peptide plus a stretch of irrelevant amino acid sequence (derived from DNA which is untranslated *in vivo*) without signal peptide characteristics.

For expressed sequence tags (ESTs) the problem can be even worse, since it is very difficult to decide whether a given sequence includes the start codon at all—it might be entirely untranslated, or correspond to an internal stretch of a protein. The last case can also produce false positive predictions, since non-cytoplasmic ends of transmembrane helices are often rather similar to signal peptide cleavage sites, and the SignalP networks have never been trained to avoid signal peptides here.

Therefore, it would be desirable to have a method which, given a nucleotide sequence, would provide a prediction of both ends of a SP, *i.e.*, the start codon and the cleavage site. Such a method does not exist yet, but a partial solution would be a score describing the probability that any given triplet is the start codon. To this end, we have developed a neural network-based method for start codon prediction in eukaryotes, NetStart (paper III). It is trained to recognise the start codon AUG against all other AUG triplets in the mRNA sequence. It performs this task by using both local context—the Kozak box (Kozak, 1984)—and long-range context in the form of implicit reading frame detection.

Chapters 5 and 6 present and discuss the results achieved within this project. Rather than repeating the information already present in the results and discussion sections of papers I–VI, these chapters focus on comparisons of the approaches taken for predictions of signal peptides, start codons, and chloroplast transit peptides. Furthermore, I discuss possible alternatives and present some results or discussions not in the papers. Chapter 5 is dedicated to the extraction and redundancy reduction of the data sets, while chapter 6 describes the construction of the prediction methods, including training, post-processing, and performance evaluation, and some examples of applications. Finally, chapter 7 discusses the future of protein sorting prediction and bioinformatics in general.

Chapter 2

Machine learning methods in bioinformatics

A wide variety of methods are used in bioinformatics. In this chapter, I will focus on the sequence analysis algorithms, *i.e.*, algorithms that treat macromolecules as strings of symbols, written in the nucleotide or amino acid alphabet. In doing this, I ignore methods that treat the molecules as three-dimensional objects, *e.g.* molecular modeling and molecular dynamics—it is a matter of definition whether these are part of the bioinformatics field.

Among the most important prerequisites for recognition of patterns are the sequence alignment methods, *i.e.*, the methods for comparing sequences, measuring their similarity, locating the matching residues, and identifying the most similar regions. Foundations in this field were established with the global (Needleman & Wunsch, 1970) and local (Smith & Waterman, 1981) pairwise alignment algorithms, and among the most important later contributions is the theory of local alignment scores and substitution matrices, which makes it possible to assess statistical significance of sequence similarities (Altschul, 1991; Altschul & Gish, 1996).

The development and refinement of pairwise and multiple alignment methods is a very important part of bioinformatics today. Key problems include refinement of scoring systems and profile alignments, aimed at improving detection of remote homologues. Functional or structural properties of a sequence are very often inferred simply by aligning it to a closely related sequence of known structure and/or function; but if the sequence similarity is too low, alignment is not sufficient for prediction. In section 5.2 and paper I, I discuss the question of how strong an alignment must be before it is safe to infer the presence of a functional feature—*in casu,* a signal peptide cleavage site—from an alignment. The focus in this chapter will be on methods which do not presume homology between sequences and therefore can be used in those cases where there are no sufficiently close homologues to rely on alignment-based inference.

In molecular biology, a sequence pattern is traditionally given as a *consensus sequence*, *i.e.*, a sequence specifying only the most frequent amino acid or nucleotide at each position. This can be generalised to pattern descriptions (*regular expressions*) that list all allowed letters at each position, and also may specify that certain positions can be skipped. The drawback of regular expressions is that quantitative relationships cannot be represented: a certain sequence either fits the pattern or not. A sequence constructed by choosing the *least* frequent of the allowed letters at each position cannot be distinguished by the consensus sequence.

Conventional expert systems work with numerical features of the entities involved. In the world of sequence analysis, this means that sequences should be described by a limited number of features, which supposedly correlate with the properties of interest. One of the most used amino acid features is the hydrophobicity, which summed over a window of a certain number of positions, can be used to predict the number and positions of transmembrane helices in a membrane protein (von Heijne, 1992). Any feature-based method involves decisions concerning the choice of features and the scale by which they are measured. Even for such a simple feature as hydrophobicity, there is a wide variety of scales based on different measurements or assumptions.

Other features include charge or volume of amino acid residues, or DNA structural parameters such as bendability or stacking energy. These may be calculated per position, per window, over a certain region defined by a set of rules, or over the entire sequence. For example, the length or the amino acid composition of a region or a whole protein could be treated as numerical features.

Some feature-based methods will be described in chapter 4, but in this chapter I will focus on methods where the sequences themselves are used as input data. This provides the possibility for a *data-driven* approach, where the prediction method is calculated directly from the examples without *a priori* assumptions about the biological mechanisms represented by the sequence patterns.

2.1 Position-weight matrices

Weight matrix methods have been widely used for pattern recognition in sequences (*e.g.*, Staden, 1984; von Heijne, 1987; Hengen *et al.*, 1997; Stormo & Fields, 1998). Briefly described, the procedure is to align a number of sequences by a particular site, and count occurrences of letters (amino acids or nucleotides) at each position of a window containing this site. The counts are normalised by dividing with the expected number of letters (according to a background distribution), and the weights are found by taking the logarithm of the normalised count:

$$w_{a,p} = \ln \frac{N_{a,p}}{\langle N_a \rangle} \tag{2.1}$$

where a denotes letter (amino acid or nucleotide) and p denotes position. The expected number, $\langle N_a \rangle$, is the average frequency of each letter times N, the number of sequences in the sample. The results is a *weight matrix*, where the number of weight values is the number of positions in a window times the number of letters in the alphabet (4 for nucleotides and 20 for amino acids).

A normalised count is also referred to as an *odds ratio*, because it is an expression of the chance of finding a letter at a specific position, relative to the chance of finding it anywhere; and the weight matrix is therefore also known as a *log-odds* matrix. The reason for using a logarithmic transformation is that it makes it possible to add the weights rather than multiply them when using the matrix for testing new sequences.

The weight matrix can be used to scan any sequence with a moving window, while looking up the weight corresponding to the letter at each window position and calculating the sum of the weights. This gives the *weight matrix score*, which is a measure of the goodness of fit for that particular window to the sites used in calculating the matrix—a positive weight is evidence for the presence of a site, a negative weight is evidence against it. The weight matrix score for position i in the sequence is calculated as the sum of the weights at each window position:

$$S_i = \sum_{j=i-l}^{i+r} w_{a(j),j-i} \tag{2.2}$$

where l and r are the left-hand and right-hand window sizes, respectively, and $a(j)$ is the letter at position j in the sequence. If the score is larger than a certain cutoff, the window is predicted to belong to the class of sites used for constructing the matrix.

If a certain letter is never observed at a certain position, the corresponding weight cannot be calculated, because the logarithm is undefined. Actually, this situation is only the most extreme instance of the wider problem of sampling errors: the amino acid or nucleotide distributions are estimated from a limited number of examples, and this tends to overestimate the deviation from a random distribution. The solution is *regularisation*: counteracting the sampling noise by modifying the distribution towards the background. In practice, this is done by adding *pseudocounts* to the observations before calculating the weights.

There are several approaches to determining the number of pseudocounts (Henikoff & Henikoff, 1996). The simplest one is to add a constant number of pseudocounts at each position corresponding to either a uniform distribution or the observed background distribution. It is also possible to use a *position-specific* regularisation, where the number of pseudocounts added depends on the conservation: at a very conserved position, the absence of a certain letter is probably significant, so only a low pseudocount value should be added. In more advanced methods, not only the *number* but also the *distribution* of pseudocounts depend on the observed letters; this can be done with a substitution matrix (as used in alignment methods) or a Dirichlet mixture, *i.e.*, a collection of alternative background distributions that are mixed according to how similar they are to the observed pattern (Durbin *et al.*, 1998, section 11.5).

2.2 Neural networks

Artificial neural networks are computational models capable of solving non-linear problems.[1] The name of these models derives from the fact that they originally were used as models of biological neurons and their interactions in the living brain, but in most contemporary applications of neural networks, the artificial neurons are often so simplified and generalised that the connection to neurobiology is merely historical and terminological. Artificial neural networks have been used for many biological sequence analysis problems (for reviews see Hirst & Sternberg, 1992; Presnell & Cohen, 1993; Wu, 1997; Baldi & Brunak, 1998).

[1] Unless otherwise stated, this section refers to Hertz *et al.* (1991) and Baldi & Brunak (1998).

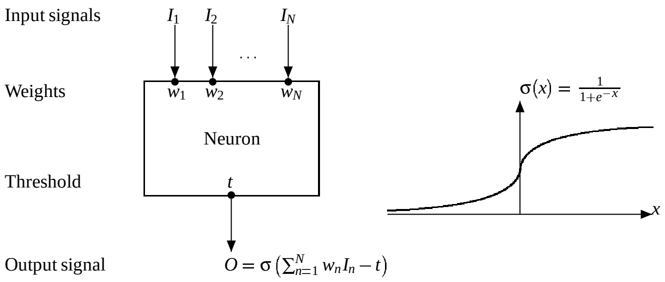

Figure 2.1: A formal neuron, showing input values, weights, threshold (bias), and sigmoid activation function.

Briefly, a neural network can be described as an interconnected assembly of simple computational units, the "formal neurons." A unit receives a number of *inputs*, multiplies each by a specific *weight* (a number representing the strength of the connection from the previous unit), and outputs a non-linear function of the weighted sum. These computational units can be combined in a wide variety of ways, referred to as the *architecture* of the network. The vast majority of neural network applications in biological sequence analysis have used the layered feed-forward network architecture, where information flows in one direction only—*i.e.*, no units receive their own output as input, directly or indirectly.

2.2.1 Artificial neurons and feed-forward networks

The output produced by an artificial neuron (see figure 2.1) can be described as:

$$O = \sigma\left(\sum_{n=1}^{N} w_n I_n - t\right) \tag{2.3}$$

where I_n are the inputs and O is the output. The weights belonging to the neuron have been denoted w_n, and t is a *threshold* or *bias* parameter.

The function σ is called the *activation function*, and is typically a sigmoid function such as:

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{2.4}$$

which gives values between 0 and 1 (see figure 2.1). The precise form of the sigmoid function is not important, but it has to be non-linear if non-linear mapping is required (see section 2.4), and it has to be differentiable in order to make back-propagation learning possible (see section 2.2.3).

The simplest neural network—the one-layer feed-forward network—is merely one or more computational units sharing the same set of inputs. More interesting properties can be obtained by inserting one or more additional layers of computational units between the input and output layer. These are called *hidden layers*, since their outputs are only sent to other units and are not visible outside the model. A network with one hidden layer (*i.e.*, a two-layer network, see figure 2.2) is represented as follows, using

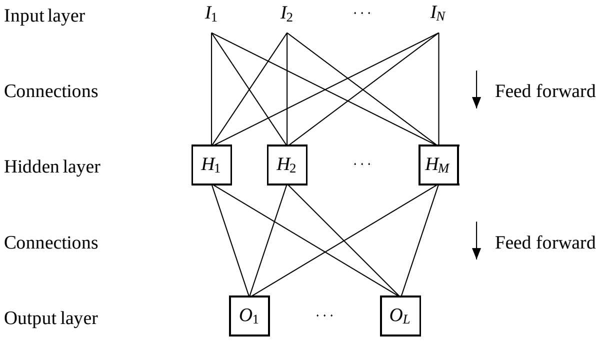

Figure 2.2: A two-layer network.

equation (2.3):

$$H_j = \sigma(\sum_{k=1}^{N} w_{jk} I_k - t_j); \quad j = 1, \ldots, M \tag{2.5}$$

$$O_i = \sigma(\sum_{j=1}^{M} w_{ij} H_j - t_i); \quad i = 1, \ldots, L \tag{2.6}$$

where H_j is the output value of hidden unit j, O_i is the output value of output unit i, and w_{ij} is the weight of the connection from unit j to unit i.

2.2.2 Encoding sequence data for neural networks

A crucial point in all neural network applications is to find a good *representation* of the data. Since the computational units take numerical inputs, sequences must be encoded as sets of numbers. These numbers can be features extracted from the single positions, regions, or an entire gene or protein. Often data are presented as a *moving window* containing the position to be classified and a number of "context" positions to the left and right, exactly like the windows used for position-weight matrices, cf. equation (2.2) on page 7.

Each position in the window must be represented by one or more input values. In some cases, physico-chemical properties such as hydrophobicity, volume, and surface area of the residues are used for encoding, this has for example been used for signal peptide prediction (Schneider & Wrede 1993, see section 4.2.3 on page 45). Several properties can be used together, so that each position in the sequence is represented by a vector of numbers. However, the use of a property vector includes a hypothesis about which properties of the amino acids are relevant to the problem.

For a more data-driven approach, where the definition of features is left to the network itself, the standard choice is *sparse encoding*, where each symbol in an alphabet of k letters is represented by k input values, of which one is "on" and the rest are "off." An amino acid residue, *e.g.*, can be represented by a 1 and nineteen 0's. This approach lets the network extract all relevant information from the raw sequences by itself—provided that the data set is good enough—and it is the only encoding scheme that does not imply differences in the pairwise distances between the symbols. The drawback is the large number of parameters required in the model. When a large alphabet is used, *e.g.* the twenty amino acids, the number of inputs is large, and a large number of training examples are needed to train the network properly (cf. the discussion in section 2.5).

The simplest feed-forward neural network with sparse encoding is actually nothing more than a single unit receiving one weight value from each of the positions in a sequence window. This is virtually identical to the position-weight matrix—the important difference being that the weights of the neural network are found by *training* rather than statistical analysis.

2.2.3 Training the network

The process of finding the appropriate weight values in a neural network is known as *training* or *learning*. The training data are a number of examples of input patterns with corresponding correct outputs (called the *targets*). Initially, the network starts with random weight values, producing totally irrelevant output values, but the weights are gradually adjusted to minimise the difference between outputs and targets.

The most widely used training algorithm for feed-forward networks is termed the "back-propagation of errors", or just *back-propagation* for short. The objective is to minimise an error function describing the difference between outputs and targets. Often, the sum of squared differences, also known as the *quadratic error measure*, is used:

$$E = \sum_{\alpha,i} (O_i^\alpha - T_i^\alpha)^2 \tag{2.7}$$

where the T's are the targets and the O's are the actual output values. This error function implicitly assumes a normal distribution of target values, and for classification problems, other alternatives are superior (Baldi & Brunak, 1998).

Applying the gradient descent algorithm, the derivative of the error function with respect to all the weights is calculated, and the weights are adjusted in the direction of the slope of the error function:

$$\Delta w = -\eta \frac{\partial E}{\partial w} \tag{2.8}$$

where η is the *learning rate*, indicating how much the weights are changed at every update. A small η gives slow learning, but a large η can make the algorithm diverge or oscillate.

If the error measure and the activation function of the neurons are differentiable, $\partial E/\partial w$ for each weight can be expressed as a function of output, target, and present weight values. The back-propagation algorithm is easily applicable in all feed-forward architectures, whether or not units are organised into fully connected layers.

Like all optimisation procedures, back-propagation can become stuck in a *local minimum*. This means that the set of weights has reached a point which is not the global minimum of the error function, but where all small weight changes will increase the error function. One way to decrease the risk of becoming stuck in a local minimum and make the training more effective is to add a stochastic element to the training procedure; *e.g.* by updating the weights after each training pattern and choosing the patterns in random order for each training cycle. Another variation, designed to reduce the oscillations during learning and the risk of getting stuck in local minima, is the momentum term: at each weight update, a constant fraction of the previous weight change is added to the currently computed weight change.

An alternative to gradient descent is the “Monte Carlo” procedure which does not compute optimal weight changes but instead, in each step, changes a weight in a random direction. The effect on the error function is calculated, whereupon the random weight change is accepted or discarded according to a set of rules. An extension of this approach is to use a *genetic algorithm*, where a “population” of networks undergo random weight changes (*mutations*), and the best-performing networks are selected for the next generation, possibly after some kind of recombination (see section 4.2.3 on page 45 for an example).

Regardless of how the weights are updated, the issue of where the training should be stopped is non-trivial. Training should not necessarily proceed to the global minimum: this point is per definition optimal for the training set, but that may not be the case for an independent data set. If performance during training is measured both on the training set and on an independent test set, a typical observation is that both training and test set error decrease at first; but after a certain point, test set error starts growing again. This is known as *overtraining* or *overlearning*, and can occur if the number of weights is large compared to the number of examples. What happens is that the effective number of free parameters increases in order to learn single examples from the data—see section 2.5 on page 17 for a further discussion of this.

Often, the training is simply stopped at a point where test set performance is optimal (*early stopping*). The power and pitfalls of this approach are discussed in section 6.1. There are also some alternative methods to avoid overtraining by reducing the number of connections either during or after training, known as *weight decay* and *pruning*, respectively.

2.3 Hidden Markov Models

Hidden Markov models (HMMs) are probabilistic models capable of representing a probability distribution over a set of sequences of symbols.[1] Although originally developed for speech recognition, they have been found useful in a wide range of bioinformatics applications. An HMM for biological sequence analysis consists of a number of *states* that are connected by *transitions*; and associated with each state is a probability distribution over the 4 nucleotides or the 20 amino acids, and a probability distribution over the possible transitions from that state. It is often useful to think of an HMMs as generative models that can “emit” sequences by following the transitions from state to state, and in each state emit a nucleotide or an amino acid, both according to the probability distributions. In analogy with the neural network terminology, the pattern of states and transitions is often referred to as the HMM *architecture*.

What is “hidden” in an HMM is the sequence of states used in a pass through an HMM—the *path*. In contrast to a Markov chain, where each state emits one unique symbol, it is generally not possible to infer the path from a sequence generated by an

[1] Unless otherwise stated, this section refers to Durbin *et al.* (1998) and Baldi & Brunak (1998).

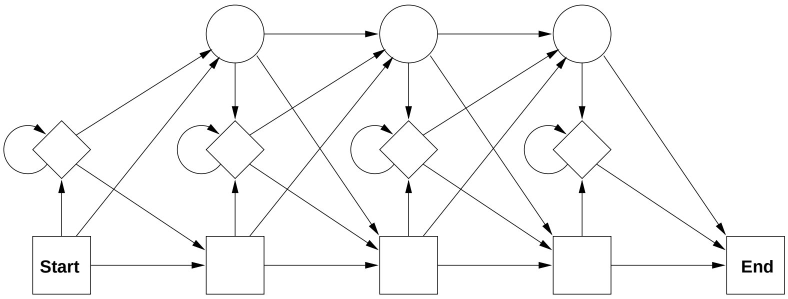

Figure 2.3: The architecture of a profi le hidden Markov model. The squares are the match-states, the diamonds are the insert-states and the circles are the silent delete-states.

HMM. However, one can calculate the most *probable* path for a given sequence, or the total probability that it was generated by the model at all (summed over all possible paths). There are simple dynamic programming solutions to these problems, known as the *Viterbi* and *forward* algorithms, respectively.

In computational biology the most commonly used HMM architecture type is the *profile HMM* which has a structure inspired by sequence profiles (multiple gapped sequence alignments; Gribskov *et al.*, 1987). It contains a number of "main states" corresponding to the typical length of the modeled sequences, and some "insert" and "delete" states representing insertions and deletions in the individual sequences with respect to the profile, see 2.3. Finding the most probable path for a sequence in a profile HMM is equivalent to finding the optimal alignment between the sequence and the profile. The delete states are silent, *i.e.*, they do not emit any nucleotide or amino acid. It is in principle always possible to eliminate silent states and make an equivalent HMM without them; but for example in the case of the profile HMM, this would make the architecture much more complicated.

In general, there are two ways an HMM can be used for classification; employing either the assigned path or the probability of a sequence. Assigning a path for a sequence through an HMM is also called *decoding*, see more details below. The probability of a sequence given a model will generally depend on the length of the sequence, so it should be compared to the probability of the same sequence given a "background" or "null" model. Often, the logarithm of the ratio of these two probabilities—the log-odds score—is calculated, for example when using a profile HMM of a protein family to search a database for new members of the family. With the log-odds score, a profile HMM with no insert or delete states is exactly equivalent to the position-weight matrix described in section 2.1.

The estimation of parameters (emission and transition probabilities) can also be done in a simple way analogous to the calculation of weights in the weight matrix, if the "correct" path of each sequence is given in the data. This is the case, *e.g.*, if a profile HMM is based on an existing multiple alignment.

If the paths are not given, the HMM must be trained by an iterative procedure. One possibility is the *Viterbi* training, where the most probable path for each sequence is used to estimate parameters (according to the equations above) for each cycle of the iteration. Alternatively, the parameters can be estimated by a probability-weighted sum over all possible paths that could have generated a sequence. This is known as *Baum-Welch* training and is an application of the expectation maximisation algorithm, optimising the probability of the training data given the model.

Regardless of the training algorithm, sampling error due to small data sets can be a problem. To remedy this, the same regularisation approaches described for the weight matrix method (page 7) can be applied. When training unlabeled data, it is also important to choose sensible starting values for the probabilities, because both the Viterbi and the Baum-Welch algorithms easily get stuck in local minima.

The HMM framework is much more general than the profiles, and several non-profile HMM architectures have been used in bioinformatics. Our HMM for discriminating between signal peptides and signal anchors, described in section 6.2, is an example of a *branched* architecture, where a sequence can be generated by a number of parallel paths with different properties. For recognition of features that can be repeated a number of times, a *cyclical* architecture is often used. One example is eukaryotic gene finding, where a gene can contain a varying number of introns: this can be modeled by placing an exon and an intron submodel in a loop, so that they be repeated any number of times (Krogh, 1997). In the same way, a model for transmembrane α-helix proteins can be made as a loop containing submodels for the in-to-out and out-to-in transmembrane helices (see section 4.4 on page 49). Cyclical HMMs have also been used for finding periodicities in DNA (Baldi *et al.*, 1995).

For decoding an HMM, the most probable (Viterbi) path is not the only possible choice. By summing over all possible paths, it is possible to calculate the *posterior state probabilities*, *i.e.*, the probability distribution over all possible states for each letter in the sequence. The assignment obtained by choosing the most probable state for each letter is not necessarily the same as the most probable path; in some cases, it may even be non-grammatical, *i.e.*, not allowed by the transitions in the model. However, it may be the best measure in applications where a specific part of the model is in focus.

In many HMM applications, it makes sense to group the states using a limited number of *labels*; *e.g.*, when building a model for predicting transmembrane protein topology, the objective will typically be a classification of each amino acid into transmembrane, cytoplasmic, and extracytoplasmic. In this context, the posterior state probabilities can be added for all those states that have the same label, giving *e.g.* the probability of being transmembrane for each residue in the amino acid sequence. This approach also provides a third decoding method: instead of finding the most probable *state path*, one can add the probabilities of all those paths that produce the same sequence of labels and thereby calculate the most probable *labeling* of the sequence (Krogh *et al.*, 1994). In a model with many parallel branches this can produce rather different results, *e.g.*, if there are many paths that model a signal peptide but only one path that models a non-signal peptide, a sequence might easily be classified as non-signal peptide by the most probable path, but as signal peptide if the probabilities of all signal peptide branches were added.

2.4 The concept of non-linearity

In bioinformatics, problems are often referred to as being “linear” or “non-linear” without specifying exactly what is meant. In this section, I will attempt a definition and discuss examples and implications of non-linearity in sequence classification problems.

A weight matrix or a feed-forward neural network performs a *mapping* between input and output examples. In the applications discussed here, this mapping is interpreted as a classification, *i.e.*, a prediction whether the input pattern belongs to a specific category. This can be described in terms of *input space*: with N input values, any input example can be represented as a point in the input space of N dimensions. The mapping is linear, if and only if all input examples belonging to one category can be separated from the rest by a hyperplane of $N-1$ dimensions. Weight matrices and neural networks without hidden layers assume that the mapping between input and output is linear, *i.e.*, that the input values influence the output independently of each other, and that their effects are additive.

To avoid misunderstandings, I should stress that a non-linear *mapping* is not the same as a non-linear *function*: the sigmoid function (equation (2.4), figure 2.1) is not linear, but may still be used in a linearly separable classification. If we use the definition of non-linearity that "small changes in input lead to large changes in output," any mapping from a continuous variable to a number of discrete classes is non-linear, because there is a discontinuity at the threshold value for changing category assignment.

The logical XOR (exclusive or) function is the simplest example of a non-linear mapping. It has the value "true" if its two inputs are unequal, but "false" if they are equal. Thus, the effect of one input depends on the value of the other input. In the plane defined by the two input values, the border between "true" and "false" points cannot be drawn as a straight line.

Let us consider a biological illustration of an XOR situation. In a hypothetical amino acid pattern there are two positions, of which one *but not both* should be arginine: "RA" and "AR" are positive examples, but "AA" and "RR" are negative (where A could be any non-Arg amino acid).

In a statistical analysis of the positive examples from this case, we would measure a *correlation* between the positions: if there is an Arg at the position 1, an Arg is not found at position 2, *i.e.*, the amino acid preference at position 2 depends on the actual amino acid found at position 1. The significance of this interdependence can be tested by the χ^2 statistic in a 20-by-20 contingency table (or 4-by-4 for nucleotide sequences). It can also be measured by *mutual information*: the two positions carry information about each other in the sense that knowing the residue at position 1 increases the chances of guessing the residue at position 2.

In bioinformatics, correlation and non-linearity are often treated as synonymous, but this is not exactly true. Remember that correlation or mutual information is measured on the positive examples alone, while the linearity of a mapping depends on the position of both positive and negative examples. Suppose we did not have any two-arginine examples in our hypothetical data set and therefore did not know how to classify them: the correlation between the positions would be the same, but since all negative examples were arginine-free, it would be possible to separate them linearly. In other words, non-linear separability implies correlation, but the converse is not necessarily true.

Note also that a linear mapping does not exclude *interaction* between positions, as long as the effects are additive. Even in a (strictly linear) weight matrix, a "bad" residue in one part of the window can be compensated by a "good" residue at another position, making the overall sum positive. Only XOR-type mappings are non-linear, OR-type and AND-type mappings are not.

It is important to stress that linearity depends on encoding. It does not really make sense to discuss whether two sets of *sequences* are linearly separable or not; the concept applies to points in a multi-dimensional space. There is no “sequence space” out there—we create it by encoding the sequences. In the examples discussed here, I assume we are talking about sequence patterns as consisting of single positions or moving windows; but if we used a set of features derived from entire sequence regions, the non-linearity might disappear.

Thus, observed non-linearity or correlation in a set of classified sequence windows may come from various sources, both natural and artificial. The possibilities in the list below are not mutually exclusive, but should rather be seen as alternative approaches for explaining or decomposing the non-linearity:

Bad alignment: if a conserved pattern occurs once per sequence, and that pattern occurs at different positions within a multiple alignment, these positions will appear correlated. In the arginine example above, we could actually be dealing with just *one* position with a completely conserved Arg and two neighbour positions where Arg was not allowed, and it could be some type of noise or mistake that had placed the Arg in two different positions in the positive examples.

Missing alignment: as an extension of the previous point, consider a situation where we do not align the sequences but simply count all dipeptides in our data and do statistics on them. If every sequence contains a motif comprised of a Trp followed by an Arg, we would observe more Trp-Arg pairs than expected from the Trp and Arg frequencies, and we would call that a Trp-Arg pair correlation—however, that correlation would disappear if we aligned the motif and calculated frequencies per position. For this reason, overlapping sequence windows used to characterise a region (*e.g.*, all positions within signal peptides) tend to be more non-linear than non-overlapping sequence windows used to characterise a specific site (*e.g.*, the signal peptide cleavage site).

Subgroups: apparent correlation in an alignment may simply be the effect of two (or more) groups of very similar sequences; knowing the residue at one position would then make it possible to predict which group the sequence belonged to, and thereby predict the residue in another position. (Therefore, computations of correlation or mutual information between positions should only be trusted if performed on homology-reduced or appropriately weighted data, see section 5.2). Even without homology in the data set, a non-linear mapping may be the result of mixing examples from two linearly separable mappings. In the Arg situation, the explanation might be that there are two enzymes recognising different patterns, one with Arg in position 1, and another with Arg in position 2.

Oligonucleotide or oligopeptide bias give rise to local correlations: the most obvious example is protein-coding DNA, where the codon usage implies that the distributions of nucleotides in the first, second, and third reading frame positions are mutually dependent. In non-coding DNA, selection for bendability or other structural properties can favour certain di- or trinucleotides. In the Arg situation, Arg–Arg dipeptides might simply be too bulky to fit into the relevant binding site.

Variation in distance between two parts of a pattern can show up as short range correlations: *e.g.*, since the length of the c-region of signal peptides is not constant, windows defined by the cleavage site will contain the C-terminal part of the h-region in varying positions, and this may result in a measurable correlation between these positions. This would disappear if both parts were properly aligned, so in a sense, the distance variation explanation is just another way of stating the bad/missing alignment explanations.

Intermediate optima: too much of a good thing can be bad. In a dynamical system of several proteins and/or nucleic acids, the components should bind each other, but not so tightly they cannot dissociate again. It is possible that the binding energy between two components can be modeled linearly with a weight matrix, but if an intermediate binding energy is optimal for function, function is not linearly separable. In the Arg situation, the explanation could be that exactly one positive charge is needed in this region.

Coevolving sites: Long distance correlations can result from interactions in three dimensions between non-adjacent parts of a chain, *e.g.* nucleotide pairings in RNA secondary structure or residue contacts in a folded protein. Substitutions in one position, which would otherwise impair the interaction, can be compensated by substitutions in the interacting position. In a recent, comprehensive analysis of coevolving residues in myoglobin sequences, Pollock *et al.* (1999) found that these were located either close in the three-dimensional structure, or on the surface at large distances where the may play a role in regulating aggregation or quaternary structure.

Neural networks can model any kind of non-linearity in their inputs, provided they are complex enough. An important limitation, however, is that correlations over distances larger than their input windows cannot be represented. To perform a non-linear mapping, a network needs at least one hidden layer of units with a non-linear activation function. Each units only manages a linear part of the problem, but the combined network is, in effect, able to decompose XOR situations into systems of AND and OR rules. The non-linear activation function is crucial; without it, all the hidden units can be reduced away by a simple operation.

Hidden Markov models can handle certain types of non-linearity, depending on their architecture. A profile HMM is easily able to deal with the “soft” non-linearity caused by variations in distance, by using the insert- and delete-states. This problem is handled much more elegantly by the HMM than by a window-based neural network, which has to recognise each distance as a separate entity, and the HMM is not limited by a window size. Local bias-type correlations, such as codon usage in a reading frame, can be represented by higher-order HMMs, where the emission probabilities of a state depends on the symbol selected in the former state(s)—this is often used in DNA applications such as gene finders, but for amino acid sequence it is very rarely used, because the number of probabilities to be estimated for each state (20×20) would require enormous data sets. Subgroup-types of correlations can be modeled by branched HMMs, with each relevant subgroup represented by a branch. Long-range coevolution correlations, however, pose severe problems for HMMs; and for secondary structures in RNA, a different class of probabilistic models is employed (Durbin *et al.*, 1998, chapter 10).

2.5 Generalisation

The interesting question for any prediction method is whether it can *generalise*—*i.e.*, extract a general mapping relating inputs to targets, and thereby give reasonable answers to "new" inputs, *i.e.*, patterns not used to construct, train, or optimise the model. A model that can reproduce its own input is not very interesting in itself from an application point of view, since any database program could do that.

This is not solely a neural network issue. The phenomenon of neural network overtraining described above may seem surprising at first, but the same phenomenon appears whenever a model with many parameters is used to describe noisy data. It is always possible to fit the model to all the data points if a sufficient number of parameters is used, but in this case the model will contain more information about the noise than about the general pattern underlying the data. As an example, if given a set of N different values of an independent variable with corresponding dependent values, it is always possible to find a polynomial of degree $N - 1$ that will go exactly through all the points. But if the data contain noise, the resulting polynomial may take the most outrageous twists and turns to hit each point and therefore give strange values for new data points.

So what is noise? If we assume that there is a "true" mapping between inputs and targets, the noise in the data is the deviation between true and observed values. In a bioinformatics example, the hypothetical "true" mapping would be determined by the molecular machine functioning in the recognition of a motif, and a total knowledge of all details in this machine would make it possible to predict the true values from the sequence—*all other things being equal*. Following this definition, noise should always be expected in biological data sets: it does not necessarily represent errors in sequencing or classification, it could just be the effect of all other things not being equal. The biological categories (*e.g.*, subcellular location) are rarely measured under strictly equivalent circumstances; the precise function of the recognition system and thereby the "true" mapping may vary according to context (cell type, species, subspecies, physiological state, time, place...). Finally, one should not forget the simple rule that in biology there are no rules without exceptions!

To test the generalisation ability of any parametric prediction model, it is therefore necessary to apply it to a *test set* of input patterns not present in the training set, and compare the outputs to the target values of the test set. The test set should not only be non-overlapping with the training set, it should also be independent, *i.e.*, sampled from the "real" sequence space in a way that is not correlated with the sampling of the training set. Exactly what this means for biological sequences is not an easy question to answer—section 5.2 is dedicated to results and discussion within this area.

An important point is that a model with an optimal training set performance does not necessarily have optimal test set performance. The "overtraining" phenomenon happens not only when training a network for a longer time, but also when adding more parameters: the models that are best at reproducing are often not best at generalising. In the case of neural networks, a sufficiently complex network can always be trained to reproduce all training examples correctly, provided that there are no "pathological" examples where identical inputs have different targets—there are even special training algorithms that guarantee 100% training performance, see section 4.2.3 for an example—but the learning of single examples tends to work against the extraction of general features from the data, especially if there is random noise in the data.

Regarding hidden Markov models, it is even simpler to illustrate this point with an example: if you insist on a model with 100% training performance, just do like this: build a branched model with as many parallel branches as there are sequences in the training set, let each branch consist of as many states as there are symbols in the respective sequence, allow only one specific symbol in each state (the observed one, of course), and give each branch the appropriate label. The result would be a computationally rather expensive implementation of a "stupid" lookup table. The generalisation ability of such a thing is guaranteed to be zero, because the probability of generating a sequence that is not in the training set is zero.

In general, more parameters are needed to represent a more non-linear mapping, whether neural networks, hidden Markov models, or other types of models are used. When the available data is limited, however, it may not be possible to estimate all parameters that would be necessary to reproduce the "true" mapping, and noise will take over the training. In other words, the complexity of the model is often a compromise between desired detail on one hand, and data scarcity, noise and bias on the other.

To summarise the comparison of machine learning technologies: as described above, hidden Markov models can deal with certain types of non-linearity in a more explicit and parameter-economic way than neural networks, provided that you know which type of non-linearity to expect. Neural networks, on the other hand, are more general and have a flexible complexity: if the learning rate is low enough, the weights are correlated in the beginning of the training process, and the effective number of free parameters is lower than the nominal number; but utilising this property requires the training to be stopped early (see also section 6.1).

Chapter 3

Molecular biology of protein secretion

Even the simplest possible cell must sort its proteins properly to at least three destinations—inside, membrane, and outside—while a eukaryotic cell contains several membrane-bound organelles and their respective membranes, each having a characteristic complement of proteins.

It should be noted that protein sorting is not only about transporting proteins into or across membranes. A membrane or a membrane-bound aqueous compartment is not necessarily homogeneous, and may contain regions of varying protein composition. Even a bacterium may have an anterior and a posterior end, and as an extreme example, the set of both soluble and membrane proteins found in the synaptic region of a neuron is very different from those of the axon, cell body, and dendritic tree. Nuclear proteins are imported after their synthesis in the cytoplasm, but this sorting event can also happen without any membrane translocation, since there are aqueous pores in the nuclear envelope connecting the cytoplasm with the nuclear matrix.

Schatz & Dobberstein (1996) have classified the membrane translocation processes into "export" systems which transport proteins away from the cytosol, and "import" systems which transport proteins into compartments that are functionally equivalent to, or evolutionarily derived from, the cytosol. The most ubiquitous export system, the initial step of the general secretory pathway, is found in the endoplasmic reticulum (ER) membrane of eukaryotes, the plasma membrane of prokaryotes, inner membranes of mitochondria and chloroplasts, and in the thylakoid membranes of chloroplasts. Import systems are found in the envelopes (*i.e.*, inner and outer membranes) of mitochondria and chloroplasts.

Integration of transmembrane α-helix proteins into the membrane is in many cases accomplished by the translocation system of the general secretory pathway; this is described in section 3.2.

A common feature of most export and import systems is that the signals for translocation are N-terminal cleavable sequences. These sequences are recognised by cytosolic or membrane-bound factors that target them to the appropriate membrane, where translocation and cleavage take place. The characteristics of the secretory (export) signals are described in section 3.3. Import transit peptides of mitochondria and chloroplasts are not similar to secretory signal peptides, but they share some characteristics with each other (see paper VI). Nuclear-encoded proteins destined for the thylakoids of chloroplasts and the intermembrane space of mitochondria are often synthesised with a *composite* signal: the most N-terminal part is an import transit peptide, and next comes an export signal peptide (similar to a secretory signal peptide) that directs a second round of targeting, translocation, and cleavage at the thylakoid membrane or mitochondrial inner membrane.

Entering the general secretory pathway does not necessarily mean secretion. In eukaryotes, proteins translocated to the ER may be retained there or transported to the Golgi, from where they may continue to the lysosomes or the outside. These sorting events happen by budding and fusion of vesicles and do not involve membrane translocation. In Gram-negative bacteria, proteins translocated across the inner membrane are further sorted to periplasm, outer membrane, pili, and outside. These processes are described in section 3.4. On the other hand, secretion does not always follow the general secretory pathway; a number of exceptions are described in section 3.5.

Since this review chapter is written from a bioinformatics point of view, I have included some notes with pointers to sequence data. In cases where families of the molecular components of the protein sorting machinery have been defined, I provide entry names from three databases: PROSITE[1] (Hofmann *et al.*, 1999), a database of protein family "signatures" mostly represented by patterns of the regular expression type (cf. page 6); PFAM[2] (Bateman *et al.*, 1999), a collection of profile HMMs (cf. page 12); or ENZYME[3] (Bairoch, 1999), a classification of enzymes according to the type of their activity. These are certainly not the only resources for protein or gene families, but from those there are links further out along the WWW. In the few cases where a high-resolution three-dimensional structure is known, I have included a pointer to the Protein Data Bank (PDB)[4] entry (Abola *et al.*, 1997).

3.1 The general secretory pathway

The mechanism and the molecular components necessary for targeting a protein to the membrane and translocating it via the general secretory pathway show wide similarities for a large variety of proteins from all three domains of life (eukaryotes, bacteria, and archaea—for reviews, see Jungnickel *et al.*, 1994; Rusch & Kendall, 1995; Pohlschröder *et al.*, 1997).

The entry to the general secretory pathway is controlled by the secretory signal peptide, an N–terminal part of the amino acid chain, which is cleaved off while the protein is translocated through the membrane. Signal peptides (also known as signal sequences) from different organisms are to some degree interchangeable (Benson *et al.*, 1985), although there are statistical differences between their properties.

[1] http://www.expasy.ch/sprot/prosite.html

[2] http://www.sanger.ac.uk/Software/Pfam/, http://www.cgr.ki.se/Pfam/, or http://pfam.wustl.edu/

[3] http://www.expasy.ch/sprot/enzyme.html

[4] http://www.pdb.bnl.gov/

There is no well-defined consensus sequence or sequence motif for signal peptides, but there is a common structure. The most characteristic feature is a stretch of seven to fifteen hydrophobic amino acids called the hydrophobic core or *h-region*. The region between the h-region and the N-terminus of the preprotein is termed the *n-region*. It is typically one to five amino acids in length, and normally carries positive charge. Between the h-region and the cleavage site is the *c-region*, which consists of three to seven polar, but mostly uncharged, amino acids. Close to the cleavage site a more specific pattern of amino acids, known as the $(-3,-1)$-rule, is found: the residues at positions -3 and -1 relative to the cleavage site must be small and neutral for cleavage to occur correctly (von Heijne, 1983, 1985). In bacterial signal peptides, the positive charge in the n-region is often balanced by a negative net charge in the c-region or in the first few residues of the mature protein (von Heijne, 1986a). The variations in signal peptide design and the specificity requirements are described in further detail in section 3.3.

Traditionally, translocation in the general secretory pathway has been described as co-translational in eukaryotes, but post-translational in prokaryotes. This is in agreement with the timing of processes: In bacteria, translation is rapid compared with the rate of translocation, so that proteins may be transported after large parts have already been synthesised or even after translation is completed (Rapoport, 1991). However, post-translational translocation has also been described in eukaryotes, and it has become increasingly clear that many crucial components of the targeting system is found in all domains of life. The general picture is that there are at least two different pathways for targeting in most, if not all, types of cells.

3.1.1 SRP targeting

Co-translational translocation in eukaryotes is dependent on a cytoplasmic protein-RNA complex called SRP (Signal Recognition Particle). Translation can be initiated on a free ribosome in the cytoplasm, but as the signal peptide emerges from the ribosome, it binds to SRP, which prevents folding of the nascent polypeptide chain and arrests the elongation step of translation (Schatz & Dobberstein, 1996). The SRP directs the ribosome complex, including mRNA and nascent protein, to the ER membrane, where the translation resumes. The remaining part of the translation takes place on ribosomes bound to the membrane of the ER, while the protein is translocated across the membrane (Rapoport, 1990).

SRP has an affinity for both signal peptides and ribosomes, and there seems to be a positive cooperativity between signal peptide binding and ribosome binding (Rapoport, 1991). At the ribosome, there is a competition between SRP and another cytoplasmic chaperone, NAC (nascent polypeptide associated complex), which preferentially binds non-secretory proteins (Wiedmann *et al.*, 1994). The role of this competition is to assure specificity in the distinction between secretory and non-secretory proteins. If NAC is depleted, even non-secretory proteins will be targeted to the membrane (Möller *et al.*, 1998).

Mammalian SRP is a complex of six polypeptides and one RNA. The polypeptides are named (after their apparent molecular weight in kDa) SRP 9, 14, 19, 54, 68, and 72; and two of these are organised as heterodimers, SRP9/14 and SRP68/72 (Althoff *et al.*, 1994). In yeast, there is an additional subunit called SRP21, and the homologue of SRP19 is known as Sec65 (Hann *et al.*, 1992). The SRP9/14 dimer is responsible for the elongation arrest (Siegel & Walter, 1986).

The subunit of SRP that binds the signal peptide has been shown to be SRP54. It apparently binds the entire signal peptide, including both the n-region and at least seven amino acids of the mature protein (Rapoport, 1990). SRP54 has a three-domain structure: an N-terminal N-domain, a GTP-binding G-domain, and a methionine-rich M-domain which binds the SRP RNA in presence of SRP19 (Rapoport, 1991; Freymann *et al.*, 1997). Both the G- and M-domains have been suggested as binding the signal peptide, but the most likely candidate for binding site is a hydrophobic pocket of the M-domain containing many methionine residues that might bury the signal peptide h-region (Rapoport, 1990, 1991; Rusch & Kendall, 1995). The total binding site may even be the cleft between the G- and M-domains (Lütcke, 1995).

The RNA part is called 7SL-RNA, or simply SRP RNA. It contains a domain homologous to the Alu repetitive sequences in human DNA; this domain seems to be responsible for the binding of SRP9/14 (Althoff *et al.*, 1994).

Prokaryotic SRP is simpler, apparently consisting of only one 48 kDa protein—called Ffh (for "fifty-four homologue") since it is homologous to eukaryotic SRP54—and a 4.5S RNA (Hartl & Wiedmann, 1993; Danese & Silhavy, 1998). It is still an open question how important SRP targeting is in bacteria, where the post-translational SecB system (see below) seems to be the preferred pathway for most secretory proteins; but bacterial SRP has nevertheless been shown to be required for targeting several transmembrane proteins (Ulbrandt *et al.*, 1997) and certain secretory proteins (*e.g.* β-lactamase, alkaline phosphatase (PhoA), and ribose-binding protein, see Danese & Silhavy, 1998). A common property of the SRP-targeted proteins is a higher hydrophobicity (de Gier *et al.*, 1997; Valent *et al.*, 1998).

In particular, it is a matter of debate whether proteins targeted by bacterial SRP are translocated co-translationally, as they are in eukaryotes. SRP-mediated targeting does not necessarily mean co-translational translocation in all systems—in chloroplasts, an SRP54 homologue (cpSRP54) targets a subset of post-translationally imported nuclear-encoded proteins across the thylakoid membrane (Dalbey & Robinson, 1999). It is also clear that translation arrest cannot occur by the same mechanism as in eukaryotes, since the SRP9/14 subunits and the corresponding domain of the SRP RNA are missing. However, Powers & Walter (1997) have reported experimental evidence for co-translational translocation and translation arrest in *E. coli*.

At the ER membrane, SRP binds a membrane protein termed the SRP receptor (SR). SRP receptor catalyses the release of SRP from the ribosome, a step requiring GTP hydrolysis. After the release, SRP can bind to another ribosome complex, thereby completing one round of what is known as the SRP cycle (Rapoport *et al.*, 1996). SRP receptor is composed of two subunits, SR-α (also known as docking protein, DP) and SR-β. While SR-β is a transmembrane protein, SR-α is probably a peripheral membrane protein in most organisms. A homologue, FtsY, is found in bacteria and occurs in both cytoplasmic and membrane-bound forms (de Leeuw *et al.*, 1997).

Interestingly, the C-terminal part of SR-α/FtsY shows a strong homology to the N-terminal N and G domains (collectively termed the "NG" domain) of SRP54/Ffh

(Althoff *et al.*, 1994).[1] Two X-ray structures of this conserved domain are available: Ffh from *Thermus aquaticus* (Freymann *et al.*, 1997)[2] and the corresponding part of FtsY from *E. coli* (Montoya *et al.*, 1997).[3]

Alignments of all SRP components can be found in a special purpose database, the signal recognition particle database, SRPDB[4] (Samuelsson & Zwieb, 1999). There is also an X-ray structure of a fusion protein corresponding to the SRP9/14 heterodimer from mouse (Birse *et al.*, 1997).[5]

3.1.2 SRP-independent targeting

SRP-independent targeting has mainly been described from bacteria. The most well described pathway is SecB-mediated targeting. The SecB protein is a homotetramer, which interacts with the signal peptide and a large region of the mature protein and keeps it in a partly unfolded "translocation-competent" conformation (Rapoport *et al.*, 1996; Danese & Silhavy, 1998).

In addition, SecB binds a specific site of the translocation protein SecA (Danese & Silhavy, 1998). SecA is a homodimer which occurs in both a cytoplasmic and a membrane-bound form. The membrane-bound form is the motor for translocation (see below) while the cytoplasmic form plays a role in targeting. Part of SecA function is analogous to FtsY: it catalyses the release of SecB from the translocating protein upon initiation of translocation.

Only a subset of secretory proteins depend on SecB for export. Others can be SRP-dependent, or targeted via other cytoplasmic proteins. One of these may be the cytoplasmic form of SecA, which apparently binds some secretory proteins without the aid of SecB. Other cytoplasmic factors, notably DnaK, DnaJ, and GrpE, play a not very well defined role in targeting (Danese & Silhavy, 1998).

The information distinguishing SecB-dependent proteins from non-secretory proteins may actually reside in the mature part of the protein, rather than in the signal peptide (Rapoport *et al.*, 1996). It has been suggested that SecB, via a kinetic mechanism, recognises proteins that are slow folders (Duong *et al.*, 1997). A more detailed investigation of the information required for SecB targeting was recently carried out by Kim & Kendall (1998), who showed that a short insert in the mature part of the otherwise SRP-dependent protein PhoA was enough to confer SecB-dependence. These results suggest that in the SecB pathway, the signal peptide is not recognised until it reaches the translocon. This might explain how mutations in the SecY gene (which encodes part of the translocon, see below) can restore secretion of proteins from which the entire signal peptide has been deleted (Prinz *et al.*, 1996).

Also in yeast, some proteins are translocated post-translationally and do not require SRP for translocation (Zheng & Gierasch, 1996). Some proteins, such as carboxypeptidase Y, are completely SRP-independent, while several others show a partial dependence, suggesting that the two targeting systems have overlapping specificities. The hy-

[1] PROSITE `PDOC00272` (`PS00300`, `SRP54`); PFAM `PF00448`

[2] PDB: `1FFH`

[3] PDB: `1FTS`

[4] `http://psyche.uthct.edu/dbs/SRPDB/SRPDB.html` or `http://www.medkem.gu.se/dbs/SRPDB/SRPDB.html`

[5] PDB: `1914`

drophobicity of signal peptides tends to be directly correlated to their SRP-dependence (Zheng & Gierasch, 1996), and mammalian cells are unable to recognise the least hydrophobic yeast signal peptides (Bird *et al.*, 1990). This hydrophobicity correlation is also analogous to the pattern seen in bacteria (see above); but the post-translational targeting systems seem to be different—no homologues of SecB and SecA have been found in eukaryotes. Instead, the Sec62, Sec63, and Sec71 gene products are needed for SRP-independent protein translocation in yeast (Zheng & Gierasch, 1996), but since these are all transmembrane proteins (Matlack *et al.*, 1998), they are hardly responsible for targeting. Better candidates for this role are Ssa1, a chaperone of the Hsp70 family[1], and its associated partner Ydj1, which shows homology to the bacterial DnaJ protein (Wilkinson *et al.*, 1997; Corsi & Schekman, 1996). Also the peripheral membrane protein Sec72, which is associated with Sec62, Sec63, and Sec71, is reported to contribute to the selective recognition of signal peptides (Feldheim & Schekman, 1994; Corsi & Schekman, 1996).

Not much is known about post-translational translocation in multicellular organisms. Mammalian cells are not able to recognise the yeast carboxypeptidase Y signal peptide; but a few small SRP-independent polypeptides have been found in mammals (Wilkinson *et al.*, 1997).

3.1.3 Translocation

Both in eukaryotes and prokaryotes, the nascent protein chain is transferred across the membrane by a complex of membrane proteins known as the *translocase* or *translocon*. The central channel component of the translocon is made from the proteins known as SecY, SecE, and SecG in bacteria, and Sec61α, β, and γ in eukaryotes (in yeast, Sec61β and γ are also known as Sbh1 and Sss1). There is homology between Sec61α and SecY,[2] and Sec61γ and SecE also share a homologous region,[3] but Sec61β and SecG do not seem to be related (Rusch & Kendall, 1995; Schatz & Dobberstein, 1996; Pohlschröder *et al.*, 1997).

The central channel component of the translocon has been visualised by freeze-fracture electron microscopy of purified Sec61 complexes from both mammalian and yeast cells, and it certainly does look like a pore (Hanein *et al.*, 1996; Beckmann *et al.*, 1997; Matlack *et al.*, 1998). The visible pores seem to consist of 3-4 copies of the Sec61 heterotrimer and have an inner diameter of approximately 20 Å (Hanein *et al.*, 1996). In contrast to this, biophysical measurements of active translocation channels have suggested a diameter of up to 60 Å (Hegde & Lingappa, 1997). A possible explanation for this discrepancy could be that a functioning translocon is a dynamic structure which can recruit more heterotrimers during translocation; this may be needed for accommodation of alpha-helices of multispanning transmembrane proteins (see section 3.2).

Recently, corresponding results were obtained for the bacterial translocon: a purified preparation of SecY and SecE from *Bacillus subtilis* was shown to be translocationally active and form pore structures similar to the Sec61 pores (Meyer *et al.*, 1999).

[1] PROSITE PDOC00269; PFAM PF00012

[2] PROSITE PDOC00612 (PS00755/SECY_1 & PS00756/SECY_2); PFAM PF00344

[3] PROSITE PDOC00818 (PS01067/SECE_SEC61G); PFAM PF00584

Interestingly, this result suggests that SecG is not a necessary component of the translocon.

Although the pore must allow passage of the nascent chain, it never forms a completely open channel; a barrier to diffusion of even small ions is kept intact during the whole translocation process. During eukaryotic co-translational translocation, the ribosome itself can provide this barrier by a tight binding to the Sec61 complex (Siegel, 1997; Matlack *et al.*, 1998). In the passive state, and in certain phases of translocation, a barrier is provided on the lumenal side by BiP (Hamman *et al.*, 1998, see also below). What constitutes the diffusion barrier in bacteria and in eukaryotic post-translational translocation is still unclear.

Some auxiliary membrane components are more or less directly associated with the translocon: these are SecD, SecF, and YajC in bacteria, or Sec62, Sec63, Sec66, and Sec72 in yeast. These are not conserved between bacteria and eukaryotes (Pohlschröder *et al.*, 1997), and generally have poorly defined functions, but most seem to be required for a translocation of a subset of substrates.

A 54 kDa membrane protein termed TRAM (for TRanslocating chain-Associating Membrane protein) is found associated with mammalian translocons (Görlich *et al.*, 1992). TRAM has been shown to be required for translocation of a subset of secretory proteins, and it may be required for correct insertion of some transmembrane segments (Hegde & Lingappa, 1997). Interestingly, the degree of TRAM-dependence of a secretory protein seems to depend on the signal peptide (Matlack *et al.*, 1998). Recent results suggest that TRAM also plays a role in regulating the exposure of the nascent chain to the cytosol (Hegde *et al.*, 1998).

An additional component in eukaryotes is the heterotetrameric translocon-associated protein (TRAP). It was formerly known as signal sequence receptor (SSR), because studies suggested a binding between one of the subunits and the signal peptide (Rapoport, 1990). According to newer results, the involvement in translocation is probably less direct; TRAP may be involved in the recycling of the translocation apparatus after completion of the translocation process or may function as a membrane-bound chaperone facilitating folding of translocated proteins (Brodsky, 1998). It has also been described as part of a complex whose function is to bind Ca^{2+} to the ER membrane and thereby regulate the retention of ER resident proteins (Hartmann & Prehn, 1994).

While some small hydrophobic proteins may be able to insert into the lipid bilayer spontaneously, the translocation of larger protein chains with extensive hydrophilic domains across the membrane is clearly thermodynamically unfavourable and requires energy. The energy may be provided by the ribosome itself in the case of co-translational translocation (Schatz & Dobberstein, 1996; Brodsky, 1998).

In post-translational translocation, obviously, the energy cannot come from the ribosome. In bacteria it is provided by the ATPase SecA,[1] a very remarkable molecule which is found both as a soluble protein, a peripheral membrane protein, and an integral membrane protein. SecA can interact with both SecB, the signal peptide, and the translocon; and it is proposed to work via a large conformational change, pushing the nascent chain through the translocon in several steps, each requiring ATP hydrolysis (Duong *et al.*, 1997).

For post-translational translocation in yeast, the energy is provided not by pushing

[1] PROSITE PDOC01016 (PS01312/SECA); PFAM PF01043

from the cytoplasmic side, but by pulling from the ER-lumenal side by BiP (known as Kar2p in yeast), a chaperone of the Hsp70 family[1] (Schatz & Dobberstein, 1996). It is not clear exactly how BiP exercises this pull, but it is known to hydrolyse ATP in interaction with a lumenal domain of Sec63. This domain, the J domain,[2] has homologues in DnaJ and Ydj1, two factors that are thought to participate in post-translational targeting in bacteria and yeast, respectively (see section 3.1.2 on page 24). Also in these cases, they stimulate a Hsp70-like chaperone—DnaK and Ssa, respectively—to hydrolyse ATP (Corsi & Schekman, 1996).

There is evidence for a specific interaction between the signal peptide and one or more of the components of the translocation apparatus. This constitutes a second recognition of the signal peptide—or, in the case of SecB targeting in bacteria possibly the first recognition (see above). The most likely candidate for signal peptide recognition is Sec61α/SecY (Wilkinson *et al.*, 1997; Plath *et al.*, 1998; Danese & Silhavy, 1998). In eukaryotes, evidence suggests that Sec61α recognises the h- and (possibly) c-region, while other components, maybe TRAM, recognises the n-region (Matlack *et al.*, 1998). In addition, there is evidence that the signal peptide interacts directly with specific lipids (van Klompenburg & de Kruijff, 1998).

3.1.4 Cleavage and glycosylation

In the translocon, the signal peptide adopts a hairpin-like conformation with the N-terminus remaining on the cytoplasmic side of the translocon. The cleavage site of the signal peptide is exposed to the ER-lumenal or periplasmic face of the membrane, where it acts as a substrate for signal peptidase (SPase), which is a transmembrane enzyme associated with the translocon.

Eukaryotic SPase has been shown to consist of up to five different subunits (Dalbey *et al.*, 1997). In mammals, the subunits are known as SPC (for Signal Peptidase Complex) 12, 18, 21, 22/23, and 25. The peptidase activity is located in the related SPC18 and SPC21 subunits. Yeast SPase is composed of at least four subunits, of which one, known as Sec11, is related to SPC18 and SPC21.

In bacteria, there are two types of signal peptidases, both monomeric. Signal peptidase I (also known as leader peptidase or Lep) cleaves secretory proteins and has a substrate specificity similar (but not identical) to that of eukaryotic SPase. It belongs to the same family of serine-type proteases as Sec11, SPC18, and SPC21.[3]

In the yeast mitochondrial inner membrane, two SPases have been found, named Imp1 and Imp2 (Dalbey *et al.*, 1997). The chloroplast thylakoid membrane also contains a SPase; its specificity is more similar to that of bacterial SPase I than that of eukaryotic SPase (Gavel & von Heijne, 1990; Howe & Wallace, 1990).

Gram-positive bacteria often have more than one type I SPase (Dalbey *et al.*, 1997). In *Bacillus subtilis*, the first Gram-positive bacterium to be sequenced, five genes for type I SPases have been found: SipS, T, U, V, and W (Kunst *et al.*, 1997). These have overlapping specificities, but seem to prefer different substrates (Tjalsma *et al.*, 1997).

[1] PROSITE `PDOC00269`; PFAM `PF00012`

[2] PROSITE `PDOC00553`; PFAM `PF00226` & `PF00684`

[3] PROSITE: `PDOC00418`; PFAM: `PF00461/signal_pept_I`; ENZYME `EC 3.4.21.89`

The periplasmic domain of *E. coli* SPase I has been crystallised in complex with a peptide analogue inhibitor (Paetzel *et al.*, 1998).[1] The catalytic mechanism of SPase had been a matter of debate, because it differs from the other serine proteases in its sensitivity to inhibitors (Dalbey *et al.*, 1997), but the solved X-ray structure confirms that the active site is a dyad of serine and lysine. The structure also confirms an old cleavage model, according to which the c-region of a signal peptide is in a β-strand-like conformation, so that the side chains of the residues in the -3 and -1 positions are in contact with the active site of the peptidase, while the -2 residue points away from it (von Heijne, 1983).

Bacterial signal peptidase II (also known as lipoprotein peptidase or Lsp)[2] is an aspartic endopeptidase, which cleaves lipoproteins upstream of a cysteine residue to which a glyceride-fatty acid lipid is attached (von Heijne, 1989). The signal peptides cleaved by SPase II do not really have a polar c-region; the cleavage site motif is located in the c-terminal part of the hydrophobic region and has the consensus sequence: Leu–Ala–(Gly or Ala) ↓ Cys (with ↓ denoting the cleavage site).[3] The n- and h-regions do not differ from those of normal signal peptides (von Heijne, 1989). The first few residues downstream from the lipid attachment Cys can determine whether the lipoprotein becomes attached to the inner or outer membrane (Gennity & Inouye, 1991).

On the lumenal side of the ER membrane, the translocating nascent chain encounters not only signal peptidase, but also an oligosaccharyltransferase performing N-linked glycosylation. This transferase, more precisely named dolichyl-diphospho-oligosaccharide-protein glycosyltransferase catalyses the transfer of a high mannose oligosaccharide from a lipid-linked oligosaccharide donor onto asparagine acceptor sites within an Asn-X-Ser/Thr consensus motif in the newly translocated protein.[4] It seems to be composed of at least three subunits in mammals, and it has been purified as a six-subunit complex in yeast (Rapoport *et al.*, 1996). Two subunits of N-oligosaccharyl transferase are sometimes called ribophorin I and II, because they were originally thought to be involved in the binding of ribosomes to the ER membrane (see, *e.g.*, Rapoport, 1991).

By in vitro expression and translocation of proteins with a putative Asn-X-Ser/Thr glycosylation motif inserted at various positions relative to a transmembrane helix, the active site of the glycosyltransferase has been mapped to be located at a distance of 12-14 residues (corresponding to approximately 30-40 Å) from the membrane, or more specifically from the lumenal end of the TM helix (Nilsson & von Heijne, 1993). By an extension of the same method—varying both the length of the hydrophobic region and the distance from hydrophobic region to glycosylation site—it was found that this distance was different for short hydrophobic regions, corresponding to cleaved signal peptides, and long hydrophobic regions, corresponding to signal anchors (see section 3.2 Nilsson *et al.* 1994).

[1] PDB entry `1B12`, on hold until Nov 25 1999.

[2] PROSITE: `PDOC00669` (`PS00855/SPASE_II`); PFAM: `PF01252/SPASE_II`; ENZYME: `EC 3.4.23.36`

[3] For the SPase II cleavage site motif, see also PROSITE `PDOC00013` (`PS00013/PROKAR_LIPOPROTEIN`)

[4] ENZYME `EC 2.4.1.119`

3.1.5 Fate of the signal peptide after cleavage

Signal peptides are generally degraded rapidly in the membrane. In *E. coli*, a transmembrane enzyme, signal peptide peptidase or protease IV,[1] is responsible for the degradation. The fragments generated by protease IV are subsequently digested by the cytosolic oligopeptidase A (Miller & Conlin, 1994). In eukaryotes, the proteases have not been characterised.

In eukaryotes, not all signal peptides are “discarded” after cleavage. Signal peptide fragments have been found to bind to calmodulin, a cytosolic protein that binds various peptides in a Ca^{2+}-dependent manner and plays a role in the regulation of many cellular processes, notably signal transduction (Martoglio & Dobberstein, 1998). The calmodulin-binding signal peptides characterised until now (specifically, those of preprolactin and HIV-1 gp160) have unusually long n-regions with many basic residues (Martoglio *et al.*, 1997).

Some signal peptides also play a role in immune recognition. Fragments of signal peptides can bind to class I molecules of the major histocompatibility complex (MHC) as epitopes. There are two pathways for signal peptide fragments to reach the binding site of the MHC molecules in the ER lumen: fragments derived from the n-region are released to the cytosol, possibly processed by the proteasome, and then transported through the ER membrane by TAP (Transporter of Antigen Presentation), an ABC (ATP-binding cassette) transporter; while fragments comprising the h- and c-regions are released directly to the ER lumen in a TAP-independent fashion (Martoglio & Dobberstein, 1998).

Apparently, MHC presentation of signal peptide fragments has two rather different roles: HLA-A, -B, and -C (classical MHC class I molecules) present epitopes derived from a wide range of cellular or viral signal peptides, while HLA-E (a non-classical MHC class I molecule) specifically presents signal peptides of HLA-A, -B, or -C! This seems to be a part of the self-presentation of a healthy cell; upon viral infection or in cancerous tissue, expression of classical MHC class I molecules may be lost, and without HLA-E-mediated presentation of their signal peptides the cell becomes a target for natural killer cells (Braud *et al.*, 1998; Long, 1998).

A potential advantage of using signal peptide-derived epitopes is that they provide a very fast way of presentation: the signal peptide is available from the moment a protein is expressed; the MHC does not have to await degradation of the protein to present fragments from it. In the case of TAP-independent epitopes, the proteasome and TAP steps are furthermore bypassed.

3.2 Insertion of transmembrane proteins

Since the translocon functions also in insertion of α-helix transmembrane proteins, it must be able to open not only “vertically” to allow translocating protein chains through, but also “laterally” to release transmembrane helices into the lipid phase of the membrane.

There are different ways in which hydrophobic stretches can interact with the translocon and signal its lateral opening, corresponding to the conventional classification of

[1] PFAM: PF01343/Peptidase_U7

α-helix transmembrane proteins into four groups (von Heijne, 1988):

Type I membrane proteins have an N-terminal signal peptide, which initiates translocation and is cleaved off, and a tract of hydrophobic amino acids further downstream. This *stop-transfer sequence* halts translocation and is released to the lipid bilayer, resulting in a membrane protein with a "N_{out}-C_{in}" topology.

Type II membrane proteins have an N-terminal hydrophobic sequence—the *signal anchor*—which initiates translocation in the same way as signal peptides do, but is not cleaved by signal peptidase. The rest of the polypeptide chain is translocated through the membrane, but the resulting protein remains anchored to the membrane with a "N_{in}-C_{out}" topology.

Type III membrane proteins have *inverted signal anchors* which do not make an N-terminal hairpin structure, *i.e.*, the N-terminal part of the chain is translocated instead of the C-terminal part. To add to the confusion, these proteins are sometimes called "signal anchor type I" proteins, because they have the same topology as type I described above (Matlack *et al.*, 1998).

Multispanning membrane proteins are a large and diverse group. The insertion of a multispanning (also called polytopic or type IV) membrane protein may be initiated by a signal peptide as a type I membrane protein, or by the first transmembrane segment as a type II or III membrane protein—or it may be inserted post-translationally by altogether different mechanisms.

One unsolved question is the translocation of long N-terminal tails: some multispanning membrane proteins have their first transmembrane segment in the N-out orientation without a signal peptide, *i.e.*, as an inverted signal anchor, but with a much longer N-terminal extracytoplasmic domain than is ever found in type III membrane proteins. Several examples of this can be found in the GPCR family of proteins with seven transmembrane helices.[1] This does not conform to the hairpin insertion geometry otherwise assumed by the signal peptide or signal anchor in the translocon, and how the translocation of this N-tail is initiated is largely unknown. In engineered proteins, a single artificial TM segment has been shown to be able to promote translocation of an up to more than 200 amino acids long N-terminal domain (see, *e.g.* McMurry & Kendall, 1998; Mitsopoulos *et al.*, 1997).

During co-translational insertion of TM proteins, the ribosome/translocon interface can shift between two states: while translocating lumenal domains, the ribosome is tightly bound to the translocon and seals the interface; but while translating cytoplasmic domains, the ribosome is more loosely tied to the membrane, and BiP is needed to keep the pore sealed on the lumenal side (Rapoport *et al.*, 1996). Hydrophobic regions are signals for switching between these two states, being recognised either as start-transfer or stop-transfer sequences. Surprisingly, this recognition seems to take place already in the ribosome rather than in the translocon: the structural changes that accompany the switch seem to take place before the hydrophobic region has entered the translocon (Siegel, 1997; Liao *et al.*, 1997).

[1] PROSITE: `PDOC00210` (`PS00237/G_PROTEIN_RECEPTOR`); PFAM `PF00001/7tm_1`; The G-protein coupled receptor database: `http://www.gcrdb.uthscsa.edu/`

While translocating a multispanning TM protein, does the translocon open laterally for every TM helix, or only when the whole protein is inserted and all its helices assembled? The most likely explanation seems to be a combination of these two, *i.e.*, a model where the translocon can accommodate a bundle of several helices, but can also open laterally during translocation to release one or more helices that comprise a structural unit (Hegde & Lingappa, 1997; Mothes *et al.*, 1997).

It should be mentioned that there is an additional class of transmembrane α-helix proteins that do not fit into the scheme outlined above: the "tail-anchored" membrane proteins. These have a membrane anchor located so close to the C-terminus that it cannot be targeted co-translationally. These proteins are inserted preferentially (maybe exclusively) into the ER membrane, but there is no definitive answer as to whether they utilise the translocon (Rapoport *et al.*, 1996).

3.3 Signal peptide properties

As described in the introduction to section section 3.1, signal peptides are composed of a positively charged n-region, a hydrophobic core or h-region, and a more polar c-region with small and neutral amino acid residues at positions -3 and -1 relative to the cleavage site.

3.3.1 Statistics

In statistical studies of signal peptides, the length variation is a recurring problem. Although the division into n-, h-, and c-regions is used by many authors, there is no universally accepted operational definition of region borders. Various statistical studies have used their own criteria for assigning regions (von Heijne, 1985; McGeoch, 1985; von Heijne, 1986a; Sjöström *et al.*, 1987; von Heijne & Abrahmsén, 1989), while one group has chosen to analyse signal peptides according to a normalised length (Shinde *et al.*, 1989; Shinde, 1990). In paper V, we introduce two definitions of region borders: one based on a decision rule, and the other built into the hidden Markov model (see section 6.2).

There are differences between the signal peptides of different groups of organisms. Eukaryotic SPs are slightly shorter than those of Gram-negative bacteria, and markedly shorter than those of Gram-positive bacteria, although there is considerable variation within each group (von Heijne 1985; von Heijne & Abrahmsén 1989; see also the length distribution in figure 1 of paper IV). The length differences between groups are seen in both the n-, h- and c-regions, while the within-group variation mostly occurs in the n-region (von Heijne 1985; von Heijne & Abrahmsén 1989; figure 4 of paper V). The amino acid composition of the three regions also differ between groups (von Heijne 1985; figure 1 of paper II), these differences will be described below.

The n-region

The n-region is characterised by at least one positive charge, and often contains Lys or Arg residues. In eukaryotes, the N-terminal of the protein itself provides one positive charge, so Lys or Arg may be absent. In prokaryotes, the N-terminal is formylated, so the presence of Lys or Arg is required for positive charge (von Heijne, 1990).

The positive charge in the n-region is balanced by a negative net charge in the c-region plus the first five amino acids of the mature protein (von Heijne, 1986a). This effect is significant for prokaryotes, but somewhat weaker for eukaryotes. A strong bias for the negatively charged residues Asp and Glu has been found at position +2 in prokaryotes (von Heijne, 1986a) and (again somewhat less strongly) at position +1 in eukaryotes (Prabhakaran, 1990). In figure 1 of paper II, it is obvious that there is more information after the cleavage site in the bacterial than in the eukaryotic pattern.

The charge distribution correlates well with the "positive inside" rule (von Heijne, 1988), which states that short cytoplasmic domains of transmembrane proteins are usually positively charged. As described in section 3.1.4, the n-region is thought to remain in contact with the cytoplasmic face of the membrane while the rest of the protein is being translocated.

The h-region

Whether the h-region simply consists of an appropriate number of appropriately hydrophobic amino acids in random order, or whether there are some specific sequence requirements, has been a matter of debate. Perlman & Halvorson (1983), investigating 39 signal peptides (both pro- and eukaryotic), found that Leu, the most common amino acid in the h-region, showed a two-peaked distribution—*i.e.*, it was less abundant in the centre of the h-region than at the edges. This was confirmed by Shinde *et al.* (1989), who found that the frequency of "helix stabilisers" (Leu and Ala) had two peaks in the h-region, while "helix destabilizers" (Val, Pro, Gly, Met, Cys, Phe, and Ile) had one peak between the two Leu-and-Ala peaks. This corresponds to an earlier finding that in prokaryotic signal peptides, the h-region often contains a centrally placed Pro or Gly residue (von Heijne, 1988). Perlman & Halvorson (1983) also reported that some pairs of adjacent amino acids (Leu–Phe, Leu–Ile, Ala–Ala, and Val–Leu) occurred less frequently than expected from the amino acid distribution. On the other hand, von Heijne (1985) examined the positions and pairwise occurrences of amino acids in actual h-regions and randomised h-regions (where the amino acids had been shuffled in random order), and found no differences between them.

The h-region tends to form α-helix in a non-polar environment but is also able to form β-sheet, according to the Chou & Fasman secondary structure prediction rules and simple propensity statistics (Perlman & Halvorson, 1983; Prabhakaran, 1990). However, the ability to form α-helix cannot be an absolute requirement, since the helix-breaking residue Pro has been found at many positions inside the h-region (Perlman & Halvorson, 1983; McGeoch, 1985). Between the h-region and the cleavage site, there is often a helix-breaking residue like Pro, Gly, or Ser (Perlman & Halvorson, 1983; von Heijne, 1983, 1986b). This correlates well with the model of cleavage site conformation (see section 3.1.4) which requires the -3 to -1 region is *not* in a helical conformation.

Cleavage specificity

The $(-3,-1)$-rule (stating that the residues at positions -3 and -1 relative to the cleavage site must be small and neutral) was formulated independently by von Heijne (1983) and Perlman & Halvorson (1983). Sometimes, it is also referred to as the "Ala-X-Ala" consensus for cleavage sites, since Ala is the most frequent residue in these position for all groups. The rule is "interpreted" much more strictly in bacteria than in eukaryotes: while position -1 has ≈80% Ala in bacteria, it has only ≈40% Ala in eukaryotes. Position -3 is less conserved than -1, and it has Val as the second most frequent residue, while Val is almost never found in -1. Gly, on the other hand, is more tolerated in -1 than -3. (von Heijne 1986b; Karamyshev *et al.* 1998; figure 1 of paper II).

von Heijne (1984) reported that position -2 is often occupied by a charged, aromatic, or large polar residue, *i.e.*, one that is not tolerated at positions -1 and -3; but the deviation from the background distribution is very weak (figure 1 of paper II). A putative preference for bulky residues at position -2 is not necessarily evidence that this position is involved in signal peptidase recognition—instead, it may be an effect of selection against cleavage ambiguity, *i.e.*, a mechanism for avoiding possible cleavage sites besides the true one. Alternative cleavage, however, is sometimes found *in vivo*; von Heijne (1984) mentions nine examples. The most famous example is bovine growth hormone, where the sequence VVGA is cleaved after the Ala in 65% of the molecules, and between the Gly and the Ala in the rest (Folz & Gordon, 1987).

Signal peptides and post-translocational sorting

Given that the signal peptide is cleaved already during translocation, one should not expect that it is able to influence the subsequent fate of the mature protein. Accordingly, most statistical studies have not found any correlation between the SP and the function or location of the mature protein—maybe because they have not looked. Sjöström *et al.* (1987), however, reported a correlation between signal peptides and final location in *E. coli*. They divided the sequences of 43 signal peptides into five groups according to final protein localisation (inner membrane, periplasmic space, outer membrane, extracellular surroundings and pili) and found that three of the groups showed statistically significant differences, measured by 20 physico-chemical properties averaged over 3 overlapping 10 aa windows and tested by partial least squares discriminant analysis. The general tendency was that proteins located further from the cytoplasm had signal peptides with more hydrophobic N-terminal parts.

The correlation between signal peptides and localisation has been confirmed in a very recent study from the same group (Edman *et al.*, 1999). Here, 29 physico-chemical properties were was reduced to three parameters after principal components analysis, and the auto- and cross-covariance terms of these three parameters were calculated for sequence separation distances of 1 to 10, and used in a partial least squares discriminant analysis. It was shown that the method could discriminate between signal peptides of different bacteria (*Escherichia coli*, *Bacillus*, and *Mycoplasma*), and more interestingly, four of the five *E. coli* protein locations (except signal peptides of inner membrane proteins) showed significant differences. The implications of this finding are not very clear—there may be some correlation between targeting and post-translocational sorting, but it is rather difficult to attach a meaningful interpretation to the auto- and cross-covariance terms.

Codon usage and signal peptides

Burns & Beacham (1985) observed that rare codons for Leu and Pro are especially common in signal peptides. This has been contradicted by Bulmer (1988), who stated that the effect has nothing to do with signal peptides as such, but simply reflects that codon usage is different in the beginning of the gene. It has also been suggested that DNA regions coding for signal peptides are characterised by a high G/C content (Arrigo *et al.* 1991, see section 4.2.3 on page 45); but this was based on a very small non-representative data set, and also in this case, the results have not been compared with N-terminal parts of non-secretory proteins. I have not seen any newer test of codon usage and signal peptides on a larger data set.

Another interesting observation was made by Képès (1996), who reported that clusters of rare codons frequently occur in the region 56–75 codons downstream from a transmembrane helix of yeast membrane proteins. The hypothetical function of this phenomenon, the "+70 pause," was suggested to be a transient slowdown in translational speed, leaving time for a hydrophobic stretch of a nascent protein to interact with SRP and/or the translocon. However, the effect was not observed downstream from signal peptides, as expected by this hypothesis. To my knowledge, the "+70 pause" has not been confirmed by other authors.

3.3.2 Mutations

A large number of experimental studies have employed mutagenesis of natural signal peptides in order to investigate the sequence requirements for signal peptide function. Almost a decade ago, there were already enough results to write a lengthy review about them (Gennity *et al.*, 1990).

It should be noted that mutation studies do not necessarily find the same limits of signal peptide variation as statistical studies of naturally occurring signal peptides do; there may be other factors than transport efficiency which influence the *in vivo* appearance of signal peptides, such as codon usage, regulation of transcription and/or translation, or evolutionary history (Laforet & Kendall, 1991). One of these factors is that in the experimental studies, the performance of a mutated signal peptide is measured by the amount and rate of translocation and cleavage. This is of course relevant from a biotechnological point of view; but it may well be the case that the fastest processed or fastest cleaved form is not optimal for the living cell.

The ability of a polypeptide to function as a signal peptide seems to be a quantitative property rather than an all-or-none question. In many of the mutagenesis studies, intermediate levels of translocation and/or signal peptide cleavage are found for some of the mutants (Gierasch, 1989). In some cases, mutants are found to be translocated and cleaved more efficiently than the wild type (see, *e.g.*, Yamamoto *et al.*, 1987; Goldstein *et al.*, 1990). However, signal peptides with an intermediate level of efficiency should be expected to be rare *in vivo*, firstly because a signal peptide is adapted to the function of its protein, which is either translocated or not; and secondly because a slowly translocated protein will occupy the translocation apparatus for a long time (Ferenci & Silhavy, 1987).

In general, the mutagenesis studies have shown a very low requirement for specific sequences. Kaiser *et al.* (1987), using fusion peptides in yeast, reported that around

20% of a library of random protein sequences were able to initiate some amount of translocation (in most cases without cleavage) of an otherwise cytoplasmic protein. The "random" protein sequences consisted of bulk human DNA cut into pieces and inserted into a gene for the enzyme invertase, containing a deletion in its signal peptide region. The value of 20% was clearly a very rough estimate, and the examples have not been sequenced. The work has been criticised (Ferenci & Silhavy, 1987), but the fact remains that there is a significant chance that a random sequence can have at least partial signal sequence function.

The n-region

In some studies, removal of the positive charge of the n-region slowed protein export, but unless it is replaced by a negative charge, translocation is not completely blocked (von Heijne, 1990; Nesmeyanova *et al.*, 1997). There is even an example where the deletion of the whole n-region did not affect translocation and cleavage in a eukaryotic cell-free system (Andrews *et al.*, 1992).

The exact placement of the positive charge can be important. One mutation study in yeast (Green *et al.*, 1989) found that replacing the N-terminal sequence Met–Arg$^+$–Phe with Met–Phe–Arg$^+$ or Met–Phe–Lys$^+$ caused a reduction in the translocation efficiency.

The h-region

Most known export-defective signal peptide mutants have amino acid alterations in the h-region. The introduction of a single charged amino acid often blocks export altogether (see von Heijne, 1988, 1990; Gierasch, 1989, for examples).

In *E. coli*, the entire h-region may be replaced by a strand of only poly-Leu (Kendall *et al.*, 1986; Chou & Kendall, 1990), poly-Ile (Kendall & Kaiser, 1988; Chou & Kendall, 1990) or poly-Phe (Rusch & Kendall, 1992) without loss of function, while poly-Trp (Rusch & Kendall, 1992) or poly-Ala (Chou & Kendall, 1990) affect signal peptide function negatively. Also in yeast, a poly-Leu h-region has been shown to function (Yamamoto *et al.*, 1987).

Thus, the presence of specific combinations of amino acids in the h-region does not generally seem to be essential. Or, in other words, the correlation between positions is apparently low. However, there are exceptions: Lehnhardt *et al.* (1987) found that the deletion of one Ala residue alone reduced export efficiency, while the deletion of the same Ala residue together with an adjacent Ile residue tended to improve it; and Ryan & Edwards (1995) found that introducing a proline at various positions in the h-region is dependent on the position in a way that correlates with position in a helical wheel.

The important requirements for the h-region seem to be a high hydrophobicity and a length within certain limits (longer than five but shorter than twenty amino acids in *E. coli* (Chou & Kendall, 1990)). In yeast, a positive correlation between h-region hydrophobicity and translocation efficiency has been demonstrated (Bird *et al.*, 1990).

Simultaneously varying composition and length of the h-region in *E. coli* suggests that the total hydrophobicity is the important variable (Chou & Kendall, 1990): artificial h-regions of Leu_{10} or Leu_{15} were functional, while a Leu_{20} h-region was inserted in the membrane and not cleaved; but on the other hand, Ala_{10} and Ala_{15} were almost totally non-functional, while Ala_{20} showed some activity.

Many mutation studies have concluded that the tendency to form α-helix is an important feature of the h-region (reviewed in von Heijne, 1990; Rusch & Kendall, 1992), but there are also examples where increased tendency of forming β-sheet has enhanced signal peptide function (see, *e.g.*, Goldstein *et al.*, 1990).

The c-region

In the c-region, a number of mutation studies have confirmed the $(-3,-1)$-rule for both *E. coli* (Laforet & Kendall, 1991; Nilsson & von Heijne, 1991; Karamyshev *et al.*, 1998) and yeast (Monod *et al.*, 1989).

As mentioned, helix-breaking residues like Pro, Ser, or Gly are often found at the border between h- and c-regions; but the evidence for the requirement of these residues is not conclusive. One mutation study found that a Pro residue at positions -4, -5 or -6 was necessary for the function of the signal sequence in yeast (Yamamoto *et al.*, 1989). On the other hand, for *E. coli*, a wide variety of artificial c-regions without any helix-breaking residues were found to function indistinguishably from the wild-type (Laforet & Kendall, 1991). One of these c-regions was a homopolymer of six Alanines, making the impressively simple signal peptide `MKQSTLLLLLLLLLLLAAAAAA` fully functional.

The region after the cleavage site

The first approximately 30 residues of the mature protein seem to have a function for protein export in *E. coli*. A study of fusion proteins showed that deletions within positions +1 to +28 lead to an export defect (Rasmussen & Silhavy, 1987). A mutation study has shown that a string of six positively charged residues blocks export (in *E. coli*) when inserted less than 38 residues from the end of the h-region (Andersson & von Heijne, 1991).

Introducing single positive charges in the first five residues of the mature protein slows down protein export in *E. coli* (Li *et al.*, 1988). In a eukaryotic system, the effects of positive charges in the +1 to +5 region was much smaller (Kohara *et al.*, 1991). One substitution found to block cleavage in both pro- and eukaryotic systems is Pro in +1 (Kohara *et al.*, 1991; Nilsson & von Heijne, 1992), presumably because it makes a correct cleavage site conformation impossible (Karamyshev *et al.*, 1998).

Interaction between the regions

In general, signal peptides from different proteins are functionally interreplaceable, indicating a low degree of correlation between the signal peptide and the mature protein (Laforet *et al.*, 1989). The different regions within the signal peptide are also generally uncorrelated; but there are exceptions, where changes in different regions are found to interact.

Apparently, there is an interaction between the positive charge in the n-region and the length of the h-region. One mutation study (Hikita & Mizushima, 1992) found that the requirement of a positive charge at the amino terminus can be compensated for by a longer hydrophobic stretch (in *E. coli*). With an h-region of Leu_9, positive charge was not important, but with a Leu_7 or Leu_8 h-region, it was strongly required.

Replacing the h-region of one signal peptide with the h-region from the signal peptide of another protein does not always yield a functional hybrid. In one example (from *E. coli*) where this was the case (Laforet *et al.*, 1989), the activity of the non-functional hybrid could be restored by removing a positive charge from position -2. This points to an interaction between the h- and c-regions: some h-regions can "tolerate" a positive charge at -2 while others cannot. Similar observations have been made on pseudo-revertants: in at least two cases, mutants with a non-functional h-region have had their export ability restored by a replacement in the mature part of the protein (reviewed in von Heijne, 1990).

In connection to the discussion about non-linearity in section 2.4, I should stress that these observed interactions do *not* necessarily imply non-linearity in the signal peptide sequence pattern. A compensation may very well be linear, if the effects of the contributions (*in casu* from the n-region and the h-region) are additive. The following interaction observed by Lehnhardt *et al.* (1987), however, seems genuinely non-linear: a signal peptide of one *E. coli* protein was fused to the mature part of two other *E. coli* proteins, and both hybrids were functional; but they reacted very differently on deletion mutations. Deletion of two amino acids from the h-region blocked secretion of the first hybrid but not the second, while deletion of one specific Ala residue slowed secretion of the second hybrid but had no effect on the first.

3.4 Subsequent sorting in the secretory pathway

In eukaryotes, secretory proteins comprise not only *secreted* proteins, but also proteins of the various compartments of the secretory pathway: the ER, the Golgi apparatus, secretory granules, and lysosomes. In Gram-negative bacteria, there is also a sorting step involved in the distinction between periplasmic proteins, outer membrane proteins, and proteins secreted to the medium. Only in very few cases are the sorting signals known.

3.4.1 The eukaryotic secretory pathway

In the ER, there are several resident proteins, some of them with functions mentioned earlier (*e.g.* the translocon constituents and BiP, section 3.1.3 and signal peptidase and oligosaccharyltransferase, section 3.1.4). Most ER proteins need a signal to be retained in the ER. The best known ER retention signal is the `KDEL` (Lys-Asp-Glu-Leu) sequence, where the initial `K` is not totally conserved, it can be `H`, `D`, `A`, or `S` in various organisms.[1] Proteins bearing the `KDEL`-type signal are not necessarily held in the ER constantly, but are selectively retrieved from a post-ER compartment by a receptor and returned to their normal location (Hurtley, 1993; Machamer, 1996).

Transport from ER to the Golgi apparatus is traditionally regarded as the "default" fate for translocated proteins. However, differences in export rates suggest that at least

[1] The `KDEL` pattern is described in the PROSITE entry `PDOC00014` (`PS00014/ER_TARGET`).

some secretory proteins have positive signals that facilitate their packaging into ER-to-Golgi transport vesicles (Nishimura & Balch, 1997; Herrmann *et al.*, 1999).

How the Golgi apparatus maintains its structure is a matter of debate. The Golgi stack can be divided into at least three compartments with different protein compositions (cis-, medial, and trans-Golgi), but there are two competing models for how the cisternae maintain their difference (Glick & Malhotra, 1998). The "vesicular transport" model, which has been predominant during the last couple of decades, regards the cisternae as stable structures, with transport vesicles carrying the bulk of secretory proteins through the stack, while resident Golgi proteins are selectively retained in their respective compartments. According to the alternative "cisternal progression/maturation" model, which is older but currently regaining support, cisternae are continuously formed from the ER and move through the Golgi stack, while their protein complement is modified by selective retrograde transport of resident proteins (Füllekrug & Nilsson, 1998; Glick & Malhotra, 1998). No well-characterised signals for retention or retrograde transport are known, but in several cases, certain transmembrane domains have been shown to be critical for Golgi localisation (Machamer, 1996; Munro, 1998; Füllekrug & Nilsson, 1998).

In the Golgi apparatus there are proteases which cleave short N-terminal peptides off (Seidah & Chrétien, 1997; Nakayama, 1997). An identification of these cleavage sites would be of great interest for signal peptide prediction, because of the possibility that some of these cleavage sites erroneously could be annotated as signal peptidase cleavage sites in the databases.

3.4.2 Bacterial secretion across the outer membrane

In Gram-negative bacteria, the export system only translocates proteins across the inner membrane to the periplasm. Proteins destined for secretion need an additional system for translocation across the outer membrane. This is known as the *Main Terminal Branch* of the general secretory pathway (Pugsley, 1993; Pugsley *et al.*, 1997), or as *type II secretion* (Russel, 1998).

This system is capable of translocating fully folded proteins (Pugsley *et al.*, 1997). A group of proteins, in many bacteria encoded together in one operon, are involved in the main terminal branch and also in biogenesis of type IV pili and competence for DNA uptake (Pugsley *et al.*, 1997; Russel, 1998). In *Klebsiella pneumoniae*, where they were first described, they are known as PulB–O named after pullulanase, PulA, a secreted enzyme located in the same operon; in other bacteria, the associated genes have names such as exe, out, xcp, or yhe.[1]

A special class of secretory proteins, the autotransporter proteins, are able to catalyse their own translocation through the outer membrane. They have cleavable signal peptides, although some of them have abnormally long n-regions with a highly conserved "IAVSELAR" motif and unusually many positively charged residues. After cleavage of the signal peptide, the C-terminal domain of the autotransporter inserts into

[1] Signatures for bacterial type II secretion system proteins C, D, E, F, and N: PROSITE: PDOC00878, PDOC00683, PDOC00567, PDOC00682, and PDOC00879; PFAM: PF00595/PDZ (PDZ domains are found in diverse signaling proteins besides secretion system proteins), PF00263/Bac_GSPproteins, PF00437/GSPII_E, PF00482/GSPII_F, and PF01203/T2SP_N.

the membrane, probably in a β-barrel conformation, the rest of the protein is threaded through the barrel, and the barrel domain is cleaved off. This remarkable mechanism is also known as "type IV secretion" (for a review, see Henderson *et al.*, 1998).

3.5 Non-classical secretion

The general secretory pathway is not the only method for protein secretion. In this section, I discuss some mechanisms that wholly or in part differs from the process outlined above.

3.5.1 Eukaryotic examples

A few examples are known of eukaryotic secreted proteins with uncleaved signal peptides: Ovalbumin and the related proteins gene Y product and plasminogen activator inhibitor II (Rapoport, 1991). All three belong to the family of serpins (serine protease inhibitors),[1] which include both secreted proteins with perfectly normal signal peptides, and cytosolic proteins. For plasminogen activator inhibitor, translocation is variable: only some of the molecules are translocated, while others remain cytosolic; the proportion depending on cell type (Belin *et al.*, 1996).

Another type of secretion without signal peptide cleavage is exemplified by interleukins, fibroblast growth factors, and transglutaminase. In contrast to the serpin examples, the secretion of these does not depend on a functional translocon. Instead, they might be exported by a system related to the ABC transporters (see below) (Rapoport *et al.*, 1996).

3.5.2 The twin-arginine translocation pathway

In bacteria and chloroplasts, some proteins are transported by a separate targeting and translocation mechanism, the TAT (Twin-Arginine Translocation) pathway (Settles & Martienssen, 1998; Dalbey & Robinson, 1999). The name refers to the special signal peptides of TAT-targeted proteins, that contain a characteristic motif with two arginines in their n-region. The TAT translocation pathway is found in the inner membrane of Gram-negative bacteria, where it is also known as the Mtt (for membrane targeting and translocation) pathway, and in chloroplast thylakoid membranes, where it is also known as the ΔpH pathway. The observation that TAT signal peptides from bacteria and thylakoids appear to be interchangeable stresses the conserved nature of the TAT system (Settles & Martienssen, 1998).

Most preproteins transported by the TAT pathway bind redox cofactors (Berks, 1996), and remarkably, they seem to be folded or even oligomerised before translocation across the membrane (Santini *et al.*, 1998). Interestingly, no clear example of an integral inner membrane protein targeted through the TAT pathway has been found so far.

Mutational analysis and database searches have led to the identification of genes that seem to be required only for the TAT pathway. Hcf106 from maize was the first TAT

[1] PROSITE `PDOC00256` (`PS00284/SERPIN`).

component to be identified (Settles *et al.*, 1997). Hcf106 has homology to open reading frames in several organisms ranging from bacteria and archaea to higher plants; as far as is known, most bacterial species have two Hcf106 homologues, while *E. coli* and *B. subtilis* seem to have three (Settles & Martienssen, 1998; Sargent *et al.*, 1998). Components of the *E. coli* TAT system are encoded by the tatABCD operon (Sargent *et al.*, 1998), also known as the mttABC operon[1] (Weiner *et al.*, 1998) and by TatE, which appears not to be part of an operon. TatA, TatB, and TatE are the *E. coli* homologues of Hcf106 (Settles & Martienssen, 1998; Sargent *et al.*, 1998).

Substrates of the TAT pathway have a characteristic, unusually long signal peptide. The n-region contains the twin-Arg motif mentioned above, with the consensus sequence (S/T)-R-R-X-F-L-K, where only the two arginines are completely conserved (Berks, 1996). In addition, most twin-Arg signal peptides have one or more positively charged residues, a "Sec-avoidance" signal, in the c-region just upstream of the signal peptidase cleavage site (Bogsch *et al.*, 1997). The cleavage site, however, conforms to the pattern of normal signal peptidase I substrates. In a recent study combining mutagenesis with a statistical investigation, we found that there is also a difference in the h-region: compared to "normal" Sec-translocated signal peptides, the h-regions of bacterial twin-Arg signal peptides are less hydrophobic and contain significantly more Gly (Cristóbal *et al.*, 1999).

3.5.3 Type IV pilins

Type IV pilins of Gram-negative bacteria have signal peptides that are cleaved *N-terminally* to their hydrophobic region by a special prepilin-like protein specific leader peptidase, PulO (Pugsley, 1993). Several proteins of the main terminal branch (see section 3.4.2) have the same type of signal peptides, which show a fairly well-conserved motif in the N-terminal region.[2] The residue immediately after the cleavage site (Phe or Met) is methylated.

Archaeal flagella seem to be related to type IV pili rather than bacterial flagella, based on sequence similarity of the constituent proteins and the presence of some archaeal ORFs with matches to proteins of the bacterial type IV pilus assembly system (Bayley & Jarrell, 1998; Bult *et al.*, 1996).

3.5.4 Type I secretion

In bacteria, there are at least two secretory pathways which are not dependent on the *Sec* genes (type I and type III secretion) and do not rely on cleavable N-terminal signal peptides (Salmond & Reeves, 1993). In Gram-negative bacteria, these transport their proteins across both membranes of the envelope at the same time.

Type I secretion uses transport channels of the ABC (ATP Binding Cassette) superfamily.[3] It is a very simple secretion mechanism, in terms of the number of components involved: in addition to the ABC transporter, some systems need only two accessory proteins, one outer membrane protein, and one inner membrane protein that may also

[1] The MttB family is found in PFAM: PF00902/UPF0032.

[2] PROSITE: PDOC00342 (PS00409/PROKAR_NTER_METHYL)

[3] PROSITE: PDOC00185; PFAM: PF00005/ABC_tran.

make contact to the outer membrane (Binet *et al.*, 1997).[1] Type I secretion secretes haemolysin and other toxins. The signal for targeting seems to reside in the C-terminal part of the protein chain, but the transport mechanism is rather specific for one or a few passenger proteins, and secretion of heterologous proteins is generally very low (Binet *et al.*, 1997). Typically, each passenger protein is encoded on an operon together with its cognate transporter (Salmond & Reeves, 1993).

Type I secretion exists also in Gram-positive bacteria, *e.g. Lactococcus lactis*. In eukaryotes, ABC transporters are known to translocate peptides across membranes, but whether they can transport entire proteins is not clear (Cleves & Kelly, 1996).

3.5.5 Type III secretion

The type III secretion system has been described mainly from pathogenic bacteria such as *Salmonella*, *Yersinia*, and *Erwinia*. In many of the known cases, type III secretion is directly involved in pathogenesis, and works almost as a molecular syringe for toxin injection: the secreted toxins are transported over the two membranes of the pathogen envelope and the host plasma membrane simultaneously. For a very extensive review, see Hueck (1998).

The exported peptides are quite diverse, and although the signal for export has been shown to be N-terminal, it has been impossible to find any shared targeting motif or common features (Silhavy, 1997). Interestingly, the explanation seems to be that the sorting signal actually resides in the mRNA. This is supported by a mutagenesis study in which two balanced frameshift mutations that completely changed the 15 N-terminal amino acids did not block secretion (Anderson & Schneewind, 1997). This implies that type III targeting must be co-translational.

Like type I secretion, components of the type III secretion system are organised into operons. Several of the proteins necessary for flagellar assembly belong to the same families or operons as inner membrane type III secretion components, while some outer membrane components are related to the type II secretion system (Hueck, 1998).[2]

[1] PROSITE: PDOC00469; PFAM: PF00529.

[2] Three families related to flagella transport and/or type III secretion: PFAM PF01312/Bac_export_2; PROSITE PDOC00763, PFAM PF00771/FHIPEP; and PROSITE PDOC00812, PFAM PF00813/FliP.

Chapter 4

Examples of protein sorting prediction

In this chapter, I will review some examples of bioinformatics applications within the protein sorting field (my own contributions are described in chapter 6). Although I pay special attention to signal peptide prediction, I will also mention two works on mitochondrial transit peptides and a number of general-purpose protein location predictors.

It should be noted that the bulk of predictions of subcellular location are probably not done using any of these methods, but by alignment to proteins with experimentally known function and/or subcellular location. In addition, several PROSITE patterns are relevant for protein sorting: some describe localisation signals such as the ER retention signal[1] or the bacterial lipoprotein peptidase cleavage and attachment site;[2] others characterise a function which is specific to one compartment—as an example, nuclear localisation could be inferred from signatures for DNA-binding proteins.

4.1 An integrated expert system

PSORT[3] (Nakai & Kanehisa, 1991, 1992; Nakai & Horton, 1999) is an integrated system of several prediction methods, using both sorting signals and global properties. Some of the components are developed within the PSORT group, others are implementations of methods published elsewhere, including selected PROSITE patterns. PSORT is the only publicly available system that shows this degree of integration, and it includes predictions for locations that no other available methods provide, *e.g.*, nuclear or peroxisomal targeting.

All the constituent predictors provide feature values, which are then integrated to produce a final prediction. In the original version, PSORT I, the integration was done in the style of a conventional knowledge base using a collection of "if-then" rules (Nakai & Kanehisa, 1991, 1992). This makes it very difficult to adjust the rules according to

[1] `PDOC00014` (`PS00014/ER_TARGET`)

[2] `PDOC00013` (`PS00013/PROKAR_LIPOPROTEIN`)

[3] `http://psort.nibb.ac.jp`

information from new data sets; so in order to be able to incorporate new data on a regular basis, the new PSORT II version uses quantitative machine learning techniques, such as probabilistic decision trees and the *k* nearest neighbours classifier to integrate scores from all the features (Horton & Nakai, 1996, 1997). At present, PSORT II is only available in a version trained on yeast data, but it should be possible to train the entire system with the user's own data.

4.2 Signal peptide prediction methods

Prediction of signal peptides involves two tasks: discrimination between signal peptides and non-secretory proteins, and predicting the position of the signal peptide cleavage site. Below, I give performance as correlation coefficient for the first task, and percent correctly placed cleavage sites for the second. However, one should take the performance values too seriously, since they are measured on very different data sets.

4.2.1 Cleavage site weight matrices

The first prediction method for signal peptide cleavage sites was described in the paper that introduced the (-3,-1)-rule (von Heijne, 1983). It is a reduced-alphabet weight matrix combined with a rule for narrowing the search region. The weight matrix covers positions -5 to +1 relative to the cleavage site, using only seven different weights at each position, corresponding to groups of amino acids. The weight values are estimated manually rather than calculated from the data. The weight matrix score is only calculated for positions 12 to 20 counted from the beginning of the h-region—defined as the first quadruplet of amino acids with at least three hydrophobic residues—and the cleavage site is assigned to the position with the highest score. This could place the cleavage site correctly in 92% of the eukaryotic data used to construct it; but measured on a larger data set, the test performance was only 64% (von Heijne, 1986b).

A "real" weight matrix—calculated with log-odds scores as described in section 2.1—was made a few years later (von Heijne, 1986b). A range of window sizes was tested: initially, positions -15 to +5 were used, but this could be narrowed to -13 to +2 without loss in performance. Separate matrices were calculated for prokaryotes and eukaryotes. The regularisation was position-dependent in a rather *ad hoc* manner: no pseudocounts were added to non-zero counts, while counts of zero were set to one before log-transformation, except in positions -1 and -3 where counts of zero were considered to be significant and were normalised to $\frac{1}{N}$ (where N is the number of sequences). When using the weight matrix for testing, the weight matrix score was calculated for the first 40 positions of the protein chain, and the cleavage site was assigned to the position with the highest score.

Training performance in cleavage site prediction was 87% for eukaryotes (N=161) and 100% for prokaryotes (N=36); test performance (seven-fold cross-validation) was 78% for eukaryotes and 89% for prokaryotes. For discrimination between signal peptides and non-signal peptides, the maximum weight matrix score in the first 40 positions was used; performance for eukaryotic sequences (not cross-validated) was 98% correct, corresponding to a correlation coefficient of 0.96.

Method	Paper(s)	1997	1998	Jan–Mar 1999
Weight matrix	von Heijne 1986b	320	247	61
SignalP	paper II, 1997	36	135	54
PSORT	Nakai & Kanehisa 1991, 1992	85	122	47

Table 4.1: The usage of three signal peptide prediction methods, measured by number of citations of the papers describing the methods, according Science Citation Index Expanded (Institute for Scientific Information™). Note that PSORT is a general protein sorting predictor, and this analysis does not show how many of the PSORT citations concern signal peptides. The newer papers describing PSORT II (Horton & Nakai, 1996, 1997; Nakai & Horton, 1999) are not included in this table, but they have been cited less than 20 times in total.

This weight matrix has found extremely wide usage. It does not exist as a WWW-server, but it has been implemented several times (see, *e.g.*, Folz & Gordon, 1987; Popowicz & Dash, 1988), it is included in PSORT (see section 4.1), and it is used in the tool SPScan, which is a part of the widely used Wisconsin Package™ (Genetics Computer Group, GCG), a commercial collection of tools for sequence analysis.[1] The von Heijne (1986b) paper is heavily cited: as of April 1, 1999, it had 3267 citations registered in Science Citation Index Expanded. Measured by citations, it is still the major signal peptide prediction method, although SignalP is now closing in, see table 4.1.

4.2.2 A feature-based method

A different approach was taken by McGeoch (1985) who tested a number of different sequence-derived features to find a combination providing good discrimination between signal peptides and other sequences. Cleavage site location was not attempted.

The two selected features were: length of the uncharged region, and maximal hydrophobicity (on the scale of Kyte & Doolittle, 1982) in an 8-amino acid window. The uncharged region was defined to begin after the last charged amino acid among the first 11 positions and end at the next charged residue, while the maximal hydrophobicity was calculated 18 positions downstream from the start of the uncharged region. A non-linear discriminative function, separating the positive and negative examples in the plane defined by these two features, was determined manually. In the training set, 110 of 114 signal peptides were correct with 39 negative examples all correct (correlation: 0.94). Test performance: 39 of 40 immune system proteins with signal peptides correct; 18 virus proteins (6 positive and 12 negative) all correct.

Originally, this method was based on a very limited data set focusing primarily on virus proteins and immune system proteins, and it could not automatically be transferred to another training set because of the subjective element involved in drawing the separating curve through feature space. However, the method has been integrated into PSORT, where it is used in combination with von Heijne's weight matrix. In PSORT I, the original two features were used for eukaryotic data (Nakai & Kanehisa, 1992), but for prokaryotic data, the method was retrained using discriminant analysis, and a third

[1] From version 9.1 of the Wisconsin Package, the weight matrix in SPScan is recalculated based on the newer SignalP data set.

feature (net charge of the charged region) was incorporated (Nakai & Kanehisa, 1991). For the newer PSORT II, the method has been further refined for yeast and *Bacillus subtilis*,[1] optimising not only the coefficients for the features in the discriminant function but also the parameters used to derive the features, *i.e.*, the length of sequence regions scanned for charged or hydrophobic residues, and the hydrophobicity scale (Nakai, 1996).

In my opinion, the refined version still suffers from a hard limitation in generalisation ability imposed by the rule for finding the start of the "uncharged region." If a signal peptide has a long n-region containing a charged residue after position 11 (there was one such example in the original data set), the "uncharged region" will not contain the h-region, but only a short arbitrary stretch from the n-region. The feature(s) derived from this will probably be totally out of range for signal peptides, leaving the method no chance of producing a reasonable answer.

4.2.3 Neural network methods

A "tiling" network

Ladunga *et al.* (1991) used a neural network for discrimination between signal peptides and cytoplasmic proteins. Cleavage site prediction was not attempted, and a moving window was not used; instead, an N-terminal part (set to 20 residues after initial testing) of each sequence was used as input. Amino acids were represented by sparse encoding. The network was trained with the tiling algorithm, a procedure which builds up the network during training, adding as many hidden neurons as necessary to classify all training data correctly (Hertz *et al.*, 1991).

The training performance was 100%—this is guaranteed by the tiling algorithm. The performance on the test set (not cross-validated) was not very good: 82% correct signal peptides (N=116) and 74% correct cytosolic proteins (N=343), yielding a correlation coefficient of 0.50. However, this could be improved to 93% and 97% (correlation 0.90) by combining the network (in some unspecified way) with the von Heijne matrix. Remarkably, when using the von Heijne matrix alone, they report a performance of only 77% and 84% (correlation 0.57). The authors do not comment on this drastic discrepancy from the performance reported by von Heijne (1986b) (correlation 0.96, see above), but the data set of Ladunga *et al.* (1991) must have been more difficult. Unfortunately, it is not clear from the paper how this data set was constructed.

In my opinion, using a fixed window disregards valuable information, especially around the cleavage site. The c-region of the signal peptide will fall at very varying positions in the input window (or outside it) and will probably not be recognised as a characteristic feature by the network. It also makes the data set much smaller than a moving window method would—this is illustrated by the fact that the most complicated network built by the tiling algorithm had only 3+2+1 computational units (in the hidden and output layers). The tiling algorithm has the valuable property of being able to adjust to the complexity of the data; but I suspect that the combination of guaranteed 100% learning and the simplest possible network carries a serious danger of overtraining (cf. section 2.5).

[1] The *B. subtilis* refinement is not yet implemented in the available server.

An unsupervised network

Arrigo *et al.* (1991) reported that an unsupervised Kohonen network unexpectedly identified the signal peptide region from a set of human insulin receptor gene data. The Kohonen network, also called a self-organising feature map, is an example of an unsupervised artificial neural network, where "training" takes place without target values in the training set (Hertz *et al.*, 1991). The Kohonen network has an input layer and a layer of computational units—the Kohonen nodes. The two layers are fully connected, so that each Kohonen node has a weight vector. The Kohonen nodes are arranged in a way that defines a neighbourhood for each node, *e.g.* a square lattice. When a training example is shown to the network, the Kohonen node whose weight vector is nearest to the input vector (in Euclidean distance) is selected. The weight vectors of the selected node and its neighbours within a certain radius are updated, so that they move closer to the input vector by a factor determined by a learning rate. The radius and learning rate decrease during training. In this way, the Kohonen nodes arrange themselves into a pattern that reflects the structure of the input data.

Arrigo *et al.* (1991) trained a network with 30 Kohonen nodes on non-overlapping windows from the cDNA of four human insulin receptor genes. In each sequence, one of the input patterns was extracted as singular in some not very clearly described way; and it turned out that the extracted pattern was wholly or partly within the DNA coding for the signal peptide for a wide range of window sizes.

However, it is not clear whether this result has anything to do with signal peptides at all. Since the approach was not tested on proteins without signal peptides, the only conclusion to be drawn from this is that the initial part of the reading frame of insulin receptors is in some way peculiar. This might be due to the signal peptide, but it might as well be the effect of correlation between codon bias and intragenic position described in by Bulmer (1988), see also section 3.3.1.

Simulated evolution of signal peptides

Schneider & Wrede (1993, 1994) trained a feed-forward neural network to predict signal peptide cleavage sites using moving windows. Instead of sparse encoding, seven physico-chemical properties were used to represent the amino acids; after training networks with a single property at a time, four of them were selected to represent amino acids in the final architecture. The training was done with a genetic algorithm rather than back-propagation. Three types of architecture were tried: with 0, 1 or 2 hidden layers; and several combinations of outputs from networks with various sizes of the one hidden layer were tested.

All these computations were performed on an extremely small data set derived from *Escherichia coli*: 17 sequences for testing and 7 for training. While comparing the architectures, performance was measured as percent correctly predicted *positions*, so it is not directly comparable to that of the von Heijne matrix. The final predictor, however, had only 3 of the 7 test cleavage site correctly placed when assigned by highest score (Schneider & Wrede, 1994).

After training the predictor, it was used in a "simulated molecular evolution" experiment: a population of 12-aa sequence fragments were subjected to random changes according to a distance metric, and then selected based on their score for being putative signal sequence cleavage sites according to the neural network. In effect, this is a genetic algorithm for optimising the sequence with respect to neural network score, just like the training was a genetic optimisation of the network weights with respect to the error function. After repeating this for many generations, a number of "optimal" cleavage sites were found, the precise sequence depending on the distance metric used (Schneider & Wrede, 1994). Remarkably, these all contained Trp, especially at positions -2 and -5, and they had h-regions dominated by Phe.

The highest-scoring cleavage site region was subsequently tested *in vivo* for their ability to promote secretion in an *E. coli* expression system (Wrede *et al.*, 1998). Indeed, the Phe- and Trp-rich construct (FFFFGWYGWA↓RE) was fully cleavable, but so were the wild type (LAGFATVAQA↓AC) and a "consensus" pattern derived from a simpler, weight matrix-like approach (VVIMSASAMA↓AC).

Although this whole process is based on statistics from only 24 sequences, the result raises an interesting point: when using a linear method, the optimal example looks like a consensus of the training examples; but for a non-linear method, this is not necessarily the case. It is remarkable that the highest-scoring neural network examples are very rich in otherwise rare amino acids. A possible explanation could be that these amino acids have extreme values in the features used for input encoding and therefore give higher scores; if this is so, the effect should not be seen when using a network trained with sparse encoding. In my opinion, a more likely possibility is that the selected amino acids are those that show the largest biases in the window positions they occupy when they do occur in the training data. And since bias is easily overestimated by sampling in small data sets, rare amino acids may have larger apparent biases, and therefore larger impact on the network output score.

So, is there any reason to expect that the non-linearly optimised "FFFFGWYGWA↓RE" is a more efficient cleavage site than the linearly optimised "VVIMSASAMA↓AC"? I think not. Even if we assume that the peculiar residues represent non-linear features found in the training data (and not just an effect of sampling error as I suspect), the highest neural network score is found in a region of sequence space not covered by the training data, implying that the network score here is an *extrapolation* rather than an *interpolation*. And since neural networks do not contain any model of how scores should vary with the input, but simply fit a non-linear function to the examples, a good generalisation in interpolation does not necessarily mean a good generalisation in extrapolation. The more non-linear the fitted function is, the less we can assume about how it should continue outside the region of the fitted data. This does not mean that I find the Wrede *et al.* (1998) work uninteresting—on the contrary, experimental investigation of examples outside the training data sequence space (*e.g.*, computationally optimised sequences) might provide a hint about the extent of the non-linearity, see section 7.3 on page 79 for a further discussion of this.

4.3 Prediction of mitochondrial transit peptides

The currently most developed method to predict mitochondrial transit peptides (mTPs) is Mitoprot[1] (Claros & Vincens, 1996). It is a feature-based method, using a linear

[1] http://websvr.mips.biochem.mpg.de/cgi-bin/proj/medgen/mitofilter

combination of a number of sequence characteristics such as amino acid abundance, maximum hydrophobicity, and maximum hydrophobic moment (α-helix amphiphilicity), that are combined into an overall score (Claros & Vincens, 1996).

Neural networks have also been used for predicting cleavage sites of mTPs. A complicating factor here is that three different types of consensus are described, with an Arg in either the -2, -3, or -10 position; where the $\mathtt{R}^{-2}$ and $\mathtt{R}^{-3}$ sites are thought to represent cleavage by MPP (mitochondrial processing peptidase), while $\mathtt{R}^{-10}$ sites presumably represent subsequent cleavage by MIP (mitochondrial intermediate peptidase). Schneider *et al.* (1998) used a Kohonen network to classify all cleavage sites into one of these three groups, and then trained feed-forward networks on data from the three groups separately.

Because of the high number of false positives seen when scanning sequences with known MPP and MIP cleavage sites, the authors conclude that the these networks are not good enough to predict mTP cleavage on their own. However, when using negative examples only from the cleavage site region, fairly high test set correlation coefficients (0.67–0.90) are obtained; so if the region to be searched for cleavage site could be narrowed in a way analogous to what we did for chloroplast transit peptides in ChloroP, I do not think prediction of mTPs cleavage sites seems at all impossible. Furthermore, preliminary work using the same approach as for ChloroP suggests that performance levels similar to those of Mitoprot can be reached with neural networks (Emanuelsson, Nielsen, & von Heijne, unpublished).

4.4 Transmembrane protein topology prediction

Recognising transmembrane proteins is, in a sense, a protein sorting prediction task, because the membranes can be viewed as compartments. However, the protein sorting aspect of transmembrane protein prediction has not been very much in focus—the discrimination between transmembrane proteins and peripheral membrane or soluble proteins is seldom reported (for exceptions, see Klein *et al.*, 1985; Rost *et al.*, 1996a). Instead, the published methods concentrate on determining the location of transmembrane helices in the sequence, and in many cases also the topology, *i.e.*, which segments are on the cytoplasmic and non-cytoplasmic faces of the membrane. A good topology predictor is probably also a good discriminator; but the actual tests are too scarce to say whether this is true.

Apart from this, scarcity is not the most conspicuous property of the transmembrane protein prediction field. In contrast to signal peptide prediction, this is a very "crowded" area with a lot of competing high-performance methods, as the non-exhaustive list below shows. I find this surprising, because the data are much more difficult to handle: for the vast majority of transmembrane proteins, topology is inferred by similarity or prediction, and in the experimentally verified cases, the borders between transmembrane and extramembrane regions ("loops") are annotated according to varying principles. Only very few (approximately ten after homology reduction) high-resolution 3D structures of transmembrane helix proteins are available; and even for these, there is no unambiguous definition of the exact borders: does a transmembrane helix end when the protein chain leaves the plane defined by the membrane surface (which in itself is a rather ill-defined concept), or when it adopts a non-helical conformation?

Because of this ambiguity in the data—and because topology prediction is rather peripheral to protein sorting prediction—I have not attempted to report performances in the list of methods below. With one exception, I have only included methods that are publicly available over the WWW.

Hydrophobicity analysis

The "canonical" method for prediction of transmembrane regions is to compute average hydrophobicity in a moving window along the sequence, and assign those segments as transmembrane that are more hydrophobic than a certain threshold. This simple approach leaves several parameters open for tweaking: the choice of hydrophobicity scale, the threshold, the window size, and heuristics for deciding whether a long hydrophobic stretch should be assigned as one transmembrane helix or two. One such method, ALOM (Klein *et al.*, 1985), is integrated into PSORT[1] (Nakai & Kanehisa, 1991, 1992) with one modification: a more stringent threshold value is used for finding the most hydrophobic segment (and thereby classifying a protein as transmembrane), and a less stringent one is used for assigning additional TM helices in a multispanning protein.

Several other methods extend the hydrophobicity analysis in various ways. SOSUI[2] (means "excluding water" in Japanese, Hirokawa *et al.*, 1998) also differentiates between most hydrophobic and additional TM helices in multispanning proteins; for the latter, it calculates an amphiphilicity index in addition to hydrophobicity.

TopPred[3] (von Heijne, 1992) uses the positive-inside rule to post-process the hydrophobicity analysis: based on two threshold values, it assigns "certain" and "putative" transmembrane helices, and then tests alternative models with and without the putative helices to find the one that maximises the difference in Lys+Arg content between inside and outside. It employs a trapezoid window for calculating hydrophobicity, *i.e.*, positions in the flanking part of the window are downweighted.

PRED-TMR[4] (Promponas *et al.*, 1998) extends the hydrophobicity analysis with a recognition of the helix "caps"—*i.e.*, the transition regions between membrane and loop domains—by a weight matrix-like approach.

Statistical methods

Instead of using a hydrophobicity scale derived from physical or chemical measurements or molecular modeling, parameters may be derived directly from a training set of transmembrane proteins. TMAP[5] (Persson & Argos, 1994) uses one set of propensity values for transmembrane regions and another for cap regions; while TmPred[6] (Hofmann & Stoffel, 1993) uses a combination of several weight matrices for scoring inside-to-outside and outside-to-inside helices, and then builds the most likely model. As an additional extension, TMAP is designed to work with multiple alignments of related transmembrane proteins, giving a better performance than on single sequences.

[1] `http://psort.nibb.ac.jp`

[2] `http://www.tuat.ac.jp/~mitaku/adv_sosui/`

[3] `http://www.biokemi.su.se/~server/toppred2/`

[4] `http://o2.db.uoa.gr/PRED-TMR`

[5] `http://130.237.130.32/tmap/`

[6] `http://www.isrec.isb-sib.ch/software/TMPRED_form.html`

The statistical approach is refined in MEMSAT[1] (Jones *et al.*, 1994), which uses propensity values for transmembrane regions, inside and outside caps, and inside and outside loops, calculated separately for single-spanning and multispanning membrane proteins. The most probable prediction is made by fitting a sequence to the set of propensity regions using dynamic programming, which makes the method very reminiscent of a hidden Markov model.

One method which is difficult to categorise is DAS[2] (Cserző *et al.*, 1997), which instead of hydrophobicity uses a special amino acid similarity matrix, the Dense Alignment Surface (DAS) matrix. The query sequence is compared to a collection of non-homologous membrane proteins using a moving window, and peaks in similarity score are assigned as transmembrane helices.

Neural networks

PHDhtm (Rost *et al.*, 1995, 1996a,b) is a neural network-based method, part of the extensive protein structure prediction server PredictProtein[3] (Rost, 1996). A primary network predicts the tendency of each position to be in a transmembrane helix, and the outputs from this are post-processed both by a second neural network and a dynamic programming optimisation to produce a topology prediction. PHDhtm is able to use information from multiple alignments, and the PredictProtein system even does the database search and alignment automatically.

Hidden Markov models

Two HMM-based methods are available: TMHMM[4] (Sonnhammer *et al.*, 1998) and HMMTOP[5] (Tusnády & Simon, 1998). These both have architectures that are reminiscent of the MEMSAT model: groups of tied states are used for transmembrane regions, caps (or "tails"), and loops; and these region models are combined into a cyclical architecture. The most important difference lies in their way of testing sequences: while TMHMM decodes an already trained model (as is the case for most HMM prediction applications), HMMTOP *retrains* its model on the query sequence, using initial parameters and pseudocounts derived from a training set. The hypothesis behind this approach is that the topology is determined by the *differences* in amino acid composition between regions rather than by the specific composition of each region. The performance can be enhanced by using multiple related sequences in the retraining, but unlike TMAP and PHDhtm, prediction is not done on an alignment.

4.5 Amino acid composition-based methods

In addition to the recognition of the sorting signals, prediction of protein sorting can exploit the fact that proteins of different subcellular compartments differ in global prop-

[1] Not on the WWW, but available as a PC program for download from `ftp://ftp.biochem.ucl.ac.uk/pub/MEMSAT/`

[2] `http://www.biokemi.su.se/~server/DAS/`

[3] `http://dodo.cpmc.columbia.edu/predictprotein/`

[4] `http://www.cbs.dtu.dk/services/TMHMM-1.0/`

[5] `http://www.enzim.hu/hmmtop/`

erties, reflected in the amino acid composition. While the signal prediction methods are probably closer to mimicking the information processing in the cell, methods based on global properties can complement imperfect signal-based methods, especially on incomplete sequences. Specifically, a composition-based method for recognising extracellular proteins can be used without knowledge of the N-terminus, and could give correct predictions for, *e.g.*, EST-derived protein fragments where the signal peptide has not even been sequenced. One drawback is that such methods will not be able to distinguish between very closely related proteins that differ in the presence or absence of a sorting signal.

Nakashima & Nishikawa (1994) used simple odds-ratio statistics to discriminate between soluble intracellular and extracellular proteins on the basis of amino acid composition and residue-pair frequencies. Performance was 88% and 84% correct in the two categories. Including biases in residue pairs ($[n, n+1]$ to $[n, n+5]$) improved performance by 8% relative to amino acid composition alone. However, none of the later methods have incorporated pair frequencies, as far as I am aware.

Cedano *et al.* (1997) extended the number of possible locations to five: intracellular, extracellular, transmembrane, membrane-anchored, and nuclear, and used the so-called Mahalanobis distance to discriminate. This metric takes interactions between amino acids into account (note: not interactions between positions; the input is only the 20 aa frequencies) and is therefore able to handle non-linear mappings in the 20-dimensional space defined by the aa composition. The reported performance of their algorithm, named ProtLoc,[1] was 76% of all 5 classes (it is not clear whether this is a test or training performance).

This approach has been refined in three recent papers by Chou & Elrod (1998, 1999a,b). They use a modified version of the Mahalanobis distance, where an extra term compensates for differences in size between the categories. The test performances were: cytoplasmic, periplasmic, and extracellular bacterial proteins: 86.5% (Chou & Elrod, 1998); five classes of transmembrane and membrane-anchored proteins: 76.4% (Chou & Elrod, 1999a); membrane proteins from nine different (plasma or organellar) membranes: 65.9% (Chou & Elrod, 1999a); "outer" *vs.* "inner" membrane proteins from both Gram-negative bacteria and eukaryotic organelles: 87.8% (Chou & Elrod, 1999a); twelve different subcellular locations (eleven aqueous compartments plus plasma membrane): 79.9% (Chou & Elrod, 1999b).

One rather disturbing aspect of the Cedano *et al.* and Chou & Elrod papers is the definition of the data sets: several classes contain both eukaryotic and prokaryotic proteins lumped together, to be distinguished from classes which would not make sense for prokaryotic proteins (such as ER or mitochondrion). This should make the classification task more difficult. When the overall performance nevertheless seems surprisingly high (in comparison, PSORT II using 11 locations reaches only 57%–63% in overall performance, Nakai & Horton 1999), I get the suspicion that it could be an overestimate due to poor homology reduction (cf. section 5.2): only proteins with the same *name* were excluded from the data set. The test performances in the Chou & Elrod papers are calculated by leave-one-out jackknife, which is extremely sensitive to data set redundancy, or by what the authors call an "independent" test set, which seems to consist of the sequences that were removed during homology reduction!

[1] Not on the WWW, but can be downloaded from `ftp://luz.uab.es/pub/ProtLoc/`

The NNPSL method[1] (Reinhardt & Hubbard, 1998) uses neural networks trained on overall amino acid composition to predict location. Test performances (cross-validated) are 81% for three bacterial compartments (cytoplasmic, periplasmic, and extracellular) and 66% for four eukaryotic compartments (cytoplasmic, extracellular, mitochondrial, and nuclear). Interestingly, plant proteins were found to be very poorly predicted, and are not included in the present method.

Why does the amino acid composition approach work, if it is not able to detect the sorting signals? It is no mystery that discrimination of transmembrane *vs.* soluble proteins is possible, since the strong hydrophobicity of the transmembrane helices influences the amino acid composition; and the discrimination of inner *vs.* outer transmembrane proteins should also be quite easy, since these are generally α-helix *vs.* β-sheet proteins, respectively. It is more surprising that discrimination between soluble proteins of different compartments by amino acid composition is possible. One contributing aspect can be the disulfide bridges, that only occur in proteins of extracellular or secretory compartments; but Cys content alone can hardly be the whole story. A more plausible explanation is that the protein surfaces reflect the chemical properties (acidity, ion concentrations, etc.) of their compartments. Andrade *et al.* (1998) found that the signal in the total amino acid composition which makes it possible to identify the subcellular location, is due almost entirely to surface residues. The surface residue signal was often strong enough to accurately predict subcellular location, given only a knowledge of which residues are at the protein surface. These results suggest that the accuracy of prediction of location from sequence might be improved by combination with a surface (solvent accessibility) prediction.

[1] `http://predict.sanger.ac.uk/nnpsl/`

Chapter 5

Results and discussion: The data sets

Within the broad range of machine learning methods available, various algorithms have different advantages in terms of their pattern recognition abilities; but they are all driven by the data used to train them. The selection of the training set is arguably the most important part in the construction of a prediction method. No matter how sophisticated the algorithm, with poor training data one will get poor results, and with high homology performances can be overestimated. In this chapter, two aspects of the data set construction are discussed using my own work as examples: how to extract the data from general purpose sequence databases, and how to reduce homology.

5.1 Data set extraction

When building a data set for protein sorting prediction, several choices have to be made concerning exactly what to include as positive and negative examples. In the signal peptide case, it is quite clear how the positive data sets should be defined—although it may be argued whether, *e.g.*, bacterial lipoproteins should be considered as positive examples—but there are many questions to be asked about negative examples: Should they comprise only soluble cytoplasmic and nuclear proteins, or include transmembrane and membrane-associated proteins? Should they be limited to N-terminal parts or include entire protein chains? In the following, I will present and discuss the choices made for the data sets used in SignalP, ChloroP, and NetStart.

Once these choices are made, another question is how to identify the sequences of interest in the database annotations. In practice, the two questions cannot be separated, because the choice of positive and negative examples is limited exactly by the availability and quality of annotations.

For proteins sorting signals, SWISS-PROT[1] (Bairoch & Apweiler, 1999) is the natural primary source of sequence data because of the rich annotations and the high level

[1] `http://expasy.hcuge.ch/sprot/`

of maintenance. Unlike the large nucleotide sequence repositories (GenBank,[1] EMBL,[2] and DDBJ[3]), SWISS-PROT is a *curated* database where entries are actively maintained by database staff. This means, among other things, that a SWISS-PROT entry actually will get updated if you report a factual error in the annotation, while GenBank entries are "owned" by their contributors and, as a rule, can only be updated by them.

Another possible source is the protein sequence database of PIR[4] (Protein Information Resource, Barker *et al.*, 1999). PIR is also a curated database, but is generally not as richly annotated as SWISS-PROT. Of course, this does not exclude the possibility that it might have higher quality of annotations for a specific feature such as signal peptides, but I have not checked that.

Even in a well-curated database such as SWISS-PROT, one cannot take all the sequence annotations at face value. In general, we have tried to secure experimental evidence for the data, but as I describe below, this is not without problems.

5.1.1 Signal and transit peptides

Information about protein sorting signals are found in the *feature tables* of the databases. Although formats vary between SWISS-PROT, PIR, and GenBank, a feature table for a sequence generally contains a number of entries giving feature type, positions (begin and end), and optional comments with qualifying information about the feature. From SWISS-PROT, signal peptides are selected by the feature type SIGNAL, and chloroplast transit peptides by the feature type TRANSIT with the description CHLOROPLAST.

Ideally, the end position of a sorting signal feature should correspond to the cleavage site of the sorting signal processing enzyme (signal peptidase or stromal processing peptidase). However, this is not always the case. First, the site given in the database may not be experimentally determined, but based on alignment to another protein, an existing prediction method, or maybe just an informed guess. We have attempted to avoid these examples by discarding entries missing begin or end positions, or with the comments POTENTIAL, PROBABLE, or BY SIMILARITY. In principle, a SWISS-PROT feature table entry without such indicators should mean that the information is experimentally determined; but we have found examples where this was not true (see papers I and VI).

Furthermore, also experimentally verified sites may be wrong, if the interpretation of the results has been faulty. Even if the end position of a sorting signal is determined by N-terminal peptide sequencing of the mature protein, it is not verified that this is the site of signal peptidase or stromal processing peptidase—only experiments performed in reconstituted *in vitro* systems provide definitive answers to this question. For eukaryotic secretory proteins, the N-terminal of the mature protein may result from subsequent cleavage by proteases in the secretory pathway recognising monobasic or dibasic cleavage sites (Seidah & Chrétien, 1997; Nakayama, 1997), and the quality of the signal peptide data set could probably be improved by a recognition of these sites.

[1] http://www.ncbi.nlm.nih.gov/Web/Genbank/index.html

[2] http://www.ebi.ac.uk/ebi_docs/embl_db/ebi/topembl.html

[3] http://www.ddbj.nig.ac.jp/

[4] http://pir.georgetown.edu/

For chloroplasts, the mature protein may be a thylakoid protein resulting from dual cleavage of a composite signal sequence; we used SignalP to scan annotated transit peptides for thylakoid signal sequences, and found 29. In addition, an *in vitro* result with purified SPP showed that 5 out of 6 precursor proteins were processed between Arg/Lys and Ala, while cleavage sites listed in SWISS-PROT tend to have Arg in positions -2 and -3, suggesting that a hypothetical stromal protease can remove one or a few N-terminal residues after the initial cleavage catalysed by SPP. This is supported by our results, showing that the most conserved pattern in most sequences occur N-terminal to the annotated cleavage site paper VI.

In some cases, the feature table suggests alternative cleavage sites. Although alternative cleavage by signal peptidase probably occurs *in vivo* (von Heijne, 1984), I did not include these entries, because experimental evidence for this phenomenon is scarce and difficult to distinguish from subsequent cleavage by other peptidases.

The SignalP training data do not include bacterial signal peptides cleaved by signal peptidase II (see section 3.1.4 on page 27), since the cleavage sites of these proteins differ considerably from those cleaved by the standard prokaryotic signal peptidase (Lep). However, experience shows that SignalP generally predicts these sequences as signal peptides, although often with a different cleavage site.

5.1.2 Negative sets for signal and transit peptides

As a background to the signal peptides, we extracted data sets comprising the N-terminal parts of cytoplasmic and (for the eukaryotes) nuclear proteins. This was done by searching for comment lines in SWISS-PROT specifying the subcellular location (see paper IV for details. Only the first 70 amino acids of each sequence were included in the data sets.

From each signal or transit peptide entry, the sequence of the signal or transit peptide and a part the mature protein were included in the data set. Thus, the initial part of the mature secreted protein, along with the cytoplasmic and nuclear proteins, served as negative data for signal peptide or transit peptide score. For SignalP, the length of this part was set to 30 amino acids after the cleavage site based on the approximate length of the hypothetical “export initiation domain,” see section 3.3.2 on page 35. For chloroplast transit, no equivalent “import initiation domain” is known, so we used the average length of the transit peptides instead. For SignalP-HMM, a constant sequence length after the cleavage site would have been a bad choice, because the HMM could have used this to predict cleavage site at a constant distance from the C-terminus without learning cleavage site properties at all; instead we used a constant length of 70 aa (paper V).

Why not simply use the entire sequences? The decision to use only the N-terminal part of each protein was based on the idea that SignalP should reproduce the recognition task met by the cell *in vivo*, where signal peptide cleavage takes place only within a certain range from the N-terminus. This has probably made the prediction problem easier, because potential false positives from other parts of the proteins have been excluded; but I have never measured the false positive rate on downstream sequences.

5.1.3 Membrane proteins

While there is an idea behind not using the entire sequences, the reason for the lack of transmembrane proteins in the negative sets is more pragmatic: since some membrane proteins do have cleaved signal peptides, it is very hard to ensure that there is experimental evidence for *absence* of cleavage. Even for membrane proteins where the location of each transmembrane segment is determined experimentally, there is no guarantee that the sequence of the N-terminal tail does *not* include a cleaved signal peptide.

An exception to this is the signal anchors: N-terminal parts of type II transmembrane proteins (see section 3.2 for a definition). These were extracted from SWISS-PROT using the feature type TRANSMEM with the description SIGNAL-ANCHOR (TYPE-II MEMBRANE PROTEIN). In several cases, the cytoplasmic domain preceding the signal anchor were marked POTENTIAL or PROBABLE even if the signal anchor itself was not, meaning that the topology is probably not experimentally verified, and the sequence might be an inverted signal anchor (type III membrane protein). Still, we regard this as experimental evidence that they are not signal peptides—otherwise, the remaining data set would have been too small for doing any statistics.

However, if the principle of only using N-terminal sequences is relaxed, it would be possible to obtain fairly large negative data sets from transmembrane proteins, simply by disregarding the N-terminal tail as "unknown" and using the rest of the sequence from the first experimentally determined transmembrane segment. N-terminal tails might also be included, if they are too short to contain a signal peptide. This would probably be a considerably more difficult discrimination task for SP prediction, since transmembrane helices of multispanning membrane proteins can have intermediate hydrophobicities reminiscent of h-regions, but I do not think transit peptide discrimination would suffer.

5.1.4 Selection of organisms

Another aspect of the choice of training set is whether sequences from all species, some group of species, or only a single organism should be included. If there is enough data, organism-specific methods should be expected to perform better than more general ones, but it is in most cases not possible to be this restrictive.

Both SignalP and SignalP-HMM are trained on three different data sets: eukaryotes, Gram-negative and Gram-positive bacteria. These three versions reflect significant differences in the characteristics of signal peptides from these groups of organisms, and each gives a better performance than a method trained on all groups together (this has been tested for SignalP-HMM; results not shown).

For SignalP, I also trained two species-specific versions on human and *Escherichia coli* SPs, and concluded that there was no significant gain in performance when testing with networks trained on a single-species data set relative to networks trained on larger groups (paper IV). This result is not definitive, however. The reason why the *E. coli*-specific network did not show an improvement compared to one trained on a larger set of Gram-negative SPs might simply be that the *E. coli* set at that time was too small to achieve the same relative performance. Regarding the human-specific network, one should note that the eukaryotic set is very dominated by mammals, *i.e.* rather close relatives to humans; so this result does not exclude the possibility that signal peptides from, *e.g.*, yeast (which are relatively underrepresented in my data set), are significantly different from those of mammals.

The division of the bacterial SignalP data into Gram-positive and Gram-negative is based mainly on a statistical study done ten years ago (von Heijne & Abrahmsén, 1989). It was found that SPs from the Gram-positive species *Bacillus*, *Staphylococcus*, *Streptococcus*, and *Streptomyces* were all quite similar and different from those of *Escherichia coli*, but the *E. coli* data were not compared to other Gram-negatives. Indeed, grouping all Gram-negatives together may be a bad choice, and newer bacterial systematics does not regard this category as a systematic group. In the newest release of SWISS-PROT (Bairoch & Apweiler, 1999), the taxonomy has been changed to that of GenBank, which does not use the Gram-positive *vs.* Gram-negative (*Firmicutes* vs. *Gracilicutes*) concept.

Mycoplasma (*Tenericutes* or *Mollicutes*) were excluded, since they may not have cleaved signal peptides at all: in the minute genome of *Mycoplasma genitalium*, no homologue of signal peptidase I has been found (Fraser *et al.*, 1995). Archaea (*Mendosicutes* in older SWISS-PROT terminology) were excluded from the training set because of lack of experimental data, and because it is not at all clear whether their SPs should be expected to be most related to eukaryotic or bacterial ones. Instead, we have attempted to characterise Archaeal SPs by using SignalP on the genome of *Methanococcus jannaschii*, see section 6.4.

Regarding the eukaryotic data, there are reasons to believe that yeast and plant SPs are special. The yeast targeting and translocation apparatus shows several deviations from the mammalian one, the most important probably being the possibility for post-translational targeting (see section 3.1.2 on page 24), and there is at least one example of a yeast signal peptide, carboxypeptidase Y, that does not function in mammalian cells (Bird *et al.*, 1987). Plant SPs showed only very slight differences from human SPs in the von Heijne & Abrahmsén (1989) study, but new neural network training results (Emanuelsson, Nielsen, & von Heijne, unpublished) suggest a lower performance in signal peptide discrimination for plant than for animal sequences, and another neural network protein sorting prediction method based on amino acid composition simply gave up on plant proteins (Reinhardt & Hubbard, 1998).

When the SignalP data set was made, there were not enough yeast examples with experimentally determined cleavage sites available to do a test on these separately. However, recent growth in databases, and especially the intense work on functional annotation of the yeast genome, may have changed this situation. In addition, experimentally determined cleavage sites are not absolutely necessary for characterisation of differences. Statistics can be made on predicted examples, as in the *Methanococcus jannaschii* work presented in section 6.4. With SignalP-HMM, it may even be possible to "bootstrap" the prediction procedure: a more general version trained on, *e.g.*, all eukaryotic sequences, could be used to extract an initial set of reliably predicted sequences from, *e.g.*, yeast, which is then used to iteratively train a species-specific version.

For the ChloroP data, it was a natural choice to limit the negative data (cytoplasmic, nuclear, secretory, and mitochondrial proteins) to plants. However, not enough different plant mitochondrial sequences were available, so in this category we included data from other eukaryotes as well. This choice can be justified by an earlier study showing that mitochondrial targeting peptides do not differ in any major way between organisms (Schneider *et al.*, 1998).

NetStart has been trained on two data sets, vertebrates and *Arabidopsis thaliana*, based on earlier findings that plant and animal start codon context is different (Lütcke *et al.*, 1987; Cavener & Ray, 1991). Statistical analyses (Pedersen, Nielsen, & Brunak, in preparation) have shown that the variation in local start codon context follows the systematic groups of eukaryotes, and that differences between various vertebrate species are indeed insignificant. This was also the case for various species of dicotyledonous plants, so in future versions, the *Arabidopsis* set should be extended. Additional groups showing significant differences to vertebrates and dicot plants (and to each other) were monocot plants (rice, wheat, maize, and barley), *Caenorhabditis elegans*, *Drosophila melanogaster*, *Saccharomyces cerevisiae*, and *Schizosaccharomyces pombe*.

5.1.5 Finding database errors

As described above, non-experimental evidence or misinterpretations of experiments should be expected in data sets derived from general databases. In addition, sequences or annotations are sometimes incorrectly entered into the database (database "typos"). Therefore, a data set should ideally be checked by hand against the primary publications. This was done with the positive set for ChloroP (*after* homology reduction, see paper VI for details), but the SignalP data sets are too large for this.

One problem with checking the references to a SWISS-PROT data set is that while one entry may have several references, there is very little information about which reference provided which feature,[1] making it necessary to leaf through several papers to find the one that mentions the signal peptide.

If the data set is too large to allow for manual inspection of all entries, some suspicious-looking examples may be identified by automated methods. First, one should look at the distribution of feature lengths (and possibly other simple statistics) and check the most extreme examples. A second possibility, which has been used with the signal peptides, is to use alignments of the unreduced set to single out pairs of sequences that show a very high similarity but discrepancies in assignment of subcellular location or cleavage site position (paper I). Another method is to use the training algorithm itself to pick out cases which are more difficult to learn than others (Brunak, 1993). All these approaches are necessarily biased; the first will only find the extremes, the second will never be able to pick up errors in sequences with no matching homologues, and both the second and the third can fail to recognise systematic errors that occur in several entries. Still, experience has shown that machine learning methods can serve as extremely useful tools for data set validation; in several cases, neural networks have been able to detect errors caused both by simple misprints and by incorrect interpretation of experiments (Brunak *et al.*, 1990b,a).

Every time a discrepancy between a curated database entry and an original reference has have been found, the error should be reported to the database—even if it is only a missing note about lack of experimental evidence. In this way, we as bioinformaticists

[1] An exception is the RP (reference position) line which gives information about which part of the sequence the reference in question has provided, and whether it was done by nucleotide or amino acid sequencing—an experimental determination of a cleavage site may show up here as a partial amino acid sequence.

can give our contribution to the improvement of database quality and hopefully avoid an explosion of incorrect annotations based on circular evidence.

5.2 Homology reduction

Nucleotide and protein sequence databases are redundant due to the presence of orthologous sequences, paralogous sequences, identical sequences submitted more than once (perhaps under different names), and sequence variants (*e.g.* different alleles or mutations). In a curated database such as SWISS-PROT, multiple submissions of the same protein from the same species are, as a rule, merged into one entry; but closely related proteins are kept separate. Furthermore, certain families of genes have been the focus of special attention and are therefore overrepresented compared to other genes.

This is problematic for two reasons. First, statistical analyses will be biased towards the large families, which may be overrepresented due to reasons that are more related to their medical or economic importance than to their biological function. Second, the performance of prediction methods will be overestimated if the test set is not independent from the training set, *i.e.*, if it contains sequences closely related to those used in the training (cf. section 2.5).

After selecting an initial set of sequences, the data set should therefore be diluted by removing examples until no pairs of too closely related sequences remain. This is most often referred to as *redundancy reduction*, for example in some of my own earlier work (papers III and IV). However, the term *redundancy* in the context of biological sequences has been overloaded by database administrators, who use it in the sense of *identical* rather than *homologous* sequences; *e.g.*, SWISS-PROT is described as a “non-redundant” database, because information about the same protein sequenced by several groups is merged into a single entry. To avoid confusion with this (much simpler) concept, I now prefer the term *homology reduction*, specifying that the goal of the process is to eliminate biases originating from homologous sequences.

However, the question of when two sequences are “too closely related” to be kept within the reduced data set is far from trivial: it raises the twin questions of how to measure sequence similarity, and how to choose a meaningful similarity cutoff. In the following, I will present two different approaches developed within this project: one, used for the SignalP data set, regards two sequences as too closely related if the prediction problem can be solved by alignment rather than prediction; the other, used for the NetStart and ChloroP data sets, employs statistically significant similarity as the criterion. In the first approach, the answer necessarily depends on the problem under consideration, because the sequence annotation is used for finding the threshold; while the second approach only takes the sequences into account.

Once these problems are dealt with, efficient methods exist for removing sequences in such a way that the remaining set does not contain any pair of “neighbours,” *i.e.*, members that have similarities above the threshold. Hobohm *et al.* (1992) proposed two algorithms for this purpose. The first, “select until done,” uses a sorted list of examples: at each iteration, it selects the top one into the data set and makes comparisons to all the others, removing those from the list that are too similar; and this is repeated until the list is empty. The second, “remove until done,” needs the total matrix of pairwise comparisons: at each iteration, it counts the number of neighbours and removes the example that has most neighbours from the data set; this is repeated until no examples with neighbours remain. For a highly redundant data set, algorithm 1 can significantly reduce the number of comparisons that have to be made, but algorithm 2 will often find a more optimal solution, *i.e.*, remove fewer sequences given the same threshold.

Removing homologous sequences is actually not the best solution. Even though redundant information should not be allowed to bias the result, it is a waste to disregard the information about variability and conservation that exists in related sequences. Instead, one could use a *weighting scheme* that regulates the influence of each example in the data set—provided that the modeling software to be used supports weighted sequences. The simplest approach is to divide the data set into clusters defined by the similarity threshold, and then let each cluster count as one effective example, *i.e.*, divide the influence of each sequence by the number of members in its cluster.

This does not remove the need for a sensible definition of sequence neighbours, and it also leaves open the choice of clustering algorithm: should a sequence be added to the cluster if it has one neighbour in the cluster (single linkage), or only if it is a neighbour to all the members of the cluster (complete linkage), or something in between?

Letting each cluster count as one may be too radical: it can be argued that large clusters should be allowed more influence, because they represent a larger variation in sequence space (but of course they should not have influence proportional to the number of members—this would not reduce the bias at all). In addition, it is hardly correct to weight all members of a large cluster evenly without taking the pattern of relations within the cluster into account. These points can be addressed by constructing a phylogenetic tree of each cluster and use a tree-based weighting scheme. This adds the question of which of the many possible phylogenetic reconstruction methods to use; and even given the correct tree, there is no consensus about how to derive the weights (see Durbin *et al.*, 1998, section 5.8, for an overview of weighting schemes).

5.2.1 Finding a threshold: The function-based approach

The idea behind the function-based approach is that if it is possible to infer a functional property of one sequence by aligning it to another having the same property, the sequences are too similar. This was inspired by the work of Sander & Schneider (1991), who used it in the context of protein structures: they suggested that if an alignment between two protein sequences had identical secondary structure in more than 70% of the positions, it would be strong enough to predict 3D structure; and they used alignments between proteins of known structure to find a sequence similarity cutoff corresponding to 70% secondary structure identity.

For application to signal peptide (SP) cleavage sites, we defined the criterion thus: if it is possible to infer the position of the cleavage site in one SP by alignment to another SP, the sequences are too similar (paper I). This makes it possible not only to evaluate threshold, but also to evaluate similarity measures: what is the correlation between the similarity value and the probability for finding the cleavage site by alignment?—or in other words, how well can the similarity score predict whether the two cleavage sites are aligned?

It should be stressed that this approach is strictly problem-specific, and that our application focuses exclusively on cleavage site location—if we had investigated the predictive power of alignments to discriminate between SPs and non-secretory proteins, the conclusions might have been totally different.

One of the interesting findings was that the choice of substitution matrix used in the pairwise alignments were very important for the results: a matrix with higher relative entropy (corresponding to smaller evolutionary distance) gave a much higher performance. It did not matter whether the substitution matrix was a simple identity matrix that only measures whether amino acids are identical or not, or a PAM matrix which weights substitutions differently. This was quite surprising, since identity matrices are known to be poor at finding related sequences in a database search.

Here, I will extend the discussion in paper I by a few considerations that may shed some light on this finding. First, SPs are less conserved than the mature protein, which means that the alignment reliability is lower in the SP. Therefore, the alignment at the cleavage site may be wrong, even though the whole alignment is statistically significant, if the significance derives from a region in the mature protein. A high-entropy matrix produces shorter local alignments, and may therefore be less disposed to this type of errors.

Second, cleavage sites may actually change position during evolution, which means that an alignment can be even evolutionary correct, but still place the two cleavage sites differently. It is not known how often this actually has happened, but a shift in cleavage site has been observed for mutagenised SPs after changing only one amino acid (see, *e.g.*, Fikes *et al.*, 1990). If it is not a rare event, the evolutionary distance between two SPs should be quite short for a cleavage site assignment to be trusted, and this might explain why a high-entropy matrix, designed for short evolutionary distances, is better.

It should be noted that the high-entropy matrix is not necessarily the one that finds most cleavage sites; but it gives the best discrimination between correctly found cleavage sites and false positives. Thus, our results have significance for the use of alignment for prediction: if you annotate sequences by similarity, you want to be able to predict whether you can trust the deduced features. And apparently, an inferred SP cleavage site can not be trusted, unless the similarity is strong enough to be detected by a high-entropy identity matrix. This could mean that many “by similarity” annotations in the databases with respect to cleavage sites are unsure—it could be very interesting to investigate how strong similarities these annotations are actually based on.

However, there are two limitations to this generalisation. First, the alignments were done with the SP plus 30 amino acids of the mature protein—these were the sequences intended for use as training data for SignalP, but for doing annotation by alignment it is a totally arbitrary choice, and other possibilities might perform better. Second, it is not necessarily true that local alignment is the best choice; global alignment could make sense if applied to the entire protein. Maybe an even stronger tool would be a semi-global approach: finding the best alignment that includes both N-termini but need not include any C-termini? This can be defined easily by a slight modification of the Smith-Waterman alignment (Durbin *et al.*, 1998, p. 28), but it is not to my knowledge implemented in any publicly available alignment program.

The function-based approach can in principle be applied to any functional sites with a specified location, but there may be practical problems. With chloroplast transit peptide cleavage sites, the problem would be the low confidence in the precise cleavage site positions; in that case, the criterion might be relaxed to a placement of the two cleavage sites within a maximum distance in the alignment. With start codons, the choice of alignment method becomes critical: should a nucleotide or amino acid alignment be used?

5.2.2 Finding a threshold: The statistical approach

Another approach, which was developed for the NetStart data set and also used for the ChloroP data set, is to ignore the annotation of the sequences and focus on the statistical properties of the sequence alignments. Alignment scores have been the subject of much theoretical work, and a statistical theory exists for ungapped alignments (Karlin & Altschul, 1990; Mott, 1992; Altschul *et al.*, 1994).

An important finding from this theory is that local alignment scores follow an *extreme value distribution*. Theoretically, this is only shown to be true for ungapped alignments, but in practice, local gapped alignments are found to follow the same distribution, provided that the gap penalties are within a reasonable range (Altschul & Gish, 1996).

Briefly, our approach is to fit the alignment scores to an extreme value distribution and choose a threshold value above which there are more observations than expected from the distribution. In an extreme value distribution the chance of observing a score greater than or equal to x purely by chance (*i.e.*, in an alignment of *unrelated* sequences) is

$$P_{score \geq x} = 1 - \exp(-e^{-\lambda(x-u)}) \tag{5.1}$$

where u is the characteristic value and λ is the decay constant. For ungapped alignments, these parameters can be calculated theoretically given the scoring system, but for gapped alignments, they must be measured empirically. Rearranging equation (5.1) gives

$$\ln(-\ln(1 - P_{score \geq x})) = -\lambda x + \lambda u \tag{5.2}$$

This means that (for a random data set) a plot of $\ln(-\ln(1-P_{score \geq x}))$ *vs.* score based on all pairwise alignments will be linear. If the data set contains similar sequence pairs this will result in more high-scoring pairs than expected from the extreme value distribution.

Figure 5.1 shows an example from the *Arabidopsis* start codon data set which contained 1040 sequences (corresponding to 540,280 pairwise alignments). For the lower values of Smith-Waterman scores this plot is linear, indicating that the scores follow a simple extreme-value distribution as expected. However, a very clear kink can be seen around a Smith-Waterman score of 103. This means that above a score of 103 there are more observed pairs than expected from the distribution of scores in the first part of the curve, and we consequently chose this value as a similarity cutoff for the *Arabidopsis* data set. After applying algorithm 2 of Hobohm *et al.* (1992) the size of the, now non-redundant, data set was 523 sequences or about half the original size. The entire $\ln(-\ln(1-P_{score \geq x}))$ plot constructed by performing all pairwise alignments for the redundancy reduced set is linear, and essentially overlaps the nonredundant part of the curve for the original set (not shown in figure 5.1).

We also tried performing all pairwise alignments on a version of the original *Arabidopsis* set where the nucleotides in each sequence have been shuffled in random order. As it can be seen in figure 5.1, all scores from this alignment follow an extreme value

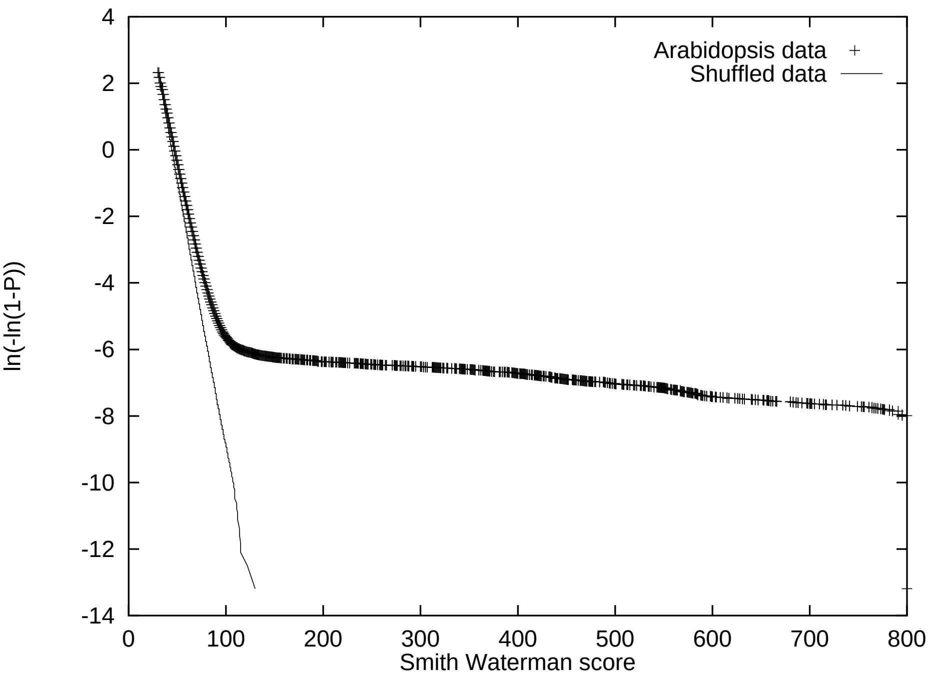

Figure 5.1: Plot of $\ln(-\ln(1-P))$ *vs.* Smith-Waterman alignment score for all pairwise alignments of the redundant *Arabidopsis* set (crosses), and for all alignments of a shuffled version of the same set (line). Note that in the low end of the alignment scores, the *Arabidopsis* plot follows a straight line indicating that the alignment scores follow an extreme value distribution. Above an alignment score of approximately 103 there are more high-scoring pairs than expected from the extreme value distribution, indicating that the set is redundant. The shuffled set has been constructed by randomly reordering the nucleotides in each sequence in the *Arabidopsis* set. As expected, the alignment scores all follow an extreme value distribution for this random set.

distribution, although the line is positioned below the line from the real *Arabidopsis* set. The lower position presumably indicates fundamental non-random sequence properties of natural DNA sequences, such as di- or trinucleotide biases.

The method used here may be of general use for construction of non-redundant data sets, and can be briefly summarised as follows: (1) construct a set of sequences that contain the feature of interest; (2) perform all pairwise alignments and make a list of the sequence pairs with corresponding alignment scores; (3) calculate $P_{score \geq x}$ for all scores; (4) plot $\ln(-\ln(1-P_{score \geq x}))$ *vs.* score; (5) locate score above which there are more alignments than expected by chance (*i.e.* the kink after the first linear part of the plot); (6) apply the redundancy reduction algorithm using this score and the list from point (2).

If the redundant data set contains sequences of widely different lengths it may be necessary to correct for the length-dependence of the alignment score (Altschul *et al.*, 1994; Pearson, 1995). As the sequences in both the start codon and chloroplast transit peptide sets were selected to be of approximate equal length we have not done this.

Chapter 6

Results and discussion: The prediction methods

This chapter summarises how the SignalP, NetStart, and ChloroP prediction methods have been constructed, and how they perform. In general, I will not elaborate on results that are already in the papers, but rather, compare the methods where approaches for the three servers differed. Two sections concern results that are not in the papers: section 6.2.1 extends the investigation of h-region length distribution from paper V, and section 6.4 describes an investigation of putative archaeal signal peptides which otherwise has only been published in a review (Nielsen *et al.*, 1999) and as a conference poster.[1]

6.1 Neural network training

The neural networks that form the central parts of SignalP, NetStart, and ChloroP have all been implemented in HOW, a neural network simulator by Søren Brunak specially designed to work with sequence data. In HOW, layered feed-forward networks are trained with back-propagation using McClelland's error function. Momentum training, weight decay, and pruning are not available. For all the three tasks, we used one hidden layer of varying size, and two output units, corresponding to two output categories.

The networks for SignalP have been trained using "early stopping" on the test set, *i.e.*, stopping the training at the point where performance on the test set was optimal. This approach has been criticised because it involves the test set for optimisation of training length, so the performance might not reflect a true generalisation ability; but practical experience in a bioinformatics application has shown the performance on a new, independent test set to be as good as that found on the data set used to stop the training (Brunak *et al.*, 1991).

For ChloroP, we avoided optimisation on the individual test sets by using a constant number of training epochs for all training sets in the cross-validation. We chose a learn-

[1] Gene Discovery *in silico*, November 6–9, 1997, Atlanta, Georgia, USA. See `http://intron.gatech.edu/~kostya/conference/poster_14.html`

ing rate that was so low that fluctuations were small and overtraining happened late (if at all) so that test performance was not very sensitive to the exact choice of stopping point. It is still true that the number of training epochs is chosen using observations done on the test set, but this is not different from practice in the neural network field. Some authors use other approaches to avoid overtraining: stopping after a certain training set performance is reached, or using weight decay learning; but these methods also have at least one parameter (performance threshold, decay rate) which is not chosen entirely independent of the test set. Furthermore, it should be noted that this discussion not only applies to training, but also to selection of other aspects of the model, *e.g.* thresholds and neural network architecture parameters. Using cross-validation performance as an estimate of generalisation performance for model selection is a common practice in statistical evaluation of artificial intelligence methods (Liu, 1995; Kearns, 1997; Prechelt, 1998).

A strictly "clean" solution to the training stopping and model selection issues would be to use a separate *validation* set for stopping, while a third part of the data is set aside to serve as a true test set (Weigend *et al.*, 1990). The validation set could then be used also for model selection, *e.g.*, optimisation of the architecture and the post-processing parameters. However, this "three-tier" data set partition is seldom used in practice, as it reduces the training set size—often the critical parameter in bioinformatics.

NetStart was originally trained and tested on just data set partition, *i.e.*, without cross-validation. Recently, however, NetStart has been retrained using cross-validation in a three-tier approach on the original data set: the data were divided into 6 equally sized parts, and for 6 different combinations it was trained on four parts, training was stopped according to performance on the fifth part, and the resulting network was tested on the sixth part. The performance measured in this way was not lower than that originally reported in paper III—in fact, it was very slightly better (correlation 0.63 instead of 0.62), but this probably just shows that we happened to pick one of the more difficult partitions in the first run.

In ChloroP, the cleavage site prediction is not done using a neural network but by a simple weight matrix. The weight matrix approach was chosen after initial NN training trials had been unsuccessful. Since a recent experimental study of the cTP processing enzyme stromal processing peptidase (SPP) suggested that the mature N-terminus of chloroplast proteins is often generated by an ill-defined proteolytic removal of one or a few extra residues after the initial SPP cleavage (Richter & Lamppa, 1998), we suspected that the cleavage sites given in SWISS-PROT do not correspond exactly to the peptidase cleavage site. To get around this problem, we used MEME (Bailey & Elkan, 1994), an automatic motif-finding algorithm that does not require pre-aligned sequences, to construct a weight matrix for the SPP cleavage site. In principle, a similar approach could have been done with neural networks, by letting the cleavage site assignment change position during training, but HOW does not implement this automatically, as MEME does. For a later ChloroP version, the MEME-predicted cleavage sites may be used as a starting point for training a cleavage site network and reassigning positions in an iterative procedure.

6.1.1 Postprocessing

The output from a neural network, trained with moving windows, is one score per position in a sequence. For protein sorting prediction, however, the user is typically interested in a conclusion concerning the *entire* sequence—is this a signal peptide, and if so, where will it be cleaved? Therefore, some form of postprocessing is needed.

The HMM does not need postprocessing; it gives both the site and the classification of the sequence immediately, by the fitting of the sequence to the model. Of course, a better performance might still be obtained by some kind of filtering of the HMM output; but the probabilistic framework in which the HMM is defined would rather encourage building all rules into the HMM itself.

Locating sites

SignalP combines two different NNs, one that has been trained to classify each residue in the sequence as either belonging or not belonging to a signal peptide (S-score), and one that has been trained only to recognise the site that is cleaved by the signal peptidase enzyme after targeting (C-score). The S-score will typically display a more or less sharp transition from a high level in the signal peptide to a low level in the mature protein, and cleavage site prediction performance can be significantly enhanced by penalising C-score peaks that are far away from the S-score transition region. This is formalised by using the "Y-score," a geometric average of the C-score and a numerical derivative of the S-score. In the example shown on the cover (and figure 3b of paper IV), the C-score has two peaks, where the upstream one is slightly higher but the downstream one occurs in the transition zone of the S-score and therefore has a higher Y-score. A thorough description of how the Y-score was defined and optimised is found in paper IV.

In ChloroP, the prediction of cleavage sites is done by a weight matrix as described above, but also in this case we found that the two types of predictions could be combined: the slope of the transit peptide score defines the region which is scanned with the weight matrix (see figure 1 of paper VI).

For NetStart, we originally evaluated the score only per position (or more precisely: per ATG, see paper III). However, newer results show that the performance can be improved if we assume that there is exactly one start codon per sequence and assign the highest scoring ATG in each sequence to be the start codon: the discrimination between start codons and and other ATG's is improved from 0.62 to 0.76, measured by correlation coefficient (still with 82% of the start codons correctly identified, but with much fewer false positives). As an interesting comparison, we evaluated the performance of simply choosing the first ATG as the start codon, regardless of score: the correlation coefficient went down to 0.53, and only 64% of the start codons were correctly identified. In other words, a start codon prediction method *is* needed, even if you know the full cDNA (as annotated in GenBank).

Classifying sequences

In SignalP, the prediction for the existence of a signal peptide can be made by the maximal value of the C-, S-, and Y-scores, or the mean S-score between the N-terminal and the predicted cleavage site. Of these, the maximal Y-score or the mean S-score give the best discrimination performance, but all four values are reported in the output. The maximal S-score also shows a high correlation coefficient (see table 6 of paper IV), but it is much more sensitive to the choice of cutoff than the other measures.

Since the conclusions based on all four measures are reported in the output, even when they do not agree, SignalP is sometimes a method in conflict with itself. In a sense, therefore, SignalP can be regarded as unfinished; and "what should I believe" is a frequently asked user question. Originally, I imagined the typical SignalP user to be working with a single sequence, or just a few, having time enough to sit and contemplate the graphical output before making her choice about what to believe. Therefore, I regarded the discrimination criteria merely as measures to evaluate performance rather than tools to be used on a daily basis. In short, I was wrong. A very large group of users turned out to be people wanting to submit whole genomes or EST data sets—hundreds or even thousands of sequences—and such users of course want one decision criterion; one measure and a cutoff value. Indeed, a good method should admit when it is in doubt; but this should rather be done with a reliability index—or a table of specificity and sensitivity per cutoff value—than with a set of conflicting answers.

In ChloroP, a different approach is taken: instead of using a maximal or average value of the score, all chloroplast transit peptide scores within the first 100 positions are used as input to a second neural network, which predicts whether or not the whole sequence is a chloroplast transit peptide. This approach contains more parameters and creates extra possibilities for overtraining; but it is conceptually simpler and leaves it to the network to decide how many inputs should be taken into account for the sequence level prediction. In our unpublished work, the postprocessing network has also turned out to be useful for a four-state prediction of plant protein localisation signals (chloroplast, mitochondrion, signal peptide, or other).

Averaging outputs

The SignalP performance is measured by cross-validation over five training/test set partitions. When implementing the finished method, the question arises: which score to report? Using just one of the partitions would bias the whole method towards that training set, and retraining the network on the whole data set would leave me without a stopping criterion. An average over the five partition scores would be more fair, and averaging has been shown to be advantageous to predictive performance (Krogh & Vedelsby, 1995).

The problem with averaging C-scores was that the training was stopped at optimal sequence level performance (% correctly placed cleavage sites, see paper IV), and for some but not all partitions this peaked earlier in training than optimal position level performance. As a result, the five partitions had different scales and thereby different optimal cutoffs. If these were mixed by a simple averaging, performance might go down instead of up. Therefore, the scores are scaled before averaging so that the optimal cutoff (for signal position correlation) is always 0.5.

Does it work? Yes, in the sense that performance measured with the full scaled and averaged ensemble on the whole data set is indeed better than the cross-validation test performance: the accuracy of cleavage site location (original release 29 version) grows from 70.2% to 76.8% for eukaryotes, from 79.3% to 85.0% for Gram-positive, and from 67.9% to 76.6% for Gram-negative bacteria. However, these figures are not comparable, since there is no test set available for the ensemble. A reliable estimate of the effect of scaling and averaging would require a three-tier data partition with "nested cross-validation:" for each test set, one should do a cross-validation over all possible training/validation set partitions and construct an averaged ensemble, and then the result of this procedure could be averaged over all test sets. This would allow averaged *vs.* non-averaged test set performances to be compared.

6.2 Building and training an HMM

As described in section 2.3 on page 11, hidden Markov models (HMMs) in bioinformatics are most often of the profile type. A profile HMM implies a multiple alignment of the sequences, either preexisting in the training data or generated by the model; and for this reason we did not regard the profile architecture as the logical choice for signal peptides, which are not related by homology but by common function. SignalP-HMM (paper V) is therefore not built with one of the available packages for training profile HMMs, but with an HMM software (written by Anders Krogh) which allows any type of architecture.

Unlike the NN-based SignalP, we have built SignalP-HMM from the preexisting conceptual knowledge about signal peptides. Thus, the three distinct regions—the positively charged n-region, the central hydrophobic h-region, and the c-region encompassing the signal peptidase cleavage site—is represented by a separate part of the model (see figure 2 of paper V). The n- and h-regions are modeled in a simple way, with all states in a region having the same amino acid composition, while the region around the cleavage site is modeled in more detail, essentially like a weight matrix. Signal anchors have an n-region, an h-region, and no cleavage site. By having two parallel submodels of the HMM, it is possible to represent differences in both length distribution and amino acid frequencies between the n- and h-region of signal peptides and signal anchors. A third branch (actually just a shortcut) is added to represent those sequences that are neither signal peptides nor signal anchors (see figure 3 of paper V). When threading a sequence through this model, one of the three branches is chosen, and this serves as the prediction of protein type.

The model was trained with the Baum-Welch algorithm (see section 2.3 on page 12). During training, the labeling of the cleavage site was used, *i.e.*, positions known to be cleavage sites were forced to use the cleavage site states. The borders between the regions, however, were not given in the training data but assigned by the model itself. Thus, SignalP-HMM provides an objective way to delineate the n-, h-, and c-regions in a signal peptide, and it may be used to compare the overall design of signal peptides from different organisms.

One of the main objectives for the SignalP-HMM work was to optimise discrimination between signal peptides and signal anchors. The difference between these lies not only in the presence or absence of cleavage site, but also in the length of the hydrophobic region. The latter difference is probably more important: experiments have shown that it is possible to convert a cleaved signal peptide into an uncleaved signal anchor merely by lengthening the hydrophobic region (Chou & Kendall, 1990; Nilsson *et al.*, 1994). Therefore, we deliberately built expectations about the region length distribu-

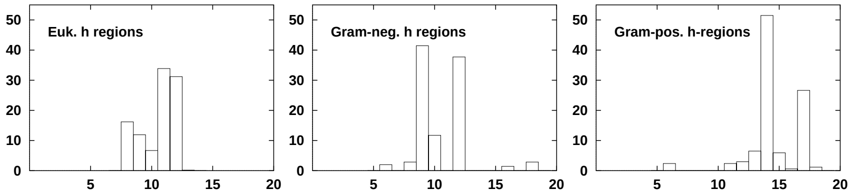

Figure 6.1: The length distributions of the h–regions of signal peptides, as assigned by the trained HMM models. The x-axis is length, and the histograms display the number of sequences in percent.

tions into the model architecture. The submodel of the signal peptide h-region contains no loop states, which gives it a hard-coded maximal length, (see figure 2 of paper V). We refer to this architecture design as *explicit length modeling*, since every possible value of the length has a corresponding transition probability. To allow also unusually long signal peptides to be recognised, the n- and c- regions are represented by hybrids of explicit length models and loop models.

6.2.1 The Twin Peaks mystery

As mentioned, one of the attractive features of the HMM for signal peptides is the opportunity to get a non-arbitrary assignment of the n-, h- and c-regions from any signal peptide. When we calculated this, the length distribution of regions (shown in figure 6.1, and figure 4 of paper V) was a surprise. For all three groups, the length of the h-regions shows a very pronounced two-peaked distribution, with peaks at 8 and 11 for eukaryotes, 9 and 12 for Gram-negative bacteria, and 14 and 17 for Gram-positive bacteria.

Is this Twin Peaks phenomenon real or an artifact? Clearly, a kind of overtraining could be involved in the explicit length modeling: if h-region length K is underrepresented, the K'th transition probability for entering the h-region is weakened; this lowers the probability that any sequence will choose that transition and thereby be assigned an h-region of length K; and as a result, K-long h-regions become even more underrepresented. In the bacterial data sets, intermediate h-regions of length 11 (Gram-negative) or 16 (Gram-positive) have so low probabilities that they are almost impossible. I have checked this with a few artificial examples (not shown): it should be possible, by mutagenesis, to engineer a signal peptide with an h-region of any length by inserting a poly-Leu stretch of that length, flanked by hydrophilic residues (see, *e.g.*, Kendall *et al.*, 1986; Chou & Kendall, 1990); but if I in this way try to present an 11 aa long h-region to the Gram-negative HMM, it simply "chooses" to assign one or more leucines to the n- or c-region, to make the sequence comply with its own "idea" of how long an h-region should be.

Conceivably, the Twin Peaks phenomenon might be an effect of this overtraining. To check this, I have tried to reproduce the overtraining in a controlled experiment:

1. The initial "rule of thumb" described in paper V was used to assign regions in Gram-negative signal peptides, and the h-region length distribution was calculated. This did not show two peaks.

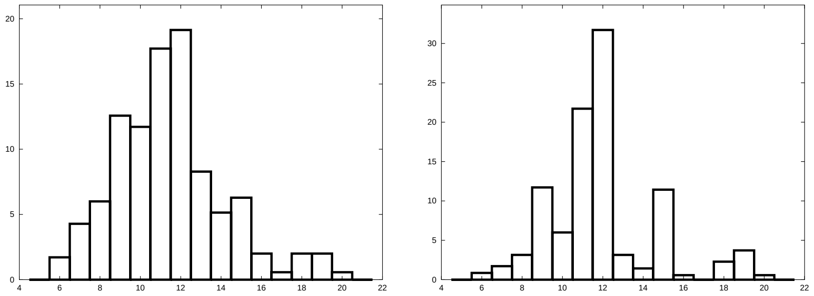

Figure 6.2: Length distribution of h-regions in an artificial data set. Left, data generated by an HMM with a hard-coded region length distribution (calculated by using a simple rule on signal peptides from Gram-negative bacteria). Right, h-region lengths of the *same* data, assigned by a training an HMM with explicit length modeling.

2. This length distribution was forced into a trained Gram-negative SignalP-HMM by replacing the transition probabilities for entering the h-region.

3. An artificial data set was generated by this modified HMM (see figure 6.2, left). Since it uses tied amino acid probabilities for all the h-region states, any sequence non-homogeneity within the h-region is due to noise.

4. This artificial data set—guaranteed "Twin Peaks-free"—was used to train a new HMM with the SignalP-HMM architecture.

5. The regions in the artificial data set were assigned by the new, possibly overtrained, HMM (see figure 6.2, right).

The resulting h-region length distribution does show a clear noise amplification: the overall shape of the distribution is sharper in the reassigned data; and lengths 10 and 14, which were slightly underrepresented in the original distribution, are now severely underrepresented. However, the overtrained distribution does not look like the Twin Peaks distributions obtained with the original data, so the phenomenon does not automatically follow from overtraining.

Two other observations also suggest that two peaks in h-region lengths is a real phenomenon, exaggerated but not created by the overtraining: (1) it was found in all cross-validation tests, although the position of one peak in a few cases was shifted by one position (not shown); (2) the distance between the two peaks was 3–4 residues in all cross-validation tests in all types of organisms.

It would be reassuring, however, to find Twin Peaks within an architecture that does not amplify noise. If the explicit length modelling is replaced by a simple one-loop model (like the h-region of the signal anchors, see figure 3 in paper V), the phenomenon disappears; but this also weakens the predictive performance. The one-loop architecture has an implicit geometric (exponentially decaying) length distribution (Durbin *et al.*, 1998, section 3.4), which does not at all resemble what is found with the "rule of thumb" (figure 6.2, left). Therefore, architectures with other implicit distribution shapes (binomial, negative binomial) should certainly be tried—maybe even the profile architecture that we initially disregarded? Alternatively, a suitable regularisation on the transition probabilities might be able to remove the noise amplification tendency of the explicit length distribution.

If Twin Peaks is real, what is the Bimodality Organising Basis (BOB)? As we suggested in the paper, it might reflect an overlapping distribution of two types of signal peptides correlated with a difference in translocation mechanism. The most obvious possibility here would be SRP-dependent *vs.* SRP-independent signal peptides (see sections 3.1.1 and 3.1.2); since the hydrophobicity of these two groups differ, this difference might be reflected in the apparent length of the h-region. However, many signal peptides probably have an affinity for both pathways and should have an in-between hydrophobicity. Furthermore, we should not expect to see this effect at all in the eukaryotic data set, since it is dominated by vertebrate examples and contains very few yeast examples, while eukaryotic SRP-independent targeting is described mainly from yeast and seems to be very rare in mammals.

A different clue may be the distance between the twin peaks, which corresponds to approximately one helical turn in all three types of organisms. Maybe we should look for BOB in the structure? It has been suggested that the h-region assumes an α-helical conformation while binding to SRP54 and/or to the translocon (Gierasch, 1989). The connection is far from clear, but one could imagine some kind of binding pocket that must accommodate a hydrophobic helix with an integral number of helical turns. If this is so, then the structure of whatever comprises the pocket must be very different in Gram-positive bacteria compared to other organisms.

The Twin Peaks phenomenon is not totally without precedents in the history of signal peptide investigations. As described in section 3.3.1 on page 31, some statistical studies have found a non-homogeneous distribution of amino acids in the h-region; most remarkably, Perlman & Halvorson (1983) and Shinde *et al.* (1989) reported a twin-peaked distribution of leucine frequencies in signal peptides. Shinde (1990) also showed that the ratio of polar to non-polar amino acids depends on helical angle in a helical wheel projection of signal peptide h+c-regions. A tendency for a helical periodicity in physico-chemical properties was also found in the statistical study by Edman *et al.* (1999) described in section 3.3.1 on page 32. However, I have tried to measure amino acid distribution at the various positions in h-regions assigned to be of equal length by SignalP-HMM without finding any significant differences (results not shown).

The results of mutagenesis studies (see section 3.3.2) are also ambiguous: while most have failed to show any requirement for specific sequences in the h-region and confirmed that simple homopolymeric h-regions can be fully functional (*e.g.,* Chou & Kendall, 1990), a newer result suggests that the h-region is not so homogeneous after all: the effect of introducing a proline at various positions in the h-region is dependent on the position in a way that correlates with position in a helical wheel (Ryan & Edwards, 1995). If the helical wheel interpretation is correct, proline-induced kinks in the h-region helix are tolerated only on one face of the helix, and this must imply that the helix does have a specified orientation relative to whatever binds it.

The conclusion compatible with most of these observations is that signal peptides do have a weak tendency for forming amphiphilic helices. BOB is certainly not a strict requirement for amphiphilicity, but possibly a larger tolerance for non-hydrophobic residues on one face of the helix, maybe only in a subgroup of signal peptides.

		Cleavage site location			Discrimination			
Method	Data				SP/non-sec			SP/SA
	(release)	Euk	G_{neg}	G_{pos}	Euk	G_{neg}	G_{pos}	Euk
NN	29	70.2%	79.3%	67.9%	0.97	0.88	0.96	(0.39)
NN	35	72.4%	83.4%	67.5%	0.97	0.89	0.96	(0.39)
HMM	35	69.5%	81.4%	64.5%	0.94	0.93	0.96	0.74

Table 6.1: Performances of SignalP in the neural network (NN) and hidden Markov model (HMM) versions. The column labeled 'Data' refers to the SWISS-PROT release number, so that the first line (NN, 29) show the performance of the original SignalP (paper I). Data sets are divided into eukaryotes (Euk), Gram-negative bacteria (G_{neg}), and Gram-positive bacteria (G_{pos}). Cleavage site location is given as percentage of signal peptide sequences where the cleavage site was placed correctly, and discrimination values between sequence types are given as correlation coefficients (Mathews, 1975). The sequence types are signal peptides (SP), soluble non-secretory—*i.e.* cytoplasmic or nuclear—proteins (non-sec), and signal anchors (SA). For SignalP-NN, cleavage site location is predicted by maximal Y-score, and discrimination performed using mean S-score; discrimination values for signal anchors are in parentheses because signal anchors were not included as negative examples in the NN training set. All values are averages over five cross-validation sets.

6.3 Measuring performance

The performance values of SignalP are shown in table 6.1, both for the original version and for a version retrained on a new data set based on SWISS-PROT release 35 instead of 29. Note that the performance for cleavage site location has improved. Since the old and new data sets are extracted by the same method, and the sizes have changed only slightly, the most probable explanation for the improvement is that the quality of SWISS-PROT annotations concerning signal peptides are better in the newer version.

There are two important points to be made about the performance values. On one hand, they should be regarded as minimal, because they are test set performances (averaged over five homology reduced cross-validation sets). These performance values should therefore be expected for a protein unrelated to anything in the data sets, while prediction accuracy on sequences with some similarity to the sequences in the data sets will in general be much higher (see the discussion of ensemble performances in section 6.1.1 on page 68).

On the other hand, these performance values are calculated under two limiting assumptions: that the correct N-terminus of the protein in question is known, and that the sequence does not contain an N-terminal transmembrane helix. The data sets on which SignalP is trained and tested contain only the N-terminal part (up to 70 amino acids) of each protein, and transmembrane proteins were not included in the negative set (see the discussion in sections 5.1.2 and 5.1.3).

These two points constitute a problem for the application of SignalP to genome and EST data. As an illustration of this, the scanning the *Haemophilus influenzae* genome which we reported in paper II produced a remarkably large variation in the estimate of the proportion of proteins with signal peptides: from 14% if using the maximal Y-score as discriminator, to 28% when using the maximal S-score, even though all these measures give high discrimination performances when used on the SignalP data set.

This means that the performance of (at least) one of these measures is considerably lower when applied to genome data; and that SignalP, when used for this purpose, should ideally be combined with a transmembrane helix prediction and a start codon prediction.

SignalP-HMM is able to discriminate between signal peptides and signal anchors with a correlation coefficient of 0.74 (see table 6.1)—far from perfect, but much better than with the NNs. In a sense, this comparison is not quite fair, because the signal anchors were not used explicitly as negative examples during training of the NN, but this would have been problematic given the small size of the signal anchor set. With the HMM, it is easy to take this limitation into account by using a simpler submodel (with a smaller number of free parameters) in the signal anchor branch than in the signal peptide branch. Regarding the identification of signal peptides *vs.* soluble non-secretory proteins, the HMMs performs on a par with the NNs—for Gram-negative bacteria even better—but they are less accurate for cleavage site prediction.

The weight matrix results are not included in table 6.1, but they can be seen in table 5 of paper IV. Note that weight matrix performances are approximately 10% lower than those reported by von Heijne (1986b) (see section 4.2.1 on page 42). This shows that the cleavage sites in the new, larger data set were less regular; but whether it reflects a wider coverage of signal peptide diversity or a higher error rate in the new data set is difficult to say.

6.4 Signal peptides of Archaea

Secretory signal peptides from eukaryotes and bacteria are well described, but only very few experimental examples are known from the third domain of life—the archaea (formerly known as archaebacteria). Although prokaryotic, they show greater similarity to eukaryotes than to bacteria in many respects, especially concerning informational cellular processes such as replication and translation (Olsen & Woese, 1997). Furthermore, their membranes exhibit very peculiar properties not found in other organisms. It is therefore not clear which, if any, of the three current organism-specific SignalP versions is valid for identification of archaeal signal peptides.

We used a “consensus” between the three SignalP versions in a first attempt to characterise the signal peptides of *Methanococcus jannaschii*, the first archaeon to be completely sequenced (Bult *et al.*, 1996). Signal peptides should indeed be expected in this organism: a signal peptidase has been identified by homology in the genome, and it shows larger homology to its eukaryotic than to its bacterial counterpart. The underlying idea is that if we are able to find sequences in the genome which could function as signal peptides in all other domains of life (*i.e.*, in eukaryotes and both groups of bacteria), they would presumably function as signal peptides in *M. jannaschii*, too.

M. jannaschii signal peptides might have been predicted by alignment to known signal peptides from other organisms, if significant matches to experimentally verified secretory proteins including the signal peptide region could be found. We made local pairwise alignments between all the predicted *M. jannaschii* protein sequences and all sequences in the SignalP data set, but found only insignificant matches. Even the best pairwise alignment scores were considerably lower than the threshold required for

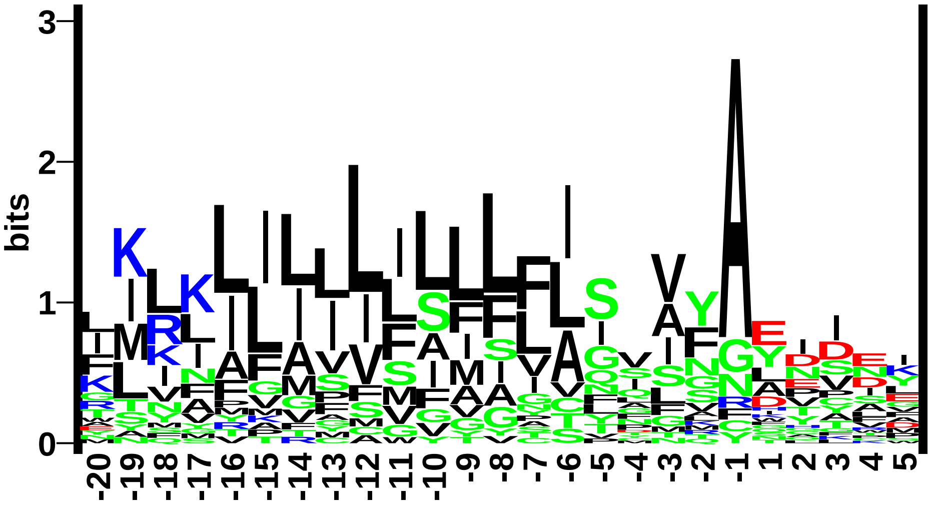

Figure 6.3: A sequence logo of 34 predicted signal peptides from *Methanococcus jannaschii*, aligned by their cleavage sites (no gaps).

using a local alignment of two signal peptide sequences to predict the location of the cleavage site (cf. paper I and section 5.2.1). This shows that we can not expect to find *M. jannaschii* signal peptides by alignment—a prediction method is indeed necessary for this task.

We selected sequences where both the maximal Y-score and the mean S-score were above their cutoff values for all three SignalP versions (eukaryotic, Gram-positive, and Gram-negative). This is a very conservative criterion: when tested on the SignalP data sets, it accepts 75% of the Gram-negative, 66% of the Gram-positive, and only 39% of the eukaryotic signal peptides. Used on the *M. jannaschii* genome, it yielded 34 putative signal peptides, none of which had known subcellular location. This number is too small to train a species-specific neural network (it might be used for an HMM but this has not been implemented), but it is enough to draw a few tentative conclusions about *M. jannaschii* signal peptides.

The 34 sequences were divided into n-, h- and c-regions, and the amino acid content compared to that of eukaryotes and bacteria. The *H. influenzae* genome (Fleischmann *et al.*, 1995) served as a reference example of a Gram-negative bacterium. In figure 6.3, the 34 putative *M. jannaschii* signal peptides are represented as a sequence logo, *i.e.*, a sequence of stacked letters, where the total height of the stack at each position shows the amount of information (conservation), while the relative height of each letter shows the relative abundance of the corresponding amino acid (Schneider & Stephens, 1990). When compared to logos of eukaryotic or bacterial signal peptides (figure 1 of paper II), the following characteristics are observed:

In the n-region, the content of Lys is very high, while Arg is relatively rare. A positively charged n-region is also found in bacterial signal peptides, but there Arg and Lys are used in more equal proportion. The Lys content of *M. jannaschii* n-regions is approximately 30% compared to 20% in *H. influenzae*. A very characteristic feature is the high content of Ile in the h-region. This is not limited to signal peptides, as Ile is strongly overrepresented in *M. jannaschii* as compared to *H. influenzae* also in transmembrane regions (16% *vs.* 12%) and in the genome as a whole (10.5% *vs.* 7.1%). However, the difference is more drastic for the h-regions (22% *vs.* 11%).

In the c-region, the dominance of Ala at position -1 is typical for both bacterial and eukaryotic signal peptide cleavage sites, whereas the tolerance of other uncharged residues such as Val, Leu, and Ile at -3 and the short length of the c-region clearly suggest a eukaryotic type of cleavage site. Around the cleavage site, a unique feature is also found: a high occurrence of Tyr (8% of the c-regions as opposed to 2% in *H. influenzae*), particularly visible at positions +1 and -2. This seems to be specific for the signal peptides, since the general Tyr content is only slightly higher in *M. jannaschii* than in *H. influenzae* (4.3% *vs.* 3.3%). Finally, the occurrence of negatively charged residues in the first few positions of the mature protein has previously been noted for bacterial but not for eukaryotic signal peptides (von Heijne, 1986a).

An open question is whether the features we find to be special for *M. jannaschii* signal peptides are related to the archaeal domain or to the hyperthermophilic condition. It will be very interesting to repeat this analysis on genomes of non-hyperthermophilic Archaea; *Sulpholobus* and *Halobacterium* should not be far off now. However, a recent paper by Haney *et al.* (1999) may provide some hints: they have compared 115 proteins of *M. jannaschii* with their homologues from mesophilic (*i.e.*, preferring moderately hot environments) *Methanococcus* species and tested which amino acid substitutions are asymmetric, *i.e.*, occurring significantly more often in the mesophilic to hyperthermophilic direction than *vice versa*. Indeed, the Leu→Ile substitution is among them, so the high Ile content in *M. jannaschii* proteins may very well be an adaptation to high temperature. This raises the question why *signal peptides* should be adapted to high temperatures—why should they need thermostability when they are not supposed to be stable?—but maybe they simply follow a genome-wide amino acid distribution preference which is selected for thermostability. With respect to the very high Lys content in the n-region, the implications are less clear: in general, *M. jannaschii* proteins have more charged amino acids than their mesophilic cousins, but Arg is more enriched than Lys and the Arg→Lys substitution is strongly *under*represented, so the high Lys/Arg ratio cannot be explained this way. Finally, Tyr is also slightly overrepresented, but most substitutions involving Tyr are too rare to show significant bias.

In conclusion, our analysis suggests that signal peptides from an archaeon have a eukaryotic-looking cleavage site, a bacterial-looking charge distribution, and a unique (possibly hyperthermophile-specific) composition of the hydrophobic region. The statistical description is of course to some extent affected by the fact that we use a consensus method, which only finds signal peptides and cleavage sites that would be acceptable in both eukaryotes and bacteria; chances are that signal peptides peculiar to archaea have gone undiscovered. In other words, we have if anything *underestimated* the unique characteristics of the *M. jannaschii* signal peptides.

Chapter 7

Future perspectives

From the previous chapters, it should be perfectly clear that the problem of signal peptide prediction is in no way exhausted by the work presented here. Apart from the simple fact that SignalP performance is still a long way from 100%, many questions remain open, concerning differences between signal peptides and their relations to other sorting signals. In this chapter, I will attempt to give an overview of the most important challenges for signal peptide prediction, and gradually widen the perspective to protein sorting prediction and bioinformatics in general.

7.1 Signal peptide diversity

A signal peptide is not just a signal peptide. In chapter 3, I mentioned several examples of signal peptides that follow variations to the general secretory pathway. The sequence patterns defining these are more or less well characterised, but no signal peptide prediction method takes them into account. Furthermore, as mentioned in section 5.1.4, we also know too little about species-specific variation in signal peptide design. Below, I have assembled a "catalogue" of signal peptide categories that it would be interesting to characterise and/or predict separately:

Targeting specificity: In bacteria and yeast, at least two parallel targeting pathways—SRP-dependent (section 3.1.1) and SRP-independent (section 3.1.2)—lead to the translocon. As mentioned on page 24, SRP-dependent targeting signals seem to be more hydrophobic than SRP-independent ones. Does this provide it possibility for predicting the SRP-dependence of a signal peptide? Probably there are too few examples with established targeting specificity, and too much overlap (many sequences seem to be able to use both systems). However, mutagenesis may provide information about which changes shift the targeting specificity—how such data could be incorporated in a prediction model is discussed in section 7.3.

Thylakoid translocation signals: As described in section 5.1.1 on page 55, it was possible to use SignalP to predict the thylakoid part of chloroplast composite signals by using the version trained on Gram-negative data. However, it has been reported that the cleavage specificity of the thylakoid signal peptidase differs from that of *E. coli* (Gavel & von Heijne, 1990; Howe & Wallace, 1990).

Mitochondrial inner membrane signals: the mitochondrial targeting system is less well described than that of chloroplasts, and I have not tested SignalP with mitochondrial composite signals. A possible complication is that the two type I signal peptidases of the mitochondrial inner membrane (see page 26) may have different substrate specificities (Dalbey *et al.*, 1997).

Prokaryotic SPase II lipoproteins already have their own prediction method (a PROSITE pattern, see page 27). However, it would be nice to have it integrated in the same model as the standard SPase I-cleaved signal peptides—and to see whether the performance of the PROSITE pattern can be improved.

TAT signal peptides of bacteria and chloroplasts: see page 38.

Type IV pilins: see page 39.

Calmodulin- and HLA-binding are two examples of functions for cleaved signal peptides in mammalian cells, see page 28. Calmodulin-binding SPs have unusually long n-regions; but other long n-regions also show peculiar sequence properties (Martoglio & Dobberstein, 1998)—maybe other functional roles are waiting to be discovered? In particular, it could be interesting to investigate the pattern of conservation and divergence in these SPs: if a region has a specific function in addition to being part of a SP, it should be expected to show a stronger conservation than SPs in general.

Viral signal peptides were not included in SignalP training. They sometimes show extremely long signal peptides, which seems peculiar since vira are otherwise very economical with their genetic material. The long SPs must therefore play a special role—two wild guesses are: maximisation of epitope diversity in order to avoid immune system recognition; or cleavage delay to aid virus particle assembly.

In addition to the *differences* between signal peptides varying function, it would be very interesting to study the pattern of *similarities* between related signal peptides. The signal peptide (and the chloroplast and mitochondrial transit peptides) should be expected to evolve faster than the rest of the protein, since the sequence requirements are rather unspecific, and substitutions should be neutral to protein function as long as they do not destroy signal peptide function. This is accordance with preliminary observations we made while doing pairwise alignments for homology reduction of both the SignalP and the ChloroP data sets (section 5.2): more strong alignments were found downstream of the cleavage site than upstream. However, we have not investigated this effect in any systematic way.

Knowing the pattern of signal or transit peptide evolution could aid the determination of the limit for reliable cleavage site inference by alignment (cf. the discussion in section 5.2.1). If the difference in conservation between sorting signal and mature protein is strong enough, it might even be used for multiple alignment-based enhancement of cleavage site prediction—provided that evolutionary shifts in cleavage site position are not too frequent.

7.2 Signal peptides and membrane proteins

SignalP-HMM does a fairly good job in discriminating between signal peptides and signal anchors, but this solves only a part of the problem, since the type II membrane proteins constitute only a minor fraction of transmembrane proteins. When scanning genome data, it would be desirable to distinguish SPs not only from signal anchors, but also from other types of transmembrane helices.

Type I membrane proteins should not pose any special problems for prediction, since their SPs are functionally equivalent to those of secretory proteins. Multispanning membrane proteins, however, could be responsible for a large number of false positives, but we have not investigated this in any systematic way yet.

Originally, I did not regard this as a serious problem for SignalP: since several high-performance transmembrane topology predictors are available (see section 4.4), it should be possible to filter out transmembrane helices before using SignalP. As the usage tends to shift towards scanning genome and EST data sets, however, it would be nice to know how many false positives to expect from TM proteins, and to see whether discrimination between SPs and TM helices can be improved by adjusting the postprocessing parameters. This could be done with a negative data set of TM helices that are not signal peptides, extracted as described in section 5.1.3.

Broome-Smith *et al.* (1994) reported that “cleavable signal peptides are rarely found in bacterial cytoplasmic membrane proteins.” This conclusion was based on the absence of known examples rather than extensive analysis of a large data set; but even after several bacterial genomes have been sequenced this statement has not, to my knowledge, been tested by anybody.

Of course, it would be preferable, both for usage on large data sets and from a theoretical point of view, to obtain one prediction of the presence and location of both SPs and transmembrane helices in the sequence. To this end, we plan to build an integrated HMM architecture based on SignalP-HMM and an HMM-based transmembrane helix prediction method, TMHMM (Sonnhammer *et al.*, 1998). Conceivably, such a combination could also improve topology prediction, since a cleaved signal peptide necessarily will leave the N-terminus on the outside.

7.3 Mutagenesis and signal peptides

Basically, there are two ways to characterise the information present in a functional sequence pattern: either one can do statistics and prediction on biological examples of the pattern, or one can engineer changes in the sequence pattern and monitor functional consequences. The first approach samples the variation of the pattern as found in nature, while the second approach samples the limits of variation allowed before a specific function disappears.

Signal peptide properties have been investigated by analysing biological (“wild-type”) examples—this is what most of this thesis is about, and the work of other groups are presented in section 3.3.1 and chapter 4—but a much higher number of papers are published on mutagenised signal peptides, as the incomplete review in section 3.3.2 shows. In a sense, it is a tremendous waste that this immense amount of experimental data is not used in the construction of prediction methods. Using signal peptides as examples, this section is a discussion of how data derived from engineered sequences could be used in bioinformatics.

Mutagenesis and bioinformatics are not very often integrated; concerning signal peptides, the only examples I am aware of are the testing of computationally optimised signal peptides (Wrede *et al.* 1998; described in section 4.2.3 on page 45), the systematic mutagenesis of cleavage regions in *E. coli* (Karamyshev *et al.*, 1998), and our own investigation on the h-regions of twin-arginine signal peptides (to appear, see section 3.5.2). For nucleotide sequence patterns, however, the combination is not as rarely seen. A collection of artificially selected binding sites for a DNA-binding protein can be obtained with an *in vitro* evolution procedure (systematic evolution of ligands by exponential enrichment, SELEX) and then analysed bioinformatically (see Shultzaberger & Schneider, 1999, for an example).

As mentioned in section 3.3.2 on page 33, the two approaches can produce different results. The *in vitro* selection of one isolated functional aspect may be rather different from the *in vivo* selection where interactions between many functional aspects are crucial for survival and multiplication.

Consider the example of targeting, translocation, and cleavage: as described in section 3.1, the signal peptide is not recognised by one specific receptor, but by several systems that operate both sequentially and in parallel. In SRP-dependent targeting and translocation, the signal peptide is recognised at least three times: by SRP54, by Sec61α/SecY (and possibly other translocon-associated components including lipids), and by signal peptidase. In the living cell, the recognition events involved in these processes are parts of an integrated system, and real signal peptides are selected to function with the combined system. Sequences that are targeted but not translocated, or translocated but not cleaved, *etc.*, are probably very rare. True, signal anchors are targeted and inserted but not cleaved, but (as described in section 6.2 on page 69) they are not merely signal peptides without a cleavage site, they have a hydrophobic region with characteristics very different from signal peptide h-regions. A signal peptide with a normal h-region but without a cleavage site is a laboratory-created monster.

However, if bioinformatics is going to participate in the task of sorting out the different recognition steps of the signal peptide, we must take these monsters into account when building models. The important point here is not that monsters will help us construct models that perform better on recognising natural sorting signals. If we are lucky, they will provide a finer mapping of the borderline between functional and non-functional examples; but maybe they will only help improve performance in those regions of sequence space where Nature never enters. The possible benefit of doing bioinformatics on engineered sequences is not, as I see it, the refinement of predictive functional genomics, but the deepening of our understanding of the sorting machineries and their interaction with the signals.

A severe problem in this context is finding and collecting the data. The mutations and their effects do not, as a rule, find their way into the general purpose sequence databases. SWISS-PROT has a “MUTAGEN” feature, and in GenBank, “/note” qualifiers may contain information about mutants; but these possibilities are not used very much. Unless this is about to change, every example of mutagenesis experiments basically has to be hunted down in a literature search, and the sequence changes and phenotypes must be manually entered from the papers.

As a possible alternative, the protein mutant database (PMD)[1] (Kawabata *et al.*, 1999) could be a promising resource. A quick search for "signal peptide" or "signal sequence" gave 160 hits representing more than 1000 mutants; and the annotation of signal peptide function, though far from complete, seems to include enough information for doing some statistics. The database construction, however, does not seem very robust: at a certain point in the history of the database, some fields have changed format without earlier entries being updated. Worse, annotation of sequence changes has ambiguities that generate errors from the database's own parsing program; meaning that a large number of the cited papers must still be consulted manually. Furthermore, the updating of the database is 3–4 years delayed: the newest entries are from papers published in 1995, and the latest approximately 20% of the entries are "under construction," *i.e.*, lacking any information about mutants or effects. Kawabata *et al.* (1999) acknowledge that the database suffers from lack of manpower, and they they are planning to reduce the amount of incoming data to be handled by concentrating on proteins with known structure—probably a reasonable choice if you are interested in active sites; but not too promising for the protein sorting business where solved structures are rare.

Suppose a "monster" data set has been obtained—what should be done with it? Simply incorporating all mutagenised sequences as positive or negative examples according to their phenotype can be problematic. First, the level of sequence similarity is necessarily very high, as mutagenesis is typically done with a limited number of reporter proteins for which there are well-documented assays, and performing homology reduction would probably discard most of the information. Second, the changes introduced in the sequences are seldom representative; except for "random mutagenesis" experiments, the mutations reflect the authors' hypotheses about which sequence properties are crucial. Third, the phenotypes are measured with different assays and under different conditions; *e.g.*, a mutation which slows export could be measured as neutral or as export-blocking depending on the time of measurement.

Alternatively, they may be used in a knowledge-based approach while designing or redesigning the models. A model can be tested with all examples from the same assay, to see whether a it can reproduce the *differences* in phenotype. This approach would also make it possible to use the information about mutant phenotypes in a *quantitative* way (such as 20% export) instead of classifying them into positive and negative examples. If a model fails to predict a phenotype difference, the nature of the mutation could be used as a hint about how the model should be modified, in order to be able to represent what happened.

A modest example of this approach was already used in the existing SignalP-HMM architecture: as described in section 6.2, the decision to put a hard upper limit on the h-region length in the signal peptide branch was based on mutation studies showing that signal peptides and signal anchors may differ only in their h-region length. Examining other examples may tell us how the architecture of SignalP-HMM should be changed in order to model the sequence recognition process more precisely. One class of modifications that the present SignalP-HMM would not be able to deal with concerns mutations that affect export by rearranging the residues in the h-region, *e.g.*, the position-dependent effect of introducing prolines mentioned in section 6.2.1, since we have used a common amino acid distribution for the entire h-region. If the explanation

[1] `http://pmd.ddbj.nig.ac.jp/`

for the two peaked h-region length distribution has something to do with the helical conformation of the h-region, we may be able to represent it in the HMM by a wheel-shaped h-region model.

In other words (cf. section 6.2.1): while a model trained on natural creatures showed us the way to Twin Peaks, it may require the assistance of monsters to achieve the level of sophistication necessary to find BOB.

7.4 Prediction of protein sorting: the future

With the recent advances in prediction methods for protein sorting, the vision of a computer program that is able to predict the subcellular location of almost any given protein with high confidence seems not entirely unrealistic. This would be an integrated system of sorting signal predictors and methods based on overall amino acid composition, and as described above, start codon prediction and transmembrane helix prediction should be included. A major use of such a program would be automatic annotation of sequence databases, including complete genomes.

On the other hand, one big integrated system of all methods may not be the most desirable solution for all users. For automated annotation of very large data sets, integrated prediction systems are of course preferable, but the biologist working on one specific gene might be better off considering comprehensive graphical output from several prediction methods separately, and then deciding which conclusion should be drawn from the possibly conflicting predictions. In some rare but interesting cases, the biologically correct answer will be something not anticipated by the method builders (*e.g.* dual targeting, double cleavage, non-standard use of sorting machineries), and uncritical use of a totally integrated prediction system could actually block new discoveries instead of promoting them.

Additionally, any given application will require careful consideration of how to strike the best balance between sensitivity and specificity. For gene hunting, one may want high sensitivity (*i.e.*, few false negatives) in order not to miss interesting candidate genes, whereas for database annotation it may be more prudent to ask for high specificity (*i.e.*, few false positives) even if this will leave many sequences un-annotated. Ideally, the cutoff for assignment of a particular feature should therefore be set by the user; but this can by difficult to implement in an integrated system.

This tradeoff illustrates a common aspect in the evaluation of prediction methods. Performances are given as percent correct, correlation coefficients *etc.*, but these depend strongly on the choice of cutoff and the selection of data sets. Just like there is no application-independent optimal cutoff, there is no single correct definition of positive and negative data (cf. the discussions in section 5.1).

This makes performance comparison of different methods a very tricky business, but also a necessary one. Clearly, common testing standards are needed for deciding whether a reported performance improvement in a new bioinformatics application represents a genuine progress in predictive power or merely a more permissive definition of correct answers. In the protein structure prediction field, this need has led to the establishment of a “competition” in the form of the biannual CASP meeting (Critical

Assessment of techniques for protein Structure Prediction).[1] Before each CASP meeting (the first was held in 1994) a number of targets are collected; these are proteins for which the structure either has been determined but not published, or will be determined before the meeting. The sequences of the targets are published, and participants can then submit structure predictions that are evaluated and compared at the meeting. The gene finding field does not have an equivalent event, but there is work in progress towards establishing a common standard for data sets.[2] In protein sorting prediction, however, comparisons are largely done *ad hoc*.

This is of course an unsatisfactory situation, and much more work is certainly needed in the definition and evaluation of performance measures and data set standards. However, I feel that the, in a very crucial sense, most informative test of a sequence-based prediction method is carried out by making it available to the biological community, both in academia and in industry, *e.g.* by implementing it as a server or a portable program. The feedback from users, either directly, or implicitly via usage and citation statistics, can provide some information about the quality of our bioinformatics work that percentages and correlation coefficients will never be able to disclose.

In general, I think we should be careful not to turn bioinformatics into a “benchmark science” where all effort is directed towards gaining a few additional points in a race towards perfect prediction of well-defined problems. To be able to keep opening new frontiers, bioinformatics should in my opinion switch from simply measuring predictive performance to building models that have an explanatory power. This will require a closer integration with the disciplines that provide the data and the various frameworks for understanding their biological implications: molecular biology, structural biology, biophysics, genetics, and evolutionary theory.

[1] CASP is organised by Lawrence Livermore National Laboratory, California, USA, `http://PredictionCenter.llnl.gov/`

[2] See `http://www.hgmp.mrc.ac.uk/Genesafe/`

Bibliography

Abola, E. E., Sussman, J. L., Prilusky, J. & Manning, N. O. (1997). Protein data bank archives of three-dimensional macromolecular structures. *Methods Enzymol.*, **277**, 556–571.

Althoff, S., Selinger, D. & Wise, J. A. (1994). Molecular evolution of SRP cycle components: functional implications. *Nucleic Acids Res.*, **22**, 1933–1947.

Altschul, S., Boguski, M. S., Gish, W. & Wootton, J. C. (1994). Issues in searching molecular sequence databases. *Nat. Genet.*, **6**, 119–129.

Altschul, S. F. (1991). Amino acid substitution matrices from an information theoretic perspective. *J. Mol. Biol.*, **219**, 555–565.

Altschul, S. F. & Gish, W. (1996). Local alignment statistics. *Methods Enzymol.*, **266**, 460–480.

Anderson, D. M. & Schneewind, O. (1997). A mRNA signal for the type III secretion of Yop proteins by *Yersinia enterocolitica*. *Science*, **278**, 1140–1143.

Andersson, H. & von Heijne, G. (1991). A 30-residue-long "export initiation domain" adjacent to the signal sequence is critical for protein translocation across the inner membrane of *Escherichia coli*. *Proc. Natl. Acad. Sci. USA*, **88**, 9751–9754.

Andrade, M. A., O'Donoghue, S. I. & Rost, B. (1998). Adaptation of protein surfaces to subcellular location. *J. Mol. Biol.*, **276**, 517–528.

Andrews, D. W., Young, J. C., Mirels, L. F. & Czarnota, G. J. (1992). The role of the N region in signal sequence and signal-anchor function. *J. Biol. Chem.*, **267**, 7761–7769.

Arrigo, P., Giuliano, F., Scalia, F., Rapallo, A. & Damiani, G. (1991). Identification of a new motif on nucleic acid sequence data using Kohonen's self-organising map. *CABIOS*, **7**, 353–357.

Bailey, T. L. & Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *ISMB*, **2**, 28–36.

Bairoch, A. (1999). The ENZYME data bank in 1999. *Nucleic Acids Res.*, **27**, 310–311.

Bairoch, A. & Apweiler, R. (1999). The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1999. *Nucleic Acids Res.*, **27**, 49–54.

Baldi, P. & Brunak, S. (1998). Bioinformatics: The machine learning approach. MIT Press, Cambridge.

Baldi, P., Brunak, S., Chauvin, Y., Engelbrecht, J. & Krogh, A. (1995). Periodic sequence patterns in human exons. *ISMB*, **3**, 30–38.

Barker, W. C., Garavelli, J. S., McGarvey, P. B., Marzec, C. R., Orcutt, B. C., Srinivasarao, G. Y., Yeh, L.-S. L., Ledley, R. S., Mewes, H.-W., Pfeiffer, F., Tsugita, A. & Wu, C. (1999). The PIR-International protein sequence database. *Nucleic Acids Res.*, **27**, 12–17.

Bateman, A., Birney, E., Durbin, R., Eddy, S. R., Finn, R. D. & Sonnhammer, E. L. L. (1999). Pfam 3.1: 1313 multiple alignments and profile HMMs match the majority of proteins. *Nucleic Acids Res.*, **27**, 260–262.

Bayley, D. P. & Jarrell, K. F. (1998). Further evidence to suggest that archaeal flagella are related to bacterial type IV pili. *J. Mol. Evol.*, **46**, 370–373.

Beckmann, R., Bubeck, D., Grassucci, R., Penczek, P., Verschoor, A., Blobel, G. & Frank, J. (1997). Alignment of conduits for the nascent polypeptide chain in the ribosome-Sec61 complex. *Science*, **278**, 2123–2126.

Belin, D., Bost, S., Vassalli, J. D. & Strub, K. (1996). A two-step recognition of signal sequences determines the translocation efficiency of proteins. *EMBO J.*, **15**, 468–478.

Benson, S. A., Hall, M. N. & Silhavy, T. J. (1985). Genetic analysis of protein export in *Escherichia coli* K12. *Annu. Rev. Biochem.*, **54**, 101–134.

Berks, B. C. (1996). A common export pathway for proteins binding complex redox cofactors? *Mol. Microbiol.*, **22**, 393–404.

Binet, R., Létoffé, S., Ghigo, J. M., Delepelaire, P. & Wandersman, C. (1997). Protein secretion by Gram-negative bacterial ABC exporters – a review. *Gene*, **192**, 7–11.

Bird, P., Gething, M.-J. & Sambrook, J. (1987). Translocation in yeast and mammalian cells: not all signal sequences are functionally equivalent. *J. Cell Biol.*, **105**, 2905–2914.

Bird, P., Gething, M.-J. & Sambrook, J. (1990). The functional efficiency of a mammalian signal peptide is directly related to its hydrophobicity. *J. Biol. Chem.*, **265**, 8420–8425.

Birse, D. E. A., Kapp, U., Strub, K., Cusack, S. & Åberg, A. (1997). The crystal structure of the signal recognition particle Alu RNA binding heterodimer, SRP9/14. *EMBO J.*, **16**, 3757–3766.

Bogsch, E., Brink, S. & Robinson, C. (1997). Pathway specificity for a ΔpH-dependent precursor thylakoid lumen protein is governed by a 'Sec-avoidance' motif in the transfer peptide and a 'Sec-incompatible' mature protein. *EMBO J.*, **16**, 3851–3859.

Braud, V. M., Allan, D. S., O'Callaghan, C. A., Söderström, K., D'Andrea, A., Ogg, G. S., Lazetic, S., Young, N. T., Bell, J. I., Phillips, J. H., Lanier, L. L. & McMichael, A. J. (1998). HLA-E binds to natural killer cell receptors CD94/NKG2A, B and C. *Nature*, **39**, 795–799.

Brodsky, J. L. (1998). Translocation of proteins across the endoplasmic reticulum membrane. *Int. Rev. Cytol.*, **178**, 277–328.

Broome-Smith, J. K., Gnaneshan, S., Hunt, L. A., Mehraein-Ghomi, F., Hashemzadeh-Bonehi, L., Tadayyon, M. & Hennessey, E. S. (1994). Cleavable signal peptides are rarely found in bacterial cytoplasmic membrane proteins. *Mol. Membr. Biol.*, **11**, 3–8.

Brunak, S. (1993). Doing sequence analysis by inspecting the order in which neural networks learn. In Soumpasis, D. M. & Jovin, T. M., (eds.) *Computation of Biomolecular Structures — Achievements, Problems and Perspectives.* Springer–Verlag, Berlin, pp. 43–54.

Brunak, S., Engelbrecht, J. & Knudsen, S. (1990a). Cleaning up gene databases. *Nature*, **343**, 123.

Brunak, S., Engelbrecht, J. & Knudsen, S. (1990b). Neural network detects errors in the assignment of pre-mRNA splice site. *Nucleic Acids Res.*, **18**, 4797–4801.

Brunak, S., Engelbrecht, J. & Knudsen, S. (1991). Prediction of human mRNA donor and acceptor sites from the DNA sequence. *J. Mol. Biol.*, **220**, 49–65.

Bulmer, M. (1988). Codon usage and intragenic position. *J. Theor. Biol.*, **133**, 67–71.

Bult, C. J., White, O., Olsen, G. J., Zhou, L., Fleischmann, R. D., Sutton, G. G., Blake, J. A., FitzGerald, L. M., Clayton, R. A., Gocayne, J. D., Kerlavage, A. R., Dougherty, B. A., Tomb, J.-F., Adams, M. D., Reich, C. I., Overbeek, R., Kirkness, E. F., Weinstock, K. G., Merrick, J. M., Glodek, A., Scott, J. L., Geoghagen, N. S. M., Weidman, J. F., Fuhrmann, J. L., Nguyen, D., Utterback, T. R., Kelley, J. M., Peterson, J. D., Sadow, P. W., Hanna, M. C., Cotton, M. D., Roberts, K. M., Hurst, M. A., Kaine, B. P., Borodovsky, M., Klenk, H.-P., Fraser, C. M., Smith, H. O., Woese, C. R., & Venter, J. C. (1996). Complete genome sequence of the methanogenic archaeon, *Methanococcus jannaschii*. *Science*, **273**, 1058–1073.

Burns, D. M. & Beacham, I. R. (1985). Rare codons in *E. coli* and *S. typhimurium* signal sequences. *FEBS Lett.*, **189**, 318–324.

Cavener, D. R. & Ray, S. C. (1991). Eukaryotic start and stop translation sites. *Nucleic Acids Res.*, **19**, 3185–3192.

Cedano, J., Aloy, P., Pérez-Pons, J. A. & Querol, E. (1997). Relation between amino acid composition and cellular location of proteins. *J. Mol. Biol.*, **266**, 594–600.

Chou, K.-C. & Elrod, D. W. (1998). Using discriminant function for prediction of subcellular location of prokaryotic proteins. *Biochem. Biophys. Res. Commun.*, **252**, 63–68.

Chou, K.-C. & Elrod, D. W. (1999a). Prediction of membrane protein types and subcellular locations. *Proteins*, **34**, 137–153.

Chou, K.-C. & Elrod, D. W. (1999b). Protein subcellular location prediction. *Protein Eng.*, **12**, 107–118.

Chou, M. M. & Kendall, D. A. (1990). Polymeric sequences reveal a functional interrelationship between hydrophobicity and length of signal peptides. *J. Biol. Chem.*, **265**, 2873–2880.

Claros, M. G. & Vincens, P. (1996). Computational method to predict mitochondrially imported proteins and their targeting sequences. *Eur. J. Biochem.*, **241**, 779–786.

Cleves, A. E. & Kelly, R. B. (1996). Rehearsing the ABCs. Protein translocation. *Curr. Biol.*, **6**, 276–278.

Corsi, A. K. & Schekman, R. (1996). Mechanism of polypeptide translocation into the endoplasmic reticulum. *J. Biol. Chem.*, **271**, 30299–30302.

Cristóbal, S., de Gier, J.-W., Nielsen, H. & von Heijne, G. (1999). Competition between the Sec and TAT protein translocation pathways in *Escherichia coli*. *EMBO J.*, to appear.

Cserző, M., Wallin, E., Simon, I., von Heijne, G. & Elofsson, A. (1997). Prediction of transmembrane α-helices in prokaryotic membrane proteins: the dense alignment surface method. *Protein Eng.*, **10**, 673–676.

Dalbey, R. E., Lively, M. O., Bron, S. & van Dijl, J. M. (1997). The chemistry and enzymology of the type I signal peptidases. *Protein Sci.*, **6**, 1129–1138.

Dalbey, R. E. & Robinson, C. (1999). Protein translocation into and across the bacterial plasma membrane and the plant thylakoid membrane. *Trends Biochem. Sci.*, **24**, 17–22.

Danese, P. N. & Silhavy, T. J. (1998). Targeting and assembly of periplasmic and outer-membrane proteins in *Escherichia coli*. *Annu. Rev. Genet.*, **32**, 59–94.

de Gier, J.-W., Valent, Q. A., von Heijne, G. & Luirink, J. (1997). The *E. coli* SRP: preferences of a targeting factor. *FEBS Lett.*, **408**, 1–4.

de Leeuw, E., Poland, D., Mol, O., Sinning, I., ten Hagen-Jongman, C. M., Oudega, B. & Luirink, J. (1997). Membrane association of FtsY, the *E. coli* SRP receptor. *FEBS Lett.*, **416**, 225–229.

Duong, F., Eichler, J., Price, A., Leonard, M. R. & Wickner, W. (1997). Biogenesis of the gram-negative bacterial envelope. *Cell*, **91**, 567–573.

Durbin, R. M., Eddy, S. R., Krogh, A. & Mitchison, G. (1998). Biological Sequence Analysis. Probabilistic models of proteins and nucleic acids. Cambridge University Press.

Edman, M., Jarhede, T., Sjöström, M., & Wieslander, Å. (1999). Different sequence patterns in signal peptides from mycoplasmas, other gram-positive bacteria, and *Escherichia coli*: A multivariate data analysis. *Proteins*, **35**, 195–205.

Emanuelsson, O. (1998). Prediction of subcellular location of plant proteins using neural networks. Master's thesis, Uppsala University and Stockholm University.

Feldheim, D. & Schekman, R. (1994). Sec72p contributes to the selective recognition of signal peptides by the secretory polypeptide translocation complex. *J. Cell Biol.*, **126**, 935–943.

Ferenci, T. & Silhavy, T. J. (1987). Sequence information required for protein translocation from the cytoplasm. *J. Bacteriol.*, **169**, 5339–5342.

Fikes, J. D., Barkocy-Gallagher, G. A., Klapper, D. G. & Bassford Jr., P. J. (1990). Maturation of *Escherichia coli* maltose-binding protein by signal peptidase I *in vivo*. Sequence requirements for efficient processing and demonstration of an alternate cleavage site. *J. Biol. Chem.*, **265**, 3417–3423.

Fleischmann, R. D., Adams, M. D., White, O., Clayton, R. A., Kirkness, E. F., Kerlavage, A. R., Bult, C. J., Tomb, J.-F., Dougherty, B. A., Merrick, J. M., McKenney, K., Sutton, G., FitzHugh, W., Fields, C., Gocayne, J. D., Scott, J., Shirley, R., Liu, L.-I., Glodek, A., Kelley, J. M., Weidman, J. F., Phillips, C. A., Spriggs, T., Hedblom, E., Cotton, M. D., Utterback, T. R., Hanna, M. C., Nguyen, D. T., Saudek, D. M., Brandon, R. C., Fine, L. D., Fritchman, J. L., Fuhrmann, J. L., Geoghagen, N. S. M., Gnehm, C. L., McDonald, L. A., Small, K. V., Fraser, C. M., Smith, H. O. & Venter, J. C. (1995). Whole-genome random sequencing and assembly of *Haemophilus influenzae* Rd. *Science*, **269**, 496–512.

Folz, R. J. & Gordon, J. I. (1987). Computer-assisted predictions of signal peptidase processing sites. *Biochem. Biophys. Res. Commun.*, **146**, 870–877.

Fraser, C. M., Gocayne, J. D., White, O., Adams, M. D., Clayton, R. A., Fleischmann, R. D., Bult, C. J., Kerlavage, A. R., Sutton, G., Kelley, J. M., Fritchman, J. L., Weidman, J. F., Small, K. V., Sandusky, M., Fuhrmann, J., Nguyen, D., Utterback, T. R., Saudek, D. M., Phillips, C. A., Merrick, J. M., Tomb, J.-F., Dougherty, B. A., Bott, K. F., Hu, P.-C., Lucier, T. S., Peterson, S. N., Smith, H. O., Hutchison III, C. A. & Venter, J. C. (1995). The minimal gene complement of *Mycoplasma genitalium*. *Science*, **270**, 397–404.

Freymann, D. M., Keenan, R. J., Stroud, R. M. & Walter, P. (1997). Structure of the conserved GTPase domain of the signal recognition particle. *Nature*, **385**, 361–364.

Füllekrug, J. & Nilsson, T. (1998). Protein sorting in the Golgi complex. *Biochim. Biophys. Acta*, **1404**, 77–84.

Gavel, Y. & von Heijne, G. (1990). A conserved cleavage-site motif in chloroplast transit peptides. *FEBS Lett.*, **261**, 455–458.

Gennity, J., Goldstein, J. & Inouye, M. (1990). Signal peptide mutants of *Escherichia coli*. *J. Bioenerg. Biomembr.*, **22**, 233–269.

Gennity, J. M. & Inouye, M. (1991). The protein sequence responsible for lipoprotein membrane localization in *Escherichia coli* exhibits remarkable specificity. *J. Biol. Chem.*, **266**, 16458–16464.

Gierasch, L. M. (1989). Signal sequences. *Biochemistry*, **28**, 923–930.

Glick, B. S. & Malhotra, V. (1998). The curious status of the Golgi apparatus. *Cell*, **95**, 883–889.

Goldstein, J., Lehnhardt, S. & Inouye, M. (1990). Enhancement of protein translocation across the membrane by specific mutations in the hydrophobic region of the signal peptide. *J. Bacteriol.*, **172**, 1225–1231.

Görlich, D., Hartmann, E., Prehn, S. & Rapoport, T. A. (1992). A protein of the endoplasmic reticulum involved early in polypeptide translocation. *Nature*, **357**, 47–52.

Green, R., Kramer, R. A. & Shields, D. (1989). Misplacement of the amino-terminal positive charge in the prepro-α-factor signal peptide disrupts membrane translocation *in vivo*. *J. Biol. Chem.*, **264**, 2963–2968.

Gribskov, M., McLachlan, A. D. & Eisenberg, D. (1987). Profile analysis: Detection of distantly related proteins. *Proc. Natl. Acad. Sci. USA*, **84**, 4355–4358.

Hamman, B. D., Hendershot, L. M. & Johnson, A. E. (1998). BiP maintains the permeability barrier of the ER membrane by sealing the lumenal end of the translocon pore before and early in translocation. *Cell*, **92**, 747–758.

Hanein, D., Matlack, K. E. S., Jungnickel, B., Plath, K., Kalies, K.-U., Miller, K. R., Rapoport, T. A. & Akey, C. W. (1996). Oligomeric rings of the Sec61p complex induced by ligands required for protein translocation. *Cell*, **87**, 721–732.

Haney, P. J., Badger, J. H., Buldak, G. L., Reich, C. I., Woese, C. R. & Olsen, G. J. (1999). Thermal adaptation analyzed by comparison of protein sequences from mesophilic and extremely thermophilic *Methanococcus* species. *Proc. Natl. Acad. Sci. USA*, **96**, 3578–3583.

Hann, B. C., Stirling, C. J. & Hewitt, E. W. (1992). Sec65 gene product is a subunit of the yeast signal recognition particle required for its integrity. *Nature*, **356**, 532–533.

Hartl, F.-U. & Wiedmann, M. (1993). A signal recognition particle in *Escherichia coli*? *Curr. Biol.*, **3**, 86–89.

Hartmann, E. & Prehn, S. (1994). The N-terminal region of the α-subunit of the TRAP complex has a conserved cluster of negative charges. *FEBS Lett.*, **349**, 324–326.

Hegde, R. S. & Lingappa, V. R. (1997). Membrane protein biogenesis: regulated complexity at the endoplasmic reticulum. *Cell*, **91**, 575–582.

Hegde, R. S., Voigt, S., Rapoport, T. A. & Lingappa, V. R. (1998). TRAM regulates the exposure of nascent secretory proteins to the cytosol during translocation into the endoplasmic reticulum. *Cell*, **92**, 621–631.

Henderson, I. R., Navarro-Garcia, F. & Nataro, J. P. (1998). The great escape: structure and function of the autotransporter proteins. *Trends Microbiol.*, **6**, 370–378.

Hengen, P. N., Bartram, S. L., Stewart, L. E. & Schneider, T. D. (1997). Information analysis of Fis binding sites. *Nucleic Acids Res.*, **25**, 4994–5002.

Henikoff, J. G. & Henikoff, S. (1996). Using substitution probabilities to improve position-specific scoring matrices. *CABIOS*, **12**, 135–143.

Herrmann, J. M., Malkus, P. & Schekman, R. (1999). Out of the ER—outfitters, escorts and guides. *Trends Cell Biol.*, **9**, 5–7.

Hertz, J., Krogh, A. & Palmer, R. G. (1991). Introduction to the theory of neural computation. Addison–Wesley, Santa Fe Institute, Studies in the Sciences of Complexity.

Hikita, C. & Mizushima, S. (1992). The requirement of a positive charge at the amino terminus can be compensated for by a longer hydrophobic stretch in the functioning of signal peptides. *J. Biol. Chem.*, **267**, 12375–12379.

Hirokawa, T., Boon-Chieng, S. & Mitaku, S. (1998). SOSUI: classification and secondary structure prediction system for membrane proteins. *Bioinformatics*, **14**, 378–379.

Hirst, J. D. & Sternberg, M. J. E. (1992). Prediction of structural and functional features of protein and nucleic acid sequences by artificial neural networks. *Biochemistry*, **31**, 7211–7218.

Hobohm, U., Scharf, M., Schneider, R. & Sander, C. (1992). Selection of representative protein data sets. *Protein Sci.*, **1**, 409–417.

Hofmann, K., Bucher, P., Falquet, L. & Bairoch, A. (1999). The PROSITE database, its status in 1999. *Nucleic Acids Res.*, **27**, 215–219.

Hofmann, K. & Stoffel, W. (1993). TMbase – a database of membrane spanning proteins segments. *Biol. Chem. Hoppe-Seyler*, **374**, 166.

Horton, P. & Nakai, K. (1996). A probabilistic classification system for predicting the cellular localization sites of proteins. *ISMB*, **4**, 109–115.

Horton, P. & Nakai, K. (1997). Better prediction of protein cellular localization sites with the *k* nearest neighbors classifier. *ISMB*, **5**, 147–152.

Howe, C. J. & Wallace, T. P. (1990). Prediction of leader peptide cleavage sites for polypeptides of the thylakoid lumen. *Nucleic Acids Res.*, **18**, 3417–3417.

Hueck, C. J. (1998). Type III protein secretion systems in bacterial pathogens of animals and plants. *Microbiol. Mol. Biol. Rev.*, **62**, 379–433.

Hurtley, S. M. (1993). Hot line to the secretory pathway. *Trends Biochem. Sci.*, **18**, 3–6. Meeting report.

Jones, D. T., Taylor, W. R. & Thornton, J. M. (1994). A model recognition approach to the prediction of all-helical membrane protein structure and topology. *Biochemistry*, **33**, 3038–3049.

Jungnickel, B., Rapoport, T. A. & Hartmann, E. (1994). Protein translocation: Common themes from bacteria to man. *FEBS Lett.*, **346**, 73–77.

Kaiser, C. A., Preuss, D., Grisafi, P. & Botstein, D. (1987). Many random sequences functionally replace the secretion signal sequence of yeast invertase. *Science*, **235**, 312–317.

Karamyshev, A. L., Karamysheva, Z. N., Kajava, A. V., Ksenzenko, V. N. & Nesmeyanova, M. A. (1998). Processing of *Escherichia coli* alkaline phosphatase: role of the primary structure of the signal peptide cleavage region. *J. Mol. Biol.*, **277**, 859–870.

Karlin, S. & Altschul, S. F. (1990). Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl. Acad. Sci. USA*, **87**, 2264–2268.

Kawabata, T., Ota, M. & Nishikawa, K. (1999). The protein mutant database. *Nucleic Acids Res.*, **27**, 355–357.

Kearns, M. (1997). A bound on the error of cross validation using the approximation and estimation rates, with consequences for the training-test split. *Neural Computation*, **9**, 1143–1161.

Kendall, D. A., Bock, S. C. & Kaiser, E. T. (1986). Idealization of the hydrophobic segment of the alkaline phosphatase signal peptide. *Nature*, **321**, 706–708.

Kendall, D. A. & Kaiser, E. T. (1988). A functional decaisoleucine-containing signal sequence. *J. Biol. Chem.*, **263**, 7261–7265.

Képès, F. (1996). The "+70 pause": hypothesis of a translational control of membrane protein assembly. *J. Mol. Biol.*, **262**, 77–86.

Kim, J. & Kendall, D. A. (1998). Identification of a sequence motif that confers SecB dependence on a SecB-independent secretory protein in vivo. *J. Bacteriol.*, **180**, 1396–1401.

Klein, P., Kanehisa, M. & DeLisi, C. (1985). The detection and classification of membrane-spanning proteins. *Biochim. Biophys. Acta*, **815**, 468–476.

Kohara, A., Yamamoto, Y. & Kikuchi, M. (1991). Alteration of N-terminal residues of mature human lysozyme affects its secretion in yeast and translocation into canine microsomal vesicles. *J. Biol. Chem.*, **266**, 20363–20368.

Kozak, M. (1984). Compilation and analysis of sequences upstream from the translational start site in eukaryotic mRNAs. *Nucleic Acids Res.*, **12**, 857–872.

Krogh, A. (1997). Two methods for improving performance of an HMM and their application for gene finding. *ISMB*, **5**, 179–186.

Krogh, A., Brown, M., Mian, I. S., Sjölander, K. & Haussler, D. (1994). Hidden Markov models in computational biology: Applications to protein modeling. *J. Mol. Biol.*, **235**, 1501–1531.

Krogh, A. & Vedelsby, J. (1995). Neural network ensembles, cross validation, and active learning. In Tesauro, G., Touretzky, D. S. & Leen, T. K., (eds.) *Advances in Neural Information Processing Systems 7*. MIT Press, Cambridge, MA, pp. 231–238.

Kunst, F., Ogasawara, N., Moszer, I., Albertini, A. M., Alloni, G., Azevedo, V., Bertero, M. G., Bessières, P., Bolotin, A., Borchert, S., Borriss, R., Boursier, L., Brans, A., Braun, M., Brignell, S. C., Bron, S., Brouillet, S., Bruschi, C. V., Caldwell, B., Capuano, V., Carter, N. M., Choi, S.-K., Codani, J.-J., Connerton, I. F., Cummings, N. J., Daniel, R. A., Denizot, F., Devine, K. M., Düsterhöft, A., Ehrlich, S. D., Emmerson, P. T., Entian, K. D., Errington, J., Fabret, C., Ferrari, E., Foulger, D., Fritz, C., Fujita, M., Fujita, Y., Fuma, S., Galizzi, A., Galleron, N., Ghim, S.-Y., Glaser, P., Goffeau, A., Golightly, E. J., Grandi, G., Guiseppi, G., Guy, B. J., Haga, K., Haiech, J., Harwood, C. R., Hénaut, A., Hilbert, H., Holsappel, S., Hosono, S., Hullo, M.-F., Itaya, M., Jones, L., Joris, B., Karamata, D., Kasahara, Y., Klaerr-Blanchard, M., Klein, C., Kobayashi, Y., Koetter, P., Koningstein, G., Krogh, S., Kumano, M., Kurita, K., Lapidus, A., Lardinois, S., Lauber, J., Lazarevic, V., Lee, S.-M., Levine, A., Liu, H., Masuda, S., Mauël, C., Médigue, C., Medina, N., Mellado, R. P., Mizuno, M., Moestl, D., Nakai, S., Noback, M., Noone, D., O'Reilly, M., Ogawa, K., Ogiwara, A., Oudega, B., Park, S.-H., Parro, V., Pohl, T. M., Portetelle, D., Porwollik, S., Prescott, A. M., Presecan, E., Pujic, P., Purnelle, B., Rapoport, G., Rey, M., Reynolds, S., Rieger, M., Rivolta, C., Rocha, E., Roche, B., Rose, M., Sadaie, Y., Sato, T., Scanlan, E., Schleich, S., Schroeter, R., Scoffone, F., Sekiguchi, J., Sekowska, A., Seror, S. J., Serror, P., Shin, B.-S., Soldo, B., Sorokin, A., Tacconi, E., Takagi, T., Takahashi, H., Takemaru, K., Takeuchi, M., Tamakoshi, A., Tanaka, T., Terpstra, P., Tognoni, A., Tosato, V., Uchiyama, S., Vandenbol, M., Vannier, F., Vassarotti, A., Viari, A., Wambutt, R., Wedler, E., Wedler, H., Weitzenegger, T., Winters, P., Wipat, A., Yamamoto, H., Yamane, K., Yasumoto, K., Yata, K., Yoshida, K., Yoshikawa, H.-F., Zumstein, E., Yoshikawa, H. & Danchin, A. (1997). The complete genome sequence of the gram-positive bacterium *Bacillus subtilis*. *Nature*, **390**, 249–256.

Kyte, J. & Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. *J. Mol. Biol.*, **157**, 105–132.

Ladunga, I., Czakó, F., Csabai, I. & Geszti, T. (1991). Improving signal peptide prediction accuracy by simulated neural network. *CABIOS*, **7**, 485–487.

Laforet, G. A., Kaiser, E. T. & Kendall, D. A. (1989). Signal peptide subsegments are not always functionally interchangeable. *J. Biol. Chem.*, **264**, 14478–14485.

Laforet, G. A. & Kendall, D. A. (1991). Functional limits of conformation, hydrophobicity, and steric constraints in prokayotic signal peptide cleavage regions. *J. Biol. Chem.*, **266**, 1326–1334.

Lehnhardt, S., Pollitt, S. & Inouye, M. (1987). The differential effect on two hybrid proteins of deletion mutations within the hydrophobic region of the *Escherichia coli* OmpA signal peptide. *J. Biol. Chem.*, **262**, 1716–1719.

Li, P., Beckwith, J. & Inouye, H. (1988). Alteration of the amino terminus of the mature sequence of a periplasmic protein can severely affect protein export in *Escherichia coli*. *Proc. Natl. Acad. Sci. USA*, **85**, 7685–7689.

Liao, S., Lin, J., Do, H. & Johnson, A. E. (1997). Both lumenal and cytosolic gating of the aqueous ER translocon pore are regulated from inside the ribosome during membrane protein integration. *Cell*, **90**, 31–41.

Liu, Y. (1995). Unbiased estimate of generalization error and model selection in neural network. *Neural Networks*, **8**, 215–219.

Long, E. O. (1998). Signal sequences stop killer cells. *Nature*, **39**, 740–743.

Lütcke, H. (1995). Signal recognition particle (SRP), a ubiquitous initiator of protein translocation. *Eur. J. Biochem.*, **228**, 531–550.

Lütcke, H. A., Chow, K. C., Mickel, F. S. & Moss, K. A. (1987). Selection of AUG initiation codons differs in plants and animals. *EMBO J.*, **6**, 43–48.

Machamer, C. E. (1996). ER-Golgi membrane traffic and protein targeting. In Hurtley, S. M., (ed.) *Protein targeting*. IRL Press, Oxford, pp. 123–151.

Martoglio, B. & Dobberstein, B. (1998). Signal sequences: more than just greasy peptides. *Trends Cell Biol.*, **8**, 410–415.

Martoglio, B., Graf, R. & Dobberstein, B. (1997). Signal peptide fragments of preprolactin and HIV-1 p-gp160 interact with calmodulin. *EMBO J.*, **16**, 6636–6645.

Mathews, B. W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim. Biophys. Acta*, **405**, 442–451.

Matlack, K. E., Mothes, W. & Rapoport, T. A. (1998). Protein translocation: tunnel vision. *Cell*, **92**, 381–390.

McGeoch, D. J. (1985). On the predictive recognition of signal peptide sequences. *Virus Res.*, **3**, 271–286.

McMurry, J. L. & Kendall, D. A. (1998). An artificial transmembrane segment directs SecA, SecB, and electrochemical potential-dependent translocation of a long amino-terminal tail. *J. Biol. Chem.*, **274**, 6776–6782.

Meyer, T. H., Ménétret, J. F., Breitling, R., Miller, K. R., Akey, C. W. & Rapoport, T. A. (1999). The bacterial SecY/E translocation complex forms channel-like structures similar to those of the eukaryotic Sec61p complex. *J. Mol. Biol.*, **285**, 1789–1800.

Miller, C. G. & Conlin, C. A. (1994). Signal peptide hydrolases. In von Heijne, G., (ed.) *Signal* *peptidases*. R. G. Landes Co., Austin, TX, pp. 49–57.

Mitsopoulos, C., Hashemzadeh-Bonehi, L. & Broome-Smith, J. K. (1997). N-tail translocation of mature β-lactamase across the *Escherichia coli* cytoplasmic membrane. *FEBS Lett.*, **419**, 18–22.

Möller, I., Beatrix, B., Kreibich, G., Sakai, H., Lauring, B. & Wiedmann, M. (1998). Unregulated exposure of the ribosomal M-site caused by NAC depletion results in delivery of non-secretory polypeptides to the Sec61 complex. *FEBS Lett.*, **441**, 1–5.

Monod, M., Haguenauer-Tsapis, R., Raused-Koenig, I. & Hinnen, A. (1989). Functional analysis of the signal sequence processing site of yeast acid phosphatase. *Eur. J. Biochem.*, **182**, 213–221.

Montoya, G., Svensson, C., Luirink, J. & Sinning, I. (1997). Crystal structure of the NG domain from the signal-recognition particle receptor FtsY. *Nature*, **385**, 365–368.

Mothes, W., Heinrich, S. U., Graf, R., Nilsson, I., von Heijne, G., Brunner, J. & Rapoport, T. A. (1997). Molecular mechanism of membrane protein integration into the endoplasmic reticulum. *Cell*, **89**, 523–533.

Mott, R. (1992). Maximum-likelihood estimation of the statistical distribution of Smith-Waterman local sequence similarity scores. *Bull. Math. Biol.*, **54**, 59–75.

Munro, S. (1998). Localization of proteins to the Golgi apparatus. *Trends Cell Biol.*, **8**, 11–15.

Nakai, K. (1996). Refinement of the prediction methods of signal peptides for the genome analyses of *Saccharomyces cerevisiae* and *Bacillus subtilis*. In Akutsu, T. *et al.*, (eds.) *Proceedings of the Seventh Workshop on Genome Informatics*. Universal Academy Press, pp. 72–81.

Nakai, K. & Horton, P. (1999). PSORT: a program for detecting sorting signals in proteins and predicting their subcellular localization. *Trends Biochem. Sci.*, **24**, 34–35.

Nakai, K. & Kanehisa, M. (1991). Expert system for predicting protein localization sites in Gram-negative bacteria. *Proteins*, **11**, 95–110.

Nakai, K. & Kanehisa, M. (1992). A knowledge base for predicting protein localization sites in eukaryotic cells. *Genomics*, **14**, 897–911.

Nakashima, H. & Nishikawa, K. (1994). Discrimination of intracellular an extracellular proteins using amino acid composition and residue-pair frequencies. *J. Mol. Biol.*, **238**, 54–61.

Nakayama, K. (1997). Furin: a mammalian subtilisin/Kex2p-like endoprotease involved in processing of a wide variety of precursor proteins. *Biochem. J.*, **327**, 625–635.

Needleman, S. B. & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, **48**, 443–453.

Nesmeyanova, M. A., Karamyshev, A. L., Karamysheva, Z. N., Kalinin, A. E., Ksenzenko, V. N. & Kajava, A. V. (1997). Positively charged lysine at the N-terminus of the signal peptide of the *Escherichia coli* alkaline phosphatase provides the secretion efficiency and is involved in the interaction with anionic phospholipids. *FEBS Lett.*, **403**, 203–207.

Nielsen, H., Brunak, S. & von Heijne, G. (1999). Machine learning approaches to the prediction of signal peptides and other protein sorting signals. *Protein Eng.*, **12**, 3–9.

Nilsson, I. & von Heijne, G. (1991). A *de novo* designed signal peptide cleavage cassette functions *in vivo*. *J. Biol. Chem.*, **266**, 3408–3410.

Nilsson, I. & von Heijne, G. (1992). A signal peptide with a proline next to the cleavage site inhibits leader peptidase when present in a *sec*-independent protein. *FEBS Lett.*, **299**, 243–246.

Nilsson, I. & von Heijne, G. (1993). Determination of the distance between the oligosaccharyltransferase active site and the endoplasmic reticulum membrane. *J. Biol. Chem.*, **268**, 5798–5801.

Nilsson, I., Whitley, P. & von Heijne, G. (1994). The COOH-terminal ends of internal signal and signal-anchor sequences are positioned differently in the ER translocase. *J. Cell Biol.*, **126**, 1127–1132.

Nishimura, N. & Balch, W. E. (1997). A di-acidic signal required for selective export from the endoplasmic reticulum. *Science*, **277**, 556–558.

Olsen, G. J. & Woese, C. R. (1997). Archaeal genomics: an overview. *Cell*, **89**, 991–994.

Paetzel, M., Dalbey, R. E. & Strynadka, N. C. J. (1998). Crystal structure of a bacterial signal peptidase in complex with a β-lactam inhibitor. *Nature*, **396**, 186–190.

Pearson, W. R. (1995). Comparison of methods for searching protein sequence databases. *Protein Sci.*, **4**, 1145–1160.

Perlman, D. & Halvorson, H. O. (1983). A putative signal peptidase recognition site and sequence in eukaryotic and prokaryotic signal peptides. *J. Mol. Biol.*, **167**, 391–409.

Persson, B. & Argos, P. (1994). Prediction of transmembrane segments in proteins utilising multiple sequence alignments. *J. Mol. Biol.*, **237**, 182–192.

Plath, K., Mothes, W., Wilkinson, B. M., Stirling, C. J. & Rapoport, T. A. (1998). Signal sequence recognition in posttranslational protein transport across the yeast ER membrane. *Cell*, **94**, 795–807.

Pohlschröder, M., Prinz, W. A., Hartmann, E. & Beckwith, J. (1997). Protein translocation in the three domains of life: variations on a theme. *Cell*, **91**, 563–566.

Pollock, D. D., Taylor, W. R. & Goldman, N. (1999). Coevolving protein residues: maximum likelihood identification and relationship to structure. *J. Mol. Biol.*, **287**, 187–198.

Popowicz, A. M. & Dash, P. F. (1988). SIGSEQ: a computer program for predicting signal sequence cleavage sites. *CABIOS*, **4**, 405–406.

Powers, T. & Walter, P. (1997). Co-translational protein targeting catalyzed by the *Escherichia coli* signal recognition particle and its receptor. *EMBO J.*, **16**, 4880–4886.

Prabhakaran, M. (1990). The distribution of physical, chemical and conformational properties in signal and nascent peptides. *Biochem. J.*, **269**, 691–696.

Prechelt, L. (1998). Automatic early stopping using cross validation: quantifying the criteria. *Neural Networks*, **11**, 761–767.

Presnell, S. R. & Cohen, F. E. (1993). Artificial neural networks for pattern recognition in biochemical sequences. *Annu. Rev. Biophys. Biomol. Struct.*, **22**, 283–298.

Prinz, W. A., Spiess, C., Ehrmann, M., Schierle, C. & Beckwith, J. (1996). Targeting of signal sequenceless proteins for export in *Escherichia coli* with altered protein translocase. *EMBO J.*, **15**, 5209–5217.

Promponas, V. J., Palaios, G. A., Pasquier, C. M., Hamodrakas, J. S. & Hamodrakas, S. J. (1998). CoPreTHi: A web tool which combines transmembrane protein segment prediction methods. *In Silico Biol.*, **1**, 0014.

Pugsley, A. P. (1993). The complete general secretory pathway in gram-negative bacteria. *Microbiol. Rev.*, **57**, 50–108.

Pugsley, A. P., Francetic, O., Possot, O. M., Sauvonnet, N. & Hardie, K. R. (1997). Recent progress and future directions in studies of the main terminal branch of the general secretory pathway in gram-negative bacteria–a review. *Gene*, **192**, 13–19.

Rapoport, T. A. (1990). Protein transport across the ER membrane. *Trends Biochem. Sci.*, **15**, 355–358.

Rapoport, T. A. (1991). Protein transport across the endoplasmic reticulum membrane: facts, models, mysteries. *FASEB J.*, **5**, 2792–2798.

Rapoport, T. A., Jungnickel, B. & Kutay, U. (1996). Protein transport across the eukaryotic endoplasmic reticulum and bacterial inner membranes. *Annu. Rev. Biochem.*, **65**, 271–303.

Rasmussen, B. A. & Silhavy, T. J. (1987). The first 28 amino acids of mature LamB are required for rapid and efficient export from the cytoplasm. *Genes Dev.*, **1**, 185–196.

Reinhardt, A. & Hubbard, T. (1998). Using neural networks for prediction of the subcellular location of proteins. *Nucleic Acids Res.*, **26**, 2230–2236.

Richter, S. & Lamppa, G. K. (1998). A chloroplast processing enzyme functions as the general stromal processing peptidase. *Proc. Natl. Acad. Sci. USA*, **95**, 7463–7468.

Rost, B. (1996). PHD: predicting 1D protein structure by profile based neural networks. *Methods Enzymol.*, **266**, 525–539.

Rost, B., Casadio, R. & Fariselli, P. (1996a). Refining neural network predictions for helical transmembrane proteins by dynamic programming. *ISMB*, **4**, 1704–1718.

Rost, B., Casadio, R., Fariselli, P. & Sander, C. (1995). Prediction of helical transmembrane segments at 95% accuracy. *Protein Sci.*, **4**, 521–533.

Rost, B., Fariselli, P. & Casadio, R. (1996b). Topology prediction for helical transmembrane proteins at 86% accuracy. *Protein Sci.*, **5**, 1704–1718.

Rusch, S. L. & Kendall, D. A. (1992). Signal sequences containing multiple aromatic residues. *J. Mol. Biol.*, **224**, 77–85.

Rusch, S. L. & Kendall, D. A. (1995). Protein transport via amino-terminal targeting sequences: common themes in diverse systems. *Mol. Membr. Biol.*, **12**, 295–307.

Russel, M. (1998). Macromolecular assembly and secretion across the bacterial cell envelope: type II protein secretion systems. *J. Mol. Biol.*, **279**, 485–499.

Ryan, P. & Edwards, C. O. (1995). Systematic introduction of proline in a eukaryotic signal sequence suggests asymmetry within the hydrophobic core. *J. Biol. Chem.*, **270**, 27876–27879.

Salmond, G. P. C. & Reeves, P. J. (1993). Membrane traffic wardens and protein secretion in Gram-negative bacteria. *Trends Biochem. Sci.*, **18**, 7–12.

Samuelsson, T. & Zwieb, C. (1999). The signal recognition particle database (SRPDB). *Nucleic Acids Res.*, **27**, 169–170.

Sander, C. & Schneider, R. (1991). Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins*, **9**, 56–68.

Santini, C.-L., Ize, B., Chanal, A., Müller, M., Giordano, G. & Wu, L.-F. (1998). A novel sec-independent periplasmic protein translocation pathway in *Escherichia coli*. *EMBO J.*, **17**, 101–112.

Sargent, F., Bogsch, E. G., Stanley, N. R., Wexler, M., Robinson, C., Berks, B. C. & Palmer, T. (1998). Overlapping functions of components of a bacterial Sec-independent protein export pathway. *EMBO J.*, **17**, 3640–3650.

Schatz, G. & Dobberstein, B. (1996). Common principles of protein translocation across membranes. *Science*, **271**, 1519–1526.

Schneider, G., Sjöling, S., Wallin, E., Wrede, P., Glaser, E. & von Heijne, G. (1998). Feature-extraction from endopeptidase cleavage sites in mitochondrial targeting peptides. *Proteins*, **30**, 49–60.

Schneider, G. & Wrede, P. (1993). Development of artificial neural filters for pattern recognition in protein sequences. *J. Mol. Evol.*, **36**, 586–595.

Schneider, G. & Wrede, P. (1994). The rational design of amino acid sequences by artificial neural networks and simulated molecular evolution: De novo design of an idealized leader peptidase cleavage site. *Biophys. J.*, **66**, 335–344.

Schneider, T. D. & Stephens, R. M. (1990). Sequence logos: A new way to display consensus sequences. *Nucleic Acids Res.*, **18**, 6097–6100.

Seidah, N. G. & Chrétien, M. (1997). Eukaryotic protein processing: endoproteolysis of precursor proteins. *Curr. Opin. Biotechnol.*, **8**, 602–607.

Settles, A. M. & Martienssen, R. (1998). Old and new pathways of protein export in chloroplasts and bacteria. *Trends Cell Biol.*, **8**, 494–501.

Settles, A. M., Yonetani, A., Baron, A., Bush, D. R., Cline, K. & Martienssen, R. (1997). Sec-independent protein translocation by the maize Hcf106 protein. *Science*, **278**, 1467–1470.

Shinde, U. P. (1990). Can the topological distribution of membrane spanning amino acid residues be responsible for the recognition of signal peptides by signal peptide peptidases? *Biosci. Rep.*, **10**, 537–546.

Shinde, U. P., Guru Row, T. N. & Mawal, Y. R. (1989). Export of proteins across membranes: The helix reversion hypothesis. *Biosci. Rep.*, **9**, 737–745.

Shultzaberger, R. K. & Schneider, T. D. (1999). Using sequence logos and information analysis of LrP DNA binding sites to investigate discrepancies between natural selection and SELEX. *Nucleic Acids Res.*, **27**, 882–887.

Siegel, V. (1997). Recognition of a transmembrane domain: Another role for the ribosome? *Cell*, **90**, 5–8.

Siegel, V. & Walter, P. (1986). Removal of the Alu structural domain from signal recognition particle leaves its protein translocation activity intact. *Nature*, **320**, 81–84.

Silhavy, T. J. (1997). Death by lethal injection. *Science*, **278**, 1085–1086.

Sjöström, M., Wold, S., Wieslander, Å. & Rilfors, L. (1987). Signal peptide amino acid sequences in *Escherichia coli* contain information related to final protein localization. A multivariate data analysis. *EMBO J.*, **6**, 823–831.

Smith, T. F. & Waterman, M. S. (1981). Identification of common molecular subsequences. *J. Mol. Biol.*, **147**, 195–197.

Sonnhammer, E. L. L., von Heijne, G. & Krogh, A. (1998). A hidden Markov model for predicting transmembrane helices in protein sequences. *ISMB*, **6**, 175–182.

Staden, R. (1984). Computer methods to locate signals in nucleic acids sequences. *Nucleic Acids Res.*, **12**, 505–519.

Stormo, G. D. & Fields, D. S. (1998). Specificity, free energy and information content in protein-DNA interactions. *Trends Biochem. Sci.*, **23**, 109–113.

Tjalsma, H., Noback, M. A., Bron, S., Venema, G., Yamane, K. & van Dijl, J. M. (1997). *Bacillus subtilis* contains four closely related type I signal peptidases with overlapping substrate specificities. Constitutive and temporally controlled expression of different *sip* genes. *J. Biol. Chem.*, **272**, 25983–25992.

Tusnády, G. E. & Simon, I. (1998). Principles governing amino acid composition of integral membrane proteins: Applications to topology prediction. *J. Mol. Biol.*, **283**, 489–506.

Ulbrandt, N. D., Newitt, J. A. & Bernstein, H. D. (1997). The *E. coli* signal recognition particle is required for the insertion of a subset of inner membrane proteins. *Cell*, **88**, 187–196.

Valent, Q. A., Scotti, P. A., High, S., de Gier, J.-W., von Heijne, G., Lentzen, G., Wintermeyer, W., Oudega, B. & Luirink, J. (1998). The *Escherichia coli* SRP and SecB targeting pathways converge at the translocon. *EMBO J.*, **17**, 2504–2512.

van Klompenburg, W. & de Kruijff, B. (1998). The role of anionic lipids in protein insertion and translocation in bacterial membranes. *J. Membr. Biol.*, **162**, 1–7.

von Heijne, G. (1983). Patterns of amino acids near signal sequence cleavage sites. *Eur. J. Biochem.*, **133**, 17–21.

von Heijne, G. (1984). How signal sequences maintain cleavage specificity. *J. Mol. Biol.*, **173**, 243–251.

von Heijne, G. (1985). Signal sequences. The limits of variation. *J. Mol. Biol.*, **184**, 99–105.

von Heijne, G. (1986a). Net N–C charge imbalance may be important for signal sequence function in bacteria. *J. Mol. Biol.*, **192**, 287–290.

von Heijne, G. (1986b). A new method for predicting signal sequence cleavage sites. *Nucleic Acids Res.*, **14**, 4683–4690.

von Heijne, G. (1987). Sequence Analysis in Molecular Biology: Treasure Trove or Trivial Pursuit? Academic Press Inc., London.

von Heijne, G. (1988). Transcending the impenetrable: How proteins come to terms with membranes. *Biochim. Biophys. Acta*, **947**, 307–333.

von Heijne, G. (1989). The structure of signal peptides from bacterial lipoproteins. *Protein Eng.*,

2, 531–534.

von Heijne, G. (1990). The signal peptide. *J. Membr. Biol.*, **115**, 195–201.

von Heijne, G. (1992). Membrane protein structure prediction: Hydrophobicity analysis and the positive-inside rule. *J. Mol. Biol.*, **225**, 487–494.

von Heijne, G. & Abrahmsén, L. (1989). Species-specific variation in signal peptide design. *FEBS Lett.*, **244**, 439–446.

Weigend, A. S., Huberman, B. A. & Rumelhart, D. E. (1990). Predicting the future: A connectionist approach. *Int. J. Neural Syst.*, **1**, 193–209.

Weiner, J. H., Bilous, P. T., Shaw, G. M., Lubitz, S. P., Frost, L., Thomas, G. H., Cole, J. A. & Turner, R. J. (1998). A novel and ubiquitous system for membrane targeting and secretion of cofactor-containing proteins. *Cell*, **93**, 93–101.

Wiedmann, B., Sakai, H., Davis, T. A. & Wiedmann, M. (1994). A protein complex required for signal-sequence-specific sorting and translocation. *Nature*, **370**, 434–440.

Wilkinson, B. M., Regnacq, M. & Stirling, C. J. (1997). Protein translocation across the membrane of the endoplasmic reticulum. *J. Membr. Biol.*, **155**, 189–197.

Wrede, P., Landt, O., Klages, S., Fatemi, A., Hahn, U. & Schneider, G. (1998). Peptide design aided by neural networks: biological activity of artificial signal peptidase I cleavage sites. *Biochemistry*, **37**, 3588–3593.

Wu, C. H. (1997). Artificial neural networks for molecular sequence analysis. *Comput. Chem.*, **21**, 237–256.

Yamamoto, Y., Taniyama, Y. & Kikuchi, M. (1989). Important role of the proline residue in the signal sequence that directs the secretion of human lysozyme in *Saccharomyces cerevisiae*. *Biochemistry*, **28**, 2728–2732.

Yamamoto, Y., Taniyama, Y., Kikuchi, M. & Ikehara, M. (1987). Engineering of the hydrophobic segment of the signal sequence for efficient secretion of human lysozyme by *Saccharomyces cerevisiae*. *Biochem. Biophys. Res. Commun.*, **149**, 431–436.

Zheng, N. & Gierasch, L. M. (1996). Signal sequences: the same yet different. *Cell*, **86**, 849–852.