ΠΑΝΕΠΙΣΤΗΜΙΟ ΣΤΕΡΕΑΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΒΙΟΙΑΤΡΙΚΗ

ΜΑΡΚΟΒΙΑΝΑ ΜΟΝΤΕΛΑ ΣΤΗΝ ΑΝΑΛΥΣΗ ΠΡΩΤΕΪΝΙΚΩΝ ΑΚΟΛΟΥΘΙΩΝ

Χάτζο Μαρία

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
Υπεύθυνος:
Μπάγκος Παντελής
Επίκουρος Καθηγητής (υπό διορισμό)

Λαμία, 2008

ΠΡΟΛΟΓΟΣ

Η παρούσα διπλωματική εργασία πραγματοποιήθηκε εξ ολοκλήρου στο Τμήμα Πληροφορικής με Εφαρμογές στη Βιοϊατρική του Πανεπιστημίου Στερεάς Ελλάδας κατά το ακαδημαϊκό έτος 2007 – 2008, υπό την επίβλεψη του Επίκουρου Καθηγητή (υπό διορισμό) Παντελή Μπάγκου.

Όσα και να γράψει κανείς στον πρόλογο της διπλωματικής του εργασίας, της πρώτης, ουσιαστικά, επαφής ενός νέου ερευνητή με το συναρπαστικό πεδίο της έρευνας, για τον άνθρωπο που τον καθοδήγησε στα πρώτα του βήματα, είναι σίγουρα λίγα. Ωστόσο, δε θα μπορούσα να παραλείψω να πω ένα πολύ μεγάλο "ευχαριστώ" στον κύριο Μπάγκο για μια σειρά από λόγους: πρώτα από όλα, γιατί με επέλεξε να εργαστώ μαζί του και με τα μέλη της ομάδας του, με τους οποίους είχαμε, όποτε χρειάστηκε, μια άψογη συνεργασία. Επιπλέον, καθόλο το διάστημα της συνεργασίας μας, ήταν για μένα πάντοτε εκεί για να μου επιλύει τις απορίες μου, να με συμβουλεύει και να με ενθαρρύνει να προσπαθώ ακόμα κι όταν απογοητευόμουν, πράγματα τα οποία πιστεύω συνέβαλαν αποφασιστικά στο να αγαπήσω αυτό με το οποίο ασχολήθηκα και να δώσω τον καλύτερό μου εαυτό, γεγονός ιδιαίτερα σημαντικό για μια φοιτήτρια που μπαίνει για πρώτη φορά στο χώρο της έρευνας στη Βιοπληροφορική. Αισθάνομαι, λοιπόν, πολύ τυχερή που είχα ως επιβλέπων μου τον κύριο Μπάγκο, καθώς νίωθω πολύ ωφελημένη από το χρόνο που πέρασα δίπλα του στο Πανεπιστήμιο και δούλευα στο θέμα της διπλωματικής μου εργασίας.

Ένα μεγάλο ευχαριστώ οφείλω αναμφίβολα και στα υπόλοιπα δύο μέλη της Τριμελούς Συμβουλευτικής Επιτροπής, την Αναπληρώτρια Καθηγήτρια κα. Αμαλία Καραγκούνη-Κύρτσου και τον Επίκουρο Καθηγητή κ. Βασίλειο Πλαγιανάκο οι οποίοι αφιέρωσαν τμήμα του πολύτιμου χρόνου τους στο να παρακολουθήσουν την παρουσίαση της διπλωματικής μου εργασίας.

Επίσης θα ήθελα να ευχαριστήσω τον κ. Αναστάσιο Ιωαννίδη για την προσοχή και τη βοήθεια που μου προσέφερε, σε διάφορα στάδια της πτυχιακής μου εργασίας. Οι παρατηρήσεις του να συνέβαλαν σημαντικά στη βελτίωση της.

Αισθάνομαι ακόμα, ότι πρέπει να ευχαριστήσω και την κυρία Αναστασία Διαμαντοπούλου, η οποία αν και δεν συνείσφερε άμεσα σε κάποιο στάδιο της

πτυχιακής μου εργασίας, υπήρξε το άτομο που με έκανε να αγαπήσω και να θέλω να ασχοληθώ με τον συναρπαστικό τομέα της Βιολογίας. Το μεγαλύτερο ποσοστό των γνώσεων που διαθέτω πάνω στη Βιολογία το οφείλω σε εκείνη. Την ευχαριστώ λοιπόν για την πίστη που έδειξε σε εμένα και τις ικανότητες μου και στον (άπειρο) χρόνο που αφιέρωσε παραδίδοντας μου μαθήματα Βιολογίας, αλλά και μαθήματα ζωής.

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω ένα επιπλέον άτομο, πολύ σημαντικό για μένα, τον Κωνσταντίνο Τσιρίγο, στην πολύτιμη βοήθεια του οποίου οφείλω πολλά. Θα ήθελα να τον ευχαριστήσω, οπότε, για την αμέριστη ηθική συμπαράσταση και τις συμβουλές που μου προσέφερε σε διάφορα στάδια της πτυχιακής εργασίας. Υπήρξε για μένα μια πηγή έμπνευσης όλο το διάστημα που συνεργαστήκαμε για τον τρόπο που χειρίζονταν τα επιστημονικά ζητήματα, την υπομονή και επιμονή που έδειχνε και για την ευφυΐα που τον διέκρινε.

Κλείνοντας δεν θα μπορούσα να μην ευχαριστήσω την οικογένεια μου και συγκεκριμένα τους γονείς μου και τη γιαγιά μου για τη στήριξη τους, την οικονομική τους ενίσχυση και τις θυσίες που έκαναν τόσα χρόνια για μένα. Οφείλω τα πάντα σ' αυτούς και κυρίως στη μητέρα μου, το σημαντικότερο άνθρωπο στη ζωή μου, η οποία ήταν πάντα δίπλα μου και στην οποία είναι αφιερωμένη αυτή η πτυχιακή εργασία.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ

Οι πρωτεΐνες διαδραματίζουν σημαντικούς και ποικίλους ρόλους σχεδόν σε όλες τις δραστηριότητες που συντελούνται μέσα σε ένα ζωντανό κύτταρο. Αυτοί ποικίλουν, από την αναγνώριση και σύνδεση κυττάρων μεταξύ τους ή και με άλλους σχηματισμούς, τη λειτουργία τους ως μοριακοί υποδοχείς, τη μεταφορά ουσιών διαμέσου των μεμβρανών, έως και την εξειδικευμένη ενζυμική δραστηριότητα.. Οι πρωτεΐνες χωρίζονται σε διάφορες κατηγορίες, από τις οποίες οι σημαντικότερες είναι οι διαμεμβρανικές, οι σφαιρικές υδατοδιαλυτές και οι ινώδης πρωτεΐνες. Τη μεγάλη πλειοψηφία των διαμεμβρανικών πρωτεϊνών, αποτελούν πρωτεΐνες των οποίων τα διαμεμβρανικά τμήματα έχουν τη δομή α-έλικας η οποία συντίθεται από υδρόφοβα αμινοξικά κατάλοιπα που διαπερνούν το υδρόφοβο περιβάλλον της λιπιδικής διπλοστιβάδας. Ορισμένες φορές, δύο α-έλικες περιελίσσονται η μια γύρω από την άλλη για να σχηματίσουν μια πολύ σταθερή δομή, η οποία είναι γνωστή ως σπειροειδές σπείραμα (coiled-coil), μια διαμόρφωση που παρουσιάζουν και πολλά άλλα είδη πρωτεϊνών, όπως οι ινώδεις πρωτεΐνες. Ειδική, πιο σπάνια και λιγότερο μελετημένη περίπτωση διαμεμβρανικών πρωτεϊνών αποτελούν οι πρωτεΐνες της εξωτερικής μεμβράνης των αρνητικών κατά Gram βακτηρίων καθώς (πιθανότατα) και των μιτοχονδρίων και των χλωροπλαστών, στις οποίες τα διαμεμβρανικά τμήματα είναι αντιπαράλληλοι κλώνοι μιας β-πτυχωτής επιφάνειας (διαμεμβρανικά β-βαρέλια).

Η μεγάλη σπουδαιότητα των πρωτεϊνών, αλλά και οι εγγενείς δυσκολίες που παρουσιάζονται στην προσπάθεια κρυστάλλωσης αυτών, καθιστούν απαραίτητη τη δημιουργία υπολογιστικών αλγορίθμων οι οποίοι θα προβλέπουν σχετικά αξιόπιστα και γρήγορα τη δευτεροταγή τους δομή αλλά και τα πιθανά λειτουργικά τους χαρακτηριστικά. Οι αλυσίδες Markov (MCs) παρέχουν ένα καλό στατιστικό πλαίσιο για τη λύση ενός μεγάλου φάσματος προβλημάτων. Η μελέτη των αλυσίδων Markov (MCs) άρχισε το 20 αιώνα από τον Markov, ο οποίος έθεσε τα θεμέλια για τη θεωρία των πιθανολογικών διαδικασιών. Μόνο προς το τέλος της δεκαετίας του '80 και στις αρχές της δεκαετίας του '90 όμως, οι αλυσίδες Markov (MCs) άρχισαν να χρησιμοποιούνται στη Μοριακή Βιολογία για υπολογιστική ανάλυση ακολουθιών και

τον προσδιορισμό των πρωτεϊνικών δομών. Έκτοτε έχουν δημιουργηθεί πολλά μοντέλα που στηρίζονται στα MCs, όπως τα: HMMs (Hidden Markov Models), VLMCs (Variable Length Markov Chains), MTDs (Mixed Memory Markov Models) και τα IMMs (Interpolated Markov Models).

Μια αλυσίδα Markov, είναι μια πιθανολογική διαδικασία, που στηρίζεται στην ιδέα ότι κάθε ένα από τα ενδεχόμενα εξαρτάται μόνο από το αμέσως προηγούμενο του, ή αλλιώς το κάθε ενδεχόμενο καθορίζει με κάποια πιθανότητα το αμέσως επόμενό του. Αν αυτή η εξάρτηση επεκταθεί και σε 2,3,...,n προηγούμενα ενδεχόμενα τότε μιλάμε για αλυσίδες Markov $2^{ης}$,$3^{ης}$, ...,$n^{ης}$ τάξης. Με άλλα λόγια, η περιγραφή της παρούσας κατάστασης παρέχει πλήρως όλες τις πληροφορίες που θα μπορούσαν να επηρεάσουν τη μελλοντική εξέλιξη της διαδικασίας.

Τα Markov Models παρά τα πολλά πλεονεκτήματά που παρουσιάζουν διαθέτουν και κάποια μειονεκτήματα. Πρώτον όταν προσπαθούμε να εφαρμόσουμε MCs μεγάλης τάξεως τότε προκύπτει ένας πολύ μεγάλος αριθμός παραμέτρων, όπως αναλύσαμε και παραπάνω. που δεν θα μπορούσαμε να υπολογίσουμε. Δεύτερον τα απλά ΜΜ, αν και δεν απαιτούν μεγάλο χρονικό διάστημα για τη δημιουργία τους, έχει αποδειχτεί ότι δεν αποδίδουν πολύ καλά σε βιολογικές εφαρμογές. Απ' την άλλη τα HMMs, τα οποία χρησιμοποιούνται ευρέως στη βιολογία τα τελευταία χρόνια και αποτελούν λύση για ένα μεγάλο φάσμα βιολογικών προβλημάτων με μεγάλα ποσοστά επιτυχίας, είναι ιδιαίτερα χρονοβόρα στη δημιουργία τους. Μια λύση στα προβλήματα αυτά αποτελούν τα VLMCs. Τα VLMCs είναι αλυσίδες Markov με το επιπρόσθετο χαρακτηριστικό ότι η μνήμη τους εξαρτάται από έναν μεταβλητό αριθμό παρελθοντικών τιμών, ανάλογα με το πώς το παρελθόν (οι προηγούμενες τιμές) μοιάζει. Τα VLMCs στηρίζονται στη δημιουργία πιθανοθεωρητικών δέντρων για των υπολογισμό των πιθανοτήτων μετάβασης. Οι ακολουθίες που θα χρησιμοποιηθούν για να εκπαιδεύσουν το VLMC δεν χρειάζεται να έχουν στοιχηθεί και μπορεί να έχουν οποιοδήποτε μήκος. Η μέθοδος είναι αυτόματη, και μπορεί επιπλέον να εφαρμοστεί, χωρίς να προϋποθέτει οποιαδήποτε άλλη βιολογική πληροφορία.

Στην αρχή έγινε μια εκτενής αναζήτηση στη βιβλιογραφία και στο διαδίκτυο, με σκοπό να βρεθούν περιλήψεις άρθρων που σχετίζονταν με το θέμα μας. Το

σύνολο των άρθρων που ανακτήθηκε, μελετήθηκε πλήρως με στόχο να αποκτήσουμε μια πλήρη εικόνα των εφαρμογών των Markov Models που υπάρχουν. Στη συνέχεια ασχοληθήκαμε εκτενώς με τα VLMCs, τα οποία απ' ότι διαπιστώσαμε έχουν εφαρμοστεί με μεγάλη επιτυχία σε οικογένειες πρωτεϊνών, που παρουσιάζουν μεγάλη ομολογία μεταξύ τους, και προσπαθήσαμε να διερευνήσουμε αν θα μπορούσαν να εφαρμοστούν με την ίδια επιτυχία σε άλλα είδη πρωτεϊνών όπως οι διαμεμβρανικές πρωτεΐνες και συγκεκριμένα αυτές που έχουν διαμόρφωση β-βαρελιού (β-barrel) και οι πρωτεΐνες με διαμόρφωση σπειροειδούς σπειράματος (coiled-coils), που παρουσιάζουν μικρότερη συντήρηση.

Για τους σκοπούς αυτούς αναπτύξαμε ένα πρόγραμμα το οποίο χρησιμοποιώντας ένα VLMC, δημιουργεί ένα πιθανοθεωρητικό μοντέλο το οποίο αποτελεί ένα καλό διαχωριστικό εργαλείο, που έχει την ικανότητα να μας δείχνει αν μια πρωτεΐνη είναι π.χ. διαμεμβρανική ή όχι. Συγκεκριμένα εφαρμόσαμε 6 διαφορετικές παραλλαγές της μεθόδου αυτή. Στις 3 παραλλαγές του προγράμματος χρησιμοποιήσαμε τον αλγόριθμο PST (Bejerano and Yona, 2001) και στις υπόλοιπες 3 χρησιμοποιήσαμε μια παραλλαγή του αλγόριθμου PST, τον SPST (Leonardi, 2006) για να δημιουργήσουμε τα ίδια πιθανοθεωρητικά μοντέλα που δημιουργήσαμε με το PST στις 3 πρώτες παραλλαγές του προγράμματος.

Για να εκπαιδεύσουμε το VLMC (είτε PST είτε SPST) ψάξαμε και βρήκαμε, κυρίως από την Swiss-Prot αλλά και από άλλες βάσεις στο διαδίκτυο καθώς και από τη βιβλιογραφία, διαμεμβρανικές πρωτεΐνες με διαμόρφωση β-βαρελιού (β-barrel) και πρωτεΐνες με διαμόρφωση σπειροειδούς σπειράματος (coiled-coils) .

ΛΕΞΕΙΣ-ΚΛΕΙΔΙΑ

Μαρκοβιανά Μοντέλα, Μεταβλητής Μνήμης Μαρκοβιανά Μοντέλα (VLMCs), πρόγνωση πρωτεϊνών.

ABSTRACT

The proteins perform a variety of very important biological functions necessary for the survival of the cell. They are involved in cellular recognition and adhesion, they act as molecular receptors, they transport substrates through membranes and they exhibit specialized enzymatic activity. The proteins are separated in various categories, from which the most important are integral membrane proteins, water soluble proteins and fibrillous proteins. The vast majority of integral membrane proteins constitute proteins whose transmembrane segments form an alpha-helix composed of mainly hydrophobic residues, spanning the lipid bilayer. Sometimes, two a-helix form a very constant structure, which looks like a spiral and is known as coiled-coil. This domain is found in many kinds of proteins, such as the fibrillous proteins. A more specialized and less well-studied case, is the case of integral membrane proteins found in the outer membrane of Gram-negative bacteria and (presumably) in the outer envelope of mitochondria and chloroplasts, proteins whose transmembrane segments are formed by amphipathic beta-strands that form a closed barrel (beta-barrels).

The importance of proteins, as well as the inherent difficulties in crystallizing and obtaining a three-dimensional structure of these proteins, dictates the need for developing computational algorithms and tools that will allow a reliable and fast prediction of their structural and functional features. Markov Chains (MCs) provide a good statistical frame-work for solving a wide range of time-series problem. The study of Markov Chains (MCs) was initiated in early 1900s by Markov, who laid the foundation for the theory of stochastic processes. But it was not till the late 1980s and early 1990s, when Marcov Chais were subsequently introduced to computational sequence analysis and protein structural modeling in molecular biology. Since then a lot of models, that are based in MCs, have been developed , such as: HMMs (Hidden Markov Models), VLMCs (Variable Length Markov Chains), MTDs (Mixed Memory Markov Models) and IMMs (Interpolated Markov Models).

Markov chain is a stochastic process, which is motivated by the observation that one symbol in a sequence is depending on all its predecessors, i. e. on the context

of symbols observed so far. If this dependence is extended in 2, 3,..., n predecessors we have 2^{nd}, 3^{rd},...,n^{th} order Markov Chain. In other words, the description of the present state provides all the information that could influence the future development of the process.

Markov Models present however certain disadvantages. Firstly when we try to apply MCs of higher order a large number of parameters, that we can not calculate, results. Secondly it has been proved that MM of low complexity do not perfom very well in biological applications, although they do not require a large amount of time to be constructed. On the other hand HMMs, which have been widely used in the biology and have been successfully applied to a large range of biological problems, have proven to be very time-consuming. A way of facing the problem of exponential parameter growth as well as the time problem, is offered by the concept of variable length Markov chains (VLMCs). VLMCs are Markov Chains with the additional characteristic that their memory depends on a variable number of its predecessors, depending on how the "past" (the context of symbols observed so far) resembles. The VLMCs use trees for the calculation of transitions probabilities. That input sequences that will be used in order to train the VLMC can be of arbitrary length and they do not need to be aligned. The method is automatic, and can be applied, without assuming any preliminary biological information, as well.

At first we searched the bibliography and the internet, in order to find summaries of articles that were related with our subject. The total number of articles that was recovered was studied in order to learn about Markov Models applications appearing in computational biology. Then we dealt extensively with the VLMCs, which can serve as a predictive tool for protein sequence classification, and for detecting conserved patterns (possibly functionally or structurally important) within protein sequences. The method was tested on the Pfam database of protein families with more than satisfactory performance. Moreover we tried to investigate if they could be applied with the same success in other kinds of proteins such as beta-barrel outer membrane proteins and coiled-coils proteins, that presents less conservation.

In order to achieve this goal we developed a program, which is based on a VLMC model. This program can discriminate for example beta-barrel outer membrane proteins from water soluble ones. As a matter of fact we developed 6 different variants of this method. In the first 3 variants of the program we used the PST algorithm (Bejerano and Yona, 2001) and in the remainder 3 we used a variant of the PST algorithm, the SPST (Leonardi, 2006) algorithm in order to create the same probabilistic models that we developed with the PST algorithm during the 3 first variants of the program. In order to train the VLMC (PST or SPST) algorithm we searched and found, mainly from the Swiss-Prot database, as well as, from other online databases and bibliography too, beta-barrel outer membrane proteins and coiled-coils proteins.

KEY WORDS

Markov Models, Variable Length Markov Modes (VLMCs), protein structure prediction

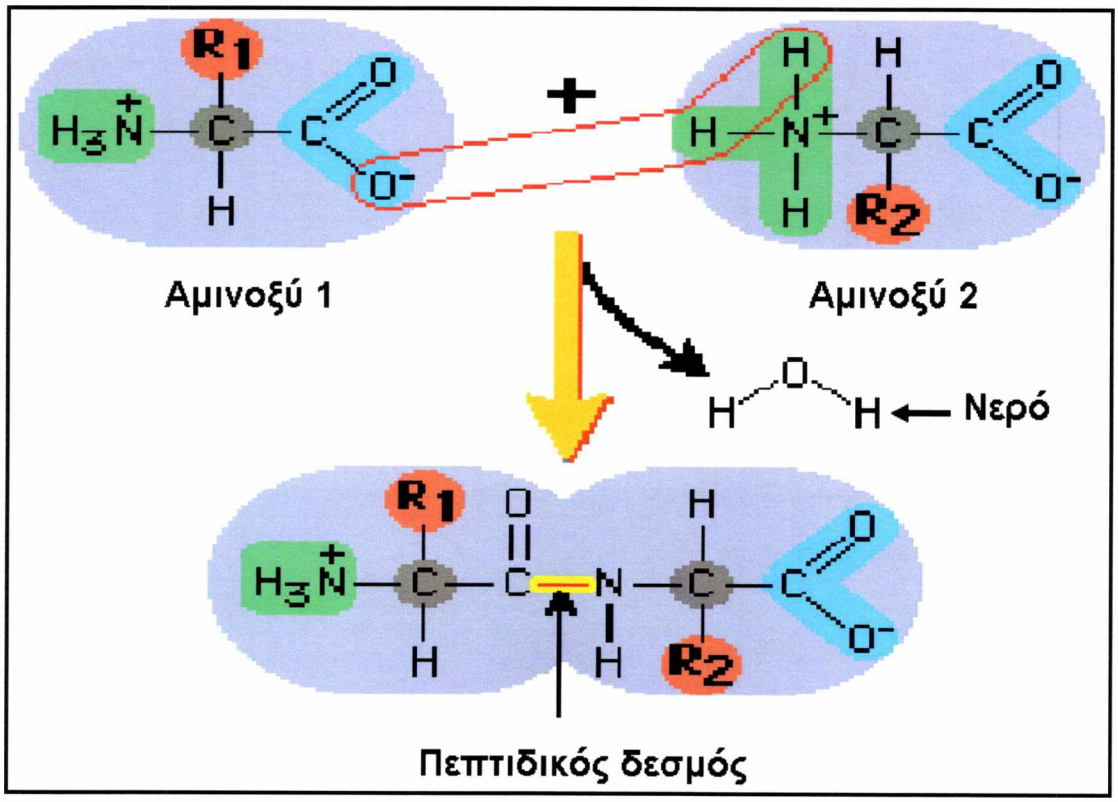
1. ΕΙΣΑΓΩΓΗ

1.1 Πρωτεΐνες

Οι πρωτεΐνες αποτελούν το μεγαλύτερο μέρος της ξηρής μάζας ενός κυττάρου. Όταν παρατηρούμε κυτταρικές δομές με το μικροσκόπιο ή όταν αναλύουμε την ηλεκτρική ή βιοχημική δραστηριότητα τους, στην πλειονότητα εξετάζουμε πρωτεΐνες. Ωστόσο οι πρωτεΐνες δεν είναι απλώς οι δομικοί λίθοι από τους οποίους συγκροτούνται τα κύτταρα, μιας και επιτελούν όλες σχεδόν τις κυτταρικές λειτουργίες. Για παράδειγμα, τα ένζυμα διαθέτουν τις περίπλοκες μοριακές επιφάνειές τους που διεκπεραιώνουν τις χημικές αντιδράσεις του κυττάρου. Ενώ οι πρωτεΐνες οι οποίες είναι ενσωματωμένες στις κυτταρικές μεμβράνες σχηματίζουν διαύλους και αντλίες που ελέγχουν τη δίοδο μικρών μορίων μέσα και έξω από αυτές στον πυρήνα ορισμένων κυττάρων (π.χ. αντλία Κ-Να).

Η δυνατότητα των πρωτεϊνών να επιτελούν πολλές και διαφορετικές λειτουργίες προκύπτει από τον τεράστιο αριθμό διαφορετικών στερεοδομών που μπορεί να προσλάβουν δίνοντας νόημα στη φράση "η λειτουργία εξαρτάται από τη δομή" (Anfinsen, 1961). Το σχήμα μιας πρωτεΐνης όμως καθορίζεται από την αλληλουχία των αμινοξέων της, όπως θα διαπιστώσουμε και παρακάτω.

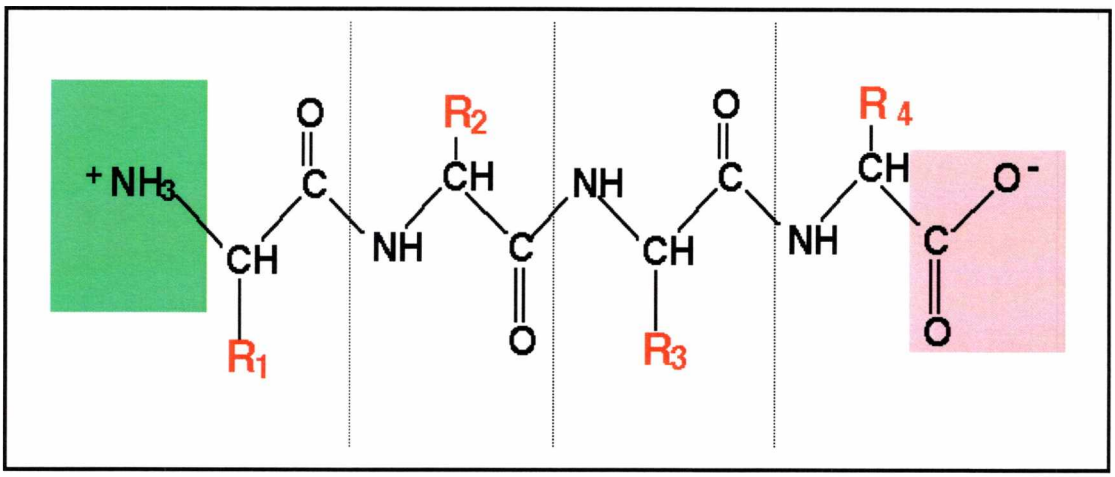
Ας ξεκινήσουμε όμως από την αρχή, από τους δομικούς λίθους των πρωτεϊνών τα αμινοξέα. Μία πρωτεΐνη αποτελείται από αμινοξέα (αμινοξικά κατάλοιπα), συνδεδεμένα σε μία γραμμική σειρά η οποία ονομάζεται αμινοξική ακολουθία ή αλληλουχία. Το κάθε αμινοξύ σε αυτήν αλληλουχία συνδέεται με το επόμενο με έναν ομοιοπολικό δεσμό και η καρβοξυλομάδα του ενός αμινοξέος ενώνεται με την αμινομάδα του επόμενου με την αποβολή ενός μορίου νερού (Εικόνα 1.1). Η δημιουργία μιας διαδοχής πεπτιδικών δεσμών, (Εικόνα 1.2) παράγει μια "κύρια αλυσίδα" ή "πρωτεϊνικό σκελετό" από την οποία προεξέχουν οι πλευρικές ομάδες των αμινοξέων. Πλευρικές ομάδες ή πλευρικές αλυσίδες καλούνται οι ομάδες εκείνες του μορίου που δεν συμμετέχουν στο σχηματισμό του πεπτιδικού δεσμού και που προσδίδουν σε κάθε αμινοξύ τις μοναδικές του ιδιότητες. Οι πλευρικές αλυσίδες των αμινοξέων μπορεί να είναι υδρόφιλες ή υδρόφοβες, πολικές ή μη πολικές, να έχουν αρνητικό ή θετικό φορτίο, να είναι δραστικές ή μη δραστικές κλπ.



Εικόνα 1.1: Σχηματική αναπαράσταση του τρόπου σύνδεσης των αμινοξέων και της δημιουργίας του πεπτιδικού δεσμού.

Πολλοί από τους ομοιοπολικούς δεσμούς σε μια μακριά αλυσίδα αμινοξέων επιτρέπουν στα άτομα που συνδέουν να περιστρέφονται ελεύθερα. Έτσι, θεωρητικά, ο πολυπεπτιδικός σκελετός μπορεί να διπλωθεί με αμέτρητους τρόπους. Κάθε διπλωμένη αλυσίδα όμως περιορίζεται από πολλά και διαφορετικά είδη ασθενών μη ομοιοπολικών δεσμών, που σχηματίζονται τόσο από τα άτομα του πολυπεπτιδικού σκελετού όσο και από άτομα των πλευρικών αλυσίδων των αμινοξέων. Ένα τέτοιο είδος ασθενούς δύναμης που παίζει πολύ σημαντικό ρόλο στον καθορισμό του σχήματος μιας πρωτεΐνης είναι η υδροφοβικότητα. Τα υδρόφοβα μόρια μεταξύ τους, και οι μη πολικές πλευρικές αλυσίδες συγκεκριμένων αμινοξέων, όταν βρεθούν σε υδατικό περιβάλλον συνωθούνται μαζί, έτσι ώστε να ελαχιστοποιηθεί η πιθανότητα διάσπασης τους στο δίκτυο λόγω της παρουσίας του νερού. Επομένως, ένας παράγοντας που κατευθύνει το δίπλωμα ή πτύχωση (folding) μιας πρωτεΐνης είναι η

κατανομή των πολικών και μη πολικών αμινοξέων της. Οι μη πολικές (υδρόφοβες) πλευρικές αλυσίδες μιας πρωτεΐνης επομένως, δηλαδή οι αλυσίδες που ανήκουν σε αμινοξέα όπως η φαινυλαλανίνη, η λευκίνη, η βαλίνη και η τρυπτοφάνη, τείνουν να συναθροίζονται στο εσωτερικό του μορίου. Με τον τρόπο αυτό αποφεύγουν την επαφή με το νερό που τις περιβάλλει στο εσωτερικό ενός κυττάρου. Αντίθετα, οι πολικές πλευρικές αλυσίδες, δηλαδή, οι αλυσίδες που ανήκουν σε αμινοξέα όπως η αργινίνη, η γλουταμίνη και η ιστιδίνη, τείνουν να διατάσσονται κοντά στην εξωτερική επιφάνεια της πρωτεΐνης, όπου μπορεί να σχηματίσουν δεσμούς υδρογόνου με τα μόρια του νερό αλλά και με άλλα πολικά μόρια. Κατά συνέπεια, κάθε αμινοξικό κατάλοιπο έχει προκαθορισμένες τάσεις να δημιουργεί δομές διαφορετικών τύπων, οι οποίες χαρακτηρίζουν μοναδικά την κάθε πρωτεΐνη.

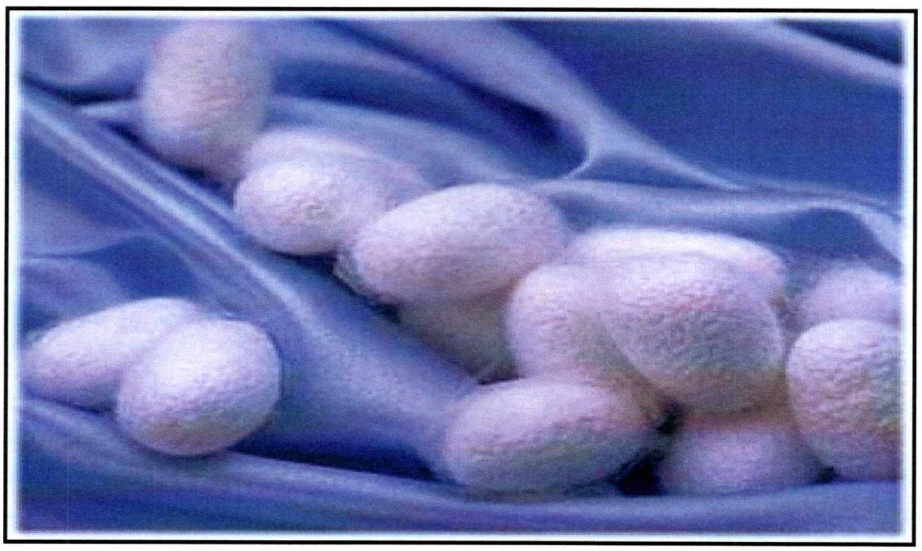


Εικόνα 1.2: Πολυπεπτιδική αλυσίδα αποτελούμενη από τέσσερα αμινοξέα.

1.2 Δομές πρωτεϊνών

Κάθε είδος πρωτεΐνης έχει μια ιδιαίτερη τρισδιάστατη δομή (πτυχωμένη δομή ή διαμόρφωση (conformation)), η οποία καθορίζεται όπως προαναφέραμε από τη σειρά των αμινοξέων στην αλυσίδα της. Το φαινόμενο αυτό της πτύχωσης των πρωτεϊνών έχει μελετηθεί εργαστηριακά με τη χρησιμοποίηση πολύ καθαρών πρωτεϊνών. Μια πρωτεΐνη μπορεί να ξεδιπλωθεί ή, όπως λέμε, να αποδιαταχθεί με την επίδραση ορισμένων διαλυτών που διασπούν τις μη ομοιοπολικές αλληλεπιδράσεις οι οποίες συγκρατούν τη δομή της διπλωμένης αλυσίδας. Η

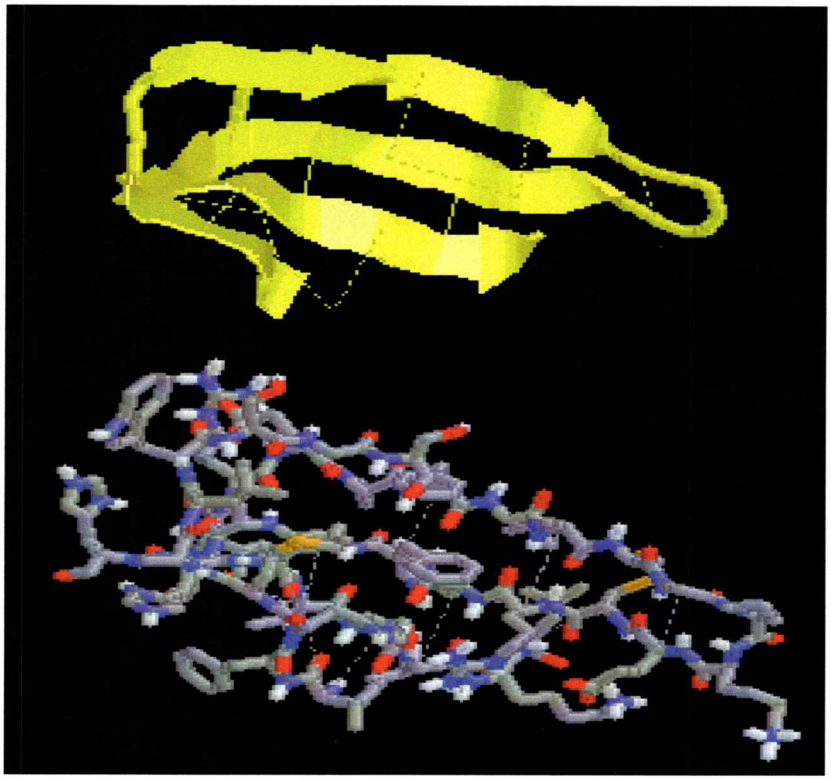
μετουσίωση (denaturation) μετατρέπει την πρωτεΐνη σε μια εύκαμπτη πολυπεπτιδική αλυσίδα που έχει χάσει το φυσικό της σχήμα. Όταν ο αποδιατακτικός παράγοντας απομακρυνθεί, η πρωτεΐνη συχνά αναδιπλώνεται αυθόρμητα ή όπως λέμε αναδιατάσσεται και ανακτά την αρχική της διαμόρφωση. Το γεγονός αυτό αποδεικνύει ότι όλες οι πληροφορίες που είναι απαραίτητες για τον καθορισμό του τρισδιάστατου σχήματος μιας πρωτεΐνης περιέχονται στην αλληλουχία των αμινοξέων της. Άρα μπορούν να αναπτυχθούν υπολογιστικές τεχνικές, που επιτρέπουν προβλέψεις, βασισμένες στην πρωτεϊνική ακολουθία αυτή καθ' αυτή.



Εικόνα 1.3: Φωτογραφία από μετάξι βασικό συστατικό του οποίου αποτελεί η πρωτεΐνη ινιδοίνη.

Οι πρωτεΐνες εμφανίζουν μεγάλη ποικιλομορφία περίπλοκων στερεοδομών. Η α-έλικα και οι β-πτυχωτές επιφάνειες είναι από τα πιο κοινά πρότυπα πτύχωσης. Τα δύο αυτά πρότυπα ανακαλύφθηκαν περίπου πριν από 50 χρόνια μελετώντας τη δομή των τριχών και του μεταξιού. Το πρώτο πρότυπο πτύχωσης που ανακαλύφθηκε, γνωστό ως α-έλικα (a-helix), βρέθηκε στην α-κερατίνη η οποία υπήρχε σε αφθονία στο δέρμα και τα παράγωγα του (τρίχες, νύχια και κέρατα). Μέσα σε ένα χρόνο από την ανακάλυψη της α-έλικας, μια δεύτερη διπλωμένη δομή, γνωστή ως β-πτυχωτή επιφάνεια (β sheet), ανακαλύφθηκε στην πρωτεΐνη ινιδοίνη (fibroin), το κύριο

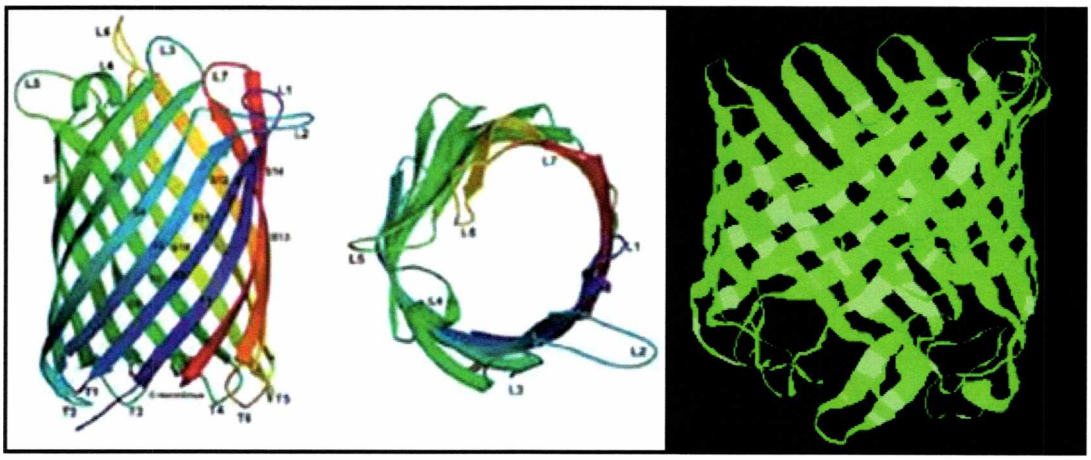
συστατικό του μεταξιού (Εικόνα 1.3). Οι πρωτεΐνες χωρίζονται σε διάφορες κατηγορίες από τις οποίες οι σημαντικότερες είναι οι διαμεμβρανικές, οι υδατοδιαλυτές και οι ινώδεις πρωτεΐνες. Το μεγαλύτερο ποσοστό των διαμεμβρανικών πρωτεϊνών αποτελούν πρωτεΐνες με διαμόρφωση α-έλικας και β-βαρελιού. Συγκεκριμένα οι α-ελικοειδείς διαμεμβρανικές πρωτεΐνες, εμφανίζονται σε μεγάλη αφθονία σε όλες σχεδόν τις κυτταρικές μεμβράνες (von Heijne, 1999), ενώ οι διαμεμβρανικές πρωτεΐνες με μορφή β-βαρελιού έχουν παρατηρηθεί έως τώρα πειραματικά μόνο στην εξωτερική μεμβράνη των αρνητικών κατά Gram βακτηρίων (Schulz, 2003).



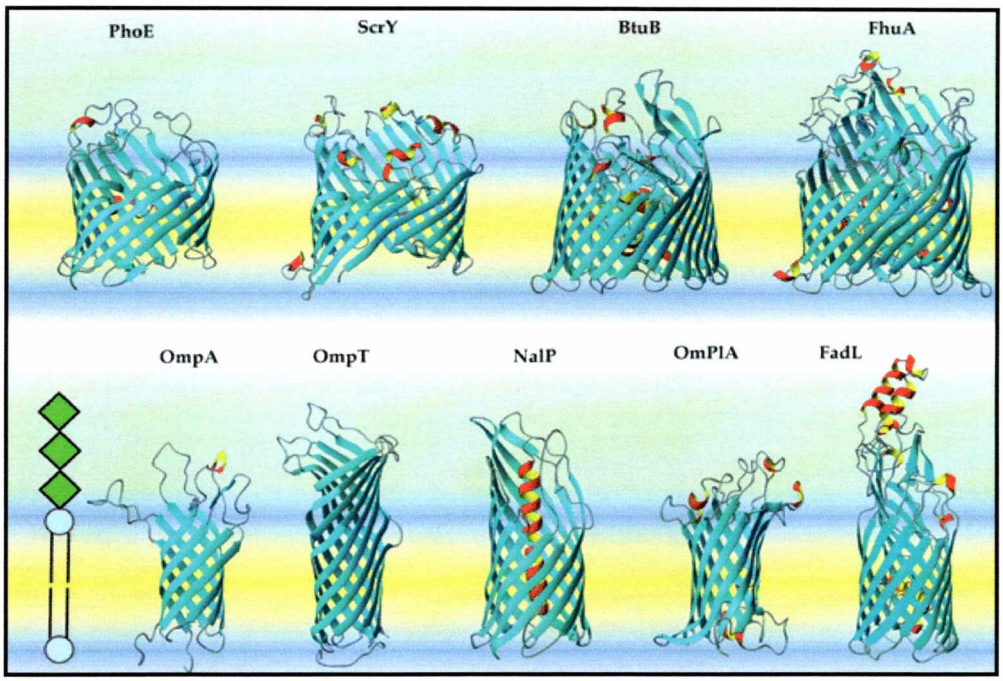
Εικόνα 1.4: Β-πτυχωτή επιφάνεια που σχηματίζεται από παρακείμενες πολυπεπτιδικές αλυσίδες που έχουν αντίθετο προσανατολισμό

Οι β-πτυχωτές επιφάνειες μπορεί να σχηματιστούν είτε από παρακείμενες πολυπεπτιδικές αλυσίδες που έχουν τον ίδιο προσανατολισμό (δηλαδή, από παράλληλες αλυσίδες), είτε από μια αλυσίδα που κάμπτεται αλλάζοντας κατεύθυνση, με συνεπεία το κάθε τμήμα της να έχει αντίθετο προσανατολισμό από τα αμέσως

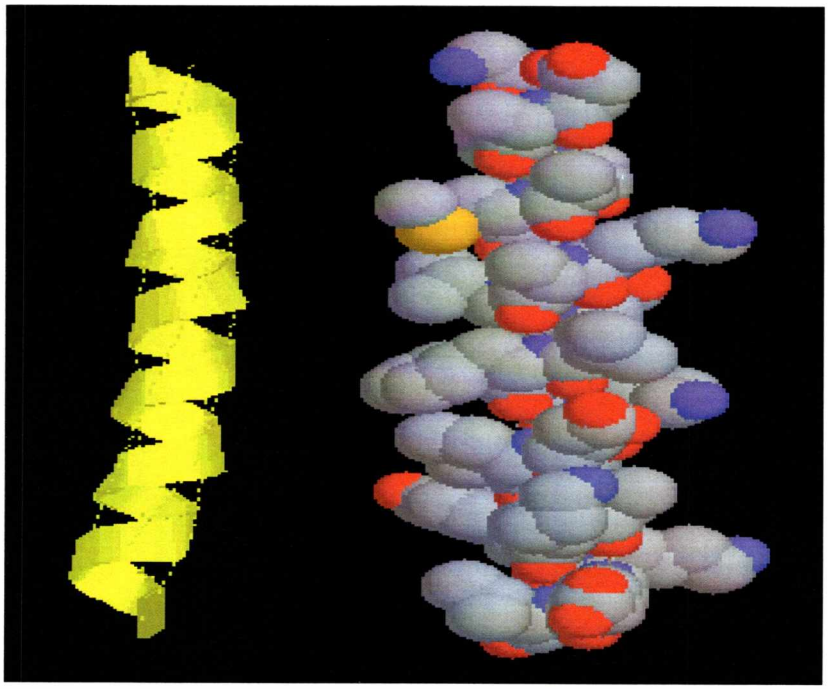
γειτονικά τμήματα (δηλαδή, από αντιπαράλληλες αλυσίδες) (Εικόνα 1.4). Και τα δυο είδη β-πτυχωτών επιφανειών σχηματίζουν μια πολύ άκαμπτη δομή, που συγκρατείται με δεσμούς υδρογόνου οι οποίοι διασυνδέουν τους πεπτιδικούς δεσμούς των παρακείμενων αλυσίδων (Εικόνα 1.5).



Εικόνα 1.5: Διαμεμβρανική πρωτεΐνη με δομή β-βαρελιού που σχηματίζεται από β-πτυχωτές επιφάνειες.



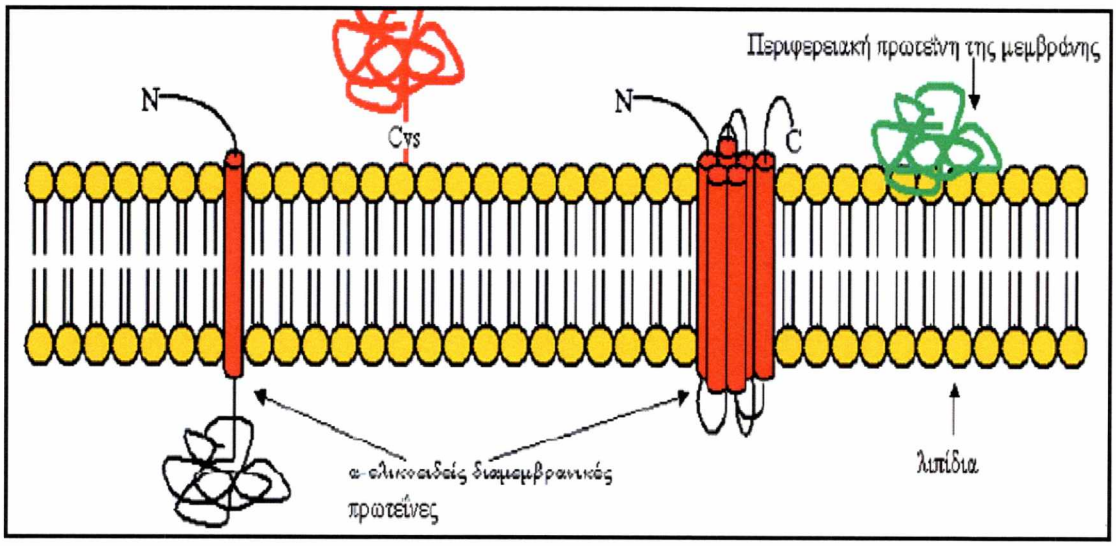
Εικόνα 1.6: Γνωστές διαμεμβρανικές πρωτεΐνες με δομή β-βαρελιού



Εικόνα 1.7: Πρωτεΐνη με διαμόρφωση α-έλικας

Μερικές διαμεμβρανικές πρωτεΐνες διαπερνούν τη μεμβράνη με τη διαμόρφωση της β-πτυχωτής επιφάνειας που καμπυλώνεται σε μια ανοιχτή κυλινδρική δομή σχηματίζοντας ένα απύθμενο β-βαρέλι (β-barrel), (Εικόνα 1.6). Όπως αναμένεται, οι πλευρικές αλυσίδες των αμινοξέων που βρίσκονται στο εσωτερικό του κυλίνδρου, και έρχονται σ' επαφή με τον υδρόφιλο πόρο, είναι κυρίως υδρόφιλες, ενώ αυτές που βρίσκονται στο εξωτερικό του κυλίνδρου και εφάπτονται με τα υδρόφοβα λιπίδια, είναι αποκλειστικά υδρόφοβες. Οπότε αν πάρουμε την πρωτεϊνική αλληλουχία μιας πρωτεΐνης που έχει διαμόρφωση β-βαρελιού αυτό που θα διαπιστώσουμε είναι τμήματα αμινοξέων, μήκους 7 με 11 αμινοξέα, στα οποία εμφανίζεται εναλλαγή υδρόφιλων και υδρόφοβων αμινοξέων.

Μια α-έλικα δημιουργείται όταν μια πολυπεπτιδική αλυσίδα περιστρέφεται γύρω από τον εαυτό της για να σχηματίσει έναν άκαμπτο κύλινδρο (Εικόνα 1.7). Στη διαμόρφωση αυτή, ένας δεσμός υδρογόνου σχηματίζεται ανά τέταρτο πεπτιδικό δεσμό, συνδέοντας το καρβονύλιο (C=O) ενός πεπτιδικού δεσμού με την αμινομάδα (N-H) ενός άλλου πεπτιδικού δεσμού. Αυτό οδηγεί σε μια κανονική έλικα η οποία πραγματοποιεί μια πλήρη περιστροφή κάθε 3-6 κατάλοιπα αμινοξέων.

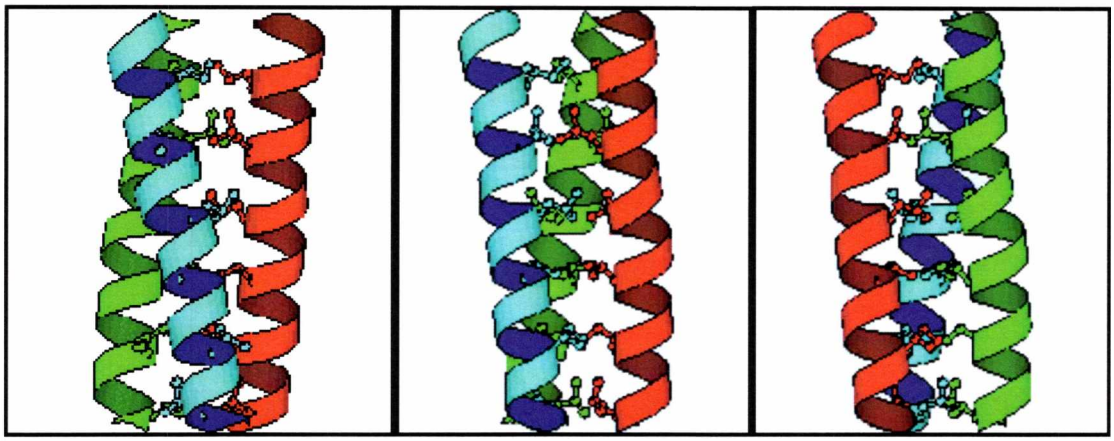


Εικόνα 1.8: Σχηματική αναπαράσταση της εξωτερικής επιφάνειας ενός αρνητικού κατά Gram βακτηρίου. Διακρίνουμε τα δυο διαφορετικά είδη α-ελικοειδών διαμεμβρανικών πρωτεϊνών που απαντώνται στην μεμβράνη του.

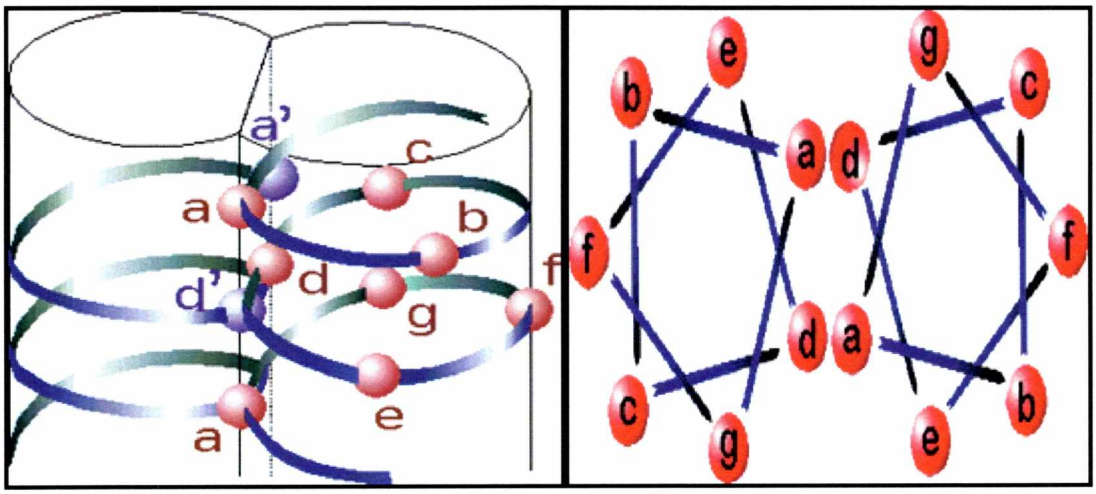
Βραχείες περιοχές με διαμόρφωση α-έλικας αφθονούν ιδιαίτερα σε πρωτεΐνες που βρίσκονται στην κυτταρική μεμβράνη, όπως οι πρωτεΐνες μεταφοράς και οι υποδοχείς (Εικόνα 1.8). Οι περιοχές μιας διαμεμβρανικής πρωτεΐνης που διαπερνούν τη διπλοστοιβάδα των λιπιδίων όταν έχουν διαμόρφωση α-έλικας αποτελούνται κυρίως από αμινοξέα με μη πολικές αλυσίδες. Στην α-έλικα, δεσμοί υδρογόνου δημιουργούνται μεταξύ ομάδων του υδρόφιλου πολυπεπτιδικού σκελετού, ο οποίος προφυλάσσεται από τον υδρόφοβο λιπιδικό περιβάλλον της μεμβράνης από τις προεξέχουσες μη πολικές πλευρικές αλυσίδες του. Όσες λοιπόν από τις διαμεμβρανικές πρωτεΐνες δεν διαπερνούν την μεμβράνη με διαμόρφωση β-βαρελιού τη διαπερνούν με διαμόρφωση α-έλικας. Σε αυτές τις α-έλικες, οι υδρόφοβες πλευρικές αλυσίδες των αμινοξέων είναι εκτεθειμένες στο εξωτερικό της έλικας όπου βρίσκονται σ' επαφή με τις υδρόφοβες λιπιδικές ουρές των λιπιδίων της μεμβράνης, ενώ τμήματα του πολυπεπτιδικού σκελετού σχηματίζουν δεσμούς υδρογόνου μεταξύ τους στο εσωτερικό της έλικας.

Ορισμένες φορές, δύο α-έλικες περιελίσσονται η μια γύρω από την άλλη για να σχηματίσουν μια πολύ σταθερή δομή, η οποία είναι γνωστή ως σπειροειδές σπείραμα (coiled-coil) (Εικόνα 1.9). Η δομή αυτή σχηματίζεται όταν οι δύο α-έλικες έχουν τις περισσότερες από τις μη πολικές (υδρόφοβες) πλευρικές αλυσίδες τους στη

μια πλευρά, έτσι ώστε να μπορεί να περιστραφούν η μια γύρω από την άλλη με τις πλευρικές αλυσίδες να κατευθύνονται προς τα μέσα. Συγκεκριμένα κάθε κλώνος μιας πρωτεϊνης με διαμμόρφωση σπειροειδούς σπειράματος μπορεί να παρασταθεί ως επαναλαμβανόμενα ολιγομερή της μορφής (a-b-c-d-e-f-g)n, όπου a, b, c,...,g είναι οι επτά διαφορετικές δομικές θέσεις στο σπειροειδές σπείραμα (Εικόνα 1.10). Υπάρχει δηλαδή επανάληψη ανα 7 κατάλοιπα. Μακριά, ραβδόσχημα σπειροειδή σπειράματα σχηματίζουν το δομικό σκελετό πολλών επιμηκών πρωτεϊνών, κυρίως ινωδών.



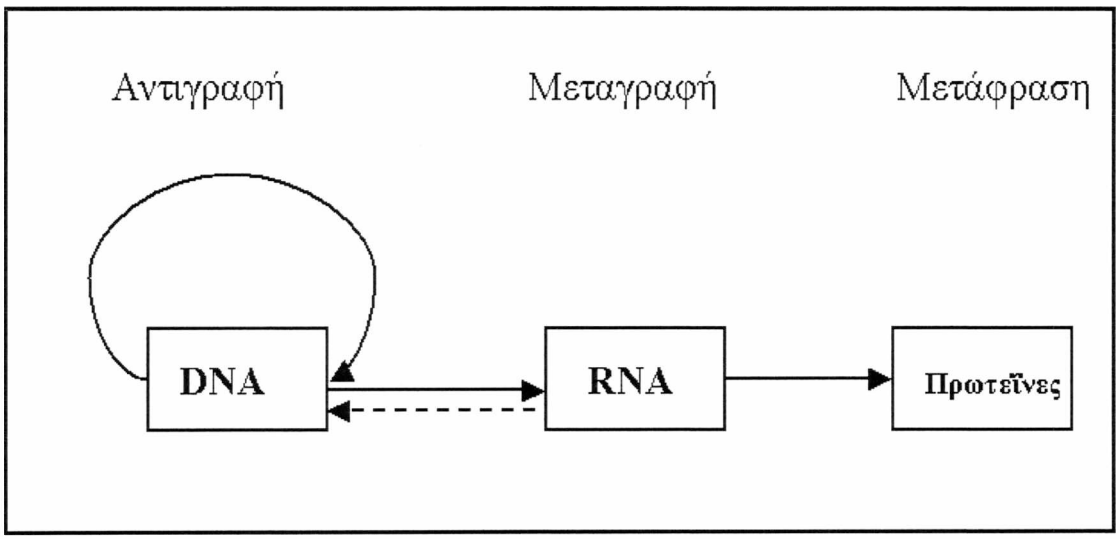
Εικόνα 1.9: Τρισδιάστατη αναπαράσταση πρωτεΐνης με διαμόρφωση σπειροειδούς σπειράματος



Εικόνα 1.10: Αναπαράσταση της δομή μιας πρωτεΐνης με διαμόρφωση σπειροειδούς σπειράματος, όπου παρουσιάζονται οι επτά δομικές θεσεις του σπειροειδούς σπειράματος.

1.3 Πρόγνωση πρωτεϊνών

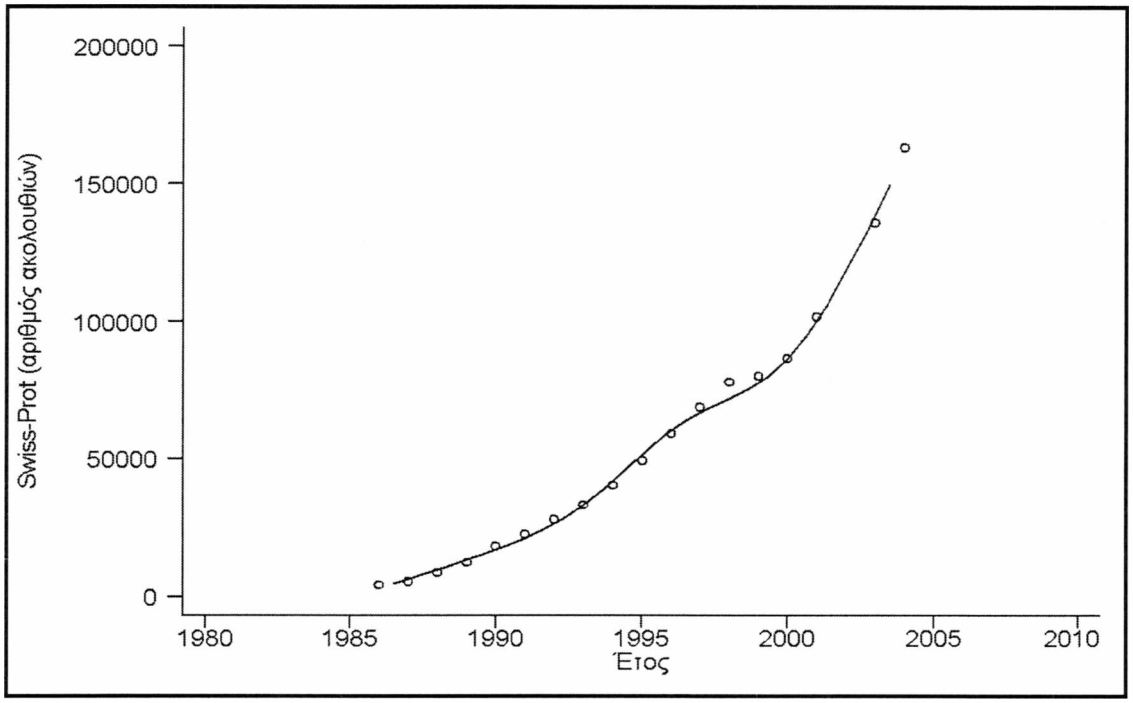
Οι πρωτεΐνες διαδραματίζουν σημαντικούς και ποικίλους ρόλους σχεδόν σε όλες τις δραστηριότητες που συντελούνται μέσα σε ένα ζωντανό κύτταρο. Σύμφωνα με το κεντρικό δόγμα της μοριακής βιολογίας (Crick, 1958; Crick et al., 1961) (Εικόνα 1.11) οι πρωτεΐνες συντίθενται μονοσήμαντα, με βάση τη γενετική πληροφορία που εμπεριέχεται στο DNA, και η οποία μεταβιβάζεται στην πρωτεϊνοσυνθετική μηχανή των ριβοσωμάτων, μέσω του RNA. Η αμινοξική αλληλουχία λοιπόν καθορίζεται από το DNA του κυττάρου στα γονίδια. Η αλληλουχία αυτή των αμινοξέων αναδιπλώνεται στο χώρο δημιουργώντας τρισδιάστατες δομές δημιουργώντας έτσι τη λειτουργική πρωτεΐνη. Οι περισσότερες πρωτεΐνες επιτελούν τη λειτουργία ή τις λειτουργίες τους, με το συνδυασμό της εξειδικευμένης διαμόρφωσης τους στο χώρο και των φυσικοχημικών ιδιοτήτων των αμινοξέων τους.



Εικόνα 1.11: Το κεντρικό δόγμα της μοριακής βιολογίας. Τα συνεχή βέλη δείχνουν τη ροή της γενετικής πληροφορίας, ενώ το στικτό βέλος απεικονίζει την ειδική περίπτωση παραγωγής DNA από RNA που απαντάται στους ρετροϊούς.

Οι πρωτεϊνικές ακολουθίες μπορούν να καθοριστούν εύκολα σήμερα, είτε χρησιμοποιώντας άμεσα το πρωτεϊνικό μόριο, είτε έμμεσα από την αλληλουχία των γονιδίων τους. Εντούτοις, η τρισδιάστατη δομή μιας πρωτεΐνης είναι δύσκολο να παρατηρηθεί πειραματικά, με την υπάρχουσα τεχνολογία, έχοντας μόνο την

αμινοξική της ακολουθία ως δεδομένο. Αυτό αποδεικνύεται και από το γεγονός ότι μέχρι σήμερα οι τρισδιάστατες δομές μόνο ενός μικρού μέρους των γνωστών πρωτεϊνών έχουν πλήρως προσδιοριστεί σε ατομική διακριτικότητα. Και χωρίς να διαθέτουμε ένα τρισδιάστατο μοντέλο για τη δομή της πρωτεΐνης, είναι δύσκολο να βρούμε το λειτουργικό ρόλο της, που είναι και ο απώτερος σκοπός μας.



Εικόνα 1.12: Η εκθετική αύξηση των πρωτεϊνικών ακολουθιών οι οποίες είναι κατατεθειμένες στην Swiss-Prot, από το 1986 έως το τέλος του 2004.

Διαφορετικές πρωτεϊνικές ακολουθίες από έναν οργανισμό, αλλά και από διαφορετικούς οργανισμούς παρουσιάζουν αξιοπρόσεκτες ομοιότητες. Πιο συγκεκριμένα, οι μέχρι τώρα γνωστές πρωτεΐνες μπορούν να οργανωθούν κατά τρόπο ιεραρχικό, με βάση τις αμινοξικές, δομικές και λειτουργικές συντηρήσεις που παρουσιάζουν. Πληθώρα νέων και υποθετικών γονιδίων ανακαλύπτονται καθημερινά. Μέχρι σήμερα είναι γνωστές μερικά εκατομμύρια πιθανών πρωτεϊνικών ακολουθιών, οπότε η ανάλυση με το χέρι αυτού του τόσο μεγάλου αριθμού πρωτεϊνών, που αυξάνει με εκθετικούς ρυθμούς, είναι πρακτικά αδύνατη (Εικόνα

1.12). Για το λόγο αυτό η ανάπτυξη αλγορίθμων για των προσδιορισμό των πρωτεϊνών αυτών κρίνεται τόσο σημαντική.

Οι Αλυσίδες Markov (Markov Chains) (Sheynin, 1988), είναι πιθανοθεωρητικά (στοχαστικά) μοντέλα, με τα οποία περιγράφουμε και αναλύουμε τις ακολουθίες βιολογικών πολυμερών όπως του DNA και των πρωτεϊνών. Πρέπει εδώ να τονιστεί ότι το μοντέλο Markov θεωρείται από πολλούς ερευνητές ως το πιο κατάλληλο για να περιγράψει αλληλουχίες μεγαλομορίων όπως του DNA και των πρωτεϊνών.

Οι αλυσίδες Markov (MCs) λοιπόν παρέχουν ένα καλό στατιστικό πλαίσιο για τη λύση ενός μεγάλου φάσματος προβλημάτων. Η μελέτη των αλυσίδων Markov (MCs) άρχισε το 20^{ο} αιώνα από τον Markov, ο οποίος έθεσε τα θεμέλια για τη θεωρία των πιθανολογικών διαδικασιών. Μόνο προς το τέλος της δεκαετίας του '80 και στις αρχές της δεκαετίας του '90 όμως, οι αλυσίδες Markov (MCs) άρχισαν να χρησιμοποιούνται στη Μοριακή Βιολογία για υπολογιστική ανάλυση ακολουθιών και τον προσδιορισμό των πρωτεϊνικών δομών. Έκτοτε έχουν δημιουργηθεί πολλά μοντέλα που στηρίζονται στα MCs, όπως τα: HMMs (Hidden Markov Models), MTD (Mixture Transition Distribution), IMMs (Interpolated Markov Models) και τα VLMCs (Variable Length Markov Chains).

1.4 Στόχοι της διπλωματικής εργασίας

Στόχος της παρούσας διπλωματικής εργασίας είναι να παρουσιάσει μια επισκόπηση των κυριότερων μοντέλων που στηρίζονται στη χρήση των αλυσίδων Markov και χρησιμοποιούνται για την πρόγνωση πρωτεϊνών και να εφαρμόσει ένα από αυτά τα μοντέλα για πρόγνωση διαμεμβρανικών πρωτεϊνών με διαμόρφωση β-βαρελιών (β-barrel) και ινωδών πρωτεϊνών με διαμόρφωση σπειροειδών σπειραμάτων (coiled-coils) .

2. ΜΑΡΚΟΒΙΑΝΑ ΜΟΝΤΕΛΑ

2.1 Αλυσίδες Markov

Μια αλυσίδα Markov (Sheynin, 1988) είναι μια πιθανολογική (στοχαστική) διαδικασία, σύμφωνα με την οποία η πιθανότητα εμφάνισης μιας μεταβλητής σε μια μελλοντική κατάσταση εξαρτάται από την προηγούμενη κατάσταση. Η θεώρηση μιας ακολουθίας ενδεχομένων ως αλυσίδα Markov, δηλαδή, στηρίζεται στην ιδέα ότι κάθε ένα από τα ενδεχόμενα εξαρτάται μόνο από το αμέσως προηγούμενο του, ή αλλιώς το κάθε ενδεχόμενο καθορίζει με κάποια πιθανότητα το αμέσως επόμενό του. Αν αυτή η εξάρτηση επεκταθεί και σε 2,3,...,n προηγούμενα ενδεχόμενα τότε μιλάμε για αλυσίδες Markov $2^{ης}$,$3^{ης}$, ...,$n^{ης}$ τάξης. Με άλλα λόγια, η περιγραφή της παρούσας κατάστασης παρέχει πλήρως όλες τις πληροφορίες που θα μπορούσαν να επηρεάσουν τη μελλοντική εξέλιξη της διαδικασίας. Κάθε μοντέλο Markov (Markov Model – MM) συνδέεται με μια πρώτης τάξεως αλυσίδα Markov (Markov Chain – MC) με κατάλληλες πιθανότητες μετάβασης μεταξύ των καταστάσεων και με μια αρχική κατανομή.

Ήδη από τη δεκαετία του 70 τα μοντέλα αυτά χρησιμοποιούνται για την αναγνώριση και επεξεργασία εικόνας, ήχου κ.α. Αυτό οφείλεται στο γεγονός ότι σε οποιοδήποτε κωδικοποιημένο σύστημα επικοινωνίας, όπως στις φυσικές γλώσσες, υπάρχει μια εσωτερική δομή η οποία καθορίζει κάποιο είδος εξάρτησης των συμβόλων. Για παράδειγμα, στην αγγλική γλώσσα το γράμμα Q ακολουθείται σχεδόν πάντοτε από το U, άρα η πιθανότητα να εμφανιστεί το U σε μια θέση δεν είναι πάντα ίδια αλλά εξαρτάται από το αν προηγήθηκε το Q. Για την ακρίβεια ο ίδιος ο Ρώσος Μαθηματικός Andrey Markov (1856-1922) οδηγήθηκε στη σύλληψη της έννοιας των ομώνυμων αλυσίδων μελετώντας τις εναλλαγές φωνηέντων και συμφώνων σε κάποιο ποίημα του Pushkin. Έστω ότι έχουμε μια αλυσίδα π.χ. DNA

```
ATTGTAATCTCACGGTGTACGCGCATGCACAGTCAGT
```

ή μια αμινοξική αλληλουχία

```
AEDGPRGSDADKLIAVCLIGFVLIFVSLVCVTYTRED
```

Ως καταστάσεις εδώ ορίζονται τα σύμβολα της ακολουθίας τα οποία ανήκουν σε ένα πεπερασμένο αλφάβητο (τα τέσσερα νουκλεοτίδια στην περίπτωση του DNA ή τα 20 αμινοξέα στην περίπτωση των πρωτεϊνών), από το οποίο η αλυσίδα Markov παίρνει τιμές σε ένα «χώρο καταστάσεων» οριζόμενο από το συγκεκριμένο αλφάβητο. Αν θεωρήσουμε μια πρωτεϊνική ακολουθία μήκους L καταλοίπων, και την ονομάσουμε $\boldsymbol{x}$, έτσι ώστε:

$$\boldsymbol{x} = x_1, x_2, \ldots, x_{L-1}, x_L$$

και θεωρήσουμε ότι τα σύμβολα (νουκλεοτίδια ή αμινοξέα) δεν είναι ανεξάρτητα μεταξύ τους, αλλά το ποιο θα ακολουθήσει εξαρτάται μόνο από το αμέσως προηγούμενο του, τότε η πιθανότητα να εμφανιστεί π.χ. κάποιο b, δεδομένου ότι το αμέσως προηγούμενο του είναι α, (πιθανότητα μεταβάσεως - transition probability) θα είναι:

$$p_{ab} = P(x_1 = b \mid x_{i-1} = a)$$

και η συνολική πιθανότητα να παρατηρηθεί η δεδομένη αλληλουχία, για τις παραπάνω ακολουθίες, θα είναι:

$$P(\boldsymbol{x}) = P(x_X \mid x_{n-1})P(x_{n-1} \mid x_{n-2})\ldots P(x_1) = P(x_1)\prod_{i=2}^{n} P(x_i \mid x_{i-1})$$

Το μοντέλο αυτό, με επεκτάσεις του σε εξάρτηση, πέραν της πρώτης τάξης, αποτελεί τη βάση όλων των μοντέλων που έχουν δημιουργηθεί και στηρίζονται στις αλυσίδες Markov, που χρησιμοποιούνται ευρέως για πρόγνωση πρωτεϊνών.

Στην περίπτωση των διαμεμβρανικών για παράδειγμα τμημάτων μιας πρωτεΐνης η πολύ απλή ερμηνεία που έχει ένα ΜΜ, είναι η εξής: είναι γνωστό ότι τα διαμεμβρανικά τμήματα απαρτίζονται κυρίως από υδρόφοβα αμινοξέα, άρα η πιθανότητα να εμφανιστεί π.χ. ισολευκίνη σε ένα διαμεμβρανικό τμήμα είναι μεγαλύτερη από ότι σε ένα μη διαμεμβρανικό. Επίσης στα διαμεμβρανικά τμήματα είναι πιο πιθανό όταν έχει προηγηθεί ένα αμινοξύ που ανήκει σε ένα τέτοιο τμήμα, το επόμενο του να είναι επίσης μέρος της διαμεμβρανικής περιοχής.

Ας δούμε όμως πιο αναλυτικά τι συμβαίνει γενικά στην περίπτωση των βιολογικών ακολουθιών. Αν θεωρήσουμε την κατανομή των αμινοξέων σε κάθε

θέση i κατά μήκος της αλληλουχίας ως τυχαία μεταβλητή, τότε μπορούμε να ορίσουμε την αλυσίδα Markov, ως μια στοχαστική διαδικασία που διαθέτει τη λεγόμενη «Μαρκοβιανή Ιδιότητα», η οποία ορίζει (σε διακριτό χρόνο) ότι η δεσμευμένη κατανομή των «μελλοντικών» παρατηρήσεων $x_{i+1}, x_{i+2}, x_{i+3}, \ldots$ δεδομένου του «παρελθόντος» $x_1, x_2, \ldots, x_{i-1}, x_i$ εξαρτάται από το παρελθόν μόνο μέσω του x_i. Αυτό τυπικά διατυπώνεται ως εξής:

$$P(x_i \mid x_{i-1}, \ldots, x_1) = P(x_i \mid x_{i-1}) \tag{2.1}$$

Μια συγκεκριμένη Αλυσίδα Markov χαρακτηρίζεται από τον πίνακα των «πιθανοτήτων μετάβασης» (transition probabilities). Τα στοιχεία αυτού του πίνακα, δίνονται από την παρακάτω σχέση:

$$a_{st} = P(x_i = t \mid x_{i-1} = s) = a_{x_{i-1} x_i} \tag{2.2}$$

η οποία δηλώνει, την πιθανότητα το κατάλοιπο t να εμφανιστεί στη θέση i της ακολουθίας, δεδομένου ότι το προηγούμενο κατάλοιπο *(i-1)* είναι s. Αν αναλογιστούμε ότι μπορούμε να γενικεύσουμε την εξάρτηση στα n προηγούμενα κατάλοιπα, είναι φυσικό η δεδομένη αλυσίδα να ονομάζεται Αλυσίδα Markov 1ης τάξεως. Η συνολική πιθανότητα μιας ακολουθίας υπολογίζεται ως εξής:

$$P(\boldsymbol{x}) = P(x_1, x_2, \ldots, x_{L-1}, x_L) = P(x_L \mid x_{L-1}, \ldots, x_1) P(x_{L-1} \mid x_{L-2}, \ldots, x_1) P(x_1)$$

και από τη σχέση (2.1) , έχουμε :

$$\begin{aligned} P(\boldsymbol{x}) &= P(x_L \mid x_{L-1}) P(x_{L-1} \mid x_{L-2}) \ldots P(x_2 \mid x_1) P(x_1) \\ &= P(x_1) \prod_{i=2}^{L} P(x_i \mid x_{i-1}) = P(x_1) \prod_{i=2}^{L} a_{x_{i-1} x_i} \end{aligned} \tag{2.3}$$

όπου $P(x_1)$ είναι η πιθανότητα για την εμφάνιση του πρώτου συμβόλου. Σύμφωνα με τον ορισμό αυτό, βλέπουμε ότι οι πιθανότητες μεταβάσεως είναι ίδιες, ανεξαρτήτως της θέσης τους στην αλυσίδα δηλαδή :

$$p_{ab}(n-1, n) = P(x_i = b \mid x_{i-1} = a) = p_{ab} \quad \text{για κάθε } n=1,2,\ldots,L$$

Η αλυσίδα αυτή λέμε ότι έχει στάσιμες πιθανότητες μεταβάσεως, ή ισοδύναμα ότι η αλυσίδα αυτή είναι ομογενής χρονικά. Ο περιορισμός αυτός χρησιμοποιείται σχεδόν πάντοτε στις περιπτώσεις μακρομορίων, κυρίως γιατί προσφέρει υπολογιστική απλότητα αλλά και γιατί δεν έχουμε, στις περισσότερες περιπτώσεις,

καμία ένδειξη που να υποστηρίζει μια τέτοια εξάρτηση από τη θέση στην αλυσίδα. Ο πίνακας ο οποίος περιέχει τις πιθανότητες μεταβάσεως, όπως είδαμε, λέγεται πίνακας πιθανοτήτων μεταβάσεως ή πίνακας μεταβάσεως 1ης τάξης και πρέπει για ένα αλφάβητο με πλήθος k να ικανοποιεί τα παρακάτω:

$$p_{a,b} \geq 0 \quad \text{για} \quad a,b = 1,2,...,k$$

και

$$\sum_{b=1}^{k} p_{a,b} = 1 \quad \text{για κάθε } b=1,2,...,k$$

Γενικότερα κάθε τετραγωνικός πίνακας που ικανοποιεί τις δυο αυτές σχέσεις, λέγεται στοχαστικός. Όπως είδαμε, από τις παραπάνω σχέσεις ορίζεται πλήρως μια αλυσίδα Markov, αρκεί να ορίσουμε επιπλέον μια πιθανότητα για την κατάσταση έναρξης της αλυσίδας (B=Begin). Η πιθανότητα αυτή ονομάζεται αρχική πιθανότητα και ορίζεται ως:

$$P(x_1 = a) = p_{Ba} \tag{2.4}$$

Όμοια μπορούμε να ορίσουμε (χωρίς όμως και να είναι απαραίτητο) μια άλλη τελική κατάσταση (E=End) για τον τερματισμό της αλυσίδας με πιθανότητα :

$$P(E \mid x_n = b) = p_{bE} \tag{2.5}$$

Έτσι πλέον μια πλήρης σχηματική αναπαράσταση του μοντέλου Markov φαίνεται στην Εικόνα 2.1 παρακάτω. Παραδοσιακά η λήξη της αλληλουχίας δεν συμπεριλαμβάνεται στο μοντέλο, θεωρούμε δηλαδή ότι η αλυσίδα μπορεί να τελειώνει οπουδήποτε. Το πλεονέκτημα του να συμπεριληφθεί αυτή η κατάσταση στο μοντέλο, είναι όταν θέλουμε να μελετήσουμε την κατανομή του μήκους της αλυσίδας. Έτσι αν

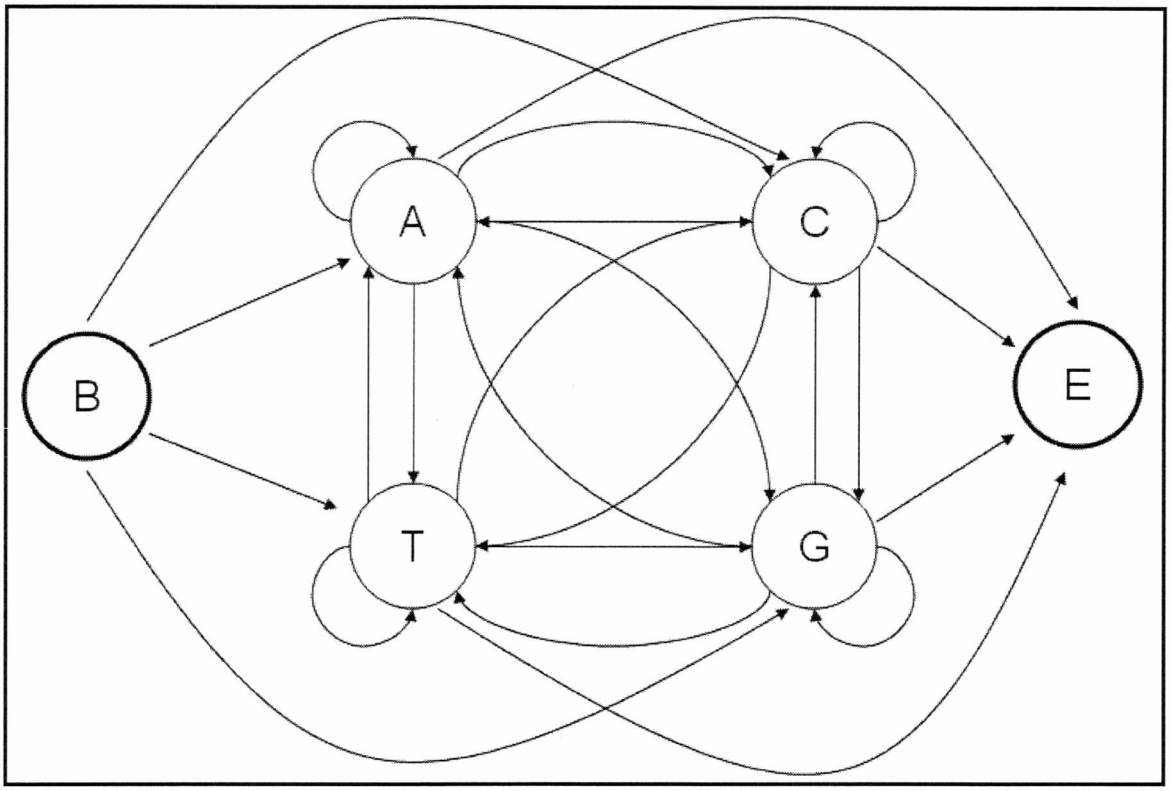
$$P(E \mid x_n = b) = p_{bE} = q$$

τότε η κατανομή του αθροίσματος των πιθανοτήτων της σχέσης (2.3) για μια ακολουθία μήκους L είναι :

$$p_{o\lambda} = q(1-q)^{L-1} \tag{2.6}$$

δηλαδή η κατανομή του αθροίσματος των πιθανοτήτων για όλες τις ακολουθίες μήκους L ακολουθεί γεωμετρική κατανομή. Αντίστοιχα το άθροισμα των πιθανοτήτων όλων των πιθανών ακολουθιών είναι (Durbin et al., 1998):

$$p_{o\lambda} = \sum_{\{x\}} P(x) = \sum \sum \ldots \sum P(x_1) \prod_{i=2}^{n} P(x_i \mid x_{i-1}) = 1 \qquad (2.7)$$



Εικόνα 2.1: Ένα τυπικό μοντέλο αλυσίδας Markov, με καταστάσεις τις 4 βάσεις του DNA. Τα βέλη συμβολίζουν τις επιτρεπτές μεταβάσεις. Β και Ε, οι καταστάσεις έναρξης και τερματισμού αντίστοιχα.

2.2 Αλυσίδες ανώτερης τάξεως

Μια $n^{\eta\varsigma}$ τάξεως αλυσίδα Markov, μπορεί να προκύψει αυτόματα από γενίκευση της Μαρκοβιανής ιδιότητας της εξισώσεως (2.1). Συγκεκριμένα, η σχέση αυτή τροποποιείται έτσι ώστε να συμπεριλάβει εξάρτηση από n προηγούμενες παρατηρήσεις:

$$P(x_i \mid x_{i-1}, \ldots, x_1) = P(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_{i-n}) = a_{x_n \ldots x_{i-1} x_i} \qquad (2.8)$$

Η εξίσωση (2.3) δείχνει ότι ένα σύμβολο σε μια ακολουθία εξαρτάται από τους προκατόχους του, δηλ. από το παρατηρηθέν πλαίσιο συμβόλων μέχρι εκείνη τη στιγμή. Σε ένα μοντέλο με σταθερό μέγεθος, για ένα τέτοιο πλαίσιο αυθαίρετου μήκους δεν μπορούμε βεβαίως να υπολογίσουμε την πιθανότητα $P(\boldsymbol{x})$. Ένας υπολογισμός της πιθανότητας $P(\boldsymbol{x})$ γίνεται επομένως με τον περιορισμό του μήκους πλαισίου σε n, το οποίο είναι η βασική ιδέα στην οποία στηρίζεται ένα $n^{ης}$ τάξεως Μαρκοβιανό μοντέλο:

$$P(\boldsymbol{x}) = P(x_1 \ldots x_{n-1}, x_n) \prod_{i=n+1}^{L} P(x_i \mid x_{i-n} \ldots x_{i-1}) \tag{2.9}$$

Λογαριθμίζοντας τον παραπάνω τύπο προκύπτει:

$$\log P(\boldsymbol{x}) = \log P(x_1, \ldots, x_{n-1}, x_n) + \log \prod_{i=n+1}^{L} P(x_i \mid x_{i-n}, \ldots, x_{i-1})$$

$$= \log P(x_1, \ldots, x_{n-1}, x_n) + \sum_{i=n+1}^{L} \log P(x_i \mid x_{i-n}, \ldots, x_{i-1})$$

Ο πρώτος όρος μπορεί να απορριφθεί από τους υπολογισμούς δεδομένου ότι μπορεί να θεωρηθεί ως την ειδική κατάσταση "begin (αρχή)" της διαδικασίας. Κατά συνέπεια, η πιθανότητα μπορεί να εκφραστεί ως:

$$\log P(\boldsymbol{x}) = \sum_{i=n+1}^{L} \log P(x_i \mid x_{i-1}, x_{i-2} \ldots, x_{i-n}) = \sum_{i=n+1}^{L} \log a_{s_n \ldots s_1 s_0}$$

$$= \sum_{s_n \ldots s_1 s_0 \forall s_0 \in k} n_{s_n \ldots s_1 s_0} \log(a_{s_n \ldots s_1 s_0}) \tag{2.10}$$

όπου $x_i = s_0$, $x_{i-1} = s_1$, $x_i = s_2, \ldots$, $x_{i-n} = s_n$ και $n_{s_k, \ldots, s_1, s_0}$ είναι ο συνολικός αριθμός των ακολουθιών της μορφής $s_k, \ldots, s_1, s_0$ στα δεδομένα μας. Το πρόβλημα που προκύπτει εδώ είναι το πως θα υπολογίσουμε τους εκτιμητές Μέγιστης Πιθανοφάνειας (Maximum Likelihood Estimates-MLEs) των πιθανοτήτων μεταβάσεως $a_{s_n \ldots s_1 s_0}$. Το πρόβλημα αυτό θα το λύσουμε παρακάτω.

Εδώ θα πρέπει επιπλέον να αναφέρουμε ότι το μοντέλο περιέχει $(k\text{-}1)^n$ ελεύθερες παραμέτρους, μία για κάθε πιθανό σύμβολο x_i μετά από κάθε πιθανό πλαίσιο μήκους n. Έχει αποδειχτεί ακόμα (Berchtold and Raftery, 2002) ότι η $n^{ης}$ τάξεως αλυσίδα Markov, είναι ισοδύναμη με μια αλυσίδα $1^{ης}$ τάξεως, αλλά με ένα

αλφάβητο της τάξης του 19^n στην περίπτωση των πρωτεϊνικών ακολουθιών. Κατά συνέπεια, απαιτεί τον υπολογισμό πινάκων μεταβάσεων μεγέθους 19^nx19^n. Άρα στην περίπτωση των πρωτεϊνών, ενώ για μια αλυσίδα $1^{ης}$ τάξεως χρειαζόμαστε να υπολογίσουμε 19^2=361 παραμέτρους για κάθε μοντέλο, για ένα μοντέλο $2^{ης}$ τάξεως χρειαζόμαστε 19^3=6859 παραμέτρους κτλ. Οι αριθμοί αυτοί των παραμέτρων που θα πρέπει να υπολογιστούν είναι υπερβολικά μεγάλοι, όπως φαίνεται και από τον πίνακα 2.1, ο οποίος στην περίπτωση των πρωτεϊνικών ακολουθιών θα απαιτούσε υπερβολικά μεγάλο αριθμό ακολουθιών να χρησιμοποιηθούν ως παραδείγματα για την εκπαίδευση των μοντέλων.

Πίνακας 2.1: Αύξηση του αριθμού παραμέτρων ανάλογα με την αύξηση της τάξης των αλυσίδων Markov για πρωτεϊνικές ακολουθίες με μέγεθος αλφάβητου 20.

Αλυσίδες Markov	**Αριθμός Παραμέτρων**
$1^{ης}$ τάξεως	$19^2 = 361$
$2^{ης}$ τάξεως	$19^3 = 6859$
$3^{ης}$ τάξεως	$19^4 = 130321$
$4^{ης}$ τάξεως	$19^5 = 2476099$
$5^{ης}$ τάξεως	$19^6 = 47045881$
$6^{ης}$ τάξεως	$19^7 = 893871739$
$7^{ης}$ τάξεως	$19^8 = 16983563041$

Όπως αναφέραμε και παραπάνω για την εκτίμηση των παραμέτρων χρησιμοποιούμε τον τύπο $(k-1)^n$, όπου k το μέγεθος του αλφάβητου και n η τάξη της αλυσίδας Markov. Έτσι περιπτώσεις αλυσίδων ανώτερης τάξης, είναι δυνατόν να εφαρμοστούν πιο εύκολα σε ακολουθίες νουκλεοτιδίων, όπου το αλφάβητο είναι μικρότερο και οι ακολουθίες πολύ μεγαλύτερες (Ellrott et al., 2002; Phillips et al., 1987) (βλ. Πίνακα 2.2). Λόγω του ότι το αλφάβητο είναι μικρό αυτό μας δίνει τη δυνατότητα να εφαρμόζουμε αλυσίδες Markov μεγαλύτερης τάξης απ' αυτές που μπορούμε να εφαρμόσουμε για πρωτεϊνικές ακολουθίες. Σε μια ενδιαφέρουσα εργασία, οι Audic και Claverie (Audic and Claverie, 1998), χρησιμοποίησαν αλυσίδες ανώτερης τάξης σε συνδυασμό με μια υπολογιστικά εντατική μεθοδολογία έτσι ώστε

να ανιχνεύσουν διαφορετικής σύστασης περιοχές σε βακτηριακά γονιδιώματα. Με αυτό τον τρόπο διαχώρισαν χωρίς την ανάγκη συνόλου εκπαίδευσης, περιοχές οι οποίες κωδικοποιούν πρωτεΐνες, περιοχές που δεν κωδικοποιούν τίποτα και περιοχές που κωδικοποιούν πρωτεΐνες αλλά στη συμπληρωματική τους αλυσίδα, σε ποσοστό που έφτανε και το 90% (Audic and Claverie, 1998).

Πίνακας 2.2: Αύξηση του αριθμού παραμέτρων ανάλογα με την αύξηση της τάξης των αλυσίδων Markov για ακολουθίες DNA με μέγεθος αλφάβητου 4.

Αλυσίδες Markov	**Αριθμός Παραμέτρων**
$1^{ης}$ τάξεως	$3^2 = 9$
$2^{ης}$ τάξεως	$3^3 = 27$
$3^{ης}$ τάξεως	$3^4 = 81$
$4^{ης}$ τάξεως	$3^5 = 243$
$5^{ης}$ τάξεως	$3^6 = 729$
$6^{ης}$ τάξεως	$3^7 = 2187$
$7^{ης}$ τάξεως	$3^8 = 6561$

2.3 Επεκτάσεις των Μαρκοβιανών Μοντέλων

Υπάρχουν επεκτάσεις των Μαρκοβιανών Μοντέλων που δημιουργήθηκαν για να μπορούμε να δημιουργούμε μοντέλα μεγάλης τάξης χωρίς να αντιμετωπίζουμε το πρόβλημα της εκθετικής αύξησης των παραμέτρων, όπως τα MTD (Mixture Transition Distribution), τα IMMs (Interpolated Markov Models) και τα VLMCs (Variable Length Markov Chains) και υπάρχουν και τα HMMς (Hidden Markov Models) τα οποία τα χρησιμοποιούμε για να δημιουργήσουμε πιο σύνθετους πίνακες μεταβάσεων.

Σε περιπτώσεις πρωτεϊνών, έχει προταθεί η προσέγγιση των πιθανοτήτων μετάβασης μεγαλύτερης τάξης. Συγκεκριμένα, η πιθανότητα μετάβασης για μια $n^{ης}$ τάξης αλυσίδα θα μπορούσε να προσεγγιστεί (Yuan, 1999), από τη σχέση:

$$P(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_{i-n}) \approx \prod_{i=1}^{n} P(x_i \mid x_{i-n}) \qquad (2.11)$$

Η σχέση (2.11) χρησιμοποιήθηκε από τον Yuan (Yuan, 1999), στην προσπάθεια να προβλεφθεί η υποκυτταρική τοποθεσία των βακτηριακών πρωτεϊνών, με αρκετή επιτυχία.

Σε γενικές γραμμές, αναμένουμε ότι με μεγαλύτερης τάξεως αλυσίδες θα έχουμε και καλύτερη διαχωριστική ικανότητα των μοντέλων, γεγονός που επιβεβαιώνεται και από τον Yuan (Yuan, 1999). Από την άλλη μεριά, μεγαλώνοντας πάρα πολύ την τάξη (>6), ακόμα και για νουκλεοτιδικές ακολουθίες, πέραν του προβλήματος υπερπροσαρμογής και της έλλειψης δεδομένων, ανακύπτει και το πρόβλημα της εισαγωγής θορύβου, από μη-σημαντικές μακρινές αλληλεπιδράσεις (Ellrott et al., 2002; Phillips et al., 1987; Yuan, 1999).

Άλλη μέθοδος, χρήσιμη κυρίως σε ακολουθίες νουκλεοτιδίων, είναι αυτή της χρήσης μη-ομογενών αλυσίδων (non-homogenous Markov chains), με την οποία χρησιμοποιούνται διαφορετικοί πίνακες μεταβάσεων, έτσι ώστε να εντοπιστούν καλύτερα οι στατιστικές προτιμήσεις στις διάφορες θέσεις της τριπλέτας βάσεων μιας κωδικής περιοχής (Borodovsky and Peresetsky, 1994). Μια ακόμα μέθοδος που δεν έχει χρησιμοποιηθεί ακόμα για κάποιο βιολογικό πρόβλημα, αλλά αποδεικνύεται πολύ ελπιδοφόρα, είναι αυτή των Mixture Transition Distribution (MTD), που αρχικά προτάθηκε από τον Raftery (Raftery, 1985) και θα αναλυθεί παρακάτω στην ενότητα 2.5. Πρόκειται για Μαρκοβιανά Μοντέλα των οποίων οι καταστάσεις προκύπτουν από το καρτεσιανό γινόμενο δύο ή περισσοτέρων τυχαίων μεταβλητών και περιγράφονται από τον παρακάτω τύπο:

$$a_{s_n \ldots s_1 s_0} = P(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_{i-n}) = \sum_{j=1}^{n} \lambda_j a_{s_j s_0}$$

Προκειμένου να έχουμε ένα μοντέλο MTD με τις παραμέτρους που καθορίζουν κατάλληλα τις πιθανότητες, πρέπει να επιβληθούν οι ακόλουθοι περιορισμοί:

$$\sum_{j=1}^{n} \lambda_j = 1$$

$$\lambda_j \geq 0$$

Μεγάλο ενδιαφέρον, επιπλέον, τόσο πρακτικό όσο και θεωρητικό, παρουσιάζουν τα μοντέλα Markov μεταβλητού μήκους (Variable length Markov Models-VMM), που θα τα αναλύσουμε παρακάτω στην ενότητα 2.8, τα οποία όπως

διατυπώθηκαν από τον Bejerano (Bejerano, 2004) είναι μια επέκταση της διατύπωσης των Στοχαστικών Πεπερασμένων Αυτομάτων (Probabilistic Finite Automata-PFA), από το Ron και τους συνεργάτες του (Ron et al., 1996). Το μοντέλο αυτό, αντί να υπολογίζει όλα τα C πλαίσια παραθύρων μήκους n, τα οποία θα καθορίσουν τις παραμέτρους της αλυσίδας n^{th} τάξης, υπολογίζει παραμέτρους μόνο για ένα υποσύνολο, το οποίο προσδιορίζεται με εκπαίδευση από τα δεδομένα και προβλέπει μεγαλύτερες εξαρτήσεις στα προηγηθέντα κατάλοιπα όταν είναι απαραίτητο ενώ μικρότερες, όταν δεν είναι. Έτσι, η προσεγγιστική σχέση που χρησιμοποιείται, είναι η εξής:

$$P(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_{i-n}) \approx P(x_i \mid \max_{n_1 \geq o} \{x_{i-n}, \ldots, x_{i-1} \in C\})$$

Με τη χρήση αυτής της σχέσης (και μιας πολύπλοκης διαδικασίας εκπαίδευσης που θα αναφερθεί παρακάτω), ο Bejerano και οι συνεργάτες του (Bejerano et al., 2001; Bejerano and Yona, 2001), κατάφεραν να κατασκευάσουν μοντέλα τα οποία να διακρίνουν με αρκετά μεγάλη ακρίβεια σχεδόν όλες τις οικογένειες πρωτεϊνών που είναι κατατεθειμένες στη βάση δεδομένων PFAM (Bateman, et al., 2004). Η προσέγγιση αυτή, έχει ενδιαφέρον γιατί έδειξε ότι απλούστερα αλλά καλής προγνωστικής αξίας μοντέλα, μπορούν να κατασκευαστούν, και να συναγωνίζονται σε επιτυχία τα πιο πολύπλοκα Hidden Markov Models (HMMs).

Τα HMMs είναι μια ακόμα κατηγορία Μαρκοβιανών Μοντέλων. Κάθε HMM συνδέεται με μια πρώτης τάξεως MC με κατάλληλες πιθανότητες μετάβασης μεταξύ των καταστάσεων και με μια αρχική κατανομή (Rabiner, 1989). Επιπλέον, κάθε κατάσταση εκπέμπει τα σύμβολα σύμφωνα με τις πιθανότητες της κατανομής. Οι πιθανότητες εκπομπής εξαρτώνται μόνο από την παρούσα κατάσταση του MC, ανεξάρτητα από τις προηγούμενες καταστάσεις. Ξεκινώντας από κάποια αρχική κατάσταση με μια αρχική πιθανότητα, μια ακολουθία καταστάσεων παράγεται πηγαίνοντας από τη μια κατάσταση στην άλλη σύμφωνα με τις πιθανότητες μετάβασης μέχρι μια τελική κατάσταση να επιτευχθεί, δημιουργώντας έτσι μια ακολουθία συμβόλων, καθώς κάθε κατάσταση εκπέμπει ένα σύμβολο. Η ακολουθία των συμβόλων είναι ορατή σε μας, ενώ οι καταστάσεις από τις οποίες περνά το

μοντέλο όχι. Για να υπολογίσουμε τη συνολική πιθανότητα, μιας ακολουθίας **x**, δεδομένου του μοντέλου, θα πρέπει να αθροίσουμε για όλες τις πιθανές αλληλουχίες καταστάσεων, δηλαδή να αθροίσουμε τη συνεισφορά στη συνολική πιθανότητα όλων των πιθανών μονοπατιών π.

$$P(\boldsymbol{x} \mid \theta) = \sum_{\pi} P(\boldsymbol{x}, \pi \mid \theta) = \sum_{\pi} a_{B\pi_1} \prod_{i=1}^{L} e_{\pi i}(x_i) a_{\pi_i \pi_{i+1}}$$

όπου: $$a_{kl} = P(\pi_i = l \mid \pi_{i-1} = k)$$

οι πιθανότητες μετάβασης από τη μια κατάσταση στην άλλη (transition probabilities) και: $$e_k(b) = P(x_i = b \mid \pi_i = k)$$

οι πιθανότητες εκπομπής των συμβόλων σε κάθε κατάσταση (emission probabilities). Κάθε HMM με *m* καταστάσεις και *k* σύμβολα μπορεί να παρασταθεί ως μια αλυσίδα Markov με *km* καταστάσεις. Συνεπώς ο αριθμός των πιθανοτήτων μετάβασης θα είναι απαγορευτικά μεγάλος της τάξης *km*km* και ο αριθμός των ελεύθερων παραμέτρων θα είναι (*km* -1)*(*km*-1). Αυτός είναι ο βασικός λόγος για τον οποίο και δημιουργήθηκαν τα υπόλοιπα είδη Μαρκοβιανών Μοντέλων που αναφέραμε παραπάνω και θα αναλύσουμε παρακάτω στις επόμενες ενότητες.

Τα Μαρκοβιανά Μοντέλα όμως παρά τα πολλά πλεονεκτήματά που παρουσιάζουν διαθέτουν και κάποια μειονεκτήματα. Πρώτον όταν προσπαθούμε να εφαρμόσουμε MCs μεγάλης τάξεως τότε προκύπτει ένας πολύ μεγάλος αριθμός παραμέτρων, όπως αναλύσαμε και παραπάνω. που δεν θα μπορούσαμε να υπολογίσουμε. Δεύτερον τα απλά ΜΜ, αν και δεν απαιτούν μεγάλο χρονικό διάστημα για τη δημιουργία τους, έχει αποδειχτεί ότι δεν αποδίδουν πολύ καλά σε βιολογικές εφαρμογές. Απ' την άλλη τα HMMs, τα οποία χρησιμοποιούνται ευρέως στη βιολογία τα τελευταία χρόνια και αποτελούν λύση για ένα μεγάλο φάσμα βιολογικών προβλημάτων με μεγάλα ποσοστά επιτυχίας, είναι ιδιαίτερα χρονοβόρα στη δημιουργία τους. Αυτό οφείλεται στο γεγονός ότι απαιτείται αρκετός χρόνος για τον καθορισμό των πιθανοτήτων μετάβασης από τη μια κατάσταση στην άλλη αλλά και των πιθανοτήτων γέννησης των συμβόλων σε κάθε κατάστασή. Μια λύση στα προβλήματα αυτά αποτελούν τα VLMCs. Τα VLMCs είναι αλυσίδες Markov με το

επιπρόσθετο χαρακτηριστικό ότι η μνήμη τους εξαρτάται από έναν μεταβλητό αριθμό παρελθοντικών τιμών, ανάλογα με το πώς το παρελθόν (οι προηγούμενες τιμές) μοιάζει. Τα VLMCs στηρίζονται στη δημιουργία πιθανολογικών δέντρων για των υπολογισμό των πιθανοτήτων μετάβασης. Οι ακολουθίες που θα χρησιμοποιηθούν για να εκπαιδεύσουν το VLMC δεν χρειάζεται να έχουν στοιχηθεί και μπορεί να έχουν οποιοδήποτε μήκος. Η μέθοδος είναι αυτόματη και εξαιρετικά γρήγορη, και μπορεί επιπλέον να εφαρμοστεί, χωρίς να προϋποθέτει οποιαδήποτε άλλη βιολογική πληροφορία.

2.4 Εκτίμηση Παραμέτρων

Ένα πιθανοθεωρητικό μοντέλο M είναι μια περιγραφή μιας κατηγορίας Ωj , όπου j οι κατηγορίες του μοντέλου, δηλ. $j = 1....\lambda$. Με δεδομένο ένα διάνυσμα σαν είσοδο, στην περίπτωσή μας μια πρωτεϊνική ακολουθία $\boldsymbol{x}$ η οποία έχει σαν χαρακτηριστικά γνωρίσματα αμινοξέα, μπορούμε να υπολογίσουμε την πιθανότητα $P_{\Omega j}(\boldsymbol{x})$ και να βρούμε σε ποια κατηγορία j ανήκει η ακολουθία, με:

$$P_{\Omega_j}(\boldsymbol{x}) := P(\boldsymbol{x} \mid \Omega_j)$$

Ένα μοντέλο περιέχει ένα σύνολο παραμέτρων θ που ρυθμίζονται κατά τη διάρκεια της εκπαίδευσης, χρησιμοποιώντας ένα σύνολο W δειγμάτων. Στην περίπτωση της εποπτευόμενης εκπαίδευσης, τα δείγματα ανήκουν σε μία κατηγορία το καθένα, έτσι το σύνολο W αποτελείται από υποσύνολα για κάθε μια από τις Ω_j κατηγορίες. Η δημιουργία ενός μοντέλου επομένως περιλαμβάνει την επιλογή των παραμέτρων, της δομής ή της τοπολογίας του μοντέλου (αν έχει), και την εκπαίδευση των παραμέτρων στη συνέχεια. Στόχος μας είναι να λάβουμε ένα σύνολο παραμέτρων που θα μας δίνει τα καλύτερα δυνατά ποσοστά αναγνώρισης των Ω_j κατηγοριών υπό εξέταση. Η επιλογή της δομής ή της τοπολογίας ή του τρόπου δημιουργίας του μοντέλου θα πρέπει να οδηγήσει σε ένα μοντέλο, το οποίο να προσαρμόζεται καλά στο δοθέν σύνολο εκπαίδευσης και ταυτόχρονα να περιγράφει γενικά τη συγκεκριμένη κατηγορία. Γενικά ένα μοντέλο με πολλές παραμέτρους αναμένεται να δώσει καλά ποσοστά όταν το ελέγχουμε έναντι του συνόλου εκπαίδευσης αλλά όταν το ελέγχουμε έναντι άλλων συνόλων θα αποτύχει γιατί θα έχει υπερπροσαρμοστεί στο σύνολο εκπαίδευσης. Ένα μοντέλο απ' την άλλη με

μικρό αριθμό παραμέτρων, δεν θα είναι σε θέση να συλλάβει πλήρως το πρόβλημα και έτσι δεν θα μας δώσει καλά αποτελέσματα όταν το ελέγχουμε έναντι ενός συνόλου με άγνωστα δείγματα .

Επιπλέον δεν είναι συχνά δυνατό να βρεθούν οι παράμετροι και να εκπαιδευτεί το μοντέλο κατά τέτοιο τρόπο ώστε το ποσοστό αναγνώρισης να βελτιστοποιείται άμεσα. Για αυτό το λόγο, χρησιμοποιείται μια αντικειμενική συνάρτηση, που αναθέτει κάποιες τιμές στις παραμέτρους, που μας δίνουν τα βέλτιστα ποσοστά επιτυχίας όταν αυτή η συνάρτηση εφαρμόζεται στο σύνολο εκπαίδευσης. Μια γνωστή αντικειμενική συνάρτηση είναι η *Μέγιστη Πιθανοφάνεια (ML)*:

$$\hat{\theta}^{ML} = \arg\max_{\theta} P(\boldsymbol{x} \mid \Omega_j) \tag{2.12}$$

όπου $\hat{\theta}$ είναι το σύνολο των θ παραμέτρων που μεγιστοποιούν την $P(\boldsymbol{x})$.

Στην περίπτωση που τα μοντέλα θα χρησιμοποιηθούν για ταξινόμηση, μπορεί να είναι καλύτερο να υιοθετήσουμε *τις χαρακτηριστικές* αντικειμενικές συναρτήσεις, όπου η έμφαση δεν δίνεται τόσο στη μοντελοποίηση μιας κατηγορίας αλλά στη σωστή ταξινόμηση των δειγμάτων. Η Μέγιστη Αμοιβαία Πληροφορία (Maximum Mutual Information -MMI) (Bahl et al, 1986) είναι ένα παραδείγματα μιας χαρακτηριστικής συνάρτησης, που μεγιστοποιεί την *a posteriori* πιθανότητα μιας κατηγορίας, υποθέτοντας ότι το τμήμα (πλαίσιο) που ανήκει σε αυτήν την κατηγορία παρατηρήθηκε.

$$\begin{aligned} \theta^{MMI} &= P(\Omega_j \mid \boldsymbol{x}) \\ &= \frac{P(\boldsymbol{x} \mid \Omega_j) P(\Omega_j)}{\sum_{j=1}^{\lambda} P(\boldsymbol{x} \mid \Omega_j) P(\Omega_j)} \end{aligned} \tag{2.13}$$

Το $P(\Omega j)$ δείχνει την *a priori* πιθανότητα της κάθε κατηγορίας j. Η αντικειμενική συνάρτηση δεν μεγιστοποιείται για κάθε κατηγορία, αντί αυτού χειρίζεται όλα τα δείγματα μαζί, και η πιθανότητα του μοντέλου για τη σωστή κατηγορία υπολογίζεται σε σχέση με τη συνολική πιθανότητα που λαμβάνεται από όλα τα μοντέλα. Ο Nadas και οι συνεργάτες του (Nadas, et al., 1988) απέδειξαν αυτού ότι η MMI μας δίνει

καλύτερα αποτελέσματα αναγνώρισης από την ML όταν έχουμε περιορισμένο μεγέθους δειγμάτων. Ο Eddy και οι συνεργάτες του (Eddy et al.,1995) περιέγραψαν μια ειδική περίπτωση της MMI και την αποκάλεσαν Μέγιστη Διάκριση (Maximum Discrimination-MD), όπου τα μοντέλα εκπαιδεύονται σύμφωνα με την MMI, αλλά χρησιμοποιώντας μόνο τα δείγματα από κάθε κατηγορία.

Ο κύριος λόγος για τον οποίο τέτοιες χαρακτηριστικές (discrimination) συναρτήσεις δεν έχουν αντικαταστήσει την εκτίμηση ML είναι η αυξανόμενη πολυπλοκότητα κατά τη διαδικασία εκπαίδευσης. Ο υπολογισμός της MMI αντικειμενικής συνάρτησης απαιτεί την εφαρμογή *όλων* των μοντέλων στο *πλήρες* σύνολο δειγμάτων, η MD απ' την άλλη απαιτεί την εφαρμογή *όλων* των μοντέλων στο υποσύνολο της αντίστοιχης κατηγορίας ενώ η ML απαιτεί να εφαρμοστεί μόνο ένα μοντέλο στο αντίστοιχο υποσύνολο δειγμάτων. Η ML, η MMI, και η MD, αποτελούν μια λύση για το πρόβλημα της εκτίμησης των παραμέτρων για ένα δεδομένο μοντέλο. Ωστόσο η επιλογή της δομής ή της τοπολογίας ή του τρόπου δημιουργίας ενός μοντέλου δεν είναι απλή, σε πολλές περιπτώσεις οι ευριστικές μέθοδοι είναι ο μόνος τρόπος για να αντιμετωπίσουμε αυτό το πρόβλημα.

Από τις παραπάνω αντικειμενικές συναρτήσεις θα επιλέξουμε τη Μέγιστη Πιθανοφάνεια (τύπος 2.12) για να βρούμε τις παραμέτρους του μοντέλου για τους λόγους που προαναφέραμε. Από τον τύπο 2.10 προκύπτει ότι η πιθανότητα $P(\boldsymbol{x})$ μιας πρωτεΐνης δίνεται από:

$$\log P(\boldsymbol{x}) = \sum_{i=n+1}^{L} \log P(x_i \mid x_{i-1}, x_{i-2} \ldots, x_{i-n}) = \sum_{i=n+1}^{L} \log a_{s_n \ldots s_1 s_0}$$

$$= \sum_{s_n \ldots s_1 s_0 \forall s_0 \in k} n_{s_n \ldots s_1 s_0} \log(a_{s_n \ldots s_1 s_0})$$

Καθορίζουμε στη συνέχεια την αρνητική log-likelihood, ως:

$$\ell = -\log P(x) = -\sum_{i=n+1}^{L} \log a_{s_n \ldots s_1 s_0}$$

Έπειτα παραγωγίζουμε τον παραπάνω τύπο ως προς τις παραμέτρους του μοντέλου:

$$\frac{\partial \ell}{\partial a_{s_n \ldots s_1 s_0}} = -\sum_{i=n+1}^{L} \frac{1}{a_{s_n \ldots s_1 s_0}}$$

Στη συνέχεια θέτουμε την παράγωγο ίση με 0:

$$-\sum_{i=n+1}^{L}\frac{1}{a_{s_n\ldots s_1 s_0}}=0$$

και λύνουμε τη εξίσωση θέτοντας τον παρακάτω περιορισμό:

$$\sum_{s_n,\ldots,s_1,s_0\,\forall s\in k} a_{s_n\ldots s_1 s_0}=1$$

Οι εκτιμητές Μέγιστης Πιθανοφάνειας (Maximum Likelihood Estimates-MLEs) των πιθανοτήτων μεταβάσεως, υπολογίζονται σύμφωνα με τη σχέση (Agresti, 2002):

$$\hat{a}_{s_n\ldots s_1 s_0}=\frac{c_{s_n\ldots s_1 s_0}}{\sum_{s_n} c_{s_n\ldots s_1 s_0}}$$

Ο αντίστοιχος εκτιμητής για πρώτης τάξης μοντέλο προκύπτει από τον παρακάτω τύπο:

$$\hat{a}_{x_{i-1}x_i}=\frac{c_{x_{i-1}x_i}}{\sum_{x_i'} c_{x_{i-1}x_i'}}$$

όπου c_{st} είναι οι παρατηρούμενες εμφανίσεις του καταλοίπου s ακολουθούμενο από το κατάλοιπο t στις ακολουθίες εκπαίδευσης, με το άθροισμα στον παρονομαστή εκτείνεται σε όλο το αλφάβητο των 20 αμινοξέων. Θεωρώντας δυο διαφορετικά μοντέλα, με τη χρήση δυο πινάκων μεταβάσεων (ένα μοντέλο + για τα λεγόμενα θετικά παραδείγματα και ένα μοντέλο – για τα λεγόμενα αρνητικά), μπορούμε να ορίσουμε ένα log-odds score, S(**x**) για ολόκληρη την ακολουθία, το οποίο είναι χρήσιμο για διαχωριστικούς σκοπούς:

$$S(x)=\log\frac{P(x\mid +)}{P(x\mid -)}=\sum_{i=1}^{L}\log\left(\frac{a_{x_{i-1}x_i}}{\overline{a}_{x_{i-1}x_i}}\right)=\sum_{i=1}^{L}\beta_{x_{i-1}x_i}$$

όπου $\beta_{x_{i-1}x_i}$, είναι το log-odds για την πιθανότητα μετάβασης από το κατάλοιπο x_{i-1} στο x_i, και είναι ένα σχετικό μέτρο της τάσης των πιθανοτήτων μετάβασης να εμφανίζονται πιο συχνά στο ένα ή το άλλο μοντέλο. Τιμές των $\beta_{x_{i-1}x_i}$ μεγαλύτερες από το 0, υποδηλώνουν προτιμήσεις των συγκεκριμένων μεταβάσεων για το μοντέλο (+), ενώ τιμές μικρότερες από το 0 προτίμηση για το μοντέλο (-). Για να

εκμηδενίσουμε την επιρροή του μήκους των ακολουθιών στο συνολικό score, περαιτέρω κοινωνικοποιούμε τις τιμές, διαιρώντας με το μήκος L της ακολουθίας έτσι ώστε να πάρουμε ένα log-odds score ανά κατάλοιπο.

$$S^{norm}(x) = \frac{S(x)}{L} = \frac{\sum_{i=1}^{L} \beta_{x_{i-1}x_i}}{L}$$

Χαρακτηριστικό παράδειγμα, 1ης τάξης μοντέλου με την παραπάνω διατύπωση, αναφέρεται στην εύρεση νησίδων CG στα ευκαρυωτικά γονιδιώματα (Durbin, et al., 1998). Δοθέντος των ML πιθανοτήτων για δύο αλυσίδες Markov πρώτης τάξης, εκπαιδεύουμε το μοντέλο μας με 48 CG ακολουθίες και άλλα αρνητικά παραδείγματα (πίνακας 2.3). Κάθε σειρά περιέχει τις πιθανότητες που προκύπτουν από το ίδιο πλαίσιο,
οι τιμές των οποίων αθροίζονται στη συνέχεια σε μια. Από τις παραμέτρους στον πίνακα 2.3, γίνεται προφανές ότι ακόμα και στο επίπεδο της πρώτης τάξης, οι παράμετροι είναι αρκετά διαφορετικές. Προσέξτε τη διαφορά πιθανοτήτων P(G|C) για τις μη CG νησίδες.

Πίνακας 2.3: Παράδειγμα, 1ης τάξης μοντέλου για εύρεση νησίδων CG στα ευκαρυωτικά γονιδιώματα. Ο πίνακας δείχνει τις παραμέτρους του μοντέλου.

	CG νησίδες				**Μη-CG νησίδες**			
πλαίσιο	***A***	***C***	***G***	***T***	***A***	***C***	***G***	***T***
A	0.180	0.274	0.426	0.120	0.300	0.205	0.285	0.210
C	0.171	0.368	0.274	0.188	0.322	0.298	0.078	0.302
G	0.161	0.339	0.375	0.125	0.248	0.246	0.298	0.208
T	0.079	0.355	0.384	0.182	0.177	0.239	0.292	0.292

2.5 HMMs (Hidden Markov Models)

2.5.1 Εισαγωγή

Η μελέτη των αλυσίδων Markov (MCs) όπως προαναφέραμε άρχισε το 19ο αιώνα (Rabiner, 1989) από τον Markov (Sheynin, 1988) και τα Hidden Markov Models (HMMs) ήταν από τα πρώτα Μαρκοβιανά Μοντέλα που εφαρμόστηκαν με μεγάλη επιτυχία για την επίλυση βιολογικών προβλημάτων. Εντούτοις, από το 1940

έως το 1960, τα HMMs μελετούνταν απλά εφαρμογή των πιθανολογικών διαδικασιών των MCs (Blackwell and Koopmans, 1957; Burke and Rosenblatt, 1958; Gilbert, 1959; Heller, 1965) και δεν ήταν ιδιαίτερα δημοφιλή μέχρι και τη δεκαετία του '70 όταν ο Baum τα εφάρμοσε επιτυχώς για αναγνώριση της ομιλίας με την ανάπτυξη ενός αποδοτικού αλγορίθμου εκπαίδευσης που βασίζονταν σε αυτά (Baum et al., 1970). Τα HMMs έγιναν πολύ δημοφιλή στην κοινότητα της υπολογιστικής βιολογίας μόνο όταν τρεις ομάδες εισήγαγαν νέες μεθόδους για τη ταξινόμηση ακολουθιών που βασίζονταν στα profile HMMs (Baldi et al., 1994; Eddy et al., 1995; Krogh et al., 1994).

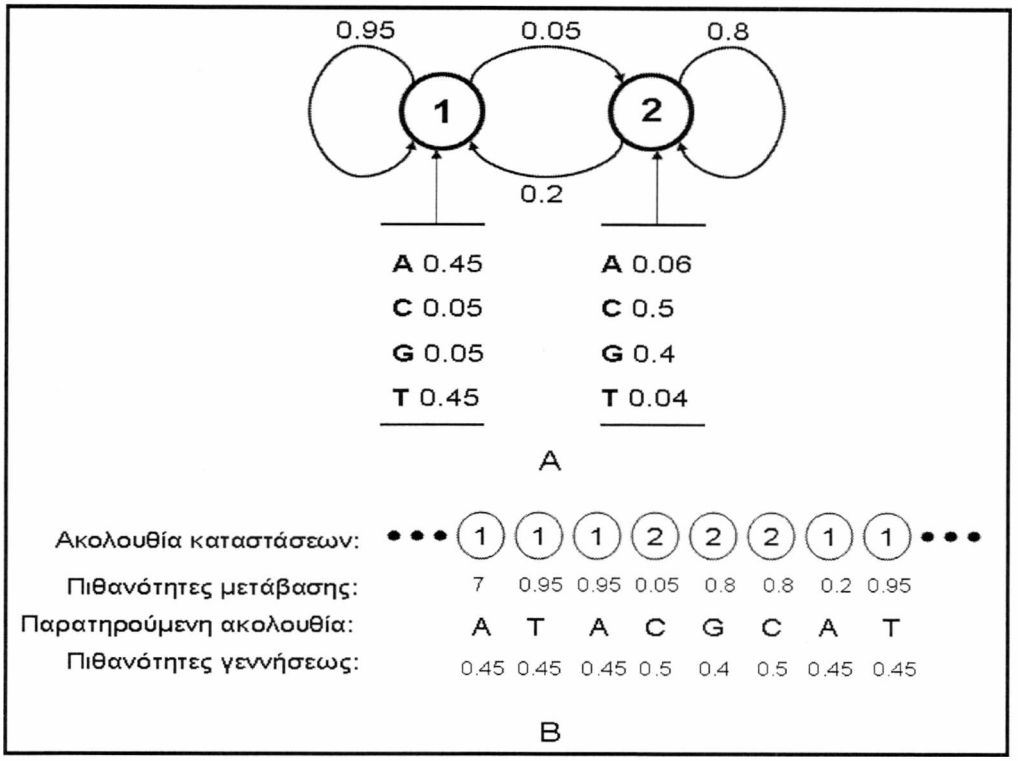
2.5.2 Θεωρία

Κάθε HMM συνδέεται με μια πρώτης τάξεως MC με κατάλληλες πιθανότητες μετάβασης μεταξύ των καταστάσεων και με μια αρχική κατανομή (Rabiner, 1989). Επιπλέον, κάθε κατάσταση εκπέμπει τα σύμβολα σύμφωνα με τις πιθανότητες της κατανομής. Οι πιθανότητες εκπομπής εξαρτώνται μόνο από την παρούσα κατάσταση του MC, ανεξάρτητα από τις προηγούμενες καταστάσεις. Ξεκινώντας από κάποια αρχική κατάσταση με μια αρχική πιθανότητα, μια ακολουθία καταστάσεων παράγεται πηγαίνοντας από τη μια κατάσταση στην άλλη σύμφωνα με τις πιθανότητες μετάβασης μέχρι μια τελική κατάσταση να επιτευχθεί, δημιουργώντας έτσι μια ακολουθία συμβόλων, καθώς κάθε κατάσταση εκπέμπει ένα σύμβολο. Η ακολουθία των συμβόλων είναι ορατή σε μας, ενώ οι καταστάσεις από τις οποίες περνά το μοντέλο όχι.

Η βασική ιδέα είναι ότι ένα HMM είναι μια γεννήτρια ακολουθίας. Ένα απλό HMM για την παραγωγή μιας ακολουθίας DNA διευκρινίζεται στην Εικόνα 2.2A. Στο μοντέλο αυτό, οι μεταβάσεις από τη μια κατάσταση στην άλλη και οι πιθανότητες αυτών υποδεικνύονται από τα βέλη και οι πιθανότητες εκπομπής συμβόλων για τα A, C, G, T σε κάθε κατάσταση είναι υποδειγμένες κάτω από την κάθε κατάσταση.

Για λόγους κατανόησης, παραλείπουμε την αρχική και την τελική κατάσταση καθώς επίσης και την αρχική κατανομή πιθανότητας. Για παράδειγμα, αυτό το πρότυπο μπορεί να παραγάγει την ακολουθία καταστάσεων που δίνεται στο σχήμα

2.2B και κάθε κατάσταση εκπέμπει ένα νουκλεοτίδιο σύμφωνα με τη πιθανότητα εκπομπής.



Εικόνα 2.2: Α. ένα απλό HMM μοντέλο για ανίχνευση ακολουθίας DNA Β. η ακολουθία που προκύπτει και η παρατηρούμενη ακολουθία DNA

Κατά την παραγωγή των ακολουθιών, μόνο τα σύμβολα παραγωγής μπορούν να παρατηρηθούν. Οι ακολουθίες των καταστάσεων που παράγονται από το MC είναι κρυμμένες και δεν μπορούν να παρατηρηθούν. Οποιαδήποτε ακολουθία μπορεί να αντιπροσωπευθεί από μια ακολουθία καταστάσεων μέσα στο μοντέλο. Η πιθανότητα κάθε ακολουθίας, που δίνεται από το μοντέλο, υπολογίζεται πολλαπλασιάζοντας τις πιθανότητες εκπομπής με αυτές τις μετάβασης κατά μήκος του μονοπατιού.

2.5.3 Τοπολογίες HMMs

Η τοπολογία ενός HMM αναφέρεται στο σύνολο των καταστάσεων, και κυρίως στις επιτρεπόμενες και απαγορευμένες μεταβάσεις μεταξύ των καταστάσεων του MC, δηλαδή στις μη-μηδενικές και τις μηδενικές τιμές των μεταβάσεων, δηλαδή

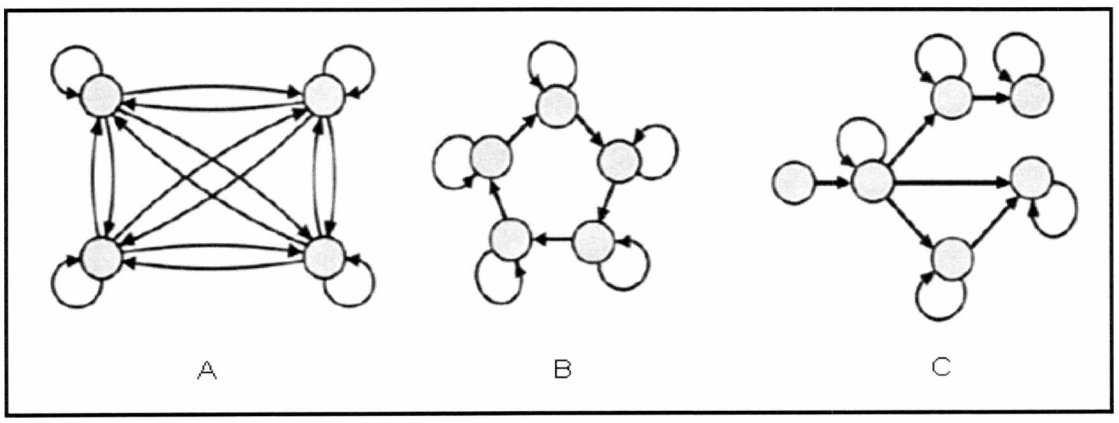
στις μεταβάσεις που επιτρέπονται και σε αυτές που απαγορεύονται. Μέχρι σήμερα, πολλές διαφορετικές τοπολογίες HMM έχουν προταθεί. Οι 3 πιο χαρακτηριστικές είναι το "***πλήρως συνδεμένο***" μοντέλο, το "***κυκλικό***" μοντέλο και το "***από αριστερά προς δεξιά***" μοντέλο.

2.5.3.1 Πλήρως συνδεδεμένο μοντέλο

Ένα HMM καλείται πλήρως συνδεδεμένο (Εικόνα 2.3Α) όταν κάθε κατάσταση συνδέεται με όλες τις άλλες καταστάσεις, δηλ. όταν από οποιαδήποτε κατάσταση μπορούμε να μεταβούμε σε οποιαδήποτε άλλη. Ακόμα σ' αυτό το μοντέλο δεν υπάρχει καμία διακριτή αρχική ή τελική κατάσταση. Με εξαίρεση τις διαγώνιες καταχωρήσεις που αντιστοιχούν σε βρόχους ή τις μεταβάσεις στον εαυτό τους οι μεταβάσεις δεν παίρνουν ποτέ μηδενικές τιμές.

2.5.3.2 Κυκλικό μοντέλο

Σε ένα κυκλικό μοντέλο (Εικόνα 2.3Β), η μια κατάσταση οδηγεί στην επόμενη ή στον εαυτό της αλλά όχι στην προηγούμενη. Επειδή οι καταστάσεις όμως συνδέονται όλες σε ένα κλειστό βρόχο (κύκλο) η κάθε κατάσταση εντέλει θα επαναληφθεί, με εξαίρεση βέβαια τις καταστάσεις με μηδενική πιθανότητα. Το μοντέλο αυτό δεν δέχεται αλλαγές μεγέθους και δεν έχει κάποια διακριτή αρχική ή τελική κατάσταση.



Εικόνα 2.3: Τοπολογίες HMM: Α. Πλήρως συνδεδεμένο μοντέλο Β. Κυκλικό μοντέλο C. Από αριστερά προς τα δεξιά μοντέλο

2.5.3.3 Από τα αριστερά προς τα δεξιά μοντέλο

Όταν το μοντέλο δεν είναι κυκλικό και έχει βέλη που πηγαίνουν μόνο προς μία κατεύθυνση, και συγκεκριμένα από αριστερά προς δεξιά, χωρίς να γυρνάνε σε κάποια προηγούμενη κατάσταση, με εξαίρεση τις μεταβάσεις στον εαυτό τους, είναι γνωστό ως από αριστερά προς δεξιά μοντέλο (Εικόνα 2.3C). Το μοντέλο αυτό έχει μια αρχική κατάσταση και μια τελική κατάσταση. Μια πιο αυστηρή μορφή αυτής της τοπολογίας είναι αυτή που απαγορεύει ακόμα και τις μεταβάσεις στον εαυτό τους, δηλαδή από μια κατάσταση στην ίδια.

2.5.4 Μοντέλα HMM

Ένα Hidden Markov Model, είναι ένα μοντέλο **M** που περιέχει 3 στοιχεία **Σ, Q, θ**.

$$\mathbf{M} = (\Sigma, Q, \theta)$$

- **Σ**, το αλφάβητο των δυνατών ενδεχομένων.
- **Q**, το σύνολο των δυνατών καταστάσεων του μοντέλου (γονίδιο – όχι γονίδιο για το DNA κλπ).
- **θ**, το σύνολο πιθανοτήτων που διέπουν το μοντέλο και μπορεί να είναι:
    1. Πιθανότητες μεταβάσεως (transitions) από κατάσταση σε κατάσταση
    2. Πιθανότητες εκπομπής-γεννήσεως (emissions), με τις οποίες παράγονται τα σύμβολα σε κάθε κατάσταση.

Πρέπει να τονιστεί, ότι σε ένα HMM η μαρκοβιανή ιδιότητα ισχύει για τις καταστάσεις του μοντέλου (states), και όχι για τα σύμβολα.

2.5.4.1 Ορισμοί

Ακολουθία συμβόλων :

$$\boldsymbol{x} = x_1, x_2, \ldots, x_{L-1}, x_L$$

Πιθανότητες μετάβασης (transition probabilities):

$$a_{kl} = P(\pi_i = l \mid \pi_{i-1} = k)$$

Πιθανότητες γεννήσεως-εκπομπής (emission probabilities):

$$e_k(b) = P(x_i = b \mid \pi_i = k)$$

Αρχικές πιθανότητες (begin probabilities):

$$a_{Bk} = P(\pi_1 = k \mid B)$$

Τελικές πιθανότητες (end probabilities):

$$a_{kE} = P(E \mid \pi_i = k)$$

Η από κοινού πιθανότητα μιας ακολουθίας **x** και του μονοπατιού $\boldsymbol{\pi}$:

$$P(\boldsymbol{x}, \pi) = P(x_L, x_{L-1}, \ldots\ldots, x_1, \pi) = a_{B\pi_1} \prod_{i=1}^{L} e_{\pi i}(x_i) a_{\pi_i \pi_{i+1}}$$

Για να υπολογίσουμε τη συνολική πιθανότητα, μιας ακολουθίας **x**, δεδομένου του μοντέλου, θα πρέπει να αθροίσουμε για όλες τις πιθανές αλληλουχίες καταστάσεων, δηλαδή να αθροίσουμε τη συνεισφορά στη συνολική πιθανότητα όλων των πιθανών μονοπατιών π.

$$P(\boldsymbol{x} \mid \theta) = \sum_{\pi} P(\boldsymbol{x}, \pi \mid \theta) = \sum_{\pi} a_{B\pi_1} \prod_{i=1}^{L} e_{\pi i}(x_i) a_{\pi_i \pi_{i+1}}$$

2.5.4.2 Generalized HMM (GHMM)

Ένα Generalized HMM (GHMM), γνωστό και ως semi-Markov model, είναι δομικά και λειτουργικά παρόμοιο με τα κανονικά HMMs. Η διαφορά του από τα υπόλοιπα HMMs είναι ότι έχει επιπλέον μια κατανομή για τη διάρκεια μιας κατάστασης, η οποία καθορίζει το χρόνο που το HMM μένει σε μια κατάσταση.

Σε ένα κανονικό HMM, η διάρκεια διανέμεται γεωμετρικά. Εντούτοις, σε ένα GHMM, η διάρκεια d μιας κατάστασης X επιλέγεται από κάποια γενικευμένη κατανομή, που συνήθως προέρχεται από τα στοιχεία της εκπαίδευσης. Κάθε κατάσταση παράγει τα σύμβολα αφού πρώτα επιλέξει τη διάρκεια κάθε κατάστασης σύμφωνα με την κατανομή διάρκειας, και έπειτα παράγει την ακολουθία. Τα GHMM μοντέλα χρησιμοποιούνται επιτυχώς σε προγράμματα εύρεσης γονιδίων.

2.5.4.3 Pair HMM (PHMM)

Αντιπροσωπεύει ακόμα μια παραλλαγή των συνηθισμένων HMMs και έχει υιοθετηθεί ευρέως για τη στοίχιση δύο ακολουθιών κατά ζεύγη (Smith, et al., 2003). Η λειτουργία των PHMM είναι ίδια με αυτή των συνηθισμένων HMM με εξαίρεση ότι κάθε κατάσταση εκπέμπει ένα ζευγάρι συμβόλων. Ένα σύνηθες πρόβλημα στη στοίχιση ακολουθιών είναι ο προσδιορισμός της σωστής στοίχισης όταν η ομοιότητα είναι μικρή. Με τη χρήση των PHMM, η πιθανότητα ένα δεδομένο ζευγάρι των ακολουθιών να συσχετίζεται μπορεί να υπολογιστεί, ανεξάρτητα από το αποτέλεσμα της στοίχισης, αθροίζοντας όλα τα πιθανά μονοπάτια χρησιμοποιώντας τον αλγόριθμο forward.

2.5.4.4 Generalized pair HMM (GPHMM)

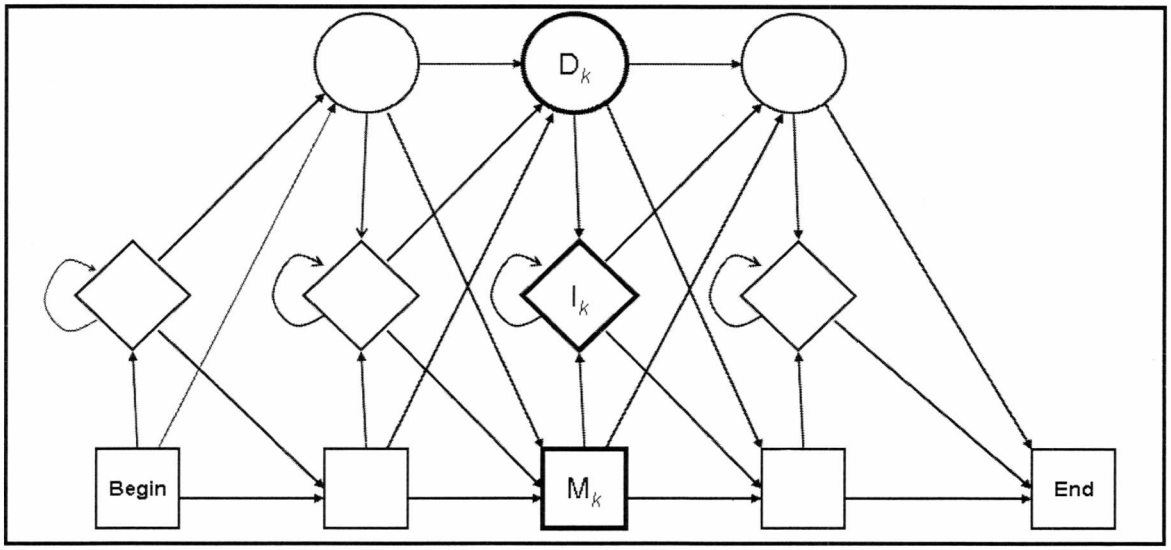
Είναι ένα υβριδικό πιθανολογικό μοντέλο που συνδυάζει το GHMM και το PHMM (Pachter et al., 2002). Ένα GPHMM μπορεί να θεωρηθεί ως μια μηχανή παραγωγής μιας ακολουθίας, που παράγει ένα ζευγάρι από τις παρατηρηθείσες ακολουθίες με διαφορετικά μήκη.

2.5.4.5 Profile HMMs

Είναι γραμμικά, από αριστερά προς δεξιά μοντέλα που χρησιμοποιούνται συνήθως για την ανίχνευση των δομικών ομοιοτήτων και των ομολογιών. Η αρχιτεκτονική των profile HMMs αποτελείται από τρεις κατηγορίες καταστάσεων (Durbin, et al., 1998) την εισαγωγή (insert state), τη σύμπτωση (match state) και την απαλοιφή (delete state), και από δύο σύνολα παραμέτρων: τις πιθανότητες μετάβασης και τις πιθανότητες εκπομπής.

Οι καταστάσεις εισαγωγής και σύμπτωσης εκπέμπουν πάντα ένα σύμβολο, ενώ η κατάσταση διαγραφής όχι, γιατί είναι σιωπηρή και ως εκ τούτου δεν έχει πιθανότητες εκπομπής. Οι μεν καταστάσεις σύμπτωσης αντιστοιχούν σε στήλες της πολλαπλής στοίχισης οι οποίες στοιχίζονται καλά και άρα αντιστοιχούν σε περιοχή με ομοιότητα, ενώ οι καταστάσεις εισαγωγής, αντιστοιχούν σε περιοχές στις οποίες

έχουμε εισαγωγή χαρακτήρων που δεν στοιχίζονται καλά. Οι περιοχές αυτές, οι οποίες δεν υπάρχουν στις υπόλοιπες ακολουθίες, εμφανίζονται ως κενά τα οποία μοντελοποιούνται μέσω των σιωπηρών καταστάσεων απαλοιφής. Το μοντέλο αρχίζει πάντα από την κατάσταση έναρξης και τελειώνει με μια τελική κατάσταση. Οι ποινές για την εισαγωγή των κενών κατά τις εισαγωγές ή τις διαγραφές, υπολογίζονται από τα δεδομένα με καθαρά πιθανοθεωρητικό τρόπο, αποκλείοντας την υποκειμενική παρέμβαση.



Εικόνα 2.4: Σχηματική αναπαράσταση ενός profile HMM.

Οι καταστάσεις που παρατηρούνται σε ένα τέτοιο μοντέλο (εκτός αυτών της εκκίνησης και του τερματισμού) χωρίζονται σε 3 κατηγορίες, όπως αυτές παρουσιάζονται και στην εικόνα:

Καταστάσεις Σύμπτωσης (Match states) M_k τετράγωνα

Καταστάσεις Εισαγωγής (Insertion states) I_k ρόμβοι

Καταστάσεις Απαλοιφής (Deletion states) D_k κύκλοι

Και συνδέονται με τις αντίστοιχες πιθανότητες μεταβάσεως, που στην εικόνα συμβολίζονται με βέλη.

Ένα κύριο μειονέκτημα των profile HMMs είναι ότι και το σήμα και ο θόρυβος αντιμετωπίζονται με τον ίδιο τρόπο, έχοντας ως συνέπεια την εκτίμηση ενός μεγάλου αριθμού παραμέτρων εκπομπής. Αυτό το πρόβλημα της υπερπροσαρμογής

(overfitting) αποφεύγεται με τη χρήση ενός συστήματος (Karplus, 1995) που αντικαθιστά την παρατηρηθείσα κατανομή των αμινοξέων με μια εκτιμώμενη κατανομή.

2.5.5 Εφαρμογές των HMMs στην υπολογιστική βιολογία

Αλγόριθμοι όπως το BLAST (Altschul, et al., 1990) ή το FASTA (Pearson and Lipman, 1988), που χρησιμοποιούνται στη σύγκριση αλληλουχιών για να συμπεράνουμε τη βιολογική λειτουργία μιας πρωτεΐνης, δουλεύουν καλά για αλληλουχίες που παρουσιάζουν μεγάλη ομοιότητα, εν τούτοις δίνουν μέτρια αποτελέσματα για τις αλληλουχίες που διαφέρουν αρκετά μεταξύ τους. Παρακάτω παρουσιάζονται πρόσφατες εφαρμογές των HMMs στους διαφορετικούς τομείς της υπολογιστικής βιολογίας που δίνουν καλύτερα αποτελέσματα σ' αυτές τις περιπτώσεις.

2.5.5.1 Στοίχιση ακολουθίας κατά ζεύγη

Η στοίχιση ακολουθίας κατά ζεύγη στηρίζεται στη στοίχιση δύο ακολουθιών με βάση την ομοιότητα μεταξύ τους προκειμένου να προσδιορίσει τη λειτουργική ομοιότητα μεταξύ τους. Με τη χρήση PHMM, το πρόβλημα της στοίχισης αντιμετωπίζεται ως τυχαία διαδικασία και υιοθετείται ένα μοντέλο πιθανότητας για τη λύση του (Smith, et al., 2003). Επιπλέον, παρουσιάζουν μια μοναδική μέθοδο εκπαίδευσης για εκτίμηση των παραμέτρων (ή της πιθανότητας) και επεκτείνουν το μοντέλο στοίχισης ώστε να επιτρέπονται πολυάριθμα σύνολα παραμέτρων, τα οποία επιλέγονται χρησιμοποιώντας HMM. Για την εκπαίδευση, δίνεται πρώτα μια συλλογή των ζευγαριών των ακολουθιών. Αφότου οριστούν μερικές από τις παραμέτρους, αρχίζει η εκπαίδευση για να μάθει τις παραμέτρους που θα παραγάγουν τις γενικές μέγιστες forward πιθανότητες για το σύνολο των ζευγαριών εκπαίδευσης.

2.5.5.2 Πολλαπλή στοίχιση ακολουθιών

Η πολλαπλή στοίχιση ακολουθιών (Multiple Sequence Alignment - MSA) χρησιμοποιείται συνήθως για την εύρεση των συντηρημένων περιοχών στις πρωτεϊνικές οικογένειες και για την πρόβλεψη των πρωτεϊνικών δομών. Τα Profile

HMMs, συγκεκριμένα, έχουν εφαρμοστεί με μεγάλη επιτυχία. Οι πολλαπλές στοιχίσεις ακολουθιών από μια ομάδα ακολουθιών μπορούν να γίνουν αυτόματα με τον αλγόριθμο Viterbi (Rabiner, 1989). Ο αλγόριθμος Viterbi υπολογίζει το μονοπάτι με τη μέγιστη πιθανότητα βρίσκοντας την πλέον πιθανή πορεία μέσω του HMM για κάθε ακολουθία. Μερικές δημοφιλείς εφαρμογές των profile HMMs περιλαμβάνουν το *SAM* (Hughey and Krogh, 1996; Karplus, et al., 1999), το *SLAM* (Alexandersson, et al., 2003) και το *HMMER* (Eddy, 1998).

2.5.5.3 Ανίχνευση πρωτεϊνικών ομολογιών

Στο πρόβλημα της ομολογίας των πρωτεϊνών, ο στόχος είναι να καθοριστεί ποιες πρωτεΐνες προέρχονται από έναν κοινό πρόγονο. Το μοντέλο του κοινού προγόνου υποθέτει ότι, σε κάποιο σημείο στο παρελθόν, κάθε πρωτεϊνική αλληλουχία από μια οικογένεια πρωτεϊνών έχει προέλθει από μια κοινή προγονική αλληλουχία. Δηλαδή σε κάθε αλληλουχία αμινοξέων, κάθε παρατηρηθέν αμινοξύ εμφανίζεται λόγω μιας μετάλλαξης (ή μιας σειράς μεταλλάξεων) από μια κοινή προγονική αλληλουχία αμινοξέων. Υπάρχουν πολλές πρωτεϊνικές αλληλουχίες που παρουσιάζουν πολλές ομοιότητες αλλά υπάρχουν και πολλές που παρουσιάζουν πολλές αποκλίσεις, με αποτέλεσμα η δομική και λειτουργική ομοιότητα δύσκολα να ανιχνεύεται μόνο από τα δεδομένα της αλληλουχίας.

Υπάρχουν πολλές μέθοδοι σύγκρισης αλληλουχιών που βασίζονται στα profile-HMMs. Μια από αυτές είναι το *HMMER* (Eddy, 1998) που παρέχει το απαραίτητο λογισμικό πακέτο για την ανίχνευση πρωτεϊνικών ομολογιών.

2.5.5.4 Πρόβλεψη πρωτεϊνικών δομών

Η καλή μαθηματική θεωρία των HMMs και η εκτενής εφαρμογή τους στη στοίχιση ακολουθιών, παρακίνησε τους ερευνητές να τα εφαρμόσουν και για την πρόβλεψη πρωτεϊνικών δομών (Karplus, et al., 1999; Karplus, et al., 1997). Ο προσδιορισμός των ομόλογων πρωτεϊνών είναι ιδιαίτερα σημαντικός δεδομένου ότι οι πρωτεΐνες που προέρχονται από κοινό πρόγονο μοιράζονται και παρόμοιες δομές και λειτουργίες.

Ορισμένα από αυτά τα εργαλεία είναι το *SAM-T98* (Karplus, et al., 1997), που βρίσκει τις δομές των πρωτεϊνών στηριζόμενο αποκλειστικά στις πληροφορίες της πρωτεϊνικής ακολουθίας. Το *SAM-T02*, που έχει παρόμοια λειτουργία με το SAM-T98 με τη διαφορά ότι περιλαμβάνει και δομικές πληροφορίες για τις πρωτεΐνες, και έτσι μας δίνει καλύτερα αποτελέσματα καθώς είναι πιο ευαίσθητη μέθοδος. Το *PRED-TMBB* (Bagos et al., 2004a) είναι μια μέθοδος που βασίζεται σε HMMs και βρίσκει τις διαμεμβρανικές πρωτεΐνες στηριζόμενο μόνο στην αμινοξική ακολουθία της πρωτεΐνης. Συγκεκριμένα βρίσκει τις βιαμεμβρανικές περιοχές των πρωτεϊνών που σχηματίζουν β-βαρέλια.

2.5.5.5 Εύρεση γονιδίων

Τα HMMs έχουν εφαρμοστεί και εφαρμόζονται όλο και περισσότερο για την εύρεση γονιδίων. Γενικά, ο υπολογιστικός τρόπος εύρεσης γονιδίων περιλαμβάνει εύρεση των πρωτεϊνικών δομών που προέρχονται από τα γονίδια και των διάφορων λειτουργιών τους.

Αλληλουχίες ολόκληρων χρωμοσωμάτων αποτελούνται από μια συλλογή γονιδίων που χωρίζονται η μια από την άλλη από μεγάλες αλληλουχίες από σκουπίδι DNA. Η υπολογιστική προσέγγιση για τον προσδιορισμό των γονιδίων στηρίζεται στη χρήση πολλών διαφορετικών πληροφοριών. Μέχρι τώρα, ο δημοφιλέστερος και πιο επιτυχής ανιχνευτής γονιδίων είναι το *GENSCAN* (Burge and Karlin, 1997), που είναι βασισμένο σε γενικευμένα HMMs. Άλλο ένα πληροφοριακό εργαλείο ανίχνευσης γονιδίων, που βασίζεται σε ένα μοντέλο pair HMM, είναι το *DOUBLESCAN* (Durbin, et al., 1998; Meyer and Durbin, 2002). Πέρα από το *GENSCAN* υπάρχουν και άλλα προγράμματα εύρεσης γονιδίων όπως το *SLAM* (Pachter et al., 2002), το *ROSETTA* (Batzoglou et al., 2000), το *SGP-1* (Wiehe et al., 2001), το *SGP-2* (Guigo et al., 2000) και το *TWIN-SCAN* (Korf et al., 2001).

2.6 MTDs (Mixture Transition Distribution Models)

Στο μοντέλο Mixture Transition Distribution (MTD) που αρχικά προτάθηκε από τον (Raftery, 1985) οι πιθανότητες μετάβασης του τύπου (2.3), ο οποίος μπορεί να γραφτεί για να μας διευκολύνει να εξηγήσουμε καλύτερα το μοντέλο αυτό και ως:

$$a_{s_n \ldots s_1 s_0}(i) = P(x_i = s_0 \mid x_{i-1} = s_1, x_i = s_2, \ldots, x_{i-n} = s_n) = a_{s_n \ldots s_1 s_0}$$

προσεγγίζονται από:

$$a_{s_n \ldots s_1 s_0} = P(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_{i-n}) = \sum_{j=1}^{n} \lambda_j a_{s_j s_0} \qquad (2.14)$$

Κατά συνέπεια, η επίδραση κάθε μιας (j = 1,2...n) lag εξετάζεται χωριστά και η υψηλότερης τάξης μεταβάσεις μοντελοποιούνται από ένα γραμμικό συνδυασμό μεταβάσεων 1ης τάξης. Στο μοντέλο MTD πολλαπλών πινάκων (MTDg), που προτάθηκε αργότερα από τον Raftery (Raftery, 1985), κάθε lag j συνοδεύεται από μια διαφορετική μήτρα πιθανοτήτων μετάβασης, α^j.

$$a_{s_n \ldots s_1 s_0} = P(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_{i-n}) = \sum_{j=1}^{n} \lambda_j a^j_{s_1 s_0}$$

Προκειμένου να έχουμε ένα μοντέλο με τις παραμέτρους που καθορίζουν κατάλληλα τις πιθανότητες, δηλαδή προκειμένου να έχουμε:

$$0 \le \sum_{j=1}^{n} \lambda_j a^j_{s_1 s_0} \le 1$$

πρέπει να επιβληθούν οι ακόλουθοι περιορισμοί:

$$\sum_{j=1}^{n} \lambda_j = 1$$

$$\lambda_j \ge 0$$

Καθορίζουμε στη συνέχεια την αρνητική πιθανοφάνεια, ως:

$$-\log P(x)$$

Χρησιμοποιώντας τώρα τον παραπάνω τύπο και τον τύπο 2.14, προκύπτει η σχέση:

$$\ell = -\sum_{i=n+1}^{L} \log \left(\sum_{j=1}^{n} \lambda_j a^j_{s_j s_0} \right)$$

Η πιθανότητα στον παραπάνω τύπο δεν μπορεί να μεγιστοποιηθεί αναλυτικά και μια επαναληπτική μέθοδος απαιτείται, ανάλογα με τις μερικές παραγώγους ως προς τις παραμέτρους του μοντέλου:

$$\frac{\partial \ell}{\partial a^{j}_{s\,s_0}} = -\sum_{i=n+1}^{L} \frac{\lambda_j}{\sum_{j=1}^{L} \lambda_j a^{j}_{s\,s_0}}$$

$$\frac{\partial \ell}{\partial \lambda_j} = -\sum_{i=n+1}^{L} \frac{\lambda_j}{\sum_{j=1}^{L} \lambda_j a^{j}_{s_j s_0}}$$

Ο Raftery (Raftery, 1985) βελτιστοποίησε την πιθανότητα χρησιμοποιώντας μια γραμμική ρουτίνα βελτιστοποίησης (NAG) που δεν είναι δημόσια διαθέσιμη. Ο Bercthold (Berchtold, 2001), πρότεινε έναν ευριστικό αλγόριθμο, που χρησιμοποιεί το διάνυσμα των μερικών παραγώγων, και παρείχε εφαρμογές για Matlab καθώς επίσης και μια αυτόνομη εφαρμογή που τρέχει σε Windows PCs. Το κύριο πρόβλημα του αλγορίθμου που προτείνεται ο Berthold, όπως ήδη αναφέρθηκε (Berchtold, 2001 Berchtold και Raftery, 2002) είναι ότι βελτιστοποιεί τις παραμέτρους του μοντέλου μια κάθε φορά, ενώ συγχρόνως, οι περιορισμοί των εξισώσεων, επιβάλλονται πάντα μετά από κάθε επανάληψη.

2.7 IMMs (Interpolated Markov Models)

Ένας επιπλέον τρόπος να αντιμετωπίσουμε το πρόβλημα της εκθετικής αύξησης των παραμέτρων είναι *παρεμβάλλοντας* αλυσίδες Markov διαφορετικής τάξης. Η βασική ιδέα των μεθόδων παρεμβολής (interpolation methods) είναι η εκτίμηση της πιθανότητας εμφάνισης υπακολουθιών μήκους μικρότερου από *n+1* εάν οι συχνότητες εμφάνισης ενός ολιγομερούς $x_i x_{i+1} \ldots x_{i+n}$ μεγέθους *(n+1)* δεν μπορούν να υπολογιστούν. Σε γενικές γραμμές, η παρεμβολή οδηγεί σε μια επανεκτίμηση των αρχικών τιμών των παραμέτρων. Δύο διαφορετικές τεχνικές παρεμβολής έχουν προταθεί. Η πρώτη είναι η *γραμμική παρεμβολή* (Jelinek, 1990). Τα μοντέλα με γραμμική παρεμβολή υπολογίζουν τις πιθανότητες εμφάνισης ολιγομερών διαφορετικού μήκους. Το πλεονέκτημα της παρεμβολής είναι ότι το μοντέλο μπορεί

να λάβει υπόψη του στατιστικές πιθανότητες μεγαλύτερης τάξης χωρίς τον κίνδυνο της υπερπροσαρμογής του μοντέλου κατά το στάδιο της εκπαίδευσης. Υπάρχουν όμως μερικές ανεπάρκειες της γραμμικής παρεμβολής. Ο τύπος της γραμμικής παρεμβολής (Jelinek, 1990) περιέχει μόνο *ένα* διάνυσμα των συντελεστών παρεμβολής, είτε όλες οι υπακολουθίες μήκους έως n εμφανίζονται πραγματικά στο σύνολο εκπαίδευσης είτε όχι. Επιπλέον, όλες οι παράμετροι αντιμετωπίζονται εξίσου, παρότι ο συντελεστής παρεμβολής, που καθορίζεται για μια παράμετρο με μεγάλη πιθανότητα εμφάνισης ολιγομερούς, θα έπρεπε να είναι μεγαλύτερος από το συντελεστή που καθορίζεται για μια ένα ολιγομερές που εμφανίζεται σπάνια. Λύση στα προβλήματα αυτά παρέχει ένα δεύτερο είδος παρεμβολής που καλείται *λογική παρεμβολή (rational interpolation)* (Schukat-Talamazzini, et al., 1997)

Το IMM γενικά υπολογίζει αρχικά την πιθανότητα καθενός ολιγομερούς το μήκος του οποίου κυμαίνεται από $0 \le \zeta \le n$, όπου ζ το μέγεθος το εκάστοτε ολιγομερούς που υπολογίζεται και n η τάξη του μοντέλου, που αποτελεί ταυτόχρονα το μέγιστο μέγεθος ενός ολιγομερούς. Συγκεκριμένα, το IMM χρησιμοποιεί έναν συνδυασμό όλο των πιθανοτήτων που βασίζονται σε 0,1,2…,ζ παρατηρηθέντα σύμβολα. Κατόπιν, για κάθε ολιγομερές μήκους ζ υπολογίζει ένα βάρος (=συντελεστής παρεμβολής) το οποίο χρησιμοποιεί για να συνδυάσει τις προβλέψεις που προκύπτουν από τα μοντέλα διαφορετικής τάξης. Μόλις υπολογιστούν τα βάρη, το IMM αξιολογεί τις νέες ακολουθίες υπολογίζοντας τις πιθανότητες που το μοντέλο M αναθέτει σε κάθε ακολουθία $\boldsymbol{x}$, $P(\boldsymbol{x} \mid M)$. Αυτή η πιθανότητα δίνεται από τον παρακάτω τύπο:

$$P(\boldsymbol{x} \mid M) = \sum_{i=1}^{L} \boldsymbol{IMM}_n(x_i)$$

όπου x_i είναι είναι ένα ολιγομερές που τελειώνει στη θέση i, και L είναι το μήκος της ακολουθίας. Οι πιθανότητες για ένα n-οστής τάξης Interpolated Markov Model, *IMMn(xi)*, υπολογίζονται όπως φαίνεται παρακάτω:

$$\boldsymbol{IMM}_\zeta(x_i) = \lambda_\zeta(x_{i\text{-}1}) \cdot P_\zeta(x_i) + [1 - \lambda_\zeta(x_i - 1)] \cdot \boldsymbol{IMM}_{\zeta\text{-}1}(x_i)$$

όπου $\lambda_\zeta(x_{i\text{-}1})$ είναι το αριθμητικό βάρος που συνδέεται με το ολιγομερές μήκους ζ που τελειώνει στη θέση i στην ακολουθία $\boldsymbol{x}$ και $P_\zeta(x_i) = P(x_i \mid x_{i-1} \ldots x_{i-\zeta})$ είναι η

εκτίμηση της πιθανότητας, που προκύπτει από τα δεδομένα εκπαίδευσης και το μοντέλο ζ τάξης αποκτηθείς από τα στοιχεία κατάρτισης της πιθανότητας της βάσης. Κατά συνέπεια, το σκορ ενός ολιγομερούς που προκύπτει από ένα n-οστής τάξης IMM είναι ένας γραμμικός συνδυασμός των προβλέψεων που προκύπτουν από όλα τα μοντέλα από τάξης *n*, τάξης *n*-1 έως και μηδενικής τάξης.

Τα IMMs, όπως παρουσιάστηκαν παραπάνω εφαρμόστηκαν στη ανάλυση μικροβιακών ακολουθιών από τον Salzberg και τους συνεργάτες του (Salzberg et al., 1998).

2.8 VLMCs (Variable Length Markov Chains).

Ένας άλλος τρόπος για να αντιμετωπίσουμε το πρόβλημα της εκθετικής αύξησης των παραμέτρων είναι οι αλυσίδες Markov μεταβλητού μήκους, οι οποίες είναι μια γενίκευση των απλών αλυσίδων Markov (Bühlmann and Wyner, 1999; Ron et al., 1996). Ο Rissanen (Rissanen, 1983) ήταν αυτός που εισήγαγε την έννοια των στοχαστικών αλυσίδων με μεταβλητή μνήμη (variable length memory) ως ένα τρόπο για τη συμπίεση δεδομένων. Ονόμασε αυτό το πρότυπο finitely generated source ή tree machine. Πρόσφατα αυτό το μοντέλο έγινε δημοφιλές στη Στατιστική βιβλιογραφία με το όνομα Variable Length Markov Chains (VLMCs) από τους Bühlmann και Wyner (Bühlmann and Wyner, 1999). Στη βιοπληροφορική είναι γνωστό και ως Probabilistic Suffix Tree - PST (Bejerano and Yona, 2001).

Τα VLMCs επιτρέπουν σε παραμέτρους με μεταβλητό μήκος πλαισίου να ανιχνεύσουν όλες τις σημαντικές ακολουθίες συμβόλων με μεγάλη ακρίβεια. Τα VLMCs μπορούν να αντιμετωπιστούν είτε ως *πιθανολογικά αυτόματα (stochastic automata)* με μια κατάσταση ανά πλαίσιο, είτε υπό μορφή *δέντρων πλαισίων (context trees).* Ένα δέντρο πλαισίων είναι μια ακυκλική γραφική παράσταση της οποίας οι κόμβοι ποικίλλουν σε βαθμό μεταξύ μηδενός και *k.* Τα βέλη που οδηγούν από έναν κόμβο σε έναν άλλο ονομάζονται με σύμβολα $x \in k$, και κάθε σύμβολο επιτρέπεται να ονομάσει το πολύ-πολύ ένα βέλος. Οι κόμβοι καθορίζονται από *ζευγάρια* $(x, P(\cdot \mid \hat{x}))$ όπου $\hat{x}$ είναι η σειρά των ετικετών στην πορεία από τον τελευταίο κόμβο u_p μέχρι τον κενό κόμβο-ρίζα e, και αντιπροσωπεύει ένα πλαίσιο, δηλ. μια

συμβολοσειρά που με τη σειρά της αντιπροσωπεύει ένα ολιγομερές, που εμπεριέχεται στο μοντέλο. Ως $P(\cdot \mid \hat{x})$ ορίζουμε την κατανομή πιθανότητας του αλφάβητου, που ορίζεται για το συγκεκριμένο πλαίσιο. Αν θέλαμε να παρουσιάσουμε ένα VLMC σαν μια απλή αλυσίδα Markov, θα μπορούσαμε να πούμε ότι ένα πλήρες δέντρο ύψους n είναι μια αλυσίδα Markov n-οστής τάξης. Ένα παράδειγμα ενός δέντρου πλαισίων δίνεται στο σχήμα της εικόνας 2.5.

Η πιθανότητα μιας ακολουθίας. Για να εξηγήσουμε πώς ένα VLMC χρησιμοποιείται για να υπολογίσει την πιθανότητα μιας ακολουθίας, χρησιμοποιούμε το δέντρο στο σχήμα 2.5 για να υπολογίσουμε την πιθανότητα της ακολουθίας 01011. Η πιθανότητα υπολογίζεται εφαρμόζοντας τον τύπο 2.3. Για κάθε σύμβολο στην ακολουθία, καθορίζουμε το μέγιστο πλαίσιο σύμφωνα με τα σύμβολα που διαπερνάμε, πηγαίνοντας ταυτόχρονα αριστερά στην ακολουθία και κάτω στο δέντρο. Σταματάμε όταν φτάνουμε σε ένα φύλλο ή όταν φτάνουμε στο πρώτο σύμβολο της ακολουθίας και υπολογίζουμε την πιθανότητα για το τρέχον σύμβολο σε αυτό το φύλλο. Για κάθε σύμβολο, πρέπει να αρχίσουμε ξανά από τη ρίζα για να μπορέσουμε να καθορίσουμε όπως προηγουμένως το κατάλληλο πλαίσιο. Στο παράδειγμα μας (Εικόνα 2.5) αυτό οδηγεί στους παρακάτω υπολογισμούς:

$$P_T(01011) = P(0 \mid e) \cdot P(1 \mid 0) \cdot P(0 \mid 01) \cdot P(1 \mid 0) \cdot P(1 \mid 01) = 0.2 \cdot 0.7 \cdot 0.6 \cdot 0.7 \cdot 0.4$$

Για το δέντρο στο σχήμα 2.5, το πλαίσιο 0101 είναι ισοδύναμο με το 01. Αυτό υποδηλώνει μια εναλλακτική άποψη σχετικά με τα δέντρα πλαισίου, δηλαδή ότι ένα δεδομένο δέντρο μέγιστου ύψους n καθορίζει μια συνάρτηση προβολής c στο μέγιστο πλαίσιο $\hat{x}_1^{L-1}$ σε οποιοδήποτε σημείο i σε μια ακολουθία με:

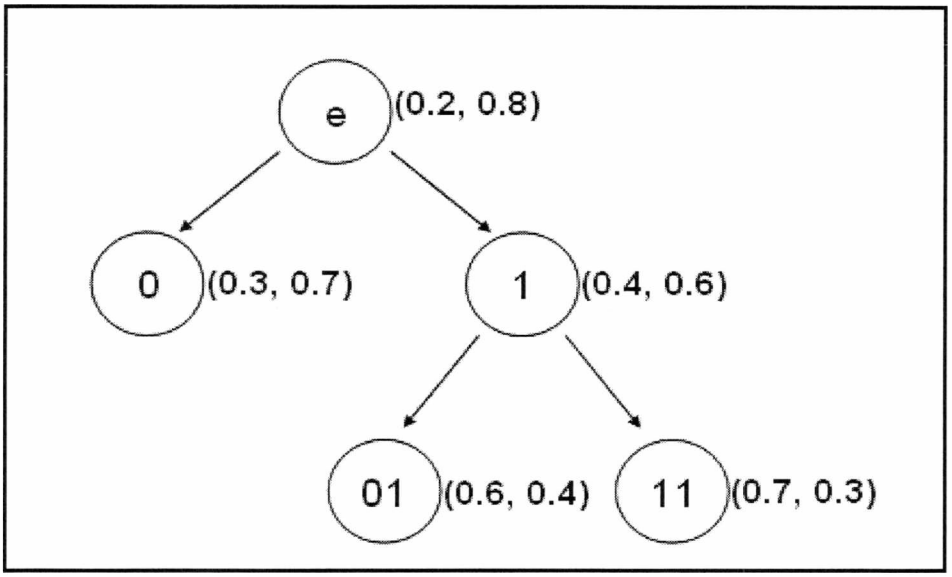
$$c : \begin{cases} k^* \rightarrow \bigcup_{i=0}^{n} k^m \\ \hat{x}_1^{L-1} \mapsto \hat{x}_{L-l}^{L-1} \end{cases}, \qquad (2.15)$$
$$l = \min\{\ h; P(x_i \mid \hat{x}_{L-h}^{L-1}), \forall x_t \in k, 0 \le h \le n\},$$

Χρησιμοποιώντας το c, η πιθανότητα μιας ακολουθίας που υπολογίζεται από ένα VLMC μπορεί τώρα να γραφτεί ως:

$$P(\boldsymbol{x}) \approx \prod_{i=1}^{L} P(x_i \mid c(x_1^{L-1})) \tag{2.16}$$

σε αναλογία με την *n*-οστής τάξης αλυσίδα Markov στον τύπο 2.9. Δεδομένου ότι οι κατανομές πιθανοτήτων περιλαμβάνονται όχι μόνο σε ένα φύλλο, αλλά και σε όλους τους εσωτερικούς κόμβους, δεν χρειάζεται να ανησυχήσουμε για την αρχή των ακολουθιών όπου το μέγιστο πλαίσιο μπορεί να μην είναι ακόμα διαθέσιμο.

Κάτι ανάλογο με τα δέντρα πλαισίων είναι και τα Πιθανολογικά Αυτόματα Επιθήματος (Probabilistic Suffix Automata-PSA), που είναι μια υποκατηγορία των Πιθανολογικών Πεπερασμένων Αυτομάτων (PFA). Εδώ, το πλαίσιο αντιπροσωπεύεται από μια κατάσταση του αυτομάτου, με τον περιορισμό ότι καμία ετικέτα δεν είναι ένα επίθημα κάποιας άλλης ετικέτας, και ότι από κάθε κατάσταση, υπάρχει μια μετάβαση για όλα τα $x \in k$. Για κάθε δέντρο πλαισίων, υπάρχει μια ισοδύναμη αναπαράσταση ως PFA.



Εικόνα 2.5 : Παράδειγμα ενός δέντρου πλαισίων με αλφάβητο {0, 1}. Οι πιθανότητες των συμβόλων δίνονται στις παρενθέσεις δίπλα στον κάθε κόμβο του δέντρου, όπου και αποθηκεύονται. Κάθε βέλος που πηγαίνει αριστερά αντιστοιχεί σε μετάβαση στο 0, ενώ κάθε βέλος που πηγαίνει δεξιά αντιστοιχεί σε μετάβαση στο 1.

Εκτίμηση του VLMCs. Σε αντίθεση με τις απλές αλυσίδες του Markov, τα VLMCs μας επιτρέπουν να συμπεριλάβουμε μεγάλα τμήματα υπακολουθιών, που έχουν μεγάλα πλαίσια, χωρίς το μοντέλο μας να υποφέρει από εκθετική αύξηση του

αριθμού των παραμέτρων. Το πρόβλημα προσανατολίζεται έπειτα στο πώς θα μπορέσουμε να αποφασίσουμε με έναν αυτοματοποιημένο τρόπο ποιά πλαίσια πρέπει να συμπεριληφθούν στο μοντέλο και ποιά όχι, δηλ. πως θα βρούμε το καλύτερο μοντέλο, εκείνο που περιγράφει καλύτερα την κατηγορία που θέλουμε. Το να βρούμε το καλύτερο μοντέλο μεταξύ όλων των πιθανών μοντέλων θα μας δημιουργούσε πάλι εκθετικό πρόβλημα αναζήτησης, το οποίο σαφώς δεν θέλουμε. Ο Ron και οι συνεργάτες του (Ron, et al., 1996) έδειξαν ότι ο αλγόριθμος της εικόνας 2.6 (Ron-Singer-Tishby (RST) algorithm) είναι σε θέση να εκπαιδεύσει ένα δέντρο πλαισίων σύμφωνα με το PAC (probably approximately correct) παράδειγμα εκπαίδευσης (Mitchell, 1997): Με μια πιθανότητα 1- δ όπου η κατανομή του εκπαιδευμένου VLMC M^L θα έχει μια απόκλιση Kullback-Leibler (KL) DKL από το σωστό VLMC M^C το περισσότερο ε ($0 \le \varepsilon, \delta \le 1$):

$$\frac{1}{L} D_{KL}[M^{L}][M^{C}] \le \varepsilon, \qquad L > 0$$

Με αυτόν τον τρόπο, η ανισότητα ισχύει για όλα τα πιθανά μήκη ακολουθίας L και ομαλοποιείται από το μήκος, καθώς η πιθανότητα μειώνεται όταν το μήκος της ακολουθίας αυξάνεται. Η απόκλιση KL μεταξύ των δύο κατανομών P και Q σε ένα σύνολο παρατηρήσεων X καθορίζεται από:

$$D_{KL}[P][Q] := \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)}$$

εδώ, το σύνολο των παρατηρήσεων είναι k^L.

Ο αλγόριθμος αυξάνει σταδιακά το δέντρο μέχρι το μέγιστο ύψος n να επιτευχθεί. Κόμβοι που αντιπροσωπεύουν ένα πλαίσιο $\hat{x}$ προστίθεται εάν τηρούν τις ακόλουθες συνθήκες:

1. Η εκτιμώμενη πιθανότητας $\bar{P}(\hat{x})$ είναι μεγαλύτερη από μια ελάχιστη πιθανότητα.
2. Για τουλάχιστον ένα σύμβολο x, η δεσμευμένη πιθανότητα που αντιστοιχεί στο $\hat{x}$ να είναι αρκετά διαφορετική από τον κόμβο-πατέρα με πλαίσιο suffix($\hat{x}$), ή το $\hat{x}$ να είναι ένας κόμβος, στην πορεία από το e σε έναν άλλο κόμβο, που είναι σημαντικά διαφορετικός.

Η πρώτη συνθήκη εμποδίζει το δέντρο από να παρουσιάσει εκθετική αύξηση, καθώς μόνο οι κόμβοι με μια ελάχιστη πιθανότητα εξετάζονται. Ενώ η τελευταία συνθήκη εξασφαλίζει ότι όλοι οι κόμβοι που είναι σημαντικά διαφορετικοί προστίθενται πράγματι στο δέντρο, ακόμα κι αν μερικοί προγονικοί κόμβοι είναι σχεδόν ίδιοι. Σαν μέτρο διαφοράς, χρησιμοποιείται η διαφορά μεταξύ των πιθανοτήτων ενός κόμβου-παιδί και του αντίστοιχου κόμβου-πατέρα. Ο αλγόριθμος έχει τις ακόλουθες πέντε παραμέτρους:

N : μέγιστο ύψος δέντρου,

Pmin : η ελάχιστη πιθανότητα ενός πλαισίου που εξετάζεται,

r : μέτρο της διαφορά μεταξύ των πιθανοτήτων ενός κόμβου-παιδί και του αντίστοιχου κόμβου-πατέρα

γmin : παράγοντας κανονικοποίησης.

α : ελάχιστη διαφορά οποιουδήποτε $P(x \mid \hat{x})$ από τον παράγοντα βελτιστοποίησης.

<table>
<tr><td colspan="3">Αρχικοποίηση: $\overline{M} = \{e\},\ \overline{S} = \{x \in k \mid \widetilde{P}(x) \geq P_{\min}\}$</td></tr>
<tr><td colspan="3">Δημιουργώντας το δέντρο πλαισίων: ΕΝΩ $S \neq 0$</td></tr>
<tr><td></td><td colspan="2">Διάλεξε & σβήσε ένα τυχαίο πλαίσιο: $\hat{x}$ από $\overline{S}$</td></tr>
<tr><td></td><td>ΕΑΝ</td><td>$\exists\ x \in k:\ \widetilde{P} = (x \mid \hat{x}) \geq (1+a)\gamma_{\min}\ \&\ \dfrac{P(x \mid \hat{x})}{\widetilde{P}(x \mid suffix(\hat{x}))} \geq r$ ή $\leq 1/r$</td></tr>
<tr><td></td><td>ΤΟΤΕ</td><td>Πρόσθεσε $\hat{x}$ στο $\overline{M}$, μαζί με όλα τα $\hat{x}'$ που είναι στο μονοπάτι από e έως $\hat{x}$</td></tr>
<tr><td></td><td>ΕΑΝ</td><td>$\mid \hat{x} \mid < n$</td></tr>
<tr><td></td><td>ΤΟΤΕ</td><td>$\forall x' \in k$</td></tr>
<tr><td></td><td></td><td>EAN: $\widetilde{P}(x'\hat{x}) \geq P_{\min}$
TOTE: $\overline{S} = \overline{S} \cup \{x'\hat{x}\}$</td></tr>
<tr><td colspan="3">$\forall \hat{x} \in \overline{M}$
$\hat{P}(x \mid \hat{x}) = (1 - \mid k \mid \gamma_{\min})\widetilde{P}(x \mid \hat{x}) + \gamma_{\min}$</td></tr>
</table>

Εικόνα 2.6: Ο αλγόριθμος RST για την εκπαίδευση του δέντρου πλαισίων $\hat{M}$ (Kulicke, 2000).

Ο Ron και οι συνεργάτες του (Ron, et al., 1996) απέδειξαν ότι ο αλγόριθμος εκπληρώνει πράγματι το κριτήριο PAC που περιγράφεται παραπάνω. Παρότι στην πράξη, το ακριβές μέγεθος του συνόλου εκπαίδευσης που απαιτείται δεν είναι γνωστό, οι Bejerano και Yona (Bejerano and Yona, 2001) δείχνουν ότι μια λογική επιλογή παραμέτρων μπορεί να οδηγήσει σε καλά αποτελέσματα στην ταξινόμηση πρωτεϊνικών ακολουθιών. Επεκτείνουν επίσης τον αλγόριθμο ώστε να συμπεριλάβουν επιπλέον γνώση σχετικά με τις πρωτεΐνες που το εφαρμόζουν. Στην εφαρμογή που πραγματοποίησαν σε οικογένειες πρωτεϊνών έθεσαν το α ίσο με μηδέν. Επιπρόσθετα σ' αυτήν την απλοποίηση, αναφέρουμε την απλοποίηση του Jeffrey σαν προσέγγιση βελτιστοποίησης, η οποία έχει σαν αποτέλεσμα ένα σύνολο τριών παραμέτρων: ύψος, κατώφλι, και ελάχιστη πιθανότητα. Μια άλλη εναλλακτική προσέγγιση για την εκπαίδευση των VLMCs παρουσιάστηκε και από τους Buhlmann και Wyner (Buhlmann and Wyner, 1999), οι οποίοι πρότειναν έναν αλγόριθμο εκπαίδευσης (αλγόριθμος BW) όπου η απόφαση για το αν θα επεκταθεί το δέντρο ή όχι δεν βασίζεται στη διαφορά μιας παραμέτρου, αλλά σε ολόκληρη την κατανομή.

Cross-validation εκτίμηση του βέλτιστου κατωφλιού. Και ο αλγόριθμος BW και ο αλγόριθμος RST δεν παρέχουν μια εκτίμηση του κατωφλίου K και r αντίστοιχα, που κυρίως καθορίζουν το καλύτερο μοντέλο VLMC, που αντιστοιχεί στο σύνολο εκπαίδευσης. Αλλά όπως ο Bühlmann (Bühlmann, 2000) επισημαίνει, το μοντέλο που είναι πιο κοντά στην αληθινή κατανομή μπορεί να μην είναι το επιθυμητό. Συχνά, ο στόχος είναι να βρεθεί ένα μοντέλο που να βελτιστοποιεί το σχετικό σφάλμα ή μια αντικειμενική συνάρτηση, όπως η ML και η MMI (βλέπε 2.4 ενότητα). Ένας εφικτός τρόπος να βρει κανείς ένα ιδανικό δέντρο είναι με το να περιορίσει την αναζήτηση στα δέντρα που παράγονται από τα διάφορα κατώφλια (K ή r), και να επιλέξει κανείς αυτό για το οποίο επιτυγχάνεται η βέλτιστη τιμή. Θα πρέπει να σημειωθεί ότι αυτή η μέθοδος δεν εγγυάται ότι θα μας δώσει το καλύτερο δέντρο, καθώς μπορεί να υπάρχουν και καλύτερα δέντρα που δεν παράγονται από ένα γενικό κατώφλι για όλα τα φύλλα του δέντρου.

Επομένως, εξακολουθούμε να αντιμετωπίζουμε το πρόβλημα του να βρούμε μια αντικειμενική συνάρτηση για τις διαφορετικές τιμές των κατωφλίων (K ή r και Pmin). Ο Bühlmann (Bühlmann, 2000) προσπάθησε να δώσει μια λύση στο

πρόβλημα εφαρμόζοντας ένα είδος δειγματοληψίας κατά τη οποία: υπολογίζει ένα αρχικό δέντρο, το χρησιμοποιεί για να παραγάγει έναν μεγάλο αριθμό ανεξάρτητων δειγμάτων, εκπαιδεύει τα δέντρα με κάθε ένα από εκείνα τα δείγματα και υπολογίζει το σχετικό σφάλμα για τα δέντρα για μια συγκεκριμένη τιμή του κατωφλίου (K στη περίπτωση του αλγόριθμου BW).

2.8.1 PST

2.8.1.1 Εισαγωγή

Το PST (Bejerano and Yona, 2001) μπορεί να χρησιμεύσει ως ένα λογισμικό εργαλείο πρόγνωσης πρωτεϊνικών ακολουθιών, και ανίχνευσης συντηρημένων περιοχών (ενδεχομένως λειτουργικά ή δομικά σημαντικές) μέσα σε πρωτεϊνικές ακολουθίες. Η μέθοδος εφαρμόστηκε στη βάση δεδομένων Pfam, που έχει πρωτεϊνικές οικογένειες, δίνοντας πάρα πολύ καλά αποτελέσματα. Οι αξιολογήσεις δείχνουν ότι το μοντέλο PST ανιχνεύει πολύ καλύτερα τις ακολουθίες από τις υπόλοιπες κατά ζεύγη μεθόδους στοίχισης όπως το Gapped-BLAST, και είναι σχεδόν όσο ευαίσθητο όσο ένα HMM, αλλά πολύ γρηγορότερο απ' αυτό.

Για να μοντελοποιήσουμε ένα μοτίβο, μια περιοχή ή μια πρωτεϊνική οικογένεια, πολλές μέθοδοι αρχίζουν κάνοντας πολλαπλή στοίχιση των ακολουθιών. Η πολλαπλή στοίχιση όμως παρουσιάζει κάποια σημαντικά μειονεκτήματα , καθώς το να κάνει κανείς στοίχιση πολλών ακολουθιών είναι πολύ χρονοβόρο και καθόλου εύκολο. Επιπλέον οι υπάρχουσες μέθοδοι εφαρμόζουν ευρεστικούς αλγορίθμους για να κάνουν πολλαπλή στοίχιση των ακολουθιών, οι όποιοι όμως δεν εγγυώνται την εύρεση της βέλτιστης στοίχισης (Bates and Sternberg, 1991; Samudrala and Moult, 1997). Τέλος, ακόμα και η βέλτιστη στοίχιση που μας δίνουν, πολλές φορές, δεν είναι βιολογικά ακριβής. Όταν στις πρωτεϊνικές ακολουθίες που σχετίζονται μεταξύ τους είναι δύσκολο να προσδιοριστεί η συντηρημένη περιοχή ή δεν μπορεί εύκολα να διακριθεί από το θόρυβο ή όταν ο αριθμός των δειγμάτων-πρωτεϊνών που χρησιμοποιούνται για να γίνει η πολλαπλή στοίχιση είναι μικρός, τότε η αξιοπιστία της είναι αμφισβητούμενη και το μοντέλο που προκύπτει (consensus pattern, profile ή HMM) μπορεί να μην είναι τόσο καλό. Η σκιαγράφηση των ορίων των περιοχών ενδιαφέροντος καθιστούν το πρόβλημα ακόμα δυσκολότερο. Επομένως, πλήρως

αυτοματοποιημένες μέθοδοι, βασισμένες σε πολλαπλή στοίχιση ακολουθιών δεν είναι πολύ αξιόπιστες για την ανίχνευση συντηρημένων περιοχών και ταξινόμηση πρωτεϊνών. Εντούτοις, το μέγεθος των σύγχρονων βάσεων δεδομένων κάνουν επιτακτική την ανάγκη ανάπτυξης αυτοματοποιημένων μεθόδων υψηλής απόδοσης.

Εδώ παρουσιάζουμε μια εναλλακτική προσέγγιση για την ανίχνευση συντηρημένων περιοχών, βασισμένη στα πιθανολογικά δέντρα (PSTs-probabilistic suffix trees), που αρχικά παρουσιάστηκε από τον Ron και τους συνεργάτες του (Ron *et al.,* 1996). Το μοντέλο στηρίζεται στην εύρεση μικρών σχετικά περιοχών (τμημάτων) που εμφανίζονται κατά μήκος μιας πρωτεϊνικής ακολουθίας και επαναλαμβάνονται σε όλες τις πρωτεΐνες που το μοντέλο δέχεται σαν είσοδο, ανεξάρτητα από τη σχετική θέση που οι περιοχές αυτές έχουν στις διάφορες πρωτεΐνες. Αυτές οι περιοχές (τμήματα) φέρουν κάποιες στατιστικές ιδιότητες που τις κάνουν να ξεχωρίζουν. Ειδικότερα, φέρουν μια κατανομή πιθανότητας βάση της οποίας καθορίζεται η εμφάνιση του επόμενου συμβόλου αμέσως μετά το τμήμα (περιοχή). Η ύπαρξη αυτής της κατανομής υπονοεί ότι ένα χαρακτηριστικό γνώρισμα είναι κοινό σε ένα μεγάλο υποσύνολο των πρωτεϊνών. Αυτό το χαρακτηριστικό γνώρισμα, που καλείται *βραχεία μνήμη* (*short memory),* και είναι κοινό σε πολλά σύνολα πρωτεϊνών, δείχνει ότι για μια συγκεκριμένη ακολουθία η κατανομή πιθανότητας για το επόμενο σύμβολο δεδομένου της προηγούμενης υπακολουθίας μπορεί να υπολογιστεί με αρκετά μεγάλη ακρίβεια παρατηρώντας τα τελευταία n σύμβολα της συγκεκριμένης υπακολουθίας.

Αυτή η παρατήρηση έχει οδηγήσει και πιο πριν στην ιδέα της μοντελοποίησης πολλών συνόλων πρωτεϊνών μέσω της χρήσης Αλυσίδων Markov τάξης n (όπου n είναι το μήκος της μνήμης του μοντέλου), ή μέσω της χρήσης Hidden Markov Models, που είναι πιο σύνθετα αλλά πιο ακριβή ως προς την ανίχνευση πιο λεπτομερών χαρακτηριστικών γνωρισμάτων. Αυτές οι δύο κατηγορίες, αν και ικανές για μοντελοποίηση πολλών πρωτεϊνικών συνόλων κατά τρόπο αποδοτικό, παρουσιάζουν κάποια βασικά μειονεκτήματα. Το μοντέλο Markov πάσχει από εκθετική αύξηση του αριθμού των καταστάσεων (παραμέτρων), όταν η μνήμη του μοντέλου αυξάνει πολύ, και από κακή εκτίμηση όταν πρόκειται για πολύ μικρής τάξεως μοντέλο. Τα HMMs απ' την άλλη πάσχουν από τα γνωστά προβλήματα

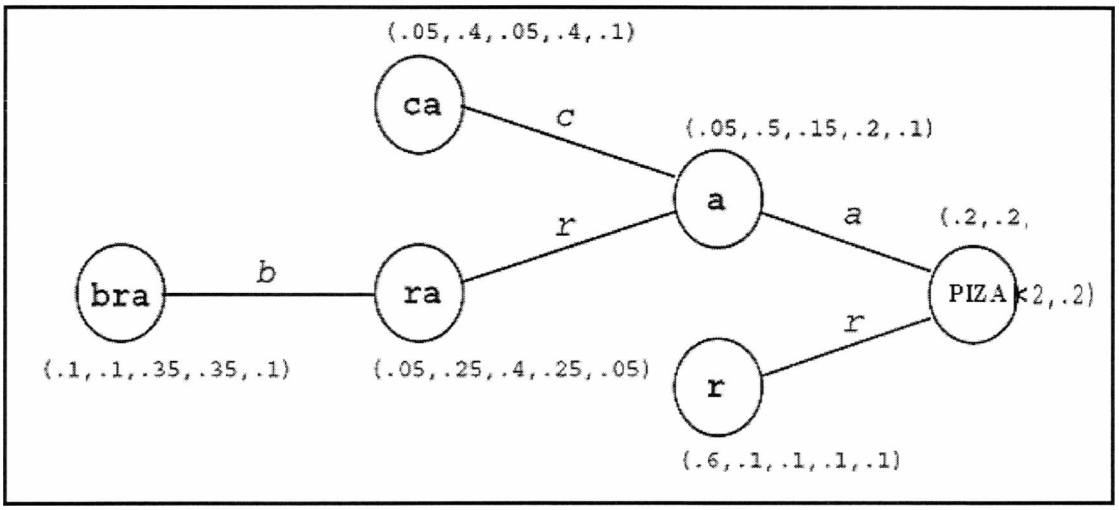
δυσκολίας εκπαίδευσης (Abe and Warmuth, 1992; Gillman and Sipser, 1994) και συνεπώς, το παραγόμενο μοντέλο δεν είναι σίγουρα το βέλτιστο. Τα πιθανοθεωρητικά δέντρα επιθεμάτων (Probabilistic Suffix Tree - PST) εμπνέονται από τον ίδιο συλλογισμό. Εντούτοις, αυτή η κατηγορία μοντέλων μπορεί να μοντελοποιήσει πολλά σύνολα πρωτεϊνών χρησιμοποιώντας ένα λογικό ποσό μνήμης. Το PST είναι ουσιαστικά ένα μοντέλο Markov μεταβλητού μήκους και το χαρακτηριστικό του, που αποτελεί και το πλεονέκτημα του, είναι ότι το μήκος μνήμης του είναι μεταβλητό, φαινόμενα που παρατηρείται και σε πολλά είδη πρωτεϊνών μέθοδος είναι απλή να εφαρμοστεί. Οι ακολουθίες που θα χρησιμοποιηθούν για να εκπαιδεύσουν το PST δεν χρειάζεται να έχουν στοιχηθεί και μπορεί να έχουν οποιοδήποτε μήκος. Η μέθοδος είναι αυτόματη, και μπορεί επιπλέον να εφαρμοστεί, χωρίς να προϋποθέτει οποιαδήποτε άλλη βιολογική πληροφορία. Παρά την απλότητά στην εφαρμογή του, το μοντέλο αυτό έχει εφαρμοστεί με μεγάλη επιτυχία σε οικογένειες πρωτεϊνών. Παρακάτω, περιγράφουμε πρώτα το μοντέλο και κατόπιν τον αλγόριθμο εκμάθησής του.

2.8.1.2 Θεωρία

Οι ορισμοί και οι εφαρμοσμένοι αλγόριθμοι είναι παραλλαγές του αλγορίθμου εκμάθησης που παρουσιάστηκε αρχικά από τον Ron και τους συνεργάτες του (Ron et al., 1996). Ένα PST πέρα από ένα αλφάβητο είναι ένα δέντρο, του οποίου οι κόμβοι ποικίλουν σε αριθμό μεταξύ του μηδενός (για τα φύλλα) και του μεγέθους του αλφάβητου. Κάθε άκρη στο δέντρο περιέχει από ένα σύμβολο του αλφάβητου, έτσι ώστε κανένα σύμβολο να μην αντιπροσωπεύεται από περισσότερες από μια άκρες, που διακλαδίζονται με οποιοδήποτε κόμβο (ως εκ τούτου ο βαθμός κάθε κόμβου οριοθετείτε από το μέγεθος του αλφάβητου). Οι κόμβοι του δέντρου περιέχουν από μια συμβολοσειρά, η οποία είναι αυτό που παράγεται αν ακολουθήσει κανείς το δέντρο ξεκινώντας από εκείνο τον κόμβο με κατεύθυνση προς τα πάνω έως τη ρίζα. Σε κάθε κόμβο ορίζεται ένα διάνυσμα κατανομής πιθανότητας για το αλφάβητο. Όταν το PST χρησιμοποιείται για να ανίχνευση τα τμήματα που εμφανίζονται με μεγαλύτερη πιθανότητα μέσα σε μια ακολουθία (δηλ. τα τμήματα με υψηλή πιθανότητα), αυτός ο πίνακας κατανομής πιθανοτήτων μπαίνει στο παιχνίδι. Ορίζει

τις πιθανότητες καθενός συμβόλου που παράγει το δέντρο, δεδομένου ότι η μεγαλύτερη υπακολουθία συμβόλων που έχει παρατηρηθεί πριν από αυτή αντιστοιχεί στη συμβολοσειρά που αποτελεί την "ετικέτα" του συγκεκριμένου κόμβου. Ένα παράδειγμα ενός PST δίνεται στο σχήμα της εικόνας .

Πρέπει να τονίσουμε εδώ ότι το PST διαφέρει από το δέντρο επιθεμάτων (suffix tree) (Gusfield, 1997). Σε ένα κλασσικό δέντρο ο πατέρας ενός κόμβου (bra) θα ήταν κόμβος (br), ενώ στο PST ο πατέρας ενός κόμβου (bra) είναι ένας κόμβος χωρίς το πρώτο σύμβολο (ra). Εδώ ο πατέρας του κόμβου(bra) είναι ο κόμβος (*ra*). Για να διευκρινίσουμε τη σχέση μεταξύ των δύο αυτών δομών δεδομένων αρκεί να πούμε ότι: ο σκελετός (κόμβοι, άκρες και ετικέτες) ενός PST για μια δεδομένη ακολουθία εισόδου είναι απλά ένα υποδέντρο του κλασσικού δέντρου με τη διαφορά ότι στο PST έχουμε *αντιστροφή* εκείνης της συμβολοσειράς.



Εικόνα 2.7 Σχηματική αναπαράσταση ενός PST. Έχει χρησιμοποιηθεί για παράδειγμα η ακολουθία abracadabra.

2.8.1.3 Δημιουργώντας το PST

Καταρχήν, ορίζουμε ως *n* το μήκος της μνήμης του PST (δηλ. το μέγιστο μήκος μιας πιθανής σειράς στο δέντρο). Έτσι ορίζουμε ότι μπορούμε να έχουμε υπακολουθίες από μήκος 1 έως μήκος *n*, ξεκινώντας με υπακολουθίες που αποτελούνται από μόνο ένα γράμμα, και σταματώντας την περαιτέρω επέκταση τους όταν η πιθανότητα μιας υπακολουθίας έχει κατέβει κάτω από ένα ορισμένο

κατώτατο όριο *(Pmin)* αν το μήκος της έχει φτάσει το μέγιστο μήκος *n*. Με την ύπαρξη του κατωφλίου *Pmin* αποφεύγομαι το να έχουμε ένα εκθετικά μεγάλο (*στο n*) διάστημα αναζήτησης.

Στην αρχή της αναζήτησης το PST αποτελείται από έναν ενιαίο κόμβο ρίζα. Κατόπιν, για κάθε υπακολουθία που αποφασίζουμε να εξετάσουμε, ελέγχουμε εάν υπάρχει κάποιο σύμβολο στο αλφάβητο για το οποίο η πιθανότητα παρατήρησης αυτού του συμβόλου αμέσως μετά από τη συγκεκριμένη υπακολουθία δεν είναι αμελητέα, και είναι επίσης σημαντικά διαφορετική από τη πιθανότητα παρατήρησης του ίδιου συμβόλου αμέσως μετά από τη συμβολοσειρά που προκύπτει από τη διαγραφή του αριστερότερο συμβόλου από την υπακολουθία μας. Όποτε αυτές οι δύο συνθήκες ικανοποιούνται, η υπακολουθία, και όλοι οι απαραίτητοι κόμβοι στην πορεία της, προστίθεται στο PST.

Ο λόγος για τον οποίο αυτή η διαδικασία περιλαμβάνει δύο στάδια (πρώτα καθορίζει όλους τους πιθανούς κόμβους που πρέπει να εξεταστούν, και έπειτα εξετάζει τον κάθε κόμβο ξεχωριστά), προκύπτει από τη φύση των PSTs. Ένα φύλλο σε ένα PST κρίνεται άχρηστο εάν ο τρόπος πρόβλεψης του είναι ίδιος (ή σχεδόν ίδιος) με τον κόμβο των γονέων του. Εντούτοις, αυτό από μόνο του δεν αρκεί ώστε να μην εξεταστούν οι κόμβοι-γιοι περαιτέρω, ψάχνοντας ταυτόχρονα και για περιοχές με υψηλή πιθανότητα. Επομένως, μπορεί, και συμβαίνει διαδοχικοί κόμβοι στο εσωτερικό PST να είναι σχεδόν ίδιοι.

Τέλος, οι προβλεπόμενες συμβολοσειρές των κόμβων προστίθεται στο προκύπτων PST, που χρησιμοποιεί την κατάλληλη εμπειρική δεσμευμένη πιθανότητα. Έπειτα αυτές οι πιθανότητες κανονικοποιούνται, χρησιμοποιώντας μια κλασσική τεχνική, έτσι ώστε κανένα σύμβολο είναι να μην είναι απολύτως αδύνατο να εμφανιστεί μετά από μια υπακολουθία.

Ορισμοί: Έστω k το αλφάβητο και $r^1, r^2, ..., r^m$ το σύνολο δειγμάτων m ακολουθιών, όπου το μήκος της i-οστης (i=1…m) ακολουθίας είναι l_i (π.χ $r^i = r_1^i r_2^i ... r_j^i \quad r_j^i \in k$).

Ορίζουμε αρχικά την εμπειρική πιθανότητα εμφάνισης μιας υπακολουθίας s στο σύνολο δειγμάτων ως το σύνολο των φορών που παρατηρήθηκε η υπακολουθία

αυτή στο σύνολο των δειγμάτων προς το μέγιστο πιθανό αριθμό που η υπακολουθία αυτή θα μπορούσε να παρατηρηθεί, δεδομένου και του μεγέθους του συνόλου δειγμάτων. Δεδομένης μια υπακολουθίας s μήκους l (s=s_1 s_2 ...s_l) η εμπειρική πιθανότητα εμφάνισης μιας υπακολουθίας s δίνεται από τον παρακάτω τύπο:

$$\chi_s^{i,j} = \begin{cases} 1 \quad if \quad s_1 s_2 ... s_l = r_j^i r_{j+1}^i ... r_{j+(l-1)}^i \\ 0 \quad \text{διαφορετικά} \end{cases}$$

για κάθε i=1…m και j=1…l_i -(l-1). Κάθε μεταβλητή $\chi_s^{i,j}$ παίρνει την τιμή 1 αν και μόνο αν η ακολουθία (συμβολοσειρά) s είναι μια υπακολουθία r^i που ξεκινάει από τη θέση j.

$$\chi_s = \sum_{i,j} \chi_s^{i,j}$$

Ο συνολικός αριθμός των υπακολουθιών μήκους $|s|$=l στο σύνολο $\{r^i\}$ είναι:

$$N_{|s|} = \sum_{i \;\; s.t. \;\; l_i \geq l} (l_i - (l-1))$$

Ορίσουμε την εμπειρική πιθανότητα της παρατηρηθείσας συμβολοσειράς s ως:

$$\tilde{P}(s) = \frac{\chi_s}{N_{|s|}}$$

Η εμπειρική αυτή πιθανότητα εξαρτάται από τον αριθμό των πιθανών φορών που εμφανίζεται στο σύνολο δειγμάτων η συμβολοσειρά s. Γενικά το να υπολογίσουμε το μέγιστο αριθμό των φορών που είναι πιθανό να εμφανιστεί στο σύνολο δειγμάτων μια συγκεκριμένη συμβολοσειρά s είναι αρκετά περίπλοκο.

Συνεχίζουμε ορίζοντας δεσμευμένη εμπειρική πιθανότητα να παρατηρήσουμε ένα σύμβολο αμέσως μετά από μια συγκεκριμένη υπακολουθία. Αυτή η πιθανότητα ορίζεται ως ο αριθμός των φορών που ένα συγκεκριμένο σύμβολο έχει εμφανιστεί αμέσως μετά από τη συγκεκριμένη υπακολουθία προς τον συνολικό αριθμό των φορών η συγκεκριμένη υπακολουθία έχει εμφανιστεί, ακολουθούμενη από οποιοδήποτε σύμβολο. Έστω λοιπόν χ_{s*} ο αριθμός εμφάνισης της συγκεκριμένης υπακολουθίας s στο σύνολο δειγμάτων $\{r^i\}$, όταν ακολουθεί οποιοδήποτε σύμβολο μετά από αυτή (non-suffix occurrences).

$$\chi_{s*} = \sum_{\sigma' \in k} \chi_{s\sigma'}$$

Τότε η δεσμευμένη εμπειρική πιθανότητα να παρατηρηθεί ένα συγκεκριμένο σύμβολο σ' μετά από μια συγκεκριμένη υπακολουθία s δίνεται από τον τύπο:

$$\tilde{P}(\sigma \mid s) = \frac{\chi_{s\sigma}}{\chi_{s*}}$$

Τέλος ορίζουμε $suf(s) = s_2 s_3 ... s_l$ και $s^R = s_l ... s_2 s_1$.

Δημιουργία PST: Παρουσιάζουμε παρακάτω τη διαδικασία δημιουργίας του PST από ένα σύνολο δειγμάτων. Η διαδικασία χρησιμοποιεί πέντε εξωτερικές παραμέτρους: n το μήκος της μνήμης, *Pmin* η ελάχιστη πιθανότητα που απαιτείται να έχει μια συμβολοσειρά (ακολουθία) για να εμφανιστεί, r που είναι ένα απλό μέτρο της διαφοράς μεταξύ της πρόβλεψης του υποψήφιου εκείνη τη στιγμή και του κόμβου πατέρα του, γmin που είναι ο παράγοντας κανονικοποίησης, και α μια παράμετρος που μαζί με τον παράγοντα κανονικοποίησης, καθορίζουν το κατώτατο όριο (κατώφλι) για την εμφάνιση ενός συμβόλου.

Χρησιμοποιούμε *το* $\overline{T}$ για να δείξουμε το δέντρο, $\overline{S}$ για να δείξουμε το σύνολο με τις (μοναδικές) συμβολοσειρές που πρέπει να ελέγξουμε και το $\overline{\gamma}_S$ για να δείξουμε την κατανομή πιθανότητας (πέρα από το επόμενο σύμβολο) που συνδέεται με τον κόμβο s.

Ο αλγόριθμος:

(1) *Αρχικοποίηση*: έστω $\overline{T}$ δέντρο με ένα κόμβο με ένα κόμβο (που δεν έχει ετικέτα)

$$\text{και έστω } \bar{S} \leftarrow \left\{ \sigma \mid \sigma \in k \quad \text{και} \quad \tilde{P}(\sigma) \leq P_{\min} \right\}$$

(2) *Δημιουργώντας το PST*: αν $\overline{S} \neq \emptyset$, επέλεξε οποιοδήποτε $s \in S$ και:

(a) Αφαίρεσε s από το $\overline{S}$

(b) Αν υπάρχει κάποιο σύμβολο $\sigma \in k$ ώστε

$$\tilde{P}(\sigma \mid s) \geq (1 + \alpha)\gamma_{\min}$$

και

$$\frac{\tilde{P}(\sigma \mid s)}{\tilde{P}(\sigma \mid suf\,(s))} \begin{cases} \geq r \\ \acute{\eta} \\ \leq 1/r \end{cases}$$

μετά πρόσθεσε στο $\overline{T}$ τον κόμβο που συνδέεται με το s και όλους τους κόμβους στο μονοπάτι για το s ξεκινώντας από τον χαμηλότερο κόμβο στο $\overline{T}$ που είναι ένα επίθεμα του s.

(c) Αν $|s| < n$ τότε πρόσθεσε τις συμβολοσειρές $\{\sigma' s \mid \sigma' \in k$ και $\tilde{P}(\sigma' s) \geq P_{\min}\}$ (αν υπάρχει) στο $\bar{S}$

(3) *Στρογγυλοποιούμε τις πιθανότητες πρόβλεψης:* για κάθε *s που είναι ετικέτα σε κάποιο κόμβο στο* $\overline{T}$, ισχύει:

$$\bar{\gamma}_s(\sigma) \equiv (1 - |\Sigma| \gamma_{\min}) \tilde{P}(\sigma \mid s) + \gamma_{\min} \qquad (1)$$

Το τελικό βήμα (το βήμα 3) του αλγορίθμου εκμάθησης είναι η διαδικασία κανονικοποίησης, η οποία εξασφαλίζει ότι η πιθανότητα εμφάνισης ενός συμβόλου δεν θα είναι ποτέ μηδενική, ανεξαρτήτως από το τι επίθεμα παρατηρείται πριν από αυτό στο δέντρο ότι. Το γ_{min} είναι η παράμετρος που καθορίζει την ελάχιστη πιθανότητα ενός συμβόλου, και οι εμπειρικές πιθανότητες πρέπει να είναι τέτοιες ώστε να ικανοποιούν αυτήν την απαίτηση. Αυτό γίνεται με τη μείωση των εμπειρικών πιθανοτήτων, έτσι ώστε το σύνολο $|k| \gamma_{\min}$ να μοιραστεί αργότερα σε όλα τα σύμβολα. Η μείωση κάθε εμπειρικής πιθανότητας γίνεται ανάλογα με την αξία της.

Επιστρέφοντας στο PST του σχήματος της εικόνας 2.8, μπορούμε τώρα να κάνουμε μερικές παρατηρήσεις στο σύνολο των ακολουθιών από το οποίο το μοντέλο εκπαιδεύεται:

- Στο σύνολο εκπαίδευσης πρέπει να υπάρξει γενικά μεγαλύτερη πιθανότητα παρατήρησης του γράμματος β μετά από το γράμμα α (όπου $\bar{\gamma}_\alpha(b) = 0.5$),

εκτός και αν το α προηγήθηκε πριν από ένα *r*, οπότε σ' αυτή την περίπτωση η πιθανότητα για είναι για *το c* (όπου $\bar{\gamma}_{ra}(c) = 0.4$).

Υποθέτοντας για μια στιγμή ότι το γmin ορίστηκε ίσο με 0.05, και εξετάζοντας το διάνυσμα πιθανότητας που συνδέεται με *το κόμβο (ca)*, μπορούμε να συμπεράνουμε ότι μόνο τρία διαφορετικά σύμβολα τα *b*, *d* και *r* , παρατηρήθηκαν στο σύνολο εκπαίδευσης δημιουργώντας την υπακολουθία ca, σε ποσότητες με αναλογία 7 : 7 : 1, αντίστοιχα μοντέλο PST δεν απαιτεί οποιαδήποτε υπόθεση για τα δεδομένα εισόδου, ούτε χρησιμοποιεί οποιαδήποτε *a priori* πληροφορία που μπορεί να έχουμε για τα δεδομένα μας. Η ενσωμάτωση τέτοιων πληροφοριών ενδεχομένως να βελτιώσουν την απόδοση του μοντέλου, δεδομένου ότι προσαρμόζει το μοντέλο στο συγκεκριμένο πρόβλημα της ανεύρεσης συγκεκριμένων επαναλαμβανόμενων περιοχών σε μακρομόρια, όπου ορισμένα χαρακτηριστικά γνωρίσματα και διαδικασίες παρατηρούνται. Συγκεκριμένα, οι πληροφορίες που θα επιθυμούσαμε να εξετάσουμε είναι των αμινοξέων που προηγούνται ενός αμινοξέος σε μια συγκεκριμένη θέση και οι πιθανότητες αντικατάστασης των αμινοξέων από άλλα αμινοξέα. Αυτές οι κατανομές είναι ενσωματωμένες στο μοντέλο PST και μερικές αλλαγές μπορούν να γίνουν για να συμπεριλάβουν και κάποιες *a priori* πληροφορίες.

Σ' αυτό το σημείο απαιτούνται λίγοι επιπρόσθετοι ορισμοί: *έστω qab* η πιθανότητα το αμινοξύ *α να* αντικατασταθεί από το αμινοξύ *β (a, β ∈ Σ)*. Επίσης, έστω χ_s ο αριθμός των *διαφορετικών* ακολουθιών που περιλαμβάνουν την υπακολουθία *s*. Αυτός ο αριθμός δεν πρέπει να υπερβαίνει ένα ορισμένο κατώτατο όριο *Nmin*, το οποίο καθορίζεται ανάλογα με το συνολικό αριθμό μ των ακολουθιών μέσα στο σύνολο δείγμα (δηλ. *Nmin = cm* όπου *το c επιλέγεται* να είναι, για παράδειγμα, 0.2 έτσι ώστε η υπακολουθία *s* να πρέπει να υπάρξει μέσα σε τουλάχιστον 20% των πρωτεϊνών μελών του συνόλου). Εναλλακτικά, αυτή η παράμετρος μπορεί να πάρει μια σταθερή τιμή ανεξάρτητα από το πραγματικό μέγεθος του συνόλου εκπαίδευσης.

Τέλος, το τελευταίο βήμα (βήμα 3) του αλγορίθμου εκμάθησης (το βήμα βελτιστοποίησης), τροποποιείται, ώστε να βασίζεται στη μέθοδο που παρουσιάστηκε από τον Henikoff και τον Henikoff (Henikoff and Henikoff, 1996). Αυτή η μέθοδος

προσθέτει τις υποθετικές τιμές στο σύνολο δείγμα προκειμένου να αποφευχθούν οι μηδενικές πιθανότητες. Ο αριθμός και η κατανομή των υποθετικών αυτών τιμών βασίζεται στη θέση των αμινοξέων στην ακολουθία (δηλ. διαφορετικά για κάθε κόμβο) και λαμβάνει υπόψη και την *ποικιλομορφία* (δηλ. τον αριθμό των διαφορετικών αμινοξέων που παρατηρούνται από μετά από κάποια υπακολουθία), καθώς επίσης και τον αριθμό των πραγματικά παρατηρηθέντων αμινοξέων.

Για έναν κόμβο που περιέχει την υπακολουθία *s*, η οποία χαρακτηρίζεται από *Rs* ποικιλομορφία (αριθμός διαφορετικών αμινοξέων παρατηρούνται μετά από *την s*), $R_s = |\{\sigma \mid \chi_{s\sigma} > 0\}|$. Δεδομένου του *Bs* ο συνολικός αριθμός των υποθετικών τιμών προστίθενται στον κόμβο s. Θέτουμε $BS = \mu Rs$, όπως προτάθηκε από τους Henikoff και Henikoff (Henikoff and Henikoff, 1996), όπου το μ μπορεί να βελτιστοποιηθεί για καλύτερη απόδοση. Κατόπιν, ο αριθμός των υποθετικών τιμών για το αμινοξύ α σε αυτόν τον κόμβο δίνεται από:

$$b_a = B_s \sum_{i=1}^{20} Pr\ ob\ (i \mid s) \cdot \Pr\ ob\ (a \mid i) = B_s \sum_{i=1}^{20} \frac{\chi_{si}}{\chi_{s*}} \cdot \frac{q_{ia}}{Q_i}$$

Όπου $Q_i = \sum_{k=1}^{20} qik$ η πιθανότητα παρατήρησης *του* α μετά από τη συμβολοσειρά s ορίζεται ως ο σταθμισμένος μέσος όρος της εμπειρικής πιθανότητας και η *a priori* πιθανότητα όπως καθορίζεται από τις υποθετικές τιμές, $P_{pse}(a \mid s) = b_a / B_s$. Η τροποποιημένη διαδικασία περιγράφεται παρακάτω:

Ο αλγόριθμος: **Build-Bio-PST** (Nmin, γ, r, n):

(1) *Αρχικοποίηση*: έστω $\overline{T}$ δέντρο με ένα κόμβο με ένα κόμβο (που δεν έχει ετικέτα)

και έστω $\bar{S} \leftarrow \left\{ \sigma \mid \sigma \in k \quad \text{και} \quad \bar{\chi}_\sigma \leq N_{\min} \right\}$

(2) *Δημιουργώντας το PST*: αν $\bar{S} \neq \varnothing$, επέλεξε οποιοδήποτε $s \in \bar{S}$ και:

(a) Αφαίρεσε s από το $\bar{S}$

(b) Αν υπάρχει κάποιο σύμβολο $\sigma \in k$ τέτοιο ώστε

$$\tilde{P}(\sigma \mid s) \geq \gamma$$

και

$$\frac{\tilde{P}(\sigma \mid s)}{\tilde{P}(\sigma \mid suf\ (s))} \begin{cases} \geq r \\ \acute{\eta} \\ \leq 1/r \end{cases}$$

μετά πρόσθεσε στο $\overline{T}$ τον κόμβο που συνδέεται με το s και όλους τους κόμβους στο μονοπάτι για το s ξεκινώντας από τον χαμηλότερο κόμβο στο $\overline{T}$ που είναι ένα επίθεμα του s.

(c) Αν $|s| < n$ τότε πρόσθεσε τις συμβολοσειρές $\{\sigma' s \mid \sigma' \in k$ και $\bar{\chi}_{\sigma' s} \geq N_{min}\}$ (αν υπάρχει) στο $\overline{S}$

(3) *Κανονικοποιώντας τις πιθανότητες πρόβλεψης:* για κάθε s *που είναι ετικέτα σε κάποιο κόμβο στο* $\overline{T}$, ισχύει:

$$\bar{\gamma}_s(\sigma) \equiv \frac{\chi_{s*}}{\chi_{s*} + B_s} \tilde{P}(\sigma \mid s) + \frac{B_s}{\chi_{s*} + B_s} P_{pse}(\sigma \mid s)$$

$$= \frac{\chi_{s*}}{\chi_{s*} + B_s} \frac{\chi_{s\sigma}}{\chi_{s*}} + \frac{B_s}{\chi_{s*} + B_s} \frac{b_\sigma}{B_s}$$

$$= \frac{\chi_{s\sigma} + b_\sigma}{\chi_{s*} + B_s}$$

Συνοψίζουμε εν συντομία τις αλλαγές όσον αφορά τις εξωτερικές παραμέτρους:

Το Nmin αντικαθιστά το *Pmin*, ενώ οι υποθετικές τιμές αντικαθιστούν το γ_{min} στην κανονικοποίηση, αφήνοντας το γ να αναλάβει, αντί για το α, και το γmin τον προσδιορισμό του κατώτατου ορίου.

Ένα χρήσιμο και πολύ πρακτικό χαρακτηριστικό γνώρισμα του αλγορίθμου εκπαίδευσης του PST είναι η δυνατότητα καθορισμού οποιουδήποτε μοντέλου PST (με το ίδιο σύνολο εκπαίδευσης από το οποίο δημιουργήθηκε αρχικά) χωρίς την ανάγκη να επαναλάβει τους υπολογισμούς που έχουν γίνει ήδη χτίζοντας το αρχικό μοντέλο. Για να το επεκτείνουμε περαιτέρω το μοντέλο PST, μπορούμε να

αλλάξουμε τις αρχικές παραμέτρους εκπαίδευσης. Συγκεκριμένα αν θέλουμε να αυξήσουμε τον αριθμό των κόμβων, που εξετάζονται για το συνυπολογισμό του PST μοντέλου, μπορούμε να χαμηλώσουμε το *Pmin* ή να αυξήσουμε *το n*. Για να μειώσουμε τα κριτήρια αποδοχής μιας συμβολοσειράς στο δέντρο μπορούμε να χαμηλώσουμε το γ προς το 1, ή το α προς το -1. Μόλις το καινούριο σύνολο παραμέτρων επιλεχτεί και αρχίσει η δημιουργία *του* PST, το βήμα αρχικοποίησης του αλγορίθμου αλλάζει στα εξής:

1. *Αρχικοποίηση:* εάν το T_0 είναι άδειο

   (a) Όπως πριν.

   (b) Έστω $T \leftarrow T_0$, και έστω

$$\bar{S} \leftarrow \{s \mid suf(s) \in T_0 \quad \text{και} \quad s \notin T_0 \quad \text{και} \quad \tilde{P}(s) \geq P_{\min} \quad \text{και} \quad |s| \leq n\}$$

Η δεύτερη παραλλαγή μπορεί να βελτιωθεί σημαντικά με τον ίδιο τρόπο.

2.8.1.4 Πρόβλεψη χρησιμοποιώντας το PST

Έχοντας μια συμβολοσειρά *s* η πρόβλεψή της από το PST γίνεται σύμβολο, σύμβολο, όπου η πιθανότητα κάθε συμβόλου υπολογίζεται ανιχνεύοντας το δέντρο και αναζητώντας του μακρύτερο επίθημα που εμφανίζεται στο δέντρο. Η δεσμευμένη πιθανότητα ενός συγκεκριμένου συμβόλου δεδομένου του συγκεκριμένου επιθήματος δίνεται από την κατανομή πιθανότητας που συνδέεται με τον αντίστοιχο κόμβο στο PST. Παραδείγματος χάριν, για προβλέψει το PST τη συμβολοσειρά *s* = *abracadabra* που δίνεται στην Εικόνα 2.4 ακολουθεί την παρακάτω διαδικασία:

$$\begin{aligned} P^T(\text{abracadabra}) &= P^T(a)P^T(b \mid \underline{a})P^T(r \mid ab)P^T(a \mid ab\underline{c})P^T(c \mid \underline{abra}) \\ &\quad \times P^T(a \mid abrac).....P^T(a \mid abracadab\ \underline{r}) \\ &= \bar{\gamma}_{root}(a)\,\bar{\gamma}_{\alpha}(b)\,\bar{\gamma}_{root}(r)\,\bar{\gamma}_{r}(a)\,\bar{\gamma}_{bra}(c)\,\bar{\gamma}_{root}(a)....\,\bar{\gamma}_{r}(a) \\ &= 0.2 \quad 0.5 \quad 0.2 \quad 0.6 \quad 0.35 \quad 0.2 \quad ... \quad 0.6 \\ &= 4.032 \cdot 10^{-6} \end{aligned}$$

Οι υπογραμμισμένες υπακολουθίες αντιπροσωπεύουν τα μακρύτερα επιθήματα που εμφανίζονται στο δέντρο (κανένας χαρακτήρας δεν υπογραμμίζεται όταν το μακρύτερο επίθημα είναι κενή συμβολοσειρά), και η πιθανότητα κάθε συμβόλου δίνεται από τη συνάρτηση πρόβλεψης που συνδέεται με τον αντίστοιχο κόμβο ($\bar{\gamma}_{root}$ (), $\bar{\gamma}_{\alpha}$ (), $\bar{\gamma}_{bra}$ () $\kappa\lambda\pi$.).

Διπλής κατεύθυνσης πρόβλεψη. Η διαδικασία πρόβλεψης που περιγραφικέ στην προηγούμενη παράγραφο πηγαίνει από τα αριστερά προς τα δεξιά, ξεκινώντας από το αριστερότερο σύμβολο. Ένα προφανές μειονέκτημα αυτής της διαδικασίας είναι ότι τα σύμβολα που βρίσκονται στην αρχή μιας περιοχής με υψηλή πιθανότητα δεν προβλέπονται με υψηλές πιθανότητες. Μόνο αν έχει παρατηρηθεί μια υπακολουθία με υψηλή πιθανότητα έχει προβλεφθεί πιο πριν τα σύμβολα που ακολουθούν προβλέπονται με υψηλή πιθανότητα.

Συνεπώς, τα (αριστερά) όρια μεταξύ περιοχών με υψηλές και χαμηλές πιθανότητες (δηλ. όρια περιοχής/μοτίβου) δεν είναι ξεκάθαρα. Για να λυθεί αυτό το πρόβλημα στο PST έχει εφαρμοστεί μια παραλλαγή του βήματος πρόβλεψης. Δεδομένου ενός συνόλου ακολουθιών, δύο PST μοντέλα δημιουργούνται, το T και το T^R. *Το* T χτίζεται από τις ακολουθίες όπως είναι, και *το* T^R χτίζεται από τις ακολουθίες ανεστραμμένες. Το βήμα πρόβλεψης επαναλαμβάνεται δύο φορές. Η ακολουθία εισαγωγής προβλέπεται χρησιμοποιώντας το T, και η αντίστροφη ακολουθία της προβλέπεται χρησιμοποιώντας *το* T^R. Κατόπιν, οι προβλέψεις συνδυάζονται και λαμβάνεται η μέγιστη πρόβλεψη και από τα δύο μοντέλα. Οπότε για $s = \tau\sigma\rho$ όπου $\sigma \in \Sigma$ και $\tau, \rho \in \Sigma^*$, $P_{T,T^R}(\sigma \mid s) = \max\{ P_T\{\sigma \mid \tau\}, P_{T^R}(\sigma \mid \rho^R) \}$.

2.8.2 SPST

Ο αλγόριθμος SPST (Leonardi, 2006) είναι μια παραλλαγή του αλγορίθμου PST. Βασίζεται και αυτός στη δημιουργία δέντρου και υπολογίζει τις πιθανότητες των αμινοξέων, με τον ίδιο τρόπο που το κάνει ο αλγόριθμος PST με τη διαφορά ότι κάνει με διαφορετικό τη βελτιστοποίηση των πιθανοτήτων που υπολογίζει. Για

n_mean ≠ 0 ο αλγόριθμος SPST μετασχηματίζεται στον αλγόριθμο SPST(FSPST). Ο αλγόριθμος SPST(FSPST) υπολογίζει αρχικά τις πιθανότητες των αμινοξέων, με τον ίδιο τρόπο που το κάνει ο αλγόριθμος SPST, και στη συνέχεια ψάχνει και βρίσκει το τμήμα της ακολουθίας με την υψηλότερη πιθανότητα, πάνω από ένα συγκεκριμένο όριο που έχει καθοριστεί κατά τη φάση της εκπαίδευσης. Το μήκος του τμήματος της ακολουθίας που αναζητά καθορίζεται από την τιμή του n_mean.

3. ΠΡΟΓΝΩΣΗ ΠΡΩΤΕΪΝΩΝ ΜΕ VLMCs

3.1 Εισαγωγή

Στο προηγούμενο κεφάλαιο κάναμε μια ανασκόπηση των ειδών των μαρκοβιανών μοντέλων που υπάρχουν με στόχο να αποκτήσουμε μια πλήρη εικόνα των εφαρμογών τους. Στη συνέχεια θα ασχοληθούμε εκτενώς με τα VLMCs, τα οποία απ' ότι διαπιστώσαμε έχουν εφαρμοστεί με μεγάλη επιτυχία σε οικογένειες πρωτεϊνών, που παρουσιάζουν μεγάλη ομολογία μεταξύ τους, και προσπαθήσαμε να διερευνήσουμε αν θα μπορούσαν να εφαρμοστούν με την ίδια επιτυχία σε άλλα είδη πρωτεϊνών όπως οι διαμεμβρανικές πρωτεΐνες και συγκεκριμένα αυτές που έχουν διαμόρφωση β-βαρελιών (β-barrel) και σπειροειδών σπειραμάτων (coiled-coils), που παρουσιάζουν μικρότερη συντήρηση.

Για τους σκοπούς αυτούς αναπτύξαμε ένα πρόγραμμα το οποίο χρησιμοποιώντας ένα VLMC, δημιουργεί ένα πιθανοθεωρητικό μοντέλο το οποίο αποτελεί ένα καλό διαχωριστικό εργαλείο. Το εργαλείο αυτό έχει την ικανότητα να μας δείχνει αν μια πρωτεΐνη είναι π.χ. διαμεμβρανικό β-βαρέλι ή όχι. Συγκεκριμένα εφαρμόσαμε 6 διαφορετικές παραλλαγές της μεθόδου αυτή. Για τη δημιουργία του προγράμματος και των παραλλαγών του στηριχθήκαμε κυρίως στις δημοσιεύσεις των Bejerano και Yona (Bejerano and Yona, 2001) και της Leonardi (Leonardi, 2006).

3.2 Υλικά και μέθοδοι

3.2.1 Συλλογή ακολουθιών και στάδια χρήσης αυτών

Για την εκπαίδευση μιας μεθόδου η οποία δεν προϋποθέτει την ύπαρξη γνωστής δομής, είναι απαραίτητο να συγκροτήσουμε ένα σύνολο δεδομένων πρωτεϊνικών ακολουθιών, για τις οι οποίες υπάρχουν αξιόπιστα δεδομένα που να τις κατατάσσουν στην κατηγορία των διαμεμβρανικών β-βαρελιών και στην κατηγορία των διαμεμβρανικών σπειροειδών σπειραμάτων. Το σύνολο δεδομένων που χρησιμοποιήθηκε για εκπαίδευση, επιλέχθηκε κυρίως από τη βιβλιογραφία. Συγκεκριμένα, από τη βιβλιογραφία επιλέχθηκαν 145 πρωτεΐνες που ανήκουν στη τάξη διπλώματος (fold) των διαμεμβρανικών β-βαρελιών ("Transmembrane beta-

barrels"). Το σύνολο αυτό, που ονομάσαμε για δική μας διευκόλυνση "omps", προήλθε από μια εργασία του Bagos και των συνεργατών του που δεν έχει δημοσιευτεί ακόμα, ύστερα από προσωπική επικοινωνία. Για παραλλαγές τις ίδιας πρωτεΐνης, επιλέχτηκε μια, ενώ οι υπόλοιπες απομακρύνθηκαν, όπως επίσης απομακρύνθηκαν και τα πολλαπλά αντίγραφα των αλυσίδων που συναντώνται σε πολυμερείς δομές. Από τη δημοσίευση των Delorenzi και Speed (Delorenzi and Speed, 2002) επιλέξαμε ένα σύνολο από 374 πρωτεΐνες με δομή σπειροειδούς σπειράματος ("Coiled-coils"), το οποίο ονομάσαμε "coils_positive". Ως αρνητικά παραδείγματα στην προσπάθεια διαχωρισμού, χρησιμοποιήθηκαν 1114 μη μεμβρανικές πρωτεΐνες με γνωστή δομή κατατεθειμένες στην PDB (Berman et al., 2002). Αυτές επιλέχθηκαν με χρήση του εξυπηρετητή του παγκοσμίου ιστού PAPIA (Noguchi and Akiyama, 2003), επιλέγοντας ως όριο για την κατά ζεύγη ομοιότητα τους το 25% σε ομοιότητα καταλοίπων (similarity), σε μήκος άνω των 80 καταλοίπων και αφού αφαιρέσαμε όλες τις πρωτεΐνες με μη προσδιορισμένα κατάλοιπα (X) στην ακολουθία τους. Το σύνολο αυτό, το οποίο ονομάσαμε "papia", χρησιμοποιήθηκε από τον Bagos και τους συνεργάτες του για το PRED-TMBB (Bagos et al., 2004a) και το MCMBB (Bagos et al., 2004b). Από τη δημοσίευση των Delorenzi και Speed (Delorenzi and Speed, 2002) επιλέξαμε ένα επιπλέον σύνολο από 1524 πρωτεΐνες οι οποίες δεν είχαν διαμόρφωση σπειροειδούς σπειράματος, που ονομάσαμε "coils_negative".

Τα σύνολα "papia" και "coils_negative" χρησιμοποιήθηκαν για την εκπαίδευση ορισμένων αρνητικών μοντέλων (μοντέλο που δημιουργήθηκε ύστερα από εκπαίδευση μιας μεθόδου από ένα σύνολο από πρωτεΐνες που δεν έχουν διαμόρφωση β-βαρελιού ή σπειροειδούς σπειράματος), που χρησιμοποιήσαμε σε κάποιες παραλλαγές του προγράμματος μας. Επιπλέον τα δύο παραπάνω σύνολα, που αποτελούνται από πρωτεΐνες που δεν έχουν τις δύο διαμορφώσεις που μας ενδιαφέρουν, τα χρησιμοποιήσαμε για την αξιολόγηση της ικανότητας της προγνωστικής μας μεθόδου να διαχωρίζει διαμεμβρανικά β-βαρέλια από τις υπόλοιπες πρωτεΐνες, καθώς και πρωτεΐνες με δομή σπειροειδούς σπειράματος από πρωτεΐνες που δεν παρουσιάζουν τέτοια διαμόρφωση. Συγκεκριμένα τρέξαμε τα προγράμματα μας έναντι αυτών των συνόλων για να δούμε πόσες πρωτεΐνες από το

ένα σύνολο θα προβλέψει λανθασμένα ότι είναι διαμεμβρανικές πρωτεΐνες με διαμόρφωση β-βαρελιού και πόσες από το δεύτερο σύνολο ότι είναι πρωτεΐνες με διαμόρφωση σπειροειδούς σπειράματος. Επιπρόσθετα σε όλες τις περιπτώσεις, για τον έλεγχο της σχετικής επιτυχίας της μεθόδου μας (τεστάρισμα), τρέξαμε τα προγράμματα και έναντι των συνόλων που συλλέξαμε ως διαμεμβρανικά β-βαρέλια (omps) και ως πρωτεΐνες με διαμόρφωση σπειροειδούς σπειράματος (coils_positive). Ακριβέστερα, για να αποφύγουμε το φαινόμενο της υπερπροσαρμογής, αφαιρούσαμε κάθε φορά μία πρωτεΐνη από το σύνολο των διαμεμβρανικών β-βαρελιών, εκπαιδεύαμε τη μέθοδο μας με τις υπόλοιπες πρωτεΐνες και τρέχαμε το πρόγραμμα μας έναντι της πρωτεΐνης που είχαμε αφαιρέσει για να δούμε αν η μέθοδος θα προβλέψει σωστά ότι η πρωτεΐνη αυτή είναι διαμεμβρανική πρωτεΐνη με διαμόρφωση β-βαρελιού ή όχι. Η ίδια διαδικασία ακολουθήθηκε και για το σύνολο των πρωτεϊνών με δομή σπειροειδούς σπειράματος.

3.2.3 Εφαρμογή του μοντέλου VLMC

Στηριζόμενοι κυρίως στις δημοσιεύσεις των Bejerano και Yona (Bejerano and Yona, 2001) και της Leonardi (Leonardi, 2006), αναπτύξαμε ένα πρόγραμμα το οποίο στηριζόμενο στα VLMCs, και συγκεκριμένα στο PST και στο SPST (κάποιες παραλλαγές) αλγόριθμο, δημιουργεί ένα πιθανοθεωρητικό μοντέλο το οποίο αποτελεί ένα καλό διαχωριστικό εργαλείο, που έχει την ικανότητα να μας δείχνει αν μια πρωτεΐνη είναι π.χ. διαμεμβρανική ή όχι. Συγκεκριμένα εφαρμόσαμε 3 διαφορετικές παραλλαγές του προγράμματος.

Όλες οι παραλλαγές των προγραμμάτων υπολογίζουν ένα σκορ για το θετικό μοντέλο (η πιθανότητα μιας πρωτεΐνης να είναι π.χ. διαμεμβρανικό β-βαρέλι) και ένα σκορ για το αρνητικό μοντέλο (η πιθανότητα της πρωτεΐνης να μην είναι π.χ. διαμεμβρανικό β-βαρέλι). Για την ακρίβεια όταν τρέχουμε το πρόγραμμα έναντι μιας πρωτεΐνης τότε το πρόγραμμα υπολογίζει για τη συγκεκριμένη πρωτεΐνη ένα σκορ για το θετικό μοντέλο και ένα σκορ για το αρνητικό μοντέλο. Έπειτα το πρόγραμμα αφαιρεί από το σκορ της πρωτεΐνης για το θετικό μοντέλο το σκορ της πρωτεΐνης για το αρνητικό μοντέλο.

$$S_{final} = S_{positive} - S_{negative}$$

Αν το αποτέλεσμα είναι θετικός αριθμός τότε η πρωτεΐνη έχει διαμόρφωση β-βαρελιού, ενώ αν το αποτέλεσμα είναι αρνητικός αριθμός τότε η πρωτεΐνη δεν έχει διαμόρφωση β-βαρελιού. Τόσο ο αλγόριθμος PST (Bejerano and Yona, 2001) όσο και ο SPST (Leonardi, 2006) γράφτηκαν σε C++ γλώσσα ενώ για τη δημιουργία των βασικών προγραμμάτων και κάποιων επιπλέον βοηθητικών προγραμμάτων χρησιμοποιήσαμε τη γλώσσα PERL.

Στην πρώτη παραλλαγή του προγράμματος χρησιμοποιήσαμε τον αλγόριθμο PST για να δημιουργήσουμε τόσο το θετικό μοντέλο όσο και το αρνητικό μοντέλο. Σε αυτή την παραλλαγή εκπαιδεύσαμε τον αλγόριθμο PST πρώτα με το σύνολο των πρωτεϊνών που ήταν διαμεμβρανικά β-βαρέλια, δημιουργώντας έτσι το θετικό μοντέλο και έπειτα με το σύνολο των πρωτεϊνών που δεν ήταν διαμεμβρανικά β-βαρέλια, δημιουργώντας έτσι το αρνητικό μοντέλο.

Πίνακας 3.1 Πιθανότητες εμφάνισης των αμινοξέων όπως υπολογίστηκαν από τη βάση δεδομένων Swiss-Prot

ΠΙΘΑΝΟΤΗΤΕΣ ΕΜΦΑΝΙΣΗΣ ΑΜΙΝΟΞΕΩΝ			
A	**0.077**	**D**	**0.058**
F	**0.040**	**H**	**0.024**
K	**0.063**	**M**	**0.022**
P	**0.046**	**R**	**0.049**
T	**0.057**	**W**	**0.015**
C	**0.018**	**E**	**0.066**
G	**0.072**	**I**	**0.056**
L	**0.086**	**N**	**0.046**
Q	**0.040**	**S**	**0.062**
V	**0.068**	**Y**	**0.035**

Στη δεύτερη παραλλαγή δημιουργήσαμε το θετικό μοντέλο εκπαιδεύοντας τον αλγόριθμο PST με το σύνολο των πρωτεϊνών που ήταν διαμεμβρανικά β-βαρέλια, όπως παραπάνω, αλλά για τη δημιουργία του αρνητικού μοντέλου έγινε με βάση τις τυχαίες πιθανότητες εμφάνισης των αμινοξέων όπως αυτές υπολογίστηκαν από τη βάση δεδομένων Swiss-Prot (Πίνακας 3.1). Συγκεκριμένα το σκορ της εκάστοτε πρωτεΐνης για το αρνητικό μοντέλο, δηλαδή η πιθανότητα της πρωτεΐνης να μην είναι διαμεμβρανικό β-βαρέλι, υπολογίζεται προσθέτοντας τις πιθανότητες εμφάνισης του

κάθε αμινοξέος και διαιρώντας τες με το σύνολο των αμινοξέων της κάθε πρωτεΐνης. Έστω a το αμινοξύ και P(α) η πιθανότητα εμφάνισης ενός αμινοξέος. Εάν μια πρωτεΐνη έχει μήκος n τότε το σκορ S για το αρνητικό μοντέλο δίνεται από τον τύπο :

$$S = \frac{\sum_{i=1}^{n} \log(P(a))}{n}$$

Για την τρίτη παραλλαγή χωρίσαμε τα 20 αμινοξέα σε 5 ομάδες (Πίνακας 3.2), μειώνοντας έτσι το αλφάβητο από 20 σύμβολα σε 4 σύμβολα. Ο διαχωρισμός των αμινοξέων σε 4 διαφορετικέ ομάδες έγινε με βάση τις ιδιότητες των αμινοξέων. Έπειτα μετατρέψαμε το σύνολο με τα διαμεμβρανικά β-βαρέλια από ένα σύνολο αμινοξικών ακολουθιών σε ένα σύνολο ακολουθιών αποτελούμενο από τα 4 σύμβολα των ομάδων. Συγκεκριμένα κάθε αμινοξικό σύμβολο το αντικαθιστούσαμε με το αντίστοιχο σύμβολο της ομάδας στην οποία ανήκει.

Πίνακας 3.2: Οι 5 ομάδες στις οποίες χωρίσαμε τα 20 αμινοξέα

ΟΜΑΔΕΣ	**ΑΜΙΝΟΞΕΑ ΚΑΘΕ ΟΜΑΔΑΣ**	**ΠΙΘΑΝΟΤΗΤΑ ΕΜΦΑΝΙΣΗΣ ΟΜΑΔΑΣ**
ΥΔΡΟΦΟΒΑ (Υ)	I, V, L, M, A	0.309
ΑΡΝΗΤΙΚΑ (Ν)	K, R	0.112
ΘΕΤΙΚΑ (Ρ)	N, Q, D, E	0.21
ΜΙΚΡΑ (Μ)	P, G, C, T, S	0.255
ΑΑΡΩΜΑΤΙΚΑ (Α)	H, F, Y, W	0.114

Στη συνέχεια δημιουργήσαμε το θετικό μοντέλο εκπαιδεύοντας τον αλγόριθμο PST με το τροποποιημένο σύνολο με τα διαμεμβρανικά β-βαρέλια, που αναφέραμε παραπάνω. Για τη δημιουργία του αρνητικού μοντέλου υπολογίσαμε τις πιθανότητες εμφάνισης των 4 συμβόλων αθροίζοντας τις τυχαίες πιθανότητες εμφάνισης των αμινοξέων που ανήκαν στην κάθε ομάδα και το άθροισμα αυτό το διαιρέσαμε με το πλήθος των αμινοξέων που υπήρχαν στην κάθε ομάδα. Το σκορ της εκάστοτε πρωτεΐνης για το αρνητικό μοντέλο, δηλαδή η πιθανότητα της πρωτεΐνης να μην είναι διαμεμβρανικό β-βαρέλι, υπολογίστηκε με παρόμοιο τρόπο με αυτόν υπολογίστηκε για το αρνητικό μοντέλο στην παραλλαγή 2. Έστω b μια από τις 5 ομάδες, *P(b)* η

πιθανότητα εμφάνισης της ομάδας b, δηλαδή η πιθανότητα εμφάνισης ενός εκ' των αμινοξέων της ομάδας b, και n το μήκος της πρωτεϊνικής ακολουθίας τότε το σκορ S' της πρωτεΐνης για το αρνητικό μοντέλο δίνεται από:

$$S' = \frac{\sum_{i=1}^{n} \log(P(b))}{n}$$

Τις υπόλοιπες παραλλαγές τις δημιουργήσαμε με τον ίδιο τρόπο όπως τις παραπάνω με τη διαφορά όμως ότι αντί για το PST χρησιμοποιήσαμε μια παραλλαγή του, το SPST (και το FSPST, που είναι ο ίδιος αλγόριθμος με το SPST με τη διαφορά ότι το n_mean παίρνει τιμές διάφορες του μηδενός) για να δημιουργήσουμε τα ίδια πιθανοθεωρητικά μοντέλα που δημιουργήσαμε παραπάνω με το PST.

Επιπλέον δημιουργήσαμε, με τον ίδιο τρόπο όπως παραπάνω, άλλα 6 προγράμματα με στόχο την πρόγνωση πρωτεϊνών με δομή σπειροειδούς σπειράματος, με τη διαφορά ότι αντί για τα σύνολα των διαμεμβρανικών β-βαρελιών και των μη διαμεμβρανικών β-βαρελιών χρησιμοποιήσαμε το σύνολο με τις πρωτεΐνες με δομή σπειράματος και το σύνολο με πρωτεΐνες που δεν είχαν αυτή τη δομή. Τα δύο σύνολα αυτά χρησιμοποιήθηκαν με τον ίδιο τρόπο που χρησιμοποιήθηκαν τα σύνολα με τα διαμεμβρανικά β-βαρέλια και τα μη διαμεμβρανικά β-βαρέλια.

3.3 Μέτρηση της επιτυχίας των μοντέλων

Για να μπορέσουμε να μετρήσουμε την επιτυχία και την αξιοπιστία των προγνώσεων που προέρχονται από μια μέθοδο, έχουν προταθεί διάφορα μέτρα. Για την αξιολόγηση των προγνώσεων σε επίπεδο πρωτεϊνών, θα αναφερθούμε στο συνολικό ποσοστό των πρωτεϊνών που έχουν προβλεφθεί σωστά, με την πρόγνωση να έχει αναχθεί σε δυο κατηγορίες (πρωτεΐνες που προβλέφθηκαν σωστά/ πρωτεΐνες που προβλέφθηκαν λάθος):

$$ΕΥΑΙΣΘΗΣΙΑ(SE) = \frac{TP}{TP + FN},$$

$$ΕΙΔΙΚΟΤΗΤΑ(SP) = \frac{TN}{TN + FP}$$

όπου ο τύπος *ΕΥΑΙΣΘΗΣΙΑ(SE)*, υπολογίζει την ευαισθησία της μεθόδου μας, δηλ. το ποσοστό των πρωτεϊνών που η μέθοδος προβλέπει σωστά ως διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα και ο τύπος *ΕΙΔΙΚΟΤΗΤΑ(SP)* υπολογίζει την ειδικότητα του μοντέλου μας, δηλ. το ποσοστό των πρωτεϊνών που η μέθοδος προβλέπει λανθασμένα ότι είναι διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα, ενώ δεν είναι. Επίσης, χρησιμοποιείται ο γνωστός συντελεστής συσχέτισης του Matthews *(MMC)* (Baldi et al.,2000):

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

Σε όλα τα παραπάνω, με *TP* (True Positives) συμβολίζουμε τον αριθμό ορθός θετικά προσδιορισμένων πρωτεϊνών, με *TN* (True Negatives) τον αριθμό των ορθός αρνητικά προσδιορισμένων πρωτεϊνών, με *FN* (False Negatives) τον αριθμό των εσφαλμένα αρνητικά προσδιορισμένων πρωτεϊνών και με *FP* (False Positives) τον αριθμό των εσφαλμένα θετικά προσδιορισμένων πρωτεϊνών. Έτσι για την αξιολόγηση της πρόγνωσης των διαμεμβρανικών β-βαρελιών ή σπειροειδών σπειραμάτων, θεωρούμε όπου *TP* τον αριθμό των πρωτεϊνών που προσδιορίστηκαν σωστά ως διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα, *TN* τον αριθμό των πρωτεϊνών που προσδιορίστηκαν σωστά ως μη διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα, *FN* τον αριθμό των πρωτεϊνών που προσδιορίστηκαν λάθος ως μη διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα ενώ ήταν, και *FP* τον αριθμό των πρωτεϊνών που προσδιορίστηκαν λάθος ως διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα ενώ δεν ήταν.

Να σημειωθεί εδώ ότι τιμές του score μεγαλύτερες από 0, υποδεικνύουν την ταξινόμηση της πρωτεΐνης σαν διαμεμβρανικό β-βαρέλι ή σπειροειδές σπείραμα, ενώ χαμηλότερες προσφέρουν ισχυρή ένδειξη ότι αυτή είναι σφαιρική πρωτεΐνη. Εξαιρούνται κάποιες παραλλαγές δεν έχει χρησιμοποιηθεί για κατώφλι το 0 (όπου αναγράφεται το κατώφλι στους πίνακες των αποτελεσμάτων), καθώς διαπιστώσαμε ότι χρησιμοποιώντας άλλα κατώφλια οι πρωτεΐνες μας διαχωρίζονταν καλύτερα από ότι όταν χρησιμοποιούσαμε το 0.

3.4 Αποτελέσματα

Τα αποτελέσματα μας δείχνουν ότι για τις συγκεκριμένες κατηγορίες πρωτεϊνών, που δεν παρουσιάζουν μεγάλη ομολογία μεταξύ τους, τα VLMCs δεν αποδίδουν τόσο καλά όσο αποδίδουν όταν εφαρμόζονται σε οικογένειες πρωτεϊνών που παρουσιάζουν μεγάλη ομολογία. Ωστόσο αποδίδουν το ίδιο καλά ή και καλύτερα από πολλά υπολογιστικά εργαλεία που βασίζονται είτε σε στατιστικούς αλγόριθμους είτε σε HMMs, τα οποία είναι πολύ πιο περίπλοκα, απαιτούν μεγάλο χρονικό διάστημα για να εκπαιδευτούν και επιπλέον δεν μπορούν να εφαρμοστούν για αλυσίδες Markov μεγάλης τάξης. Σε μια σύγκριση που έγινε με άλλα υπολογιστικά (Πίνακας 3.4) διαπιστώσαμε ότι στο σύνολο των διαμεμβρανικών πρωτεϊνών βγάζουμε καλύτερα ή ίδια σχεδόν αποτελέσματα με τα υπάρχοντα εργαλεία με εξαίρεση το PRED-TMBB (Bagos et al., 2004a), που βγάζει τα μεγαλύτερα ποσοστά επιτυχίας. Σύγκρισή με υπολογιστικά εργαλεία (Πίνακες 3.3 – 3.4) που προβλέπουν αν μια πρωτεΐνη έχει δομή σπειροειδούς σπειράματος δεν έγινε γιατί τα υπάρχοντα εργαλεία κάνουν πρόβλεψη της τοπολογίας, δηλ. της θέσης που βρίσκεται το σπειροειδές σπείραμα σε μια πρωτεΐνη που γνωρίζουμε ότι εκ των προτέρων ότι εμφανίζει μια τέτοια δομή. Δεν υπάρχει επομένως κάποιο υπολογιστικό εργαλείο που να προβλέπει αν μια άγνωστη πρωτεΐνη είναι διαμεμβρανικό σπειροειδές σπείραμα ή όχι, πράγμα που κάνει η μέθοδος μας.

Πίνακας 3.3: Αποτελέσματα σύγκρισης με κάποια από τα υπάρχοντα υπολογιστικά εργαλεία

ΥΠΟΛΟΓΙΣΤΙΚΑ ΕΡΓΑΛΕΙΑ	ΕΥΑΙΣΘΗΣΙΑ	ΕΙΔΙΚΟΤΗΤΑ	ΣΥΝΤΕΛΕΣΤΗΣ ΣΥΣΧΕΤΙΣΗΣ
Bomp	80% (116/29)	96.4% (1074/40)	0.74
MCMBB	87.8% (127/18)	92.8% (1034/80)	0.692
PSORTb(v.2.0.4)	66.9% (97/48)	99.2% (1105/9)	0.759
PRED-TMBB	100% (145/0)	96.8% (1078/36)	0.88
TMBETA-SVM	73% (106/39)	92.4% (1029/85)	0.582
TMBETADISC-RBF	84% (122/23)	94% (1047/67)	0.70
OMP_WED	64.8% (94/51)	97.7% (1088/26)	0.679

Πίνακας 3.4: Ηλεκτρονικές διευθύνσεις των υπολογιστικών εργαλείων με τα οποία έγινε σύγκριση

ΥΠΟΛΟΓΙΣΤΙΚΑ ΕΡΓΑΛΕΙΑ	*URL*	*ΑΝΑΦΟΡΕΣ*
BOMP	http://www.bioinfo.no/tools/bomp	Berven et al., 2004
MCMBB	http://athina.biol.uoa.gr/bioinformatics/mcmbb/	Bagos et al., 2004a
PRED-TMBB	http://bioinformatics.biol.uoa.gr/PRED-TMBB/	Bagos et al., 2004b
PSORT-B	http://www.psort.org/psortb/	Gardy et al., 2003
Tmbeta-SVM	http://tmbeta-svm.cbrc.jp/	Park et al., 2005
TMBETADISC-RBF	http://rbf.bioinfo.tw/~sachen/OMP.html	Ou et al., 2008
OMP_WED	http://www.cs.usu.edu/~cyan/OMP_WED/	Hu and Yan, 2008

Το γεγονός ότι η μέθοδος μας βγάζει καλύτερα ή ίδια σχεδόν αποτελέσματα με τα υπάρχοντα εργαλεία είναι πολύ σημαντικό δεδομένου ότι η εκπαίδευση ενός HMM, μοντέλα στα οποία στηρίζονται πολλά από τα παραπάνω υπολογιστικά εργαλεία, είναι πολύ χρονοβόρα όπως προαναφέραμε, όταν η εκπαίδευση ενός VLMC διαρκεί μόλις λίγα λεπτά, αλλά και για πολλούς άλλους λόγους που θα αναλύσουμε παρακάτω. Συγκεκριμένα και ανεξάρτητα από τη σύγκριση με τα υπάρχοντα εργαλεία, τα αποτελέσματα μας δείχνουν ότι τα ποσοστά επιτυχίας της μεθόδου μας κυμαίνονται κοντά στο 91% για τα β-βαρέλια και κοντά στο 95% για τα σπειροειδή σπειράματα (Πίνακες 3.5-3.11).

Πίνακας 3.5: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο PST για τις 3 παραλλαγές που περιγράψαμε παραπάνω, για διάφορες τιμές παραμέτρων.

Αποτελέσματα για διαμεμβρανικά β-βαρέλια με τον αλγόριθμο PST								
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**		
	Pmin	alpha	gamma	P_ratio	order	***SE***	***SP***	***MCC***
1η παραλλαγή	0.05	0	0.001	1.05	3	91%	91.3%	0.681
2η παραλλαγή	0.05	0	0.001	1.05	3	90.3%	91.9%	0.69
3η παραλλαγή	0.05	0	0.001	1.05	3	86.9%	81%	0.48

Πίνακας 3.6: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο SPST για τις 3 παραλλαγές που περιγράψαμε παραπάνω, για διάφορες τιμές παραμέτρων.

Αποτελέσματα για διαμεμβρανικά β-βαρέλια με τον αλγόριθμο SPST								
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**		
	Nmin	r_max	gamma	n_mean	order	***SE***	***SP***	***MCC***
1η παραλλαγή	1	1.5	0.001	0	1	88.3%	93.8%	0.681
2η παραλλαγή	3	3.8	0.001	0	1	87.5%	92.4%	0.69
3η παραλλαγή	3	3.8	0.001	0	2	86.2%	84%	0.489

Πίνακας 3.7: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο SPST (FSPST) για τις 3 παραλλαγές που περιγράψαμε παραπάνω, για διάφορες τιμές παραμέτρων.

Αποτελέσματα για διαμεμβρανικά β-βαρέλια με τον αλγόριθμο SPST(FSPST)								
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**		
	Nmin	r_max	gamma	n_mean	order	***SE***	***SP***	***MCC***
1η παραλλαγή	1	1.5	0.001	120	1	88.3%	60.2%	0.28
2η παραλλαγή (κατώφλι=0.01)	3	3.8	0.001	120	1	97.2%	64%	0.395
3η παραλλαγή (κατώφλι=0.01)	1	1.5	0.001	120	3	91%	42.5%	0.22

Πίνακας 3.8: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο PST για τις 3 παραλλαγές που περιγράψαμε παραπάνω, για διάφορες τιμές παραμέτρων.

Αποτελέσματα για σπειροειδή σπειράματα με τον αλγόριθμο PST								
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**		
	Pmin	alpha	gamma	p_ratio	order	***SE***	***SP***	***MCC***
1η παραλλαγή	0.05	0	0.001	1.05	3	86.4%	18.6%	0.578
2η παραλλαγή	0.05	0	0.001	1.05	3	86.3%	86.4%	0.645
3η παραλλαγή (κατώφλι=2)	0.05	0	0.001	1.05	3	86%	92.1%	0.736

Πίνακας 3.9: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο SPST για τις 3 παραλλαγές που περιγράψαμε παραπάνω, για διάφορες τιμές παραμέτρων.

Αποτελέσματα για σπειροειδή σπειράματα με τον αλγόριθμο SPST								
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**		
	Nmin	r_max	gamma	n_mean	Order	***SE***	***SP***	***MCC***
1η παραλλαγή	1	1.5	0.001	0	1	81.2%	85.35	0.53
2η παραλλαγή	3	3.8	0.001	0	2	82.3%	91.1%	0.7
3η παραλλαγή	3	3.8	0.001	0	3	88.2%	80%	0.575

Πίνακας 3.10: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο SPST(FSPST) για τις 3 παραλλαγές που περιγράψαμε παραπάνω, για διάφορες τιμές παραμέτρων.

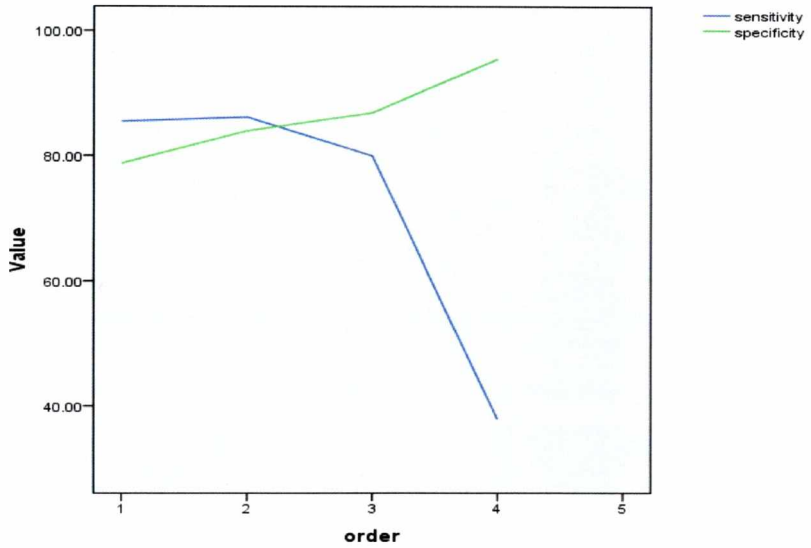
Αποτελέσματα για σπειροειδή σπειράματα με τον αλγόριθμο SPST(FSPST)								
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**		
	Nmin	r_max	gamma	n_mean	Order	***SE***	***SP***	***MCC***
1η παραλλαγή	1	1.5	0.001	100	1	85.6%	15.1%	0.73
2η παραλλαγή (Κατ.=0.03) (Κατ.=0.04) (Κατ.=0.05)	1	1.5	0.001	100	1	 97% 91.2% 84%	 89.9% 95.9% 98%	 0.777 0.847 0.847
3η παραλλαγή (Κατ.=0.03) (Κατ.=0.035) (Κατ.=0.029)	1	1.5	0.001	100	1	 89.5% 79.6% 91.2%	 92.1% 95.4% 90.9%	 0.77 0.755 0.751

Πίνακας 3.11: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο SPST(FSPST) για τις 3 παραλλαγές που περιγράψαμε παραπάνω, για διάφορες τιμές παραμέτρων.

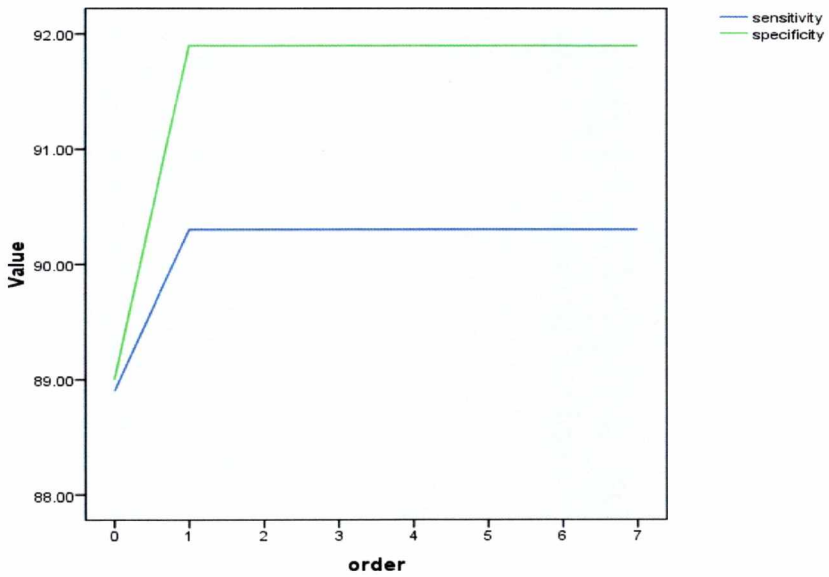
Αποτελέσματα για σπειροειδή σπειράματα με τον αλγόριθμο SPST(FSPST)								
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**		
	Nmin	r_max	gamma	n_mean	Order	***SE***	***SP***	***MCC***
1η παραλλαγή	1	1.5	0.001	80	1	85.8%	15.1%	0.1
2η παραλλαγή	1	1.5	0.001	80	1			
(Κατ.=0.03)						98.3%	80.5%	0.658
(Κατ.=0.04)						95.7%	90.9%	0.784
(Κατ.=0.05)						90.9%	96.2%	0.851
3η παραλλαγή	1	1.5	0.001	80	1			
(Κατ.=0.03)						97.9%	62%	0.476
(Κατ.=0.035)						93,3%	81.5%	0.632
(Κατ.=0.029)						80.5%	94.3%	0.299

Από τους παραπάνω πίνακες παρατηρούμε επιπλέον ότι ο αλγόριθμος SPST(FSPST) βγάζει πολύ καλά αποτελέσματα για τα σπειροειδή σπειράματα ενώ για τα β-βαρέλια δεν έχει καθόλου καλή απόδοση. Αυτό συμβαίνει γιατί ο αλγόριθμος SPST(FSPST) υπολογίζει αρχικά τις πιθανότητες των αμινοξέων, με τον ίδιο τρόπο που το κάνει ο αλγόριθμος SPST, και στη συνέχεια ψάχνει και βρίσκει το τμήμα της ακολουθίας με την υψηλότερη πιθανότητα. Επειδή το μήκος των σπειροειδών σπειραμάτων κυμαίνεται κοντά στα 100 αμινοξέα χωρίς μεγάλες αποκλίσεις, ο αλγόριθμος αυτός πιάνει τα τμήματα με δομή σπειροειδούς σπειράματος. Συγκεκριμένα φαίνεται από τα αποτελέσματά μας ότι όταν ο αλγόριθμος SPST(FSPST) αναζητά τμήματα με δομή σπειροειδούς σπειράματος μήκους 100 αμινοξέων (n_mean=100) (Πίνακας 3.10) αποδίδει πολύ καλά για τις πρωτεΐνες που παρουσιάζουν αυτή τη δομή. Μάλιστα αποδίδει πολύ καλύτερα απ' ότι για τμήματα μήκους 80 αμινοξέων (Πίνακας 3.10). Αντίθετα για τα διαμεμβρανικά β-βαρέλια ο αλγόριθμος αυτός αποτυγχάνει να μας δώσει παρόμοια αποτελέσματα καθώς το μήκος των διαμεμβρανικών β-βαρελιών κυμαίνεται μεταξύ 140 και 400 αμινοξέων. Έτσι εμείς όταν λέμε στον SPST(FSPST) αλγόριθμο να αναζητήσει το τμήμα μήκους 120 αμινοξέων (n_mean=120) (Πίνακας 3.6) με τη

μεγαλύτερη πιθανότητα δεν πιάνει τα β-βαρέλια που είναι πολύ μικρά ή τα β-βαρέλια που είναι πολύ μεγάλα σε μήκος.



Διάγραμμα 3.1: Μεταβολές του sensitivity και του specificity ανάλογα με την τάξη του μοντέλου για την 3η παραλλαγής-αλγόριθμος SPST του προγράμματος με τιμές παραμέτρων N_min=1, r_max=1.5, gamma=0.001, n_mean=0. Στον x'x άξονα έχουμε τις μεταβολές της τάξης του μοντέλου και στον y'y άξονα στα ποσοστά % του sensitivity και του specificity.



Διάγραμμα 3.2: Μεταβολές του sensitivity και του specificity ανάλογα με την τάξη του μοντέλου για την 2η παραλλαγής-αλγόριθμος PST του προγράμματος με τιμές παραμέτρων Pmin=1, alpha=0, gamma=0.001, p_ratio=1.05. Στον x'x άξονα έχουμε τις μεταβολές της τάξης του μοντέλου και στον y'y άξονα στα ποσοστά % του sensitivity και του specificity.

Ένα ακόμα συμπέρασμα που προκύπτει είναι ότι η μέθοδος μας βγάζει τα καλύτερα αποτελέσματα για πρώτης και δεύτερης τάξης μοντέλο και ενίοτε και για τρίτης τάξης, με κάποιες από τις παραλλαγές που εφαρμόσαμε. Γεγονός που αποδεικνύει ότι δεν θα είχε νόημα να εφαρμόσουμε αλυσίδες Markov τάξης μεγαλύτερης της $3^{ης}$ για πρόγνωση των διαμεμβραβικών πρωτεϊνών με διαμόρφωση β-βαρελιού ή σπειροειδών σπειραμάτων. Ένα χαρακτηριστικό παράδειγμα που ενισχύει την άποψη αυτή δίνεται στο διάγραμμα 3.1, στο οποίο βλέπουμε πως μεταβάλλεται το sensitivity και το specificity για τα διαμεμβρανικά β-βαρέλια όταν τα τρέχουμε έναντι της $3^{η}$ παραλλαγής-αλγόριθμος SPST του προγράμματος (Πίνακας 3.6) (χρησιμοποιήσαμε αυτή την παραλλαγή, παρότι δεν είναι η καλύτερη, γιατί μας έδινε τα πιο χαρακτηριστικά αποτελέσματα για τις μεταβολές του sensitivity και του specificity ανάλογα με την τάξη του μοντέλου). Στο διάγραμμα βλέπουμε ότι το sensitivity του μοντέλου αυξάνει μέχρι η τάξη του μοντέλου να γίνει 2 και έπειτα αρχίζει να πέφτει, από τη άλλη το specificity αυξάνει συνεχώς. Αυτό που επίσης παρατηρούμε είναι ότι και το $3^{ης}$ τάξης μοντέλο μας δίνει αρκετά καλές τιμές sensitivity και specificity, για μεγαλύτερης όμως τάξης βλέπουμε ότι η διαχωριστική ικανότητα του μοντέλου μας χάνεται εντελώς καθώς το sensitivity μειώνεται ραγδαία και το specificity αυξάνει, που σημαίνει ότι ανιχνεύει όλες τις πρωτεΐνες μας ως μη διαμεμβρανικά β-βαρέλια. Θα πρέπει να τονίσουμε επίσης εδώ ότι ο αλγόριθμος SPST δεν λειτουργεί για $0^{ης}$ τάξης μοντέλο, γι' αυτό και δεν έχει συμπεριληφθεί στο διάγραμμα, ενώ ο αλγόριθμος PST λειτουργεί.

Σε κάποιες παραλλαγές βέβαια η εικόνα της μεταβολής του sensitivity και του specificity ανάλογα με την τάξη του μοντέλου είναι αρκετά διαφορετική. Συγκεκριμένα το sensitivity και του specificity αυξάνουν ως ένα σημείο και μετά δεν μεταβάλλονται πλέον ανάλογα με τις μεταβολές της τάξης, έχουμε δηλαδή μια γραμμή, η οποία αυξάνει ως ένα ορισμένο σημείο και μετά είναι μια ευθεία γραμμή η οποία δεν μεταβάλλεται με την αλλαγή της τάξης του μοντέλου (διάγραμμα 3.2). Αυτό συμβαίνει γιατί ορισμένοι συνδυασμοί παραμέτρων συντελούν στο να "κοπεί" το δέντρο, αν μπορούμε να χρησιμοποιήσουμε αυτό τον όρο, δηλαδή αναγκάζουν το δέντρο να έχει συγκεκριμένο ύψος (βλέπε 2.8.1.2 Θεωρία PST), με αποτέλεσμα το δέντρο να μεγαλώνει πάντα ως ένα σημείο και από κει και πέρα να μην μεταβάλλεται

πλέον, ανεξαρτήτως αν η τάξη του μοντέλου αυξάνει. Συνήθως επιλέγουμε τέτοιους συνδυασμούς παραμέτρων γιατί παρουσιάζουν μεγάλα ποσοστά επιτυχίας στην αναγνώριση των δομών που μας ενδιαφέρουν.

Πίνακας 3.12: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο SPST για τις 2 παραλλαγές που περιγράψαμε παραπάνω και διαπιστώσαμε ότι βγάζουν καλύτερα αποτελέσματα για τα διαμεμβρανικά β-βαρέλια.

Αποτελέσματα με τον SPST αλγόριθμο						
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**
	Nmin	r_max	gamma	n_mean	order	***SE***
Αποτελέσματα για το "glob" σύνολο πρωτεϊνών						
1η παραλλαγή	1	1.5	0.001	80	1	91.7%
2η παραλλαγή	3	3.8	0.001	80	1	88%
Αποτελέσματα για το "omp121" σύνολο πρωτεϊνών						
1η παραλλαγή	1	1.5	0.001	80	1	66.1%
2η παραλλαγή	3	3.8	0.001	80	1	65.3%

Πίνακας 3.13: Αποτελέσματα της μεθόδου μας με τον αλγόριθμο PST για τις 2 παραλλαγές που περιγράψαμε παραπάνω και διαπιστώσαμε ότι βγάζουν καλύτερα αποτελέσματα για τα διαμεμβρανικά β-βαρέλια.

Αποτελέσματα με τον PST αλγόριθμο						
Προγράμματα	**Παράμετροι**					**Αποτελέσματα**
	Pmin	alpha	gamma	p_ratio	order	
Αποτελέσματα για το "glob" σύνολο πρωτεϊνών						
1η παραλλαγή	0.05	0	0.001	1.05	3	88%
2η παραλλαγή	0.05	0	0.001	1.05	3	89%
Αποτελέσματα για το "omp121" σύνολο πρωτεϊνών						
1η παραλλαγή	1	1.5	0.001	80	1	69.4%
2η παραλλαγή	1	1.5	0.001	80	1	66.93%

Παραπάνω παρουσιάζονται (Πίνακες 3.12 – 3.13) κάποια επιπλέον αποτελέσματα που προκύπτουν όταν τρέχουμε το πρόγραμμα μας έναντι δυο επιπλέον συνόλων πρωτεϊνών, που ονομάζουμε αντίστοιχα "glob" και "omp121". Το

πρώτο σύνολο, "glob", αποτελείται από μη διαμεμβρανικές πρωτεΐνες που δεν παρουσιάζουν διαμόρφωση β-βαρελιού και το δεύτερο σύνολο αποτελείται από διαμεμβρανικά β-βαρέλια.

4. ΣΥΖΗΤΗΣΗ-ΣΥΜΠΕΡΑΣΜΑΤΑ

Οι πρωτεΐνες αποτελούν το μεγαλύτερο μέρος της ξηρής μάζας ενός κυττάρου και επιτελούν όλες σχεδόν τις κυτταρικές λειτουργίες. Ένα μεγάλο ποσοστό των πρωτεϊνών αποτελούν οι διαμεμβρανικές πρωτεΐνες, οι οποίες πέρα από τις ζωτικές για το κύτταρο λειτουργίες που επιτελούν, αποτελούν και στόχους φαρμάκων γι' αυτό και η μελέτη τους χρήζει ιδιαίτερης σημασίας. Ένα σημαντικό ποσοστό, απ' την άλλη μεριά, των διαμεμβρανικών πρωτεϊνών αποτελούν πρωτεΐνες των οποίων τα διαμεμβρανικά τμήματα έχουν τη δομή β-βαρελιού (β-barrel) και σπειροειδούς σπειράματος (coiled-coil). Η ανάλυση όμως με το χέρι του ήδη μεγάλου αριθμού των γνωστών πρωτεϊνών μέχρι σήμερα, ο οποίος αυξάνει κιόλας με εκθετικούς ρυθμούς, είναι πρακτικά αδύνατη. Για το λόγο αυτό η ανάπτυξη αλγορίθμων για τον προσδιορισμό των πρωτεϊνών αυτών κρίνεται τόσο σημαντική. Επιπλέον καθώς ο αριθμός των κρυσταλλογραφικώς λυμένων δομών άρχισε να αυξάνει, έγινε φανερό ότι το πρόβλημα της πρόγνωσης των διαμεμβρανικών β-βαρελιών αλλά και των α-ελίκων, και κατ' επέκταση των σπειροειδών σπειρωμάτων, ήταν πιο σύνθετο από τον απλό εντοπισμό της εναλλαγής υδρόφοβων-πολικών καταλοίπων, άρα η ανάλυση με το χέρι έγινε αυτόματα πολύ πιο σύνθετη και χρονοβόρα.

Με την έκρηξη της Βιοπληροφορικής τη δεκαετία του '90, νέες μέθοδοι Μηχανικής Μάθησης (Machine Learning), όπως τα Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks-ANNs) και τα Κρυφά Μοντέλα Markov (Hidden Markov Models-HMMs), υιοθετήθηκαν για να επιλύσουν γνωστά βιολογικά προβλήματα, ανάμεσα τους και αυτά. Οι μέθοδοι αυτές, είναι σε γενικές γραμμές ικανές να αναγνωρίζουν μη γραμμικές συσχετίσεις των αμινοξέων σε μια πρωτεΐνη, και αποδίδουν καλύτερα από απλές στατιστικές τεχνικές και αναλύσεις βασισμένες στη χρήση φυσικοχημικών παραμέτρων και αμινοξικής σύστασης. Επιπλέον, οι μαθηματική θεμελίωση και ο φορμαλισμός που παρέχουν, τις κάνει να είναι πιο σταθερές και να αποτελούν ένα ασφαλέστερο σημείο εκκίνησης για την αντιμετώπιση δυσεπίλυτων προβλημάτων.

Στην παρούσα πτυχιακή εργασία επιλέξαμε να ασχοληθούμε με μία κατηγορία μεθόδων Μηχανικής Μάθησης, τα Μαρκοβιανά Μοντέλα, τα οποία έχουν

χρησιμοποιηθεί με επιτυχία κατά το παρελθόν για την πρόγνωση δομών και την ταξινόμηση άγνωστων πρωτεϊνών. Στα πλαίσια του θέματος που επιλέξαμε κάναμε αρχικά μια ανασκόπηση στα είδη των Μαρκοβιανών Μοντέλων που υπάρχουν και στη συνέχεια συγκεντρώσαμε το ενδιαφέρον μας στα Μαρκοβιανά Μοντέλα Μεταβλητής Μνήμης (VLMCs – Variable Length Markov Models) και αναπτύξαμε ένα πρόγραμμα το οποίο συνδυάζει με μοναδικό τρόπο:

1. ένα μοντέλο VLMCs, PST ή SPST
2. κατάλληλους αλγόριθμους εκπαίδευσης του μοντέλου και
3. εκπαίδευση πάνω σε δεδομένα τα οποία έχουν προέλθει από παρατηρήσεις των γνωστών δομών, μέσω της οποίας βελτιστοποιείται η απόδοση του μοντέλου.

Γενικά, δείξαμε ότι αξιοποιώντας τα δομικά χαρακτηριστικά των πρωτεϊνών αυτών, μπορούμε να εφαρμόσουμε τα VLMCs με αρκετά μεγάλη επιτυχία και σε άλλα είδη πρωτεϊνών όπως οι διαμεμβρανικές πρωτεΐνες και συγκεκριμένα αυτές που έχουν διαμόρφωση β-βαρελιών (β-barrel) αλλά και σε άλλα είδη όπως οι ινώδεις πρωτεΐνες με δομή σπειροειδούς σπειράματος (coiled-coils). Το γεγονός αυτό είναι ιδιαίτερα σημαντικό καθώς τα VLMCs έχουν εφαρμοστεί, με μεγάλη ομολογούμενος επιτυχία, μόνο σε οικογένειες πρωτεϊνών, που παρουσιάζουν μεγάλη ομολογία μεταξύ τους σε αντίθεση με τα διαμεμβρανικά β-βαρέλια και σπειροειδή σπειράματα, που παρουσιάζουν μικρότερη συντήρηση. Το πιο σημαντικό ίσως αποτέλεσμα που προέκυψε από αυτήν την πτυχιακή εργασία είναι η δημιουργία του προγράμματος που στηρίζεται στα VLMCs και κάνει πρόγνωση των διαμεμβρανικών πρωτεϊνών με δομή β-βαρελιού και των πρωτεϊνών με δομή σπειροειδούς σπειράματος.

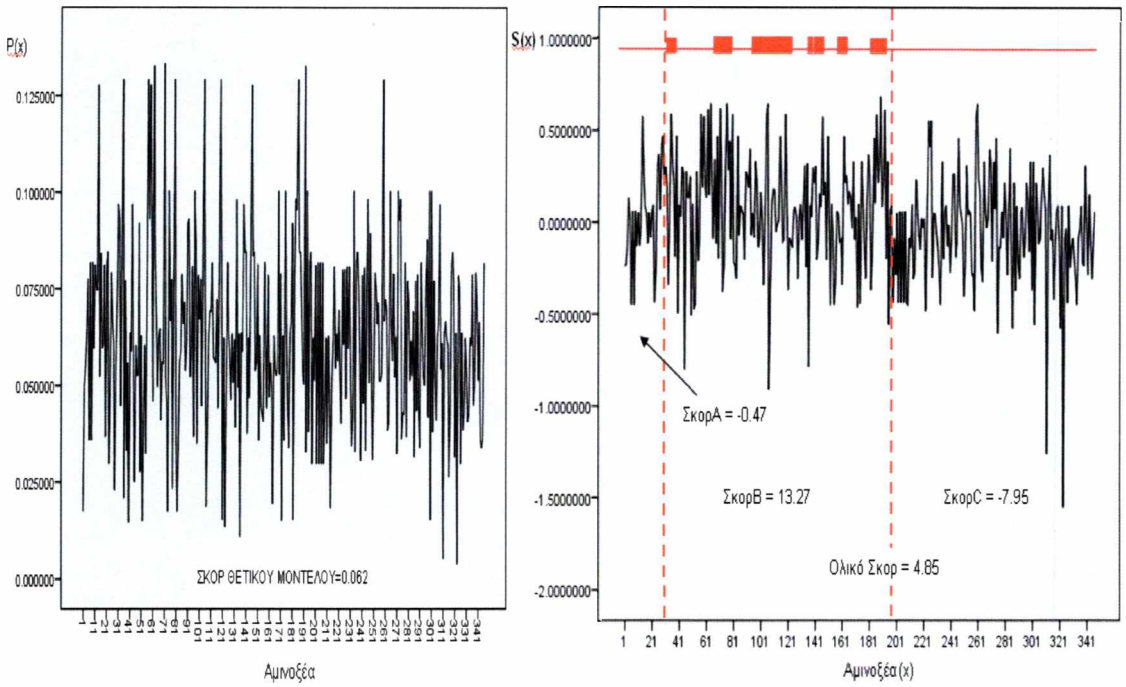
Η δημιουργία του προγράμματος είναι τόσο σημαντική για τρεις λόγους. Ο πρώτος είναι ο χρόνος που απαιτείται για να δημιουργηθεί το μοντέλο και να εκπαιδευθεί, ο οποίος ανέρχεται σε ελάχιστα λεπτά, καθώς το πρόγραμμα χρησιμοποιώντας τον PST ή τον SPST αλγόριθμο δημιουργεί με αυτόματο τρόπο το μοντέλο μαθαίνοντας από τις ακολουθίες εκπαίδευσης, με μέγιστη ανθρώπινη παρέμβαση τον καθορισμό των 5 παραμέτρων του μοντέλου. Αυτό είναι πολύ σημαντικό δεδομένου ότι η δημιουργία ενός HMM μπορεί να πάρει π.χ. και 2 χρόνια, καθώς πρέπει ο ερευνητής να υπολογίσει της πιθανότητες του μοντέλου πριν αρχίσει

το σχεδιασμό του. Επιπλέον η εκπαίδευση ενός HMM είναι πολύ χρονοβόρα, όταν η εκπαίδευση ενός VLMC διαρκεί μόλις λίγα λεπτά. Ο δεύτερος λόγος αφορά την τάξη του μοντέλου, καθώς τα VLMCs δεν παρουσιάζουν κάποιο περιορισμό στην τάξη του μοντέλου, το πρόγραμμα μας θα μπορούσε να τρέξει για οποιασδήποτε τάξης μοντέλο, παρόλο που διαπιστώσαμε ότι έως και $3^{ης}$ τάξης μόνο βγάζει ικανοποιητικά αποτελέσματα για τα διαμεμβρανικά β-βαρέλια και τα σπειροειδή σπειράματα. Γεγονός που οφείλεται στο ότι οι πρωτεΐνες αυτές δεν παρουσιάζουν μεγάλη συντήρηση, δηλαδή μεγάλα σε μήκος κοινά τμήματα, όπως παρουσιάζουν οι οικογένειες πρωτεϊνών, με αποτέλεσμα για μεγαλύτερης τάξης μοντέλο να αποτυγχάνει να πιάνει τέτοιες περιοχές. Ο τελευταίος και βασικότερος λόγος είναι τα αποτελέσματα μας, τα οποία δείχνουν ότι τα ποσοστά επιτυχίας κυμαίνονται κατά μέσο όρο κοντά στο 92%, τόσο για τις διαμεμβρανικές πρωτεΐνες με δομή β-βαρελιού όσο και για αυτές τις πρωτεΐνες με δομή σπειροειδούς σπειράματος, ποσοστό που ξεπερνάμε κιόλας με κάποιες παραλλαγές. Επιπλέον σε σύγκριση που έγινε με τα υπάρχοντα υπολογιστικά εργαλεία, που κάνουν πρόγνωση διαμεμβρανικών β-βαρελιών, διαπιστώσαμε ότι βγάζουμε καλύτερα ή ίδια αποτελέσματα με αυτά, με εξαίρεση το PRED-TMBB, που βγάζει τα καλύτερα αποτελέσματα.

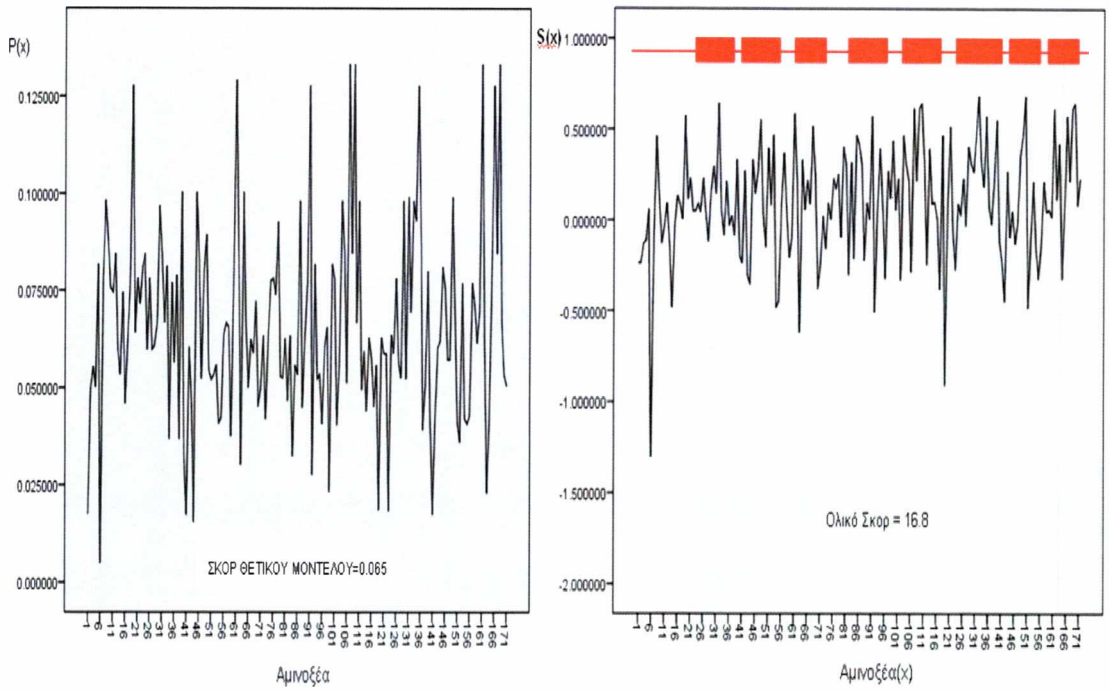
Επιπρόσθετα λόγω των πλεονεκτημάτων που παρουσιάζει, η μέθοδος μας θα μπορούσε να εφαρμοστεί και για πρόβλεψη άλλων πρωτεϊνών. Τέλος, λόγω της ευελιξίας που παρουσιάζει θα μπορούσε να χρησιμοποιηθεί μελλοντικά ώστε να κάνει πρόγνωση της θέσης που εμφανίζεται η δομή που μας ενδιαφέρει, δηλαδή θα μπορούσε να χρησιμοποιηθεί για πρόγνωση της τοπολογίας των πρωτεϊνών. Έτσι με τη μέθοδο αυτή, θα μπορούσε να προβλέπεται με ακρίβεια η τοποθεσία των διαμεμβρανικών β-κλώνων αλλά και των σπειροειδών σπειραμάτων, όπως φαίνεται στις εικόνες 4.1 – 4.6.

Στις εικόνες αυτές στα αριστερά φαίνεται το διάγραμμα με τις πιθανότητες των αμινοξέων βάση του θετικού μοντέλου, δηλ. του μοντέλου που δημιουργήσαμε με τον PST αλγόριθμο, και δεξιά το διάγραμμα με τις πιθανότητες των αμινοξέων που προκύπτουν από το πρόγραμμα μας, έχοντας χρησιμοποιήσει για τη δημιουργία του αρνητικού μοντέλου τις τυχαίες πιθανότητες εμφάνισης των αμινοξέων όπως

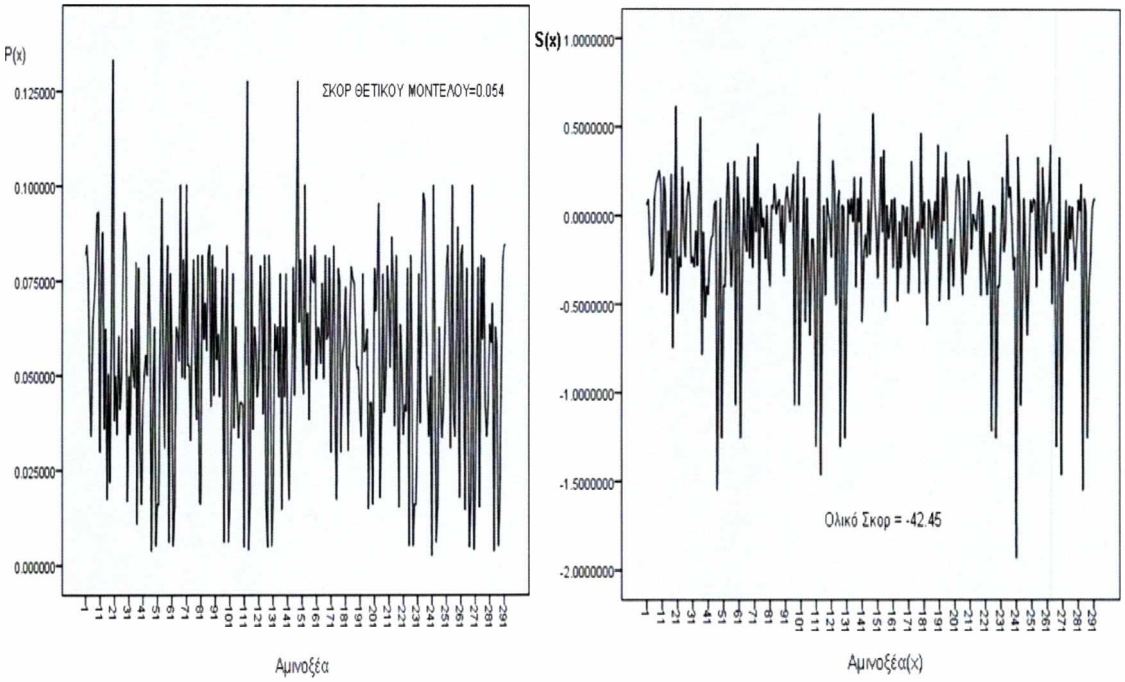
αυτές υπολογίστηκαν από τη βάση δεδομένων Swiss-Prot (βλ. 2η παραλλαγή). Στον άξονα y'y απεικονίζονται οι πιθανότητες των αμινοξέων P(x) στις εικόνες αριστερά και το σκορ των αμινοξέων S(x) στις εικόνες δεξιά και στον άξονα x'x απεικονίζεται ο αριθμός των αμινοξέων της αντίστοιχης ακολουθίας. Επιπλέον με T συμβολίζουμε τα τμήματα της ακολουθίας που δεν παρουσιάζουν τη διαμόρφωση που μας ενδιαφέρει, με B τα τμήματα της ακολουθίας με διαμόρφωση β-βαρελιού και με C τα τμήματα της ακολουθίας με διαμόρφωση σπειροειδούς σπειράματος. Για καθένα από τα τμήματα αυτά υπολογίζουμε σε κάθε διάγραμμα το αντίστοιχο σκορ και το αντίστοιχο συνολικό της κάθε πρωτεΐνης. Επιπλέον στο διάγραμμα στα δεξιά έχει σημειωθεί για τα β-βαρέλια η θέση των κλόνων καθώς και το τέλος του β-βαρελιού, ενώ για τις πρωτεΐνες με δομή σπειροειδούς σπειράματος έχουν σημειωθεί οι θέσεις της αρχής και του τέλους του σπειροειδούς σπειράματος, όπως αυτές έχουν υπολογιστεί πάλι από τη βάση δεδομένων Swiss-Prot.



Εικόνα 4.1: Πρόβλεψη των διαμεμβρανικών β-κλώνων για τη διαμεμβρανική πρωτεΐνη OMPA_ECOLI.



Εικόνα 4.2: Πρόβλεψη των διαμεμβρανικών β-κλώνων για τη διαμεμβρανική πρωτεΐνη OMPX_ECOLI.



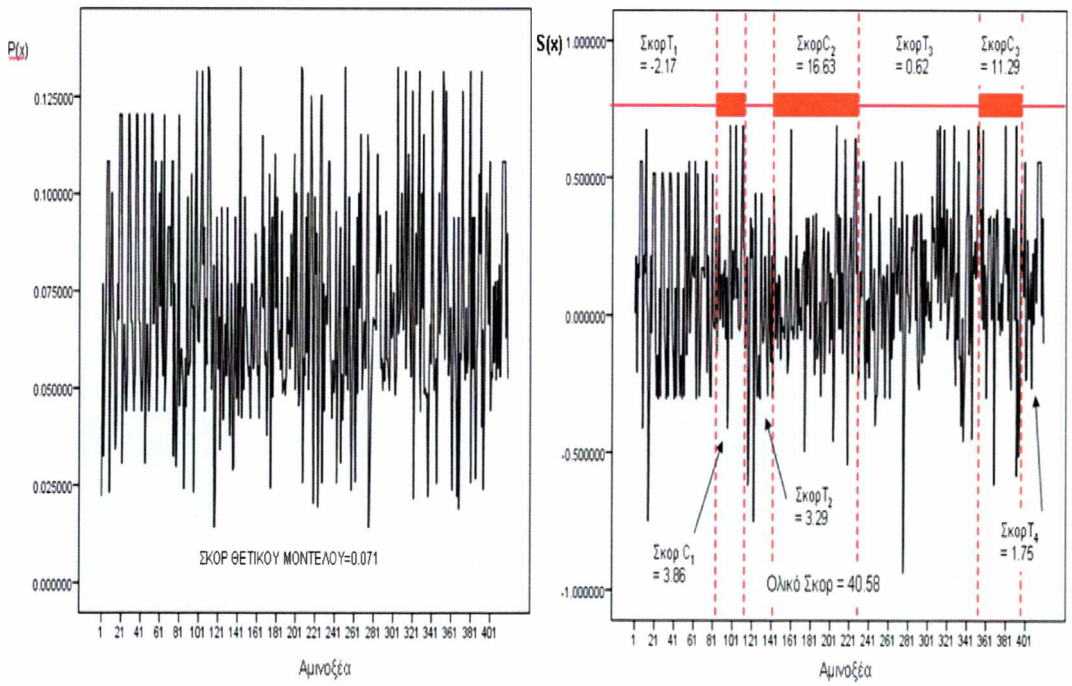
Εικόνα 4.3: Πρόβλεψη των διαμεμβρανικών β-κλώνων για τη μη διαμεμβρανική πρωτεΐνη 19HC_A.

Από τα παραπάνω διαγράμματα διαπιστώνουμε ότι στις διαμεμβρανικές που έχουν διαμόρφωση β-βαρελιού (Εικόνες 4.1-4.2) οι πιθανότητες εμφάνισης των αμινοξέων είναι μεγαλύτερες συγκριτικά με τη μη διαμεμβρανική πρωτεΐνη (Εικόνα 4.3). Επιπλέον στην Εικόνα 4.1 που το β-βαρέλι ξεκίνα από το 27^{o} αμινοξύ και πηγαίνει ως το 190^{o} αμινοξύ, διαπιστώνουμε ότι οι πιθανότητες σε αυτήν την περιοχή είναι υψηλότερες συγκριτικά με την υπόλοιπη ακολουθία και επιπλέον παρατηρούμε ότι προς το τέλος της δομής υπάρχει μια απότομη πτώση της πιθανότητας. Υπάρχουν ενδείξεις ότι, αυτό ίσως οφείλεται, στην παρουσία ενός αρωματικού καταλοίπου, συνήθως Φαινυλαλανίνης, στην τελευταία θέση του καρβοξυτελικού β-κλώνου του βαρελιού, η οποία είναι σημαντική για τη σωστή συγκρότηση αυτού και την εισαγωγή του στη διπλοστιβάδα (Struyve et al., 1991). Συγκεκριμένα υπάρχει ένα πρότυπο που περιγράφει αυτό το καρβοξυτελικό β-κλώνο, το οποίο είναι:

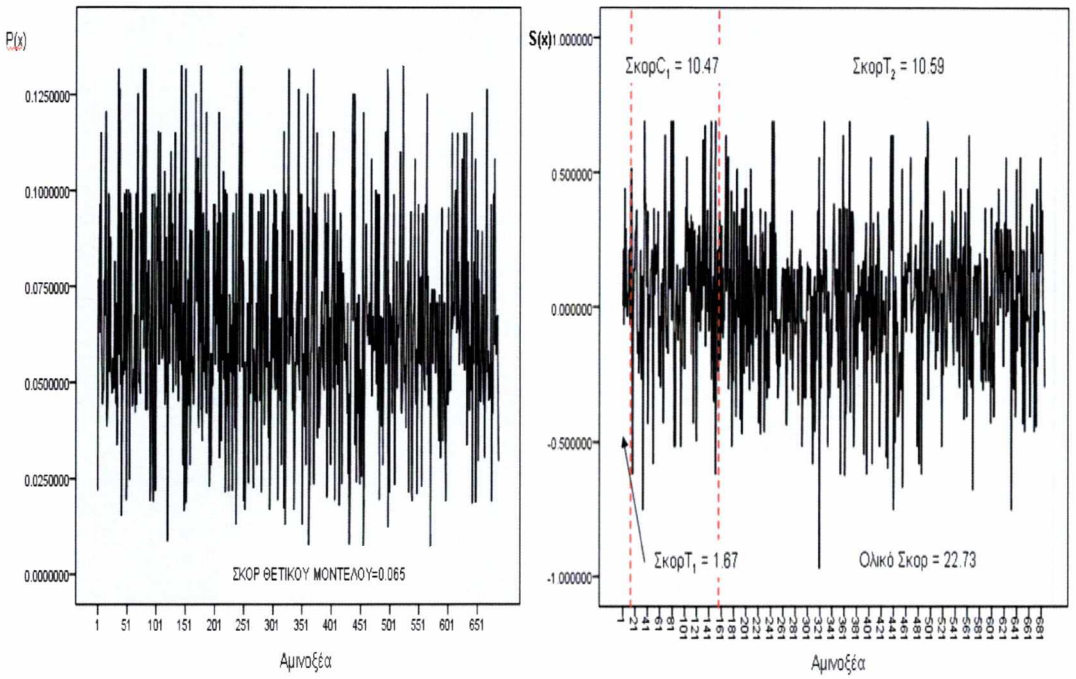
`.{100} [^C] [YFWKLHVITMAD] [^C] [YFWKLHVITMAD] [^C] [YFWKLHVITMAD] [^C] [YFWKLHVITMAD] [^C] [FYW]`

Το συγκεκριμένο πρότυπο, είναι αρκετά εύκαμπτο καθώς επιτρέπει την παρουσία αρκετών καταλοίπων `(YFWKLHVITMAD)`, στις θέσεις που αντικρίζουν τη μεμβράνη, ενώ από την άλλη επιτρέπει όλα τα κατάλοιπα εκτός Κυστεΐνης στις εσωτερικές θέσεις του βαρελιού. Πρέπει εδώ να τονίσουμε, ότι η Κυστεΐνη, δεν είναι παρούσα σε καμία διαμεμβρανική περιοχή των πρωτεϊνών με γνωστή δομή και επιπλέον έχει μεγαλύτερη προτίμηση για τις σφαιρικές πρωτεΐνες (Liu et al., 2003). Η πληροφορία αυτή είναι πολύ σημαντική καθώς το πρότυπο αυτό θα μπορούσε μελλοντικά να ενσωματωθεί στο πρόγραμμά μας ώστε να βελτιωθεί η ανίχνευση της τοπολογίας του βαρελιού.

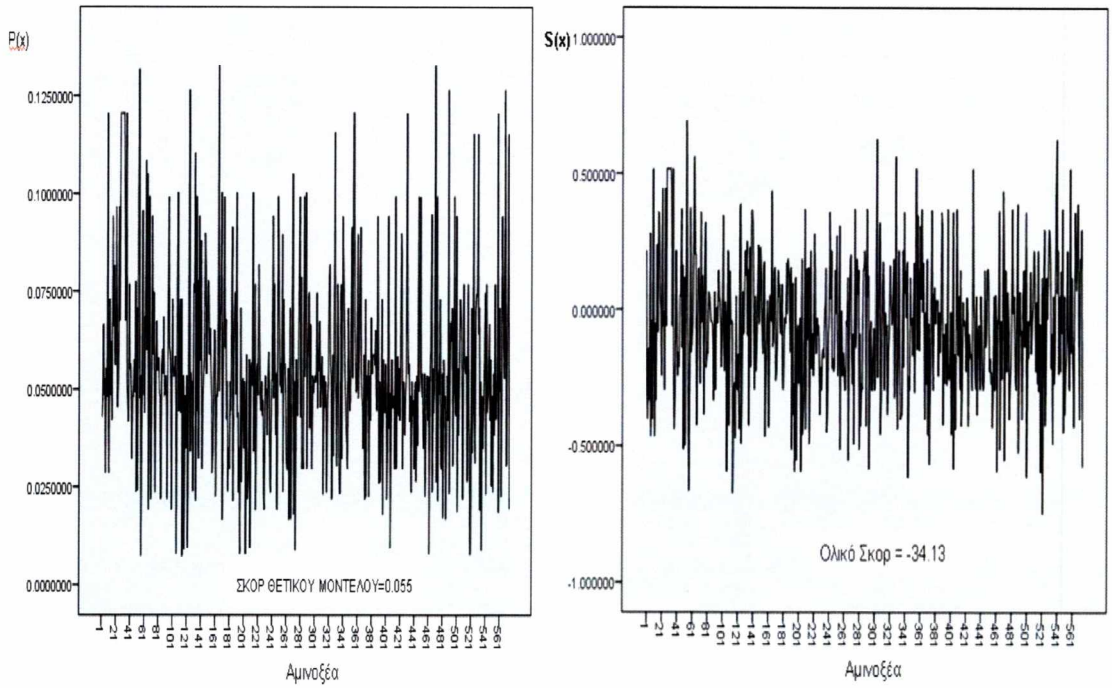
Θα μπορούσαμε έτσι να χρησιμοποιήσουμε τη μέθοδο μας για προσδιορισμό πρώτα της (κατά προσέγγιση) τοποθεσίας του διαμεμβρανικού β-βαρελιού, και κατόπιν θα μπορούσαμε να επιχειρούσε πρόγνωση των διαμεμβρανικών β-κλώνων. Με αυτόν τον τρόπο, θα κατορθώναμε να αυξήσουμε κατά πολύ τα ποσοστά της επιτυχούς πρόγνωσης, τόσο όσον αφορά τα διαμεμβρανικά τμήματα αλλά και τη διάκριση των πρωτεϊνών της εξωτερικής μεμβράνης, ειδικά σε περιπτώσεις πρωτεϊνών με πολλαπλά αυτοτελή στοιχεία (domains), εκ των οποίων ένα μόνο απαρτίζει το διαμεμβρανικό β-βαρέλι.



Εικόνα 4.4: Πρόβλεψη των σπειροειδών σπειραμάτων για την πρωτεΐνη K1C4_XENLA.



Εικόνα 4.5: Πρόβλεψη των σπειροειδών σπειραμάτων για την πρωτεΐνη KNLC_STRPU.



Εικόνα 4.6: Πρόβλεψη των σπειροειδών σπειραμάτων για την πρωτεΐνη 4DPV, που δεν παρουσιάζει διαμόρφωση σπειροειδούς σπειράματος.

Η μέθοδος όπως διαπιστώνουμε παραπάτω (Εικόνες 4.4 - 4.6) μπορεί να χρησιμοποιηθεί επιλέον και για πρόγνωση της τοπολογίας των τμημάτων μιας πρωτεΐνης με διαμόρφωση σπειροειδούς σπειράματος. Βλέπουμε παραπάνω ότι τα σκορ στα τμήματα των πρωτεϊνών που παρουσιάζουν αυτή τη διαμόρφωση παρουσιάζουν σημαντική διαφορά στο σκορ από τα υπόλοιπα τμήματα που δεν την παρουσιάζουν.

5. ΒΙΒΛΙΟΓΡΑΦΙΑ

Abe, N. and Warmuth, M. (1992) On the computational complexity of approximating distributions by probabilistic automata, *Machine Learning*, **9**, 205–260.

Agresti, A. (2002) *Categorical Data Analysis*. John Wiley & Sons.

Alexandersson, M., Cawley, S. and Pachter, L. (2003) SLAM: cross-species gene finding and alignment with a generalized pair hidden Markov model, *Genome Res*, **13**, 496-502.

Altschul, S.F., Gish, W., Miller, W., Myers, E.W. and Lipman, D.J. (1990) Basic local alignment search tool, *J Mol Biol*, **215**, 403-410.

Anfinsen, C.B. (1961) Genetic control of protein structure in bacteriophages, *Fin Lakaresallsk Handl*, **20**, 634-640.

Audic, S. and Claverie, J.M. (1998) Self-identification of protein-coding regions in microbial genomes. Proc, *Natl Acad Sci U S A*, **95**, 10026-10031.

Bagos, P.G., Liakopoulos, T.D., Spyropoulos, I.C. and Hamodrakas, S.J. (2004a) PRED-TMBB: a web server for predicting the topology of beta-barrel outer membrane proteins, *Nucleic Acids Res*, **32**, W400-404.

Bagos, P.G., Liakopoulos, T.D. and Hamodrakas, S.J. (2004b) Finding beta-barrel outer membrane proteins with a markov chain model, *WSEAS Transactions on Biology and Biomedicine*, **1**, 186-189.

Baldi, P., Chauvin, Y., Hunkapiller, T. and McClure, M. (1994) Hidden Markov models of biological primary sequence information, *Proc. Natl. Acad. Sci. USA*, **91**, 1059-1063.

Bateman, A., Coin, L., Durbin, R., Finn, R.D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E.L., Studholme, D.J., Yeats, C. and Eddy, S.R. (2004) The Pfam protein families database, *Nucleic Acids Res*, **32**, D138-141.

Bates, P.A. and Sternberg, M.J.E. (1991) Model building by comparison at CASP3: using expert knowledge and computer automation, *Proteins*, **Suppl 3**, 47–54.

Batzoglou, S., Pachter, L., Mesirov, J.P., Berger, B. and Lander, E.S. (2000) Human and mouse gene structure: comparative analysis and application to exon prediction, *Genome Res*, **10**, 950-958.

Baum, L.E., Petrie, T., Soules, G. and Weiss, N. (1970) A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains, *Ann. Math. Stat.*, **41**, 164-171.

Bejerano, G. and Yona, G. (2001) Variations on probabilistic suffix trees: statistical modeling and prediction of protein families, *Bioinformatics*, **17**, 23-43.

Berchtold, A. and Raftery, E.A. (2002) The Mixture Transition Distribution Model for High-Order Markov Chains and Non-Gaussian Time Series *Statistical Science*, **17**, 328–356.

Berman, H.M., Battistuz, T., Bhat, T.N., Bluhm, W.F., Bourne, P.E., Burkhardt, K., Feng, Z., Gilliland, G.L., Iype, L., Jain, S., Fagan, P., Marvin, J., Padilla, D., Ravichandran, V., Schneider, B., Thanki, N., Weissig, H., Westbrook, J.D. and Zardecki, C. (2002) The Protein Data Bank, *Acta Crystallogr D Biol Crystallogr*, **58**, 899-907.

Berven, F., Flikka, K., Jensen, H. and Eidhammer, I. (2004) BOMP: a program to predict integral beta-barrel outer membrane proteins encoded within genomes of Gram-negative bacteria, *Nucleic Acids Res.*, **32**, 394-399.

Blackwell, D. and Koopmans, L. (1957) On the identifiable problem for functions of finite Markov chains, *Ann. Math. Stat.*, **28**, 1011-1015.

Borodovsky, M. and Peresetsky, A. (1994) Deriving non-homogeneous DNA Markov chain models by cluster analysis algorithm minimizing multiple alignment entropy, *Comput Chem*, **18**, 259-267.

Bühlmann, P. (2000) Model selection for variable length Markov chains and tuning the context algorithm, *Annals of the Institute of Statistical Mathematics*, **52**, 287–315.

Bühlmann, P. and Wyner, A.J. (1999) Variable length Markov chains., *Ann. Math. Stat.*, **27**, 480–513.

Burge, C. and Karlin, S. (1997) Prediction of complete gene structures in human genomic DNA, *J Mol Biol*, **268**, 78-94.

Burke, C.J. and Rosenblatt, M. (1958) A Markovian function of a Markov chain, *Ann. Math. Stat.*, **29**, 1112-1120.

Crick, F.H. (1958) On protein synthesis, *Symp Soc Exp Biol*, **12**, 138-163.

Crick, F.H., Barnett, L., Brenner, S. and Watts-Tobin, R.J. (1961) General nature of the genetic code for proteins, *Nature*, **192**, 1227-1232.

Delorenzi, M. and Speed, T. (2002) An HMM model for coiled-coil domains and a comparison with PSM-based predictions, *Bioinformatics*, **18**, 617-625.

Durbin, R., Eddy, S.R., Krogh, A. and Mitchison, G. (1998) Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids, *Cambridge University Press, Cambridge, UK.*

Eddy, S.R. (1998) Profile hidden Markov models, *Bioinformatics*, **14**, 755-763.

Eddy, S.R., Mitchison, G. and Durbin, R. (1995) Maximum discrimination hidden Markov models of sequence consensus, *J Comput Biol*, **2**, 9-23.

Ellrott, K., Yang, C., Sladek, F.M. and Jiang, T. (2002) Identifying transcription factor binding sites through Markov chain optimization, *Bioinformatics*, **18 Suppl 2**, S100-S109.

Gardy, L.J., Spencer, C., Wang, K., Ester, M., Tusnady, E.G., Simon, I., Hua, S., de Fays, K., Lambert, C., Nakai, K. and Brinkman, S.L.F. (2003) PSORT-B: improving protein subcellular localization prediction for Gram-negative bacteria, Nucleic Acids Research, 31, 3613-3617

Gilbert, E.J. (1959) On the identifiablity problem for functions of finite Markov chains, *Ann. Math. Stat.*, **30**, 688-697.

Gillman, D. and Sipser, M. (1994) Inference and minimization of hidden Markov chains. *In Proc. Annual ACM Conf. Computational Learning Theory*. ACM Press, New Brunswick, New Jersey, U.S.A, 147–158.

Guigo, R., Agarwal, P., Abril, J.F., Burset, M. and Fickett, J.W. (2000) An assessment of gene prediction accuracy in large DNA sequences, *Genome Res*, **10**, 1631-1642.

Gusfield, D. (1997) *Algorithms on Strings, Trees, and Sequences*. Cambridge University Press, New York, U.S.A.

Heller, A. (1965) On stochastic processes derived from Markov chains, *Ann. Math. Stat.*, **36**, 1286-1291.

Hendrych, R. (1995) Permutierte Polygramme zur grammatischen Sprachmodellierung. University of Erlangen-Nuremberg.

Henikoff, J.G. and Henikoff, S. (1996) Using substitution probabilities to improve position-specific scoring matrices, *Comput Appl Biosci*, **12**, 135-143.

Hughey, R. and Krogh, A. (1996) Hidden Markov models for sequence analysis: extension and analysis of the basic method, *Comput. Appl. Biosci.*, **12**, 95-107.

Jelinek, F. (1990) Self–organized Language Modeling for Speech Recognition. In Waibel, A. and Lee, K.-F. (eds), *Readings in Speech Recognition*. Morgan Kaufmann Publishers Inc, San Mateo, 450–506.

Karplus, K. (1995) Evaluating regularizers for estimating distributions of amino acids, *Proc Int Conf Intell Syst Mol Biol*, **3**, 188-196.

Karplus, K., Barrett, C., Cline, M., Diekhans, M., Grate, L. and Hughey, R. (1999) Predicting protein structure using only sequence information, *Proteins*, **Suppl 3**, 121-125.

Karplus, K., Sjolander, K., Barrett, C., Cline, M., Haussler, D., Hughey, R., Holm, L. and Sander, C. (1997) Predicting protein structure using hidden Markov models, *Proteins*, **Suppl 1**, 134-139.

Korf, I., Flicek, P., Duan, D. and Brent, M.R. (2001) Integrating genomic homology into gene structure prediction, *Bioinformatics*, **17 Suppl 1**, S140-148.

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

Leonardi, F.G. (2006) A generalization of the PST algorithm: modeling the sparse nature of protein sequences, *Bioinformatics*, **22**, 1302-1307.

Liu, Q., Zhu, Y., Wang, B. and Li, Y. (2003) Identification of beta-barrel membrane proteins based on amino acid composition properties and predicted secondary structure, *Comput Biol Chem*, **27**, 355-361.

Meyer, I.M. and Durbin, R. (2002) Comparative ab initio prediction of gene structures using pair HMMs, *Bioinformatics*, **18**, 1309-1318.

Mitchell, T.M. (1997) *Machine Learning*. McGraw Hill, New York.

Nadas, A., Nahamoo, D. and Picheny., M.A. (1988) On a model-robust training method for speech recognition, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **36**, 1432–1435.

Noguchi, T. and Akiyama, Y. (2003) PDB-REPRDB: a database of representative protein chains from the Protein Data Bank (PDB) in 2003, *Nucleic Acids Res*, **31**, 492-493.

Ou, Y.-Y., Gromiha, M.M., Chen, S.-A. and Suwa, M. (2008) TMBETADISC-RBF: Discrimination of beta-barrel membrane proteins using RBF networks and PSSM profiles, *Comput Biol Chem.*, **32**, 227-231

Pachter, L., Alexandersson, M. and Cawley, S. (2002) Applications of generalized pair hidden Markov models to alignment and gene finding problems, *J Comput Biol*, **9**, 389-399.

Park, K.-J., Gromiha, M.M., Horton, P. and Suwa, M. (2005) Discrimination of Outer Membrane Proteins using Support Vector Machines, *Bioinformatics*, **21**, 4223-4229.

Pearson, W.R. and Lipman, D.J. (1988) Improved tools for biological sequence comparison, *Proc. Natl. Acad. Sci. USA*, **85**, 2444-2448.

Phillips, G.J., Arnold, J. and Ivarie, R. (1987) Mono- through hexanucleotide composition of the Escherichia coli genome: a Markov chain analysis, *Nucleic Acids Res*, **15**, 2611-2626.

Rabiner, L.R. (1989) A tutorial on hidden Markov models and selected applications in speech recognition, *Proc. IEEE*, **77**, 257-286.

Raftery, A. (1985) A model for high-order Markov chains, *Journal of the Royal Statistical Society B*, **47**, 528–539.

Rissanen, J. (1983) A universal data compression system, *IEEE Trans. Inform. Theory*, **29(5)**, 656–664.

Ron, D., Singer, Y. and Tishby, N. (1996) The power of amnesia: Learning probabilistic automata with variable memory length, *Machine Learning*, **25**, 117–149.

Salzberg, S.L., Delcher, A.L., Kasif, S. and White, O. (1998) Microbial gene identification using interpolated Markov models, *Nucleic Acids Res*, **26**, 544-548.

Samudrala, R. and Moult, J. (1997) Handling context-sensitivity in protein structures using graph theory: bona fide prediction, *Proteins*, **Suppl 1**, 43-49.

Schukat-Talamazzini, E.G. (1995) *Automatische Spracherkennung*. Vieweg, Braunschweig.

Schukat-Talamazzini, E.G., Gallwitz, F., Harbeck, S. and Warnke, V. (1997) Rational interpolation of Maximum Likelihood predictors in stochastic language

modeling. *In Proc. European Conf. on Speech Communication and Technology*. Rhodes, Greece, 2731–2734.

Schulz, G.E. (2003) Transmembrane beta-barrel proteins, *Adv Protein Chem*, **63**, 47-70.

Sheynin, O. (1988) A Markov's work on probability, *Arch. Hist. Exact Sci.*, **39**, 337-377.

Smith, L., Yeganova, L. and Wilbur, W.J. (2003) Hidden Markov models and optimized sequence alignments, *Comput Biol Chem*, **27**, 77-84.

Struyve, M., Moons, M. and Tommassen, J. (1991) Carboxy-terminal phenylalanine is essential for the correct assembly of a bacterial outer membrane protein, *J Mol Biol*, **218**, 141-148.

von Heijne, G. (1999) Recent advances in the understanding of membrane protein assembly and structure, *Q Rev Biophys*, **32**, 285-307.

Warnke, V., Harbeck, S., Noth, E., Niemann, H. and Levit, M. (1999) Discriminative estimation of interpolation parameters for language model classifiers. *In Proc ICASSP-Proceedings of the Acoustics, Speech, and Signal Processing, on IEEE International Conference*. IEEE Computer Society, Phoenix, 525–528.

Wiehe, T., Gebauer-Jung, S., Mitchell-Olds, T. and Guigo, R. (2001) SGP-1: prediction and validation of homologous genes based on sequence alignments, *Genome Res*, **11**, 1574-1583.

Yuan, Z. (1999) Prediction of protein subcellular locations using Markov chain models, *FEBS Lett*, **451**, 23-26.

6. ΠΑΡΑΡΤΗΜΑ

6.1 Αλλαγές στον PST αλγόριθμο

Ο αλγόριθμος PST μας έδεινε τις πιθανότητες για κάθε αμινοξύ της πρωτεΐνης. Για να υπολογίσουμε το τη συνολική πιθανότητα μιας πρωτεΐνης κάναμε κάποιες μετατροπές στον κώδικα. Συγκεκριμένα τροποποιήσαμε την κλάση "*predict_entry*" του αρχείου *pst.c* (βλ. παρακάτω), η οποία μας δίνει την πιθανότητα του κάθε αμινοξέος. Με τις τροποποιήσεις που κάναμε κατορθώσαμε να λογαριθμίσουμε την πιθανότητα του κάθε αμινοξέος της πρωτεΐνης και στη συνέχεια να τις προσθέσουμε όλες μαζί ώστε να υπολογίσουμε έτσι την πιθανότητα της κάθε πρωτεΐνης.

```
void predict_entry (char *AB, pst_type T, char *S, const int lwrite,
                FILE *lfp, const int swrite, FILE *sfp, char *D)
{
  char c;
  int i;
  int lri;
  int lenS, sumdepth;
  pst_node *leafp;
  double loglike, persymbol,m,t[50000]; # ορισμός μεταβλητών m & t[]
```

ως m ορίζουμε την πιθανότητα κάθε αμινοξέος μιας πρωτεΐνης και ως t[] ορίζουμε τον πίνακα που κρατάει τις πιθανότητες όλων των αμινοξέων μιας πρωτεΐνης

```
  sumdepth = 0;
  loglike = 0.0;
  lenS = strlen(S);
  if (lwrite)
    fprintf(lfp, "%7d\t%s\t", lenS, D);

  for (i=0; i<lenS; i++)
    {
      t[0] = 0;                          # αρχικοποιεί το t
      c = S[i];
      S[i] = '\0';
      traverse_pst(AB, T, S, &leafp, &lri);
      persymbol = leafp->p_c[ABind(AB,c)]; # η πιθανότητα

      if (lwrite)
        if (persymbol==0) # αν η πιθανότητα ενός αμινοξέος είναι 0
        {                   τότε επειδή δεν ορίζεται το log(0) θέτει
          m=-10000;         την πιθανότητα του αμινοξέος ίση με -10000
```

```
        }

        else
        {
          m=log(persymbol);      # αν η πιθανότητα ενός αμινοξέος δεν είναι 0
                                   τότε λογαριθμίζει την πιθανότητα
        }

      t[i+1]=t[i]+m;           # αθροίζει όλες τις πιθανότητες των αμινοξέων για
                                 μια πρωτεΐνη

      sumdepth += (i - lri);
      loglike += log(persymbol);
      S[i] = c;
    }

    fprintf(lfp, "%1.8f\n", t[lenS]);    # τυπώνει το αποτέλεσμα
  if (swrite)
    fprintf(sfp, "%7d\t%6d\t%1.8f\t%-40s<\n",
lenS,sumdepth,loglike,D);
  return;
}
```

6.2 Προγράμματα που δημιουργήθηκαν

Για τη δημιουργία των τελικών προγραμμάτων δημιουργήσαμε διάφορα μικρότερα προγράμματα τα οποία εξυπηρετούν διάφορους σκοπούς.

Για την 1η παραλλαγή δημιουργήθηκαν τα εξής προγράμματα:

• ***crossval.pl*** , το οποίο κάνει crossvalidation τα σύνολα των πρωτεϊνών που χρησιμοποιούνται ταυτόχρονα για εκπαίδευση και έλεγχο. Συγκεκριμένα χωρίζει το δοθέν σύνολο πρωτεϊνών σε *n* υποσύνολα που τα καθορίζουμε εμείς. Στη συνέχεια εξαιρεί ένα υποσύνολο, εκπαιδεύει το μοντέλο με τα υπόλοιπα *n-1* σύνολα και κάνει τεστ στο υποσύνολο που εξαιρέθηκε. Η διαδικασία αυτή, που ονομάζεται crossvalidation, επαναλαμβάνεται για όλα τα *n* υποσύνολα. Το πρόγραμμα αυτό παίρνει σαν είσοδο το σύνολο πρωτεϊνών για crossvalidation, τον αριθμό *n* των υποσυνόλων, που επιθυμούμε να χωρίσουμε το σύνολα μας, το σύμβολο + για να μας δώσει το ποσοστό των πρωτεϊνών που ανιχνεύτηκαν σωστά και τις παραμέτρους που απαιτούνται για την εκπαίδευση του μοντέλου.

```
use POSIX;
```

```
$input1 ="$ARGV[0]";      # Σύνολο πρωτεϊνών για crossvalidation σε one-line
                            format
$size   ="$ARGV[1]";      # Αριθμός n των υποσυνόλων
$input2 ="$ARGV[2]";      # +

$q_min ="$ARGV[3]";       # Παράμετροι μοντέλου
$alpha="$ARGV[4]";
$gama="$ARGV[5]";
$p_ratio="$ARGV[6]";
$order="$ARGV[7]";

if ($input1=~/.*\/(.*)/)
{
      $test_file=$1;
}
else
{
      $test_file=$input1;
}

open FILE , "$input1";
$n=0;
while (<FILE>)
{
      if ($_=~/^>(.*)/)
      {

            $id=$1;
            $seq=<FILE>;
            push @seq, $seq;
            push @id, $id;
            $n++;
      }
}
close FILE;

print `mkdir TRAIN`;  # Φτιάχνει τους φακέλους TRAIN & RESULTS
print `mkdir RESULTS`;

open OUT, ">>RESULTS/RES";   #Δημιουργεί το αρχείο αποτελεσμάτων RES και
                               τυπώνει σε αυτό αρχικά τις παραμέτρους και
                               το n
print OUT "q_min=".$q_min."\t"."alpha=".$alpha."\n";
print OUT "gama=".$gama."\n"."p_ratio=".$p_ratio."\n";
print OUT "CLUSTERS=".$size."\t"."order=".$order."\n\n\n";

$nclust=ceil($n/$size);
#Παρακάτω κάνει τη διαδικασία του crossvalidation που περιγράψαμε παραπάνω
foreach $c(0..$nclust-1)
{
      open TEST, ">tmp.test.$c";
      open TRAIN, ">tmp.train.$c";
            foreach $i(0..$n-1)
      {
```

```
            if(  $i>=$c*$size and $i < ( $c+1 )*$size  )
            {

                  print  TEST ">$id[$i]\n$seq[$i]";

            }
            else
            {
                  print  TRAIN ">$id[$i]\n$seq[$i]";

            }
      }
      close TEST;
      close TRAIN;
      #Καλεί τα προγράμματα  ttrain_null-pst.pl & ttest_null.pl

      print  `perl  train-pst.pl  tmp.train.$c  $input2  $p_min  $alpha
               $gamma_min $p_ratio $order`;
      Print    `perl    test-pst.pl    tmp.test.$c    tmp.train.$c.pst
               $input2.pst`;

      print `mv tmp.test.$c TRAIN/ `;
      print `mv tmp.train.$c TRAIN/ `;
}

print `rm -r TRAIN/ `;

#Υπολογίζει το ποσοστό επιτυχίας
$number1=`grep -c ">.*-" RESULTS/RES`;
$number_2=`wc -l RESULTS/RES`;
$number2=$number_2-7;
$number3=$number2-$number1;


if ($input3 eq "+")
{
      $cros=$number3/$number2;
      $val=$cros*100;
      $val1=sprintf ("\n\n CROSSVALIDATION: %.1f \n\n\n", $val);
      open OUT, ">>RESULTS/RES";
      print OUT $val1;
      close OUT;

}
else
{
      $cros=$number1/$number2;
      $val=100 - $cros*100;
      $val1=sprintf ("\n\n CROSSVALIDATION: %.1f \n\n\n", $val);
      open OUT, ">>RESULTS/RES";
      print OUT $val1;
      close OUT;

}
```

• ***test-pst.pl,*** το οποίο καλείται από το *crossval.pl* και υπολογίζει την πιθανότητα για διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα αλλά και τις για πρωτεΐνες που δεν παρουσίαζαν αυτές τις διαμορφώσεις. Το πρόγραμμα αυτό δεν δέχεται τίποτα από το χρήστη σαν είσοδο καθώς καλείται όπως προαναφέραμε από το *crossval.pl* από το οποίο παίρνει σαν είσοδο το σύνολο των πρωτεϊνών για έλεγχο και το θετικό και το αρνητικό μοντέλο που έχουμε δημιουργήσει είτε με το PST είτε με το SPST αλγόριθμο.

```
$input="$ARGV[0]";          # Σύνολο πρωτεϊνών για έλεγχο
$file1="$ARGV[1]";          # Θετικό μοντέλο PST
$file2="$ARGV[2]";          # Αρνητικό μοντέλο SPST

#Υπολογίζει την πιθανότητα της πρωτεΐνης για το θετικό μοντέλο
print `../predict TRAIN/$file1 $input @ TRAIN/$input.pos @ @`;
#Υπολογίζει την πιθανότητα της πρωτεΐνης για το αρνητικό μοντέλο
print `../predict TRAIN/$file2 $input @ TRAIN/$input.neg @ @`;

open FILE1 , " TRAIN/$input.pos" ;
open FILE2 , " TRAIN/$input.neg" ;

while(<FILE1>)
{
if ($_=~/.*(>.*)\t(.*)/)    # Μας δίνει την πιθανότητα της κάθε πρωτεΐνης για
                              το θετικό μοντέλο
      {
            $log1=$2;
            $name=$1;
            push @name, $name;
            $log1{$name}=$log1;
      }
}
while(<FILE2>)    # Μας δίνει την πιθανότητα της κάθε πρωτεΐνης για το αρνητικό
                    μοντέλο
{
if ($_=~/.*(>.*)\t(.*)/)
      {
            $log2=$2;
            $name=$1;
            $log2{$name}=$log2;
      }
}

foreach $x(@name) # Υπολογίζει το σκορ της πρωτεΐνης αφαιρώντας από την
                    πιθανότητα της πρωτεΐνης για το αρνητικό μοντέλο την
                    πιθανότητα της πρωτεΐνης για το θετικό μοντέλο

{
```

```
        $logor=$log1{$x} - $log2{$x};
        print       "$x\t $logor\n";
        $output="RESULTS/RES";
        open FILEHANDLE,">>$output";
              print FILEHANDLE $x."\t".$logor."\n";
}

close FILE1;
close FILE2;
unlink "$test_file.pos" ;
unlink "$test_file.neg" ;
```

- ***train-pst.pl,*** το οποίο είναι ίδιο με το *train_null-pst.pl*, καλείται από το *crossval_null.pl* και δημιουργεί για λογαριασμό του το θετικό μοντέλο είτε με τον αλγόριθμο PST είτε με το αλγόριθμο SPST. Για την εκπαίδευση του θετικού μοντέλου χρησιμοποιεί το σύνολο με τις πρωτεΐνες που παρουσιάζουν τη διαμόρφωση που μας ενδιαφέρει (διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα). Το πρόγραμμα αυτό δεν δέχεται τίποτα από το χρήστη σαν είσοδο καθώς καλείται όπως προαναφέραμε από το *crossval_null.p*, από το οποίο παίρνει σαν είσοδο το σύνολο των πρωτεϊνών με τα διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα και τις παραμέτρους του μοντέλου.

```
$input1="$ARGV[0]";        # Σύνολο των πρωτεϊνών που παρουσιάζουν τη
                           τη διαμόρφωση που μας ενδιαφέρει (διαμεμβρανικά β-
                           βαρέλια ή σπειροειδή σπειράματα) (σε one-line format)
$q_min ="$ARGV[1]"; # Παράμετροι του θετικού μοντέλου
$alpha="$ARGV[2]";
$gama="$ARGV[3]";
$p_ratio= "$ARGV[4]";
$order="$ARGV[5]";

$input1="$ARGV[0]";        # Σύνολο των πρωτεϊνών που παρουσιάζουν τη
                           τη διαμόρφωση που μας ενδιαφέρει (διαμεμβρανικά β-
                           βαρέλια ή σπειροειδή σπειράματα) (σε one-line format)
$input2="$ARGV[1]";        # Σύνολο των πρωτεϊνών που δεν παρουσιάζουν τη
                           διαμόρφωση που μας ενδιαφέρει (μη διαμεμβρανικά β-
                           βαρέλια ή σπειροειδή σπειράματα) (σε one-line format)
$p_min ="$ARGV[2]"; # Παράμετροι του θετικού και αρνητικού μοντέλου
$alpha ="$ARGV[3]";
$gamma_min="$ARGV[4]";
$p_ratio="$ARGV[5]";
$order="$ARGV[6]";


if ($input1=~/.*\/(.*)/)
```

```
{
        $file1=$1;
}
else
{
        $file1=$input1;
}
if ($input2=~/.*\/(.*)/)
{
        $file2=$1;
}
else
{
        $file2=$input2;
}
#Καλεί τον PST αλγόριθμο για να δημιουργήσει τα δύο μοντέλα, αρνητικό και θετικό

print   `../train   ab.PROTEINS_BZX   $input1   @   @   TRAIN/$file1.pst
        TRAIN/STATISTICS:100   @   1   $p_min   $alpha   $gamma_min   $p_ratio
        $order`;

print   `../train   ab.PROTEINS_BZX   $input2   @   @   TRAIN/$file2.pst
        TRAIN/STATISTICS:100 @ 1 $p_min $alpha $gamma_min $p_ratio 0`;
```

Για την 2η παραλλαγή δημιουργήθηκαν τα εξής προγράμματα:

• ***crossval_null.pl*** , το οποίο κάνει crossvalidation τα σύνολα των πρωτεϊνών που χρησιμοποιούνται ταυτόχρονα για εκπαίδευση και έλεγχο. Συγκεκριμένα χωρίζει το δοθέν σύνολο πρωτεϊνών σε *n* υποσύνολα που τα καθορίζουμε εμείς. Στη συνέχεια εξαιρεί ένα υποσύνολο, εκπαιδεύει το μοντέλο με τα υπόλοιπα *n-1* σύνολα και κάνει τεστ στο υποσύνολο που εξαιρέθηκε. Η διαδικασία αυτή, που ονομάζεται crossvalidation, επαναλαμβάνεται για όλα τα *n* υποσύνολα. Το πρόγραμμα αυτό παίρνει σαν είσοδο το σύνολο πρωτεϊνών για crossvalidation, τον αριθμό *n* των υποσυνόλων, που επιθυμούμε να χωρίσουμε το σύνολο μας, το σύμβολο + για να μας δώσει το ποσοστό των πρωτεϊνών που ανιχνεύτηκαν σωστά και τις παραμέτρους που απαιτούνται για την εκπαίδευση του μοντέλου.

```
use POSIX;
$input1  ="$ARGV[0]";      # Σύνολο πρωτεϊνών για crossvalidation σε one-line
                             format
$size    ="$ARGV[1]";    # Αριθμός n των υποσυνόλων
$input2 ="$ARGV[2]";     # +

$q_min ="$ARGV[3]";      # Παράμετροι μοντέλου
```

```
$alpha="$ARGV[4]";
$gama="$ARGV[5]";
$p_ratio="$ARGV[6]";
$order="$ARGV[7]";

if ($input1=~/.*\/(.*)/)
{
      $test_file=$1;
}
else
{
      $test_file=$input1;
}

open FILE , "$input1";
$n=0;
while (<FILE>)
{
      if ($_=~/^>(.*)/)
      {

            $id=$1;
            $seq=<FILE>;
            push @seq, $seq;
            push @id, $id;
            $n++;
      }
}
close FILE;

print `mkdir TRAIN`;  # Φτιάχνει τους φακέλους TRAIN & RESULTS
print `mkdir RESULTS`;

open OUT, ">>RESULTS/RES";   #Δημιουργεί το αρχείο αποτελεσμάτων RES και
                                 τυπώνει σε αυτό αρχικά τις παραμέτρους και
                                 το n
print OUT "q_min=".$q_min."\t"."alpha=".$alpha."\n";
print OUT "gama=".$gama."\n"."p_ratio=".$p_ratio."\n";
print OUT "CLUSTERS=".$size."\t"."order=".$order."\n\n\n";

$nclust=ceil($n/$size);
#Παρακάτω κάνει τη διαδικασία του crossvalidation που περιγράψαμε παραπάνω
foreach $c(0..$nclust-1)
{
      open TEST, ">tmp.test.$c";
      open TRAIN, ">tmp.train.$c";
            foreach $i(0..$n-1)
      {
            if(  $i>=$c*$size and $i < ( $c+1 )*$size  )
            {
                  print  TEST ">$id[$i]\n$seq[$i]";
            }
            else
            {
                  print  TRAIN ">$id[$i]\n$seq[$i]";
```

```
            }
      }
      close TEST;
      close TRAIN;

#Καλεί τα προγράμματα  ttrain_null-pst.pl & ttest_null.pl
print  `perl  ttrain_null-pst.pl  tmp.train.$c  $q_min  $alpha  $gama
        $p_ratio $order`;
print   `perl ttest_null.pl tmp.test.$c tmp.train.$c.pst`;

print `mv tmp.test.$c TRAIN/`;
print `mv tmp.train.$c TRAIN/`;
print `rm -r TRAIN/`;
}

$number1=`grep -c ">.*-" RESULTS/RES`;
$number_2=`wc -l RESULTS/RES`;
$number2=$number_2-6;
$number3=$number2-$number1;

#Υπολογίζει το ποσοστό επιτυχίας, δηλαδή πόσες πρωτεΐνες ανιχνεύτηκαν σωστά
ως διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα
if ($input2 eq "+")
{
      $cros=$number3/$number2;
      $val=$cros*100;
      $val1=sprintf ("\n\n CROSSVALIDATION: %.1f \n\n\n", $val);
      open OUT, ">>RESULTS/RES";
            print OUT $val1;
}
close OUT;
```

- ***test_null-pst***, το οποίο δημιουργεί το αρνητικό μοντέλο με βάση τις τυχαίες πιθανότητες εμφάνισης των αμινοξέων όπως αυτές υπολογίστηκαν από τη βάση δεδομένων Swiss-Prot (Πίνακας 3.1), υπολογίζει την πιθανότητα των πρωτεϊνών που δεν παρουσιάζουν τη διαμόρφωση που μας ενδιαφέρει και επιπλέον υπολογίζει το ποσοστό επιτυχίας, δηλαδή πόσες πρωτεΐνες ανιχνεύτηκαν σωστά ως μη διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα. Το πρόγραμμα αυτό παίρνει σαν είσοδο το σύνολο των πρωτεϊνών που δεν παρουσιάζουν τη διαμόρφωση που μας ενδιαφέρει, το θετικό μοντέλο που έχουμε δημιουργήσει είτε με το PST είτε με το SPST αλγόριθμο και τις παραμέτρους με τις οποίες δημιουργήσαμε το θετικό μοντέλο, τις οποίες τις δίνουμε όχι για υπολογιστικούς σκοπούς αλλά για να τυπωθούν στο αρχείο των αποτελεσμάτων.

```
$input="$ARGV[0]";          # Σύνολο των πρωτεϊνών που δεν παρουσιάζουν τη
                            διαμόρφωση που μας (σε one-line format)
$file1="$ARGV[1]";          #Θετικό μοντέλο

$q_min ="$ARGV[2]";   # Παράμετροι με τις οποίες δημιουργήσαμε το θετικό
μοντέλο
$alpha="$ARGV[3]";
$gama="$ARGV[4]";
$p_ratio="$ARGV[5]";
$order="$ARGV[6]";
```

Παρακάτω δημιουργεί το αρνητικό μοντέλο με βάση τις τυχαίες πιθανότητες εμφάνισης των αμινοξέων όπως αυτές υπολογίστηκαν από τη βάση δεδομένων Swiss-Prot

```
%hash = (
'A' => '0.077',
'C' => '0.018',
'D' => '0.058',
'E' => '0.066',
'F' => '0.040',
'G' => '0.072',
'H' => '0.024',
'I' => '0.056',
'K' => '0.063',
'L' => '0.086',
'M' => '0.022',
'N' => '0.046',
'P' => '0.046',
'Q' => '0.040',
'R' => '0.049',
'S' => '0.062',
'T' => '0.057',
'V' => '0.068',
'W' => '0.015',
'Y' => '0.035'
);
```

#Υπολογίζει την πιθανότητα της πρωτεΐνης για το θετικό μοντέλο

```
print `../predict TRAIN_NEG/$file1 $input @ TRAIN_NEG/$input.pos @
@`;

open FILE1 , "TRAIN_NEG/$input.pos" ;
open FILE2, "$input";

while(<FILE1>) # Μας δίνει την πιθανότητα της κάθε πρωτεΐνης για το θετικό
μοντέλο
{
      if ($_=~/.*(>.*)\t(.*)/)
      {
            $log1=$2;
            $name=$1;
```

```
            push @name, $name;
            $log1{$name}=$log1;
      }
}

while(<FILE2>)   #Υπολογίζει την πιθανότητα της πρωτεΐνης για το αρνητικό μοντέλο
{
      if($_=~/^>/)
      {
            $id=$_; chomp $id;
            $wanted_id=substr($id,0,40);
            $seq=<FILE2>;      #print $id."\n";
       }
            @split_seq=split(//,$seq);

            for($i=0;$i<=$#split_seq;$i++)
            {
                  $amino=$split_seq[$i];
                  if (exists($hash{$amino}))
                        {
                              $prob=log($hash{$amino});

                              $total_prob=$total_prob+$prob;
                        }
            }
$log2{$wanted_id}=$total_prob;
$total_prob=0;
}

foreach  $x(@name)  #Υπολογίζει το σκορ της πρωτεΐνης αφαιρώντας από την πιθανότητα της πρωτεΐνης για το αρνητικό μοντέλο την πιθανότητα της πρωτεΐνης για το θετικό μοντέλο
{
      $logor=$log1{$x} - $log2{$x};

       $output="RESULTS/RES.$input";
       open FILEHANDLE,">>$output";
            print FILEHANDLE $x."\t".$logor."\n";
}
close FILE1;
close FILE2;
```

#Υπολογίζει το ποσοστό επιτυχίας, δηλαδή πόσες πρωτεΐνες ανιχνεύτηκαν σωστά ως μη διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα

```
$number1=`grep -c ">.*-" RESULTS/RES.$input`;
$number2=`wc -l RESULTS/RES.$input`;
$number3=$number2-$number1;

$cros=$number1/$number2;
$val=$cros*100;
$val1=sprintf ("\n\n CROSSVALIDATION: %.1f \n\n\n", $val);
open OUT, ">>RESULTS/RES.$input";
```

```
print OUT $val1;
close OUT;
```

#Τυπώνει στο αρχείο αποτελεσμάτων τις παραμέτρους

```
open NEW, ">>RESULTS/RES.$input";

print NEW "q_min=".$q_min."\t"."alpha=".$alpha."\n";
print NEW "gama=".$gama."\n"."p_ratio=".$p_ratio."\n";
print NEW "order=".$order."\n\n\n";
```

- ***train_null-pst.pl***, το οποίο δημιουργεί το θετικό μοντέλο είτε με τον αλγόριθμο PST είτε με το αλγόριθμο SPST και χρησιμοποιείται για την εκπαίδευση του συνόλου των πρωτεϊνών που παρουσιάζουν τη διαμόρφωση που μας ενδιαφέρει (διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα). Το πρόγραμμα αυτό παίρνει σαν είσοδο το σύνολο των πρωτεϊνών με τα διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα και τις παραμέτρους του μοντέλου.

```
$input1="$ARGV[0]";        # Σύνολο των πρωτεϊνών που παρουσιάζουν τη
                           τη διαμόρφωση που μας ενδιαφέρει (διαμεμβρανικά β-
                           βαρέλια ή σπειροειδή σπειράματα) (σε one-line format)
$q_min ="$ARGV[1]"; # Παράμετροι του θετικού μοντέλου
$alpha="$ARGV[2]";
$gama="$ARGV[3]";
$p_ratio="$ARGV[4]";
$order="$ARGV[5]";

print `mkdir TRAIN_NEG` ;

if ($input1=~/.*\/(.*)/)
{
      $file1=$1;
}
else
{
      $file1=$input1;
}
```

#Καλεί τον PST αλγόριθμο

```
print `../train ab.PROTEINS_BZX $input1 @ @ TRAIN_NEG/$file1.pst
      STATISTICS:100 @ 1 $q_min $alpha $gama $p_ratio $order`;
```

- ***ttest_null-pst.pl,*** το οποίο καλείται από το *crossval_null.pl* και είναι σχεδόν ίδιο με το *test_null-pst.pl* με ελάχιστες διαφορές στον κώδικα αλλά όχι σε αυτά που κάνει. Συγκεκριμένα η βασική διαφορά του από το *test_null-pst.pl* είναι ότι υπολογίζει την

πιθανότητα για διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα ενώ το *test_null-pst.pl* υπολόγιζε την πιθανότητα για πρωτεΐνες που δεν παρουσίαζαν αυτές τις διαμορφώσεις. Το πρόγραμμα αυτό δεν δέχεται τίποτα από το χρήστη σαν είσοδο καθώς καλείται όπως προαναφέραμε από το *crossval_null.pl* από το οποίο παίρνει σαν είσοδο το σύνολο των πρωτεϊνών που παρουσιάζουν τη διαμόρφωση που μας ενδιαφέρει και το θετικό μοντέλο που έχουμε δημιουργήσει είτε με το PST είτε με το SPST αλγόριθμο.

```
$input="$ARGV[0]";          # Σύνολο των πρωτεϊνών που παρουσιάζουν τη
                            διαμόρφωση που μας (σε one-line format)
$file1="$ARGV[1]";          #Θετικό μοντέλο

#Παρακάτω δημιουργεί το αρνητικό μοντέλο με βάση τις τυχαίες πιθανότητες
εμφάνισης των αμινοξέων όπως αυτές υπολογίστηκαν από τη βάση δεδομένων Swiss-
Prot

%hash = (
'A' => '0.077',
'C' => '0.018',
'D' => '0.058',
'E' => '0.066',
'F' => '0.040',
'G' => '0.072',
'H' => '0.024',
'I' => '0.056',
'K' => '0.063',
'L' => '0.086',
'M' => '0.022',
'N' => '0.046',
'P' => '0.046',
'Q' => '0.040',
'R' => '0.049',
'S' => '0.062',
'T' => '0.057',
'V' => '0.068',
'W' => '0.015',
'Y' => '0.035'
);

#Υπολογίζει την πιθανότητα της πρωτεΐνης για το θετικό μοντέλο
print `../predict TRAIN/$file1 $input @ TRAIN/$input.pos @ @`;

open FILE1 , "TRAIN/$input.pos" ;
open FILE2, "$input";


while(<FILE1>) #Μας δίνει την πιθανότητα της πρωτεΐνης για το θετικό μοντέλο
{
```

```
        if ($_=~/.*(>.*)\t(.*)/)
        {
                $log1=$2;
                $name=$1;
                push @name, $name;
                $log1{$name}=$log1;
        }
}

while(<FILE2>) #Υπολογίζει την πιθανότητα της πρωτεΐνης για το αρνητικό
μοντέλο
{
        if($_=~/^>/)
        {
                $id=$_; chomp $id;
                $wanted_id=substr($id,0,40);
                $seq=<FILE2>;      #print $id."\n";
         }
                @split_seq=split(//,$seq);

                for($i=0;$i<=$#split_seq;$i++)
                {
                        $amino=$split_seq[$i];
                        if (exists($hash{$amino}))
                                {
                                        $prob=log($hash{$amino});

                                        $total_prob=$total_prob+$prob;
                                }
                }
$log2{$wanted_id}=$total_prob;
$total_prob=0;
}

foreach $x(@name)    #Υπολογίζει το σκορ της πρωτεΐνης αφαιρώντας από την
                     πιθανότητα της πρωτεΐνης για το αρνητικό μοντέλο την
                     πιθανότητα της πρωτεΐνης για το θετικό μοντέλο
{
        $logor=$log1{$x} - $log2{$x};
        $output="RESULTS/RES";
         open FILEHANDLE,">>$output";
           print FILEHANDLE $x."\t".$logor."\n";
}


close FILE1;
close FILE2;
```

- ***ttrain_null-pst.pl,*** το οποίο είναι ίδιο με το *train_null-pst.pl*, καλείται από το *crossval_null.pl* και δημιουργεί για λογαριασμό του το θετικό μοντέλο είτε με τον αλγόριθμο PST είτε με το αλγόριθμο SPST. Για την εκπαίδευση του θετικού

μοντέλου χρησιμοποιείται το σύνολο των πρωτεϊνών που παρουσιάζουν τη διαμόρφωση που μας ενδιαφέρει (διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα). Το πρόγραμμα αυτό δεν δέχεται τίποτα από το χρήστη σαν είσοδο καθώς καλείται όπως προαναφέραμε από το *crossval_null.pl*, από το οποίο παίρνει σαν είσοδο το σύνολο των πρωτεϊνών με τα διαμεμβρανικά β-βαρέλια ή σπειροειδή σπειράματα και τις παραμέτρους του μοντέλου.

```
$input1="$ARGV[0]";        #Σύνολο των πρωτεϊνών που παρουσιάζουν τη
                           τη διαμόρφωση που μας ενδιαφέρει (διαμεμβρανικά β-
                           βαρέλια ή σπειροειδή σπειράματα) (σε one-line format)
$q_min ="$ARGV[1]"; #Παράμετροι του θετικού μοντέλου
$alpha="$ARGV[2]";
$gama="$ARGV[3]";
$p_ratio= "$ARGV[4]";
$order="$ARGV[5]";

if ($input1=~/.*\/(.*)/)
{
      $file1=$1;
}
else
{
      $file1=$input1;
}

# Καλεί PST αλγόριθμο
print   `../train   ab.PROTEINS_BZX   $input1   @   @   TRAIN/$file1.pst
         STATISTICS:100 @ 1 $q_min $alpha $gama $p_ratio $order`;
```

Για την 3η παραλλαγή τα προγράμματα που δημιουργήθηκαν ήταν ίδια με εκείνα της 2ης παραλλαγής με τη διαφορά ότι για αυτή την παραλλαγή δημιουργήσαμε ένα επιπλέον πρόγραμμα που μετατρέπει τις πρωτεΐνες από ακολουθίες αμινοξέων, δηλ. Ακολουθίες 20 συμβόλων σε ακολουθίες 5 συμβόλων.

- ***abrakatabra.pl***, το οποίο αντικαθιστά τα αμινοξικά σύμβολα που συναντά κατά μήκος μιας πρωτεϊνικής ακολουθίας με το αντίστοιχο σύμβολο της ομάδας που ανήκει.

```
while (<>)
{
      if($_=~/^>/)
      {
            $id=$_;
            $seq=<>;
```