Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Χημικών Μηχανικών

Διπλωματική Εργασία

Μελέτη και σχεδιασμός μεθόδων Εξόρυξης Δεδομένων και εφαρμογές σε προβλήματα Μεταβολομικής

Γεράσιμος Α. Χουρδάκης

Επιβλέπων :
Αν. Καθηγητής ΕΜΠ Σαρίμβεης Χαράλαμπος

Φεβρουάριος 2014

(η σελίδα αυτή αφέθηκε σκόπιμα λευκή)

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Χημικών Μηχανικών

Διπλωματική Εργασία

Μελέτη και σχεδιασμός μεθόδων Εξόρυξης Δεδομένων και εφαρμογές σε προβλήματα Μεταβολομικής

Γεράσιμος Α. Χουρδάκης

Επιβλέπων :
Αν. Καθηγητής ΕΜΠ Σαρίμβεης Χαράλαμπος

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 19η Φεβρουαρίου 2014

Αν. Καθηγητής ΕΜΠ	Καθηγητής ΕΜΠ	Καθηγητής ΕΚΠΑ
Σαρίμβεης Χαράλαμπος	Κυρανούδης Χρήστος	Μικρός Εμμανουήλ
.....	.....	.....

Γεράσιμος Α. Χουρδάκης
Διπλωματούχος Χημικός Μηχανικός Ε.Μ.Π.

.....

Χουρδάκης, Γεράσιμος Α.

Μελέτη και σχεδιασμός μεθόδων Εξόρυξης Δεδομένων και εφαρμογές σε προβλήματα Μεταβολομικής

Διπλωματική εργασία, Σχολή Χημικών Μηχανικών ΕΜΠ

Τομέας ΙΙ: Ανάλυσης, Σχεδιασμού και Ανάπτυξης Διεργασιών και Συστημάτων

Επιβλέπων: Αν. Καθηγητής ΕΜΠ Σαρίμβεης Χαράλαμπος

DDC 006.312: Data Mining

σελ. xvi + 117, 21×30 cm

Επικοινωνία: `makishourdakis@gmail.com`

Περίληψη

Η διπλωματική αυτή εργασία εστίασε στην επιστημονική περιοχή της Εξόρυξης Δεδομένων, με ενδελεχή μελέτη των διαθέσιμων αλγορίθμων, ανάπτυξη αλγόριθμου επιλογής μεταβλητών και εφαρμογές σε προβλήματα Μεταβολομικής. Συγκεκριμένα, μια σειρά μεθόδων μηχανικής μάθησης εφαρμόστηκαν σε δύο αρκετά διαφορετικά σύνολα δεδομένων, με στόχο την ταξινόμηση αγνώστων δειγμάτων σε προκαθορισμένες κλάσεις. Το πρώτο προέρχεται από δημοσιευμένη εργασία σχετικά με την πρόβλεψη της μετεγχειρητικής οξείας νεφρικής βλάβης (AKI). Περιέχει 106 φάσματα NMR από ανθρώπινα ούρα, με 701 χαρακτηριστικά και 2 κλάσεις. Εξετάστηκαν αρχικά αλγόριθμοι ταξινόμησης στο λογισμικό εξόρυξης δεδομένων WEKA και έπειτα εξήχθησαν συναινετικά μοντέλα με βάση τα αποτελέσματα από τα προηγούμενα μοντέλα, δημιουργώντας κατάλληλο λογιστικό φύλλο. Το δεύτερο σύνολο δεδομένων παραχωρήθηκε από το Τμήμα Φαρμακευτικής του Εθνικού Καποδιστριακού Πανεπιστημίου Αθηνών, και αφορά την επίδραση της ολευρωπαΐνης στην χρόνια καρδιακή ανεπάρκεια που προκαλεί η χορήγηση αδριαμυκίνης. Περιέχει 40 φάσματα NMR από εκχυλίσματα ιστών επιμύων, με 38 χαρακτηριστικά και 6 κλάσεις. Αρχικά χρησιμοποιήθηκε το εξειδικευμένο λογισμικό μεταβολομικής ανάλυσης MetaboAnalyst για τη διερεύνηση της ικανότητας διαχωρισμού των δεδομένων με συμβατικές μεθόδους. Στη συνέχεια, εξετάστηκαν αλγόριθμοι ταξινόμησης στη WEKA καθώς και συναινετικά μοντέλα που προέκυψαν χρησιμοποιώντας τα διαθέσιμα σε αυτήν εργαλεία. Τέλος, αναπτύχθηκε ένας αλγόριθμος επιλογής μεταβλητών με γενετική έρευνα και εκπαίδευση μοντέλων με μια υλοποίηση μηχανών διανυσμάτων υποστήριξης (Support Vector Machines, SVM). Τα αποτελέσματα της διπλωματικής εργασίας, έδειξαν ότι οι μέθοδοι μηχανικής μάθησης μπορούν να δώσουν λύσεις σε προβλήματα ανάλυσης δεδομένων Μεταβολομικής, με την ανάπτυξη μοντέλων μεγαλύτερης ακρίβειας σε σχέση με αυτά που παράγονται από συμβατικές στατιστικές μεθόδους.

Λέξεις-κλειδιά

Εξόρυξη Δεδομένων, Μεταβολομική, Μεταβονομική, Μηχανική Μάθηση, Support Vector Machines, SVM, Επιλογή μεταβλητών, Γενετική Έρευνα, WEKA, MetaboAnalyst

Abstract

This diploma thesis focused on the scientific area of Data Mining, with in-depth study of the available algorithms, on the development of a variable selection algorithm and on applications to Metabolomics. Namely, a series of machine learning methods was applied to two very different datasets, in order to classify unknown samples to pre-set classes. The first one comes from a published work about predicting Acute Kidney Injury (AKI). It contains 106 human urine NMR spectra, with 701 attributes and 2 classes. At first, classification algorithms of the data mining software WEKA were used. Then, consensus models were built using the results of the previous models, creating a suitable spreadsheet. The second dataset was given from the faculty of Pharmacy of the University of Athens and concerns the effect of the Oleuropein to chronic doxorubicin-induced cardiomyopathy. It contains 40 NMR spectra of rat tissue extracts, with 38 attributes and 6 classes. At first, the Metabolomics-specific software MetaboAnalyst was used to investigate the ability to separate the data with conventional methods. Then, WEKA classification algorithms were examined, as well as consensus modelling using its tools. Finally, a variable selection algorithm was developed using genetic search and a support vector machines (SVM) implementation. The results of this diploma thesis showed that the machine learning methods can provide solutions to Metabolomics data analysis problems, by building models of higher accuracy than those built from conventional statistical methods.

Keywords

Data Mining, Metabolomics, Metabonomics, Machine Learning, Support Vector Machines, SVM, Variable Selection, Genetic Search, WEKA, MetaboAnalyst

Περιεχόμενα

Κατάλογος σχημάτων

Κατάλογος πινάκων

Πρόλογος και ευχαριστίες

Η διπλωματική εργασία που κρατάτε στα χέρια σας είναι η πρώτη προσπάθεια «παντρέμματος» της εξόρυξης δεδομένων και της μηχανικής μάθησης με την μεταβολομική για την εργαστηριακή μονάδα Αυτόματης Ρύθμισης και Πληροφορικής της Σχολής Χημικών Μηχανικών ΕΜΠ. Ως τέτοια, δεν μπορεί παρά να είναι «αναγνωριστική», εξετάζοντας ένα εύρος δυνατοτήτων παρά εστιάζοντας και εμβαθύνοντας σε πολύ συγκεκριμένα σημεία. Ελπίζω, ωστόσο, ότι καταφέραμε να εντοπίσουμε και να σας παρουσιάσουμε τις βασικότερες δυνατότητες που προσφέρει ο σχετικά νέος αυτός επιστημονικός χώρος. Καθώς η διεθνής κοινότητα στρέφεται ολοένα περισσότερο προς τη μεταβολομική, αντιλαμβάνεται το άνοιγμα συνόρων που μπορούν να της προσφέρουν οι τελευταίες εξελίξεις στη μηχανική μάθηση. Το εργαστήριό μας, έχοντας από καιρό εφαρμόσει ή δημιουργήσει αλγορίθμους μηχανικής μάθησης σε άλλους τομείς και, έχοντας πρόσβαση σε φοιτητές με γενικότερες γνώσεις χημικής μηχανικής, πιστεύω πως μπορεί να ακολουθήσει μια ιδιαίτερα αξιόλογη πορεία σε αυτόν τον τομέα. Ελπίζω η εργασία αυτή να καταφέρει να καθοδηγήσει τους συμφοιτητές μου που ίσως διαλέξουν να ασχοληθούν με αντίστοιχα θέματα σε δικές τους εργασίες.

Θα ήθελα να ευχαριστήσω βαθύτατα τον επιβλέποντά μου, αναπληρωτή καθηγητή Χ. Σαρίμβεη για την αστείρευτη υπομονή και την άψογη συνεργασία μας από την πλευρά του. Μαζί με αυτόν ευχαριστώ και τους διδάκτορες Φ. Δογάνη και Π. Σωπασάκη για την καθοδήγησή τους στα κρίσιμα πρώτα βήματα. Για την ευγενική προσφορά των δεδομένων, τον χρόνο και την όρεξη που διέθεσε σε σχετικές συζητήσεις ευχαριστώ τον καθηγητή του τμήματος Φαρμακευτικής ΕΚΠΑ, Εμμ. Μικρό, με τον οποίο εύχομαι να συνεχιστεί η συνεργασία.

Κλείνοντας αυτόν τον 5ετή κύκλο των προπτυχιακών μου σπουδών, θα ήθελα να θυμηθώ μερικούς ανθρώπους που με διαμόρφωσαν. Καταρχάς τους συνταξιούχους πλέον καθηγητές Ι. Παλυβό και Κ. Σπυρόπουλο που μου έδωσαν μια καταπληκτική ευκαιρία να ασχοληθώ βαθύτερα με τον προγραμματισμό, καθώς και τον καθηγητή Α. Μπουντουβή και όλο το προσωπικό του Υπολογιστικού Κέντρου για την καρποφόρα συνεργασία μας όλα αυτά τα χρόνια. Πολύ σημαντική ήταν και η επιρροή του λέκτορα Α. Καραντώνη ο οποίος γνωρίζει ότι τον ευχαριστώ βαθύτατα. Κινδυνεύοντας να αδικήσω πολλούς αξιολογότατους ανθρώπους με τη μη αναφορά τους, το λιγότερο που μπορώ να κάνω είναι να ευχαριστήσω τους φίλους, τους συμφοιτητές και τους, με ή χωρίς τον τυπικό τίτλο, δασκάλους μου.

Ελπίζω οι ελπίδες της οικογένειας και των στενών μου φίλων να μην πήγαν χαμένες. Τους ευχαριστώ πολύ για την υπομονή, τη στήριξη και την όρεξη που μου έδιναν κάθε στιγμή αυτά τα πέντε χρόνια.

Στον αδερφό μου, Μανώλη,
με ευχές για μια λαμπρή πορεία.

Κεφάλαιο 1

Εισαγωγή

Για να κατανοήσουμε τον μηχανισμό πίσω από κάποιο φαινόμενο που παρατηρούμε, συνήθως διατυπώνουμε κάποιες υποθέσεις και διεξάγουμε τα κατάλληλα πειράματα ώστε να επιβεβαιώσουμε ή να απορρίψουμε την ευστάθειά τους. Η διατύπωση όμως καλών υποθέσεων δεν είναι πάντα εύκολη, ενώ πολλές φορές τα πειράματα είναι ιδιαίτερα δύσκολα, με πολύ υψηλό κόστος. Ειδικότερα όταν αντικείμενο πειραμάτων είναι ο οργανισμός ανθρώπων ή πειραματοζώων, οι υποθέσεις που δοκιμάζονται επιβάλλεται να έχουν κάποια σημαντική πιθανότητα ευστάθειας. Σε αυτήν την περίπτωση, τα συστήματα που ζητούμε να κατανοήσουμε είναι από τη φύση τους περίπλοκα, περιέχουν όμως πάρα πολλές «κρυμμένες» πληροφορίες. Σκοπός του ερευνητή είναι να καταφέρει να συνδυάσει τα διαθέσιμα δεδομένα με τον κατάλληλο τρόπο έτσι ώστε να προκύψει ένα αντιπροσωπευτικό μοντέλο που να εξηγεί με ικανοποιητική ακρίβεια τα φαινόμενα που παρατηρεί. Ο ανθρώπινος εγκέφαλος μπορεί και αναγνωρίζει πρότυπα που υπάρχουν σε σχετικά μικρά σύνολα δεδομένων, ωστόσο πολλά σύγχρονα προβλήματα παρέχουν τεράστιο όγκο δεδομένων, τα οποία δεν μπορεί να τα διαχειριστεί με προφανή τρόπο για την εξόρυξη της επιθυμητής πληροφορίας. Η υπολογιστική διαδικασία της αναγνώρισης προτύπων σε (μεγάλα) σύνολα δεδομένων ονομάζεται «εξόρυξη δεδομένων» (data mining) και είναι ένας κόμβος όπου συναντιούνται πολλοί διαφορετικοί κλάδοι επιστημών, με βασικότερους την μηχανική μάθηση (machine learning) και την στατιστική. Η μηχανική μάθηση εστιάζει στους τρόπους «μάθησης». Είναι ένα δυνατό θεωρητικό εργαλείο. Η εξόρυξη δεδομένων ωστόσο δεν ενδιαφέρεται τόσο πολύ για την θεωρητική περιγραφή της μάθησης, όσο για την πρακτική αναγνώριση δομικών προτύπων σε συγκεκριμένα δεδομένα και τη δυνατότητα εξαγωγής προβλέψεων από αυτά. [1]

Τεχνικές εξόρυξης δεδομένων έχουν υιοθετηθεί με επιτυχία για εξήγηση φαινομένων, ενδεικτικά, στους εξής τομείς [1]:

- στην συμπεριφορά ψηφοφόρων και τελικά στην πρόβλεψη εκλογικού αποτελέσματος, με βάση ιστορικά, γεωγραφικά, οικονομικά κ.α. στοιχεία.
- στην συμπεριφορά καταναλωτών σύμφωνα με το ιστορικό αγορών τους.
- στη λήψη αποφάσεων σχετικά με χορήγηση δανείων, προβλέποντας την πιθανότητα επιστροφής του δανείου από συγκεκριμένο πελάτη.

- σε μηχανισμούς ασθενειών και δράσης φαρμάκων, όπως και στη διάγνωση ασθενειών χρησιμοποιώντας μεταβολομικά δεδομένα.
- στην πρόρρηση τοξικότητας μορίων με βάση δομικά χαρακτηριστικά τους.
- στην αναγνώριση προτύπων σε εικόνες, όπως π.χ. ανθρώπινα πρόσωπα από κάμερες.
- στην πρόρρηση φορτίου σε ηλεκτρικά δίκτυα, σύμφωνα με ιστορικά δεδομένα.
- στον παγκόσμιο ιστό, για την κατάταξη ιστοσελίδων σύμφωνα με την δημοτικότητά τους και την καλύτερη αναζήτηση σε αυτές.

Στην εργασία αυτή θα εστιάσουμε στην εξόρυξη δεδομένων μεταβολομικής. Τα δεδομένα αυτά προέρχονται γενικώς από αναλύσεις προϊόντων μεταβολισμού (αίμα, ούρα, εκχυλίσματα ιστών κ.α.) με τεχνικές που αποδίδουν πλήρη φάσματα μεταβολιτών, όπως η φασματομετρία πυρηνικού μαγνητικού συντονισμού (NMR) ή η φασματομετρία μάζας (MS), συνήθως συνδυαζόμενη με υγρή χρωματογραφία (LC/MS). [2] Συνήθης εικόνα ενός συνόλου δεδομένων μεταβολομικής είναι ένας πίνακας δυο διαστάσεων, του οποίου κάθε γραμμή αναπαριστά ένα διαφορετικό άτομο, ενώ οι στήλες φιλοξενούν τιμές π.χ. συγκεντρώσεων για διαφορετικές ουσίες-μεταβολίτες ή εμβαδά διαφορετικών κορυφών ενός φάσματος. Υπάρχει επίσης μια επιπλέον στήλη, μέσω της οποίας αντιστοιχίζεται κάθε άτομο σε μια κλάση (π.χ. «υγιές» ή «ασθενές»). Χρησιμοποιώντας αυτά τα δεδομένα έχουμε πλήρη εικόνα της κατάστασης στην οποία βρίσκονται τα κύτταρα που μελετάμε. Αναλύοντας, αντιθέτως, το γονιδίωμα ενός ανθρώπου μπορούμε μόνο να προσδιορίσουμε τα χαρακτηριστικά που έχει αποκτήσει κατά τη γέννησή του. Δεν έχουμε όμως καμία πληροφορία για το τι τροφή έχει καταναλώσει ή από ποιες ασθένειες πάσχει. Ένα κύτταρο παράγει διαφορετικά προϊόντα μεταβολισμού σε διαφορετικές συνθήκες ζωής. Έτσι, γνωρίζοντας τις ουσίες που παράγει και συνδυάζοντας τα μεταβολικά προφίλ πολλών κυττάρων του ίδιου είδους-κλάσης και κυττάρων άλλων ειδών-κλάσεων, μπορούμε να συμπεράνουμε τις συνθήκες που οδήγησαν στην παραγωγή των συγκεκριμένων μεταβολιτών. Επεκτείνοντας, μπορούμε να εξάγουμε ένα γενικότερο μοντέλο συμπεριφοράς, το οποίο θα μπορεί να αποφασίζει για την τωρινή, μελλοντική ή παρελθούσα (ανάλογα με το πρόβλημα) κατάσταση του κυττάρου (ή ενός ολόκληρου οργανισμού) με σημαντική στατιστική πιθανότητα. Και όλα αυτά χωρίς να χρειάζεται η γνώμη κάποιου ειδικού γιατρού. Μάλιστα, οι αποφάσεις μπορεί να βασίζονται σε μικρές αλλαγές του μεταβολισμού, οι οποίες δεν θα έδιναν ευανάγνωστα συμπτώματα, καθώς και σε πολύ μικρά δείγματα π.χ. βιολογικών υγρών.

Βασικά εργαλεία της μεταβολομικής είναι (μεταξύ άλλων) οι μέθοδοι διεξαγωγής των σχετικών πειραμάτων, οι μέθοδοι χημικής ανάλυσης των δειγμάτων και οι μέθοδοι επεξεργασίας και ανάλυσης των δεδομένων με στόχο την εξόρυξη νέων πληροφοριών. Στην παρούσα εργασία το ενδιαφέρον επικεντρώθηκε αποκλειστικά στο τελευταίο κομμάτι. Εξετάστηκαν δεδομένα για δυο προβλήματα. Το πρώτο πρόβλημα αντλήθηκε από τη βιβλιογραφία [3,4] και αφορά ανθρώπους που εμφάνιζαν μετεγχειρητική οξεία βλάβη του ήπατος (Acute Kidney Injury - AKI). Το δεύτερο πρόβλημα αφορά σε επίμυες (ποντίκια) στα οποία χορηγούνταν το αντικαρκινικό φάρμακο αδριαμυκίνη (Doxorubicin - DXR), το οποίο όμως οδηγεί σταδιακά σε καρδιακή ανεπάρκεια. Εξετάζεται σε αυτήν την περίπτωση αν η χορήγηση ολευρωπαΐνης,

του βασικού συστατικού των φύλλων της ελιάς, συμπληρωματικά με την DXR, επαναφέρει τα καρδιακά κύτταρα σε φυσιολογικές συνθήκες. Τα δεδομένα αυτά παρασχέθηκαν ευγενικά από τον Τομέα Φαρμακευτικής Χημείας του Τμήματος Φαρμακευτικής του Εθνικού Καποδιστριακού Πανεπιστημίου Αθηνών (καθηγητής Εμμ. Μικρός). Στο πρόβλημα έχει ήδη γίνει σημαντική μελέτη [5, 6], ωστόσο παραμένουν ανοιχτά ερωτήματα, στα οποία ευελπιστούμε να συνεισφέρει η παρούσα εργασία.

Κύριος στόχος και στα δυο προβλήματα από την πλευρά μας ήταν η σύγκριση μεθόδων εξόρυξης δεδομένων και, κυρίως, διαφορετικών αλγορίθμων μηχανικής μάθησης. Κύριο ερώτημα ήταν το πόσο καλά αποτελέσματα προβλέψεων μπορούμε να παράγουμε με κάθε αλγόριθμο σε κάθε πρόβλημα. Όπως θα δούμε, τα δυο προβλήματα παρουσιάζουν σημαντικές δομικές διαφορές ως προς το πλήθος των δεδομένων, καθώς και ως προς το πλήθος και τη σχέση των κλάσεων. Οι διαφορές αυτές οδηγούν και σε σημαντική διαφορά στην γενικότερη επιτυχία προβλέψεων.

Η ανάλυση των δεδομένων βασίστηκε κυρίως σε έτοιμα πακέτα λογισμικού. Μάλιστα, όλα ήταν ελεύθερα προσβάσιμα και ανοιχτού κώδικα. Συνοπτικά, χρησιμοποιήσαμε:

- την online συλλογή εργαλείων MetaboAnalyst, η οποία είναι εξειδικευμένη για δεδομένα μεταβολομικής. [7, 8]
- τη σουίτα WEKA, η οποία περιέχει εργαλεία data mining γενικής χρήσης. [9]
- την υλοποίηση LibSVM για Support Vector Classification, τόσο αυτόνομη, όσο και ως πρόσθετη στη σουίτα WEKA. [10, 11]
- κώδικα σε Octave (συμβατός με MATLAB™) που αναπτύχθηκε στα πλαίσια της εργασίας για επιλογή μεταβλητών με βάση έναν γενετικό αλγόριθμο και τον οποίο μπορείτε να δείτε στο παράρτημα Β΄.

Λόγω του μικρού όγκου των δεδομένων δεν χρειάστηκε κάποιο ισχυρό υπολογιστικό σύστημα. Όλες οι αναλύσεις έγιναν σε σύγχρονο προσωπικό Η/Υ σε πολύ μικρό χρόνο. Χρησιμοποιήθηκε λειτουργικό σύστημα Linux 64bit, ωστόσο όλα τα προγράμματα που αναφέρθηκαν είναι διαθέσιμα για όλες τις βασικές πλατφόρμες. Αναλυτικότερες πληροφορίες δίνονται στο παράρτημα Α΄.

Επιλογή μας ήταν η εργασία αυτή να δομηθεί με τρόπο που να είναι κατανοητός σε επιστήμονες που δεν έχουν εμπειρία στη μεταβολομική και τη μηχανική μάθηση και ευχόμαστε να είναι ένας καλός οδηγός στα πρώτα βήματα όσων αποφασίσουν να ασχοληθούν. Θεωρούμε όμως ταυτόχρονα, ότι η εργασία αυτή παρουσιάζει εργαλεία και μεθόδους που μπορούν να προκαλέσουν το ενδιαφέρον και πιο έμπειρων ερευνητών στις περιοχές της μεταβολομικής και της μηχανικής μάθησης.

Κεφάλαιο 2

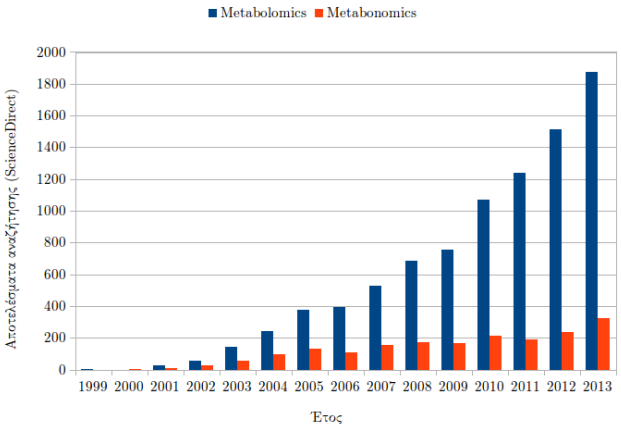
Μεταβολομική

2.1 Βασικές αρχές

Η επιστημονική κοινότητα ασχολείται πολύ έντονα τα τελευταία χρόνια με την «τριλογία των omics», όπως χαρακτηρίζει την τριάδα Genomics, Proteomics και Metabolomics το εισαγωγικό review του περιοδικού Nature Reviews: Molecular Cell Biology. [2] Ο όρος Genomics αναφέρεται σε ανάλυση δεδομένων που προέρχονται από DNA και mRNA. Ο όρος Proteomics αναφέρεται σε ανάλυση δεδομένων που προέρχονται από πρωτεΐνες. Ο όρος Metabolomics αναφέρεται σε ανάλυση δεδομένων που προέρχονται από προϊόντα μεταβολισμού κυττάρων (μεταβολίτες). Συχνά ο όρος συναντάται και ως «μεταβονομική». Ωστόσο δεν υπάρχει ιδιαίτερα σαφής διαχωρισμός μεταξύ των δυο όρων. Είναι κυρίως θέμα σύμβασης και συχνά χρησιμοποιούνται ως συνώνυμοι. Στο εξής θα χρησιμοποιούμε τον όρο «μεταβολομική» και για τις δυο περιπτώσεις.

Ο όρος «μεταβολομική» καθιερώθηκε πρωταρχικά από τον J. Nicholson ως «ο ποσοτικός προσδιορισμός της δυναμικής, πολυπαραμετρικής, μεταβολικής απόκρισης των ζωντανών συστημάτων σε παθοφυσιολογικά (θρεπτικά, ξενοβιοτικά, χειρουργικά ή τοξικά) ερεθίσματα ή γενετικές αλλοιώσεις» [12] Παρουσιάζοντας τον όρο λιγότερο αυστηρά, θα λέγαμε ότι αναφέρεται στην ανάλυση πολλών μεταβολιτών ταυτόχρονα με κατάλληλες τεχνικές και την στατιστική ανάλυση των παραγόμενων δεδομένων. [5] Στο σχήμα 2.1 φαίνεται η χρονική κατανομή στα αποτελέσματα αναζήτησης μέσω της μηχανής ScienceDirect. Οι δυο όροι συναντώνται για πρώτη φορά στην αρχή της προηγούμενης δεκαετίας και από τότε η εξέλιξη είναι ραγδαία.

Ένα γονίδιο ή μια πρωτεΐνη δεν μπορεί να καθορίσει την κατάσταση στην οποία βρίσκεται ένα κύτταρο και, κατ' επέκταση, ένας οργανισμός. Αφ' ενός μεν, υπόκεινται σε ανεξέλεγκτες τροποποιήσεις. Αφ' ετέρου, περιέχουν μόνο την πληροφορία που έχει αποκτήσει το κύτταρο από την προηγούμενη γενιά ή άμεσα παράγωγα αυτής. Οι διαταραχές στα βιοχημικά-μεταβολικά μονοπάτια, οι οποίες μπορεί να προκαλούνται π.χ. από μια ασθένεια, εμφανίζονται στα προϊόντα μεταβολισμού. Έτσι, αν κάποια ασθένεια επεμβαίνει σε κάποιο μεταβολικό μονοπάτι, μπορούμε να ανιχνεύσουμε τους μεταβολίτες που επηρεάζονται, να μάθουμε στη συνέχεια σε ποια μεταβολικά μονοπάτια συμμετέχουν και να σχεδιάσουμε ένα κατάλληλο φάρμακο. Αυτό δεν είναι τόσο απλό. Κάθε οργανισμός έχει το δικό του



Σχήμα 2.1: Αποτελέσματα αναζήτησης για τους όρους "Metabolomics" και "Metabonomics" (περιοδικά, βιβλία και έργα αναφοράς) στη μηχανή ScienceDirect. Τα πρώτα άρθρα εμφανίζονται το 1999. Για το 2013, τα αποτελέσματα για Metabolomics ανέρχονται σε 1877.

μεταβολικό προφίλ, το οποίο επηρεάζεται και από το γονιδίωμά του. Πρέπει λοιπόν να γίνει κατάλληλη πολυπαραμετρική ανάλυση ώστε να βρεθούν μόνο οι συγκεκριμένοι αυτοί μεταβολίτες οι οποίοι επηρεάζονται περισσότερο από τη διαταραχή που μελετάμε.

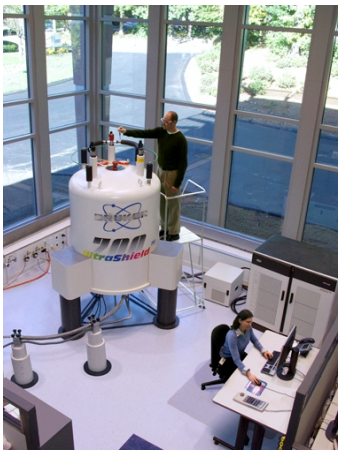
Τα μεταβολομικά πειράματα χαρακτηρίζονται συνήθως ως στοχευμένα (targeted) ή μη στοχευμένα (untargeted). [2] Στα στοχευμένα πειράματα, εστιάζουμε το ενδιαφέρον μας σε συγκεκριμένα μεταβολικά μονοπάτια και μεταβολίτες που έχουμε από πριν επιλέξει. Στα μη στοχευμένα πειράματα, αναλύουμε όλο το δυνατό φάσμα μεταβολιτών και παρατηρούμε, χωρίς προκατάληψη, τους μεταβολίτες που φαίνεται να σχετίζονται με το πείραμά μας. Με αυτή τη μέθοδο, πολλές φορές έρχονται στην επιφάνεια νέοι, άγνωστοι μέχρι πριν, μεταβολίτες. Ανάλογα με την αναλυτική μέθοδο που χρησιμοποιείται, τα δεδομένα από μη στοχευμένα πειράματα μπορούν να έχουν μέγεθος της τάξης gigabytes ανά δείγμα. Μη στοχευμένα πειράματα χρησιμοποιούνται κυρίως όταν θέλουμε να ελέγξουμε βιοχημικές υποθέσεις, όταν θέλουμε να κατανοήσουμε βαθύτερα ένα βιοχημικό μονοπάτι ή την εξάρτηση μιας ασθένειας από συγκεκριμένες ουσίες. Τα μη στοχευμένα πειράματα ωστόσο, μπορούν να παρέχουν πολύτιμες πληροφορίες σε συστήματα για τα οποία δεν έχουμε ιδιαίτερα σαφή γνώση. Δηλαδή τα στοχευμένα πειράματα εκπορεύονται από υποθέσεις, ενώ τα μη στοχευμένα πειράματα δημιουργούν υποθέσεις.

2.2 Αναλυτικές μέθοδοι

Στη μεταβολομική χρησιμοποιούνται κυρίως η φασματομετρία πυρηνικού μαγνητικού συντονισμού (Nuclear Magnetic Resonance - NMR, σχήμα 2.2) και η φασματομετρία μάζας (Mass Spectrometry - MS), συνδυαζόμενη με την υγρή χρωματογραφία (Liquid Chromatography - LC). Η συνδυασμένη τεχνική (LC/MS) μπορεί να παράγει δεδομένα εξαιρε-

τικά υψηλής ανάλυσης, και μπορεί να εντοπίσει τους περισσότερους μεταβολίτες. Και οι δύο μέθοδοι προτιμούνται για την ακρίβεια, την επαναληψιμότητα και την αναπαραγωγισιμότητα που προσφέρουν σε ποσοτικούς προσδιορισμούς, καθώς και για το μεγάλο εύρος ουσιών που μπορούν να ανιχνεύσουν ταυτόχρονα, σε ελάχιστο χρόνο και με πολύ μικρή ποσότητα δείγματος. Σαφέστατα, μπορούν να χρησιμοποιηθούν και άλλες κλασσικές τεχνικές, όπως η φασματομετρία υπεριώδους-ορατού (UV-Vis) ή η φασματομετρία ιονισμού φλόγας για τη μέτρηση μεταβολιτών. [2]

Δείγματα που αναλύονται συνήθως είναι βιολογικά υγρά (πλάσμα αίματος, ούρα) ή εκχυλίσματα ιστών. Είναι πολύ βασικό το κάθε δείγμα να μπορεί να συγκριθεί ως προς τη σύστασή του με τα υπόλοιπα με ακρίβεια. Γι' αυτό το λόγο, εκτός από την υψηλή ακρίβεια που πρέπει να παρέχει η χρησιμοποιούμενη αναλυτική τεχνική, πρέπει να δοθεί πολύ μεγάλη προσοχή στην προετοιμασία των δειγμάτων. Όλα πρέπει να έχουν υποστεί την κατάλληλη αραίωση ώστε να είναι συγκρίσιμα. Για παράδειγμα, δείγματα ούρων μπορεί να παρουσιάζουν μεγάλες διακυμάνσεις στις μετρούμενες συγκεντρώσεις λόγω διαφορετικής πρόσληψης υγρών πριν τη δειγματοληψία. Σε αυτήν την περίπτωση χρησιμοποιείται η συγκέντρωση κάποιου συστατικού που θεωρείται ότι είναι κοινή και σταθερή για τη διόρθωση (π.χ. κρεατινίνη για δείγματα ούρων). [13] Επίσης πρέπει να ελέγχεται αν άλλες συνθήκες, όπως π.χ. το pH επηρεάζουν τη μέθοδο. Σε αυτήν την εργασία, θα ασχοληθούμε αποκλειστικά με την φασματομετρία Πυρηνικού Μαγνητικού Συντονισμού (NMR) ως αναλυτική τεχνική για πειράματα μεταβολομικής.



Σχήμα 2.2: Διάταξη NMR. *(Πηγή: public domain)*

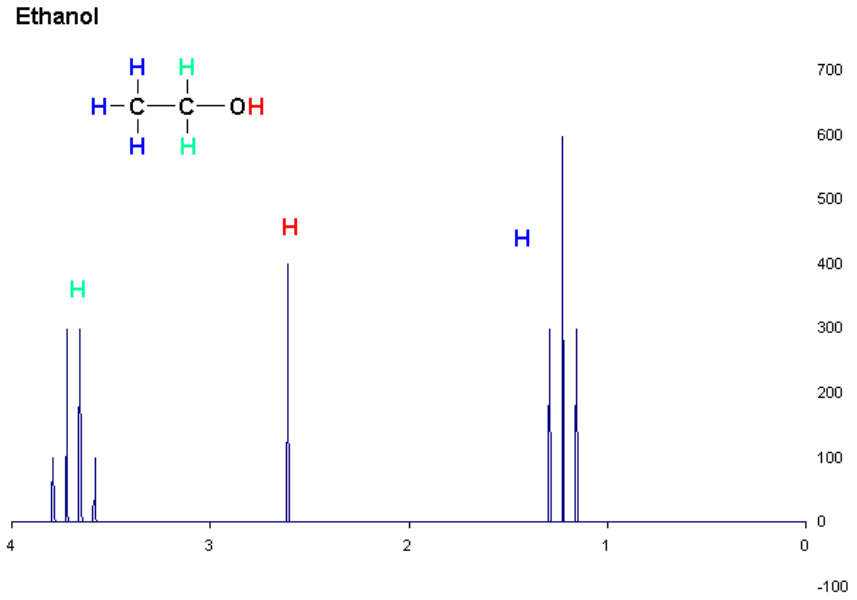
2.2.1 Φασματομετρία Πυρηνικού Μαγνητικού Συντονισμού

Για να κατανοήσουμε ένα διάγραμμα NMR, πρέπει πρώτα να κατανοήσουμε την αρχή λειτουργίας της NMR. Οι πυρήνες ατόμων με περιττό μαζικό αριθμό (π.χ. το υδρογόνο ^{1}H ή το ισότοπο του άνθρακα ^{13}C), ως φορτισμένα και κινούμενα σωματίδια, έχουν την ικανότητα να συντονίζονται από ένα μαγνητικό πεδίο και οι άξονές τους να στρέφονται παράλληλα ή αντιπαράλληλα ως προς αυτό. [14, 15] Ο παράλληλος και ο αντιπαράλληλος προσανατολισμός αντιστοιχούν σε διαφορετικές ενεργειακές στάθμες, με τον παράλληλο προσανατολισμό να ευνοείται. Οι στάθμες αυτές απέχουν τόσο περισσότερο, όσο ισχυρότερο είναι το μαγνητικό πεδίο που επιβάλλεται. Εάν, ταυτόχρονα με το μαγνητικό πεδίο, εφαρμόσουμε κατάλληλο πεδίο ραδιοσυχνοτήτων, μπορούμε να ωθήσουμε τους πυρήνες να μεταβούν στην υψηλότερη ενεργειακή στάθμη (και, στη συνέχεια, να επανεκπέμψουν την αντίστοιχη ακτινοβολία).

Δεν απαιτούν όλοι οι πυρήνες, ωστόσο, την ίδια ενέργεια για αυτή τη μετάβαση. Κατά την εφαρμογή του μαγνητικού πεδίου, τα ηλεκτρόνια που υπάρχουν στην ευρύτερη περιοχή του ατόμου, τόσο τα δικά του όσο και γειτονικών ατόμων, δημιουργούν ένα αντίθετης φοράς μαγνητικό πεδίο. Το αποτέλεσμα είναι να «εξουδετερώνεται» μέρος του μαγνητικού πεδίου που επιβάλλεται και άρα ο πυρήνας να βρίσκεται σε ένα τοπικό, χαμηλότερης ισχύος μαγνη-

τικό πεδίο, και άρα να απαιτεί μικρότερη ενέργεια. Το φαινόμενο αυτό καλείται «προστασία» του πυρήνα.

Στο σχήμα 2.3 φαίνεται ένα απλό διάγραμμα NMR πρωτονίου (υδρογόνου), το οποίο αντιστοιχεί στην αιθανόλη. Ο κατακόρυφος άξονας αντιστοιχεί στην ένταση της ακτινοβολίας που απορροφάται (δηλαδή στην ενεργειακή διαφορά των δυο καταστάσεων). Αν δεν υπήρχε το φαινόμενο της προστασίας, θα είχαμε ένα σημείο έντασης στον άξονα, αφού απλώς θα αθροίζονταν οι απαιτούμενες ενέργειες για όλους τους πυρήνες. Λόγω όμως της προστασίας των πυρήνων, κάθε πυρήνας συντονίζεται στην (σταθερή) επιβαλλόμενη συχνότητα για διαφορετική ισχύ του επιβαλλόμενου μαγνητικού πεδίου. Ο οριζόντιος άξονας λοιπόν αναπαριστά την ισχύ, ή ακριβέστερα, τη συχνότητα του επιβαλλόμενου μαγνητικού πεδίου. Στο μηδέν της κλίμακας βρίσκεται η κορυφή ενός προτύπου (δεν απεικονίζεται εδώ), συνήθως του τετραμέθυλοσιλανίου (TMS). Στα αριστερά του άξονα, η συχνότητα του μαγνητικού πεδίου μειώνεται κατά το απεικονιζόμενο ποσοστό (1 ppm = 1 εκατομμυριοστό). Τα διαγράμματα αυτά δεν εξαρτώνται από τη μέγιστη συχνότητα της διάταξης. Έτσι, μια κορυφή που βρίσκεται στη χημική μετατόπιση π.χ. δ=1 ppm σε μια διάταξη 300 MHz, θα βρίσκεται στην ίδια θέση και σε μια διάταξη 60 MHz. Αυτό είναι σημαντικό για την αναπαραγωγισιμότητα των μετρήσεων. Συνήθως δεν απαιτείται κάποια ιδιαίτερη προετοιμασία του δείγματος για ανάλυση. Αν χρειαστεί διαλύτης, τότε χρησιμοποιείται κατά προτίμηση κάποιος που να είναι «αόρατος» στο NMR, όπως το δευτεριωμένο χλωροφόρμιο.



Σχήμα 2.3: Φάσμα ^{1}H NMR της αιθανόλης. Διακρίνονται οι κορυφές που αντιστοιχούν στα διαφορετικά υδρογόνα του μορίου. Ο κατακόρυφος άξονας αντιστοιχεί στην ένταση της ακτινοβολίας που απορροφάται και ο οριζόντιος άξονας στην χημική μετατόπιση. *(Πηγή: T.vanschaik, wikimedia commons - άδεια CC BY-SA)*

Κάθε κορυφή αντιστοιχεί σε μια ομάδα «ισοδύναμων» πυρήνων. Π.χ. η ομάδα $-CH_3$ της αιθανόλης θα δώσει μια κοινή κορυφή. Η κορυφή αυτή μπορεί να είναι πολλαπλή, κάτι που εξηγείται από το φαινόμενο «σχάση spin-spin» ή «λεπτή υφή». [14, 15] Ένα μόριο τελικά, αναπαρίσταται από πολλές κορυφές. Το διάγραμμα γίνεται συνθετότερο όταν αναλύουμε ταυτόχρονα πολλές ουσίες. Δεν βλέπουμε άμεσα συγκεκριμένα μόρια, αλλά χαρακτηριστικές ομάδες μορίων. Μπορούμε ωστόσο να εστιάσουμε το ενδιαφέρον μας σε κορυφές που να χαρακτηρίζουν ξεκάθαρα ένα και μόνο μόριο. Βαθμονομώντας διαφορετικά διαγράμματα με ένα πρότυπο (π.χ. TMS), μπορούμε πλέον να έχουμε συγκρίσιμες εικόνες τόσο της ποιοτικής, όσο και της ποσοτικής σύστασης διαφορετικών δειγμάτων.

2.3 Προεπεξεργασία δεδομένων

Τα φάσματα NMR που προκύπτουν έχουν τεράστιο όγκο δεδομένων, λόγω της υψηλής τους ανάλυσης. Για τη διαχείρισή τους συνήθως μειώνουμε την ανάλυση, έτσι ώστε να έχουμε διακριτές περιοχές εύρους π.χ. 0.02 ppm. Μπορούμε από αυτές τις περιοχές να εξαιρέσουμε τμήματα του διαγράμματος που γνωρίζουμε ότι δεν παρέχουν σημαντική πληροφορία ή που γνωρίζουμε ότι θα δημιουργήσουν πρόβλημα στην υπόλοιπη ανάλυση. Παραδείγματος χάριν, αν χρησιμοποιήσουμε έναν διαλύτη ορατό στο NMR, τότε αυτός θα απεικονίζεται ως μια πολύ υψηλή κορυφή στο διάγραμμα. Αυτή η κορυφή θα προκαλεί «συμπίεση» των υπολοίπων κατά την οπτική παρατήρηση και αριθμητικά προβλήματα κατά την επεξεργασία με H/Υ.

Σε μη στοχευμένα πειράματα χρησιμοποιούμε όσο το δυνατόν υψηλότερη ανάλυση και μεγαλύτερο εύρος του φάσματος. Ενδιαφέρον θα είχε μάλιστα η έρευνα για πληροφορία που χάνεται μέσα στο θόρυβο. Ορισμένες φορές, μπορεί ο κρίσιμος μεταβολίτης που αναζητούμε, να εμφανίζει μια κορυφή τόσο μικρή που δύσκολα να ξεχωρίζει οπτικά από τον θόρυβο. [16] Όταν μια κορυφή είναι πολύ κοντά στο όριο ανίχνευσης, τότε πρέπει να αντιμετωπίζεται σαν το εμβαδό της να είναι άγνωστο (missing value). Οι άγνωστες τιμές είναι κάτι ιδιαίτερα ανεπιθύμητο, αν και υπάρχουν τεχνικές που μπορούν να αντιμετωπίσουν τέτοιες περιπτώσεις, π.χ. αντικαθιστώντας με την μέση τιμή για αυτήν την κορυφή-μεταβλητή. [13, 16]

Σε στοχευμένα πειράματα απομονώνουμε συγκεκριμένες κορυφές ενδιαφέροντος. Οι κορυφές αυτές πρέπει να γνωρίζουμε ότι είναι χαρακτηριστικές μονάχα για τις ουσίες που θέλουμε να παρακολουθήσουμε. Στην περίπτωση που εξετάζουμε ολόκληρο το φάσμα, οι παρατηρούμενες μεταβλητές ή χαρακτηριστικά (attributes) που εξετάζουμε μπορεί να είναι εκατοντάδες έως χιλιάδες. Εξετάζοντας μόνο συγκεκριμένους μεταβολίτες, οι μεταβλητές περιορίζονται συνήθως σε μερικές δεκάδες. Στην ενότητα 3.7 θα δούμε πώς οι μεταβλητές αυτές μπορούν να μειωθούν ακόμα περισσότερο.

Πριν τα δεδομένα αναλυθούν περαιτέρω, είναι καλό να μετασχηματιστούν, έτσι ώστε να είναι εύκολες οι συγκρίσεις μεταξύ τους. Μάλιστα, οι στατιστικές τεχνικές προϋποθέτουν ότι τα δεδομένα υπακούν σε κάποια κατανομή. Σε αυτήν την περίπτωση, πρέπει να θυμόμαστε ότι τυχόν νέα δεδομένα δεν είναι πλέον άμεσα συγκρίσιμα. Εκτός από την προεπεξεργασία ως προς τα διαφορετικά δείγματα, θα πρέπει να γίνει και ένας μετασχηματισμός στους μεταβολίτες, έτσι ώστε ουσίες με γενικώς πολύ υψηλές συγκεντρώσεις να μην «επιβάλλονται» σε ουσίες με μικρότερες. Συνήθως εφαρμόζεται η κλασσική κανονικοποίηση (μέσος όρος μηδέν και τυπική απόκλιση ίσα με τη μονάδα) ή κάποια άλλη τεχνική (π.χ.

Pareto scaling, Range scaling κ.α.). [13] Προσοχή χρειάζεται επίσης στην ανίχνευση και αντιμετώπιση ακραίων δειγμάτων (outliers).

2.4 Ανάλυση δεδομένων

Στις εργασίες μεταβολομικής χρησιμοποιούνται συνήθως μέθοδοι πολυπαραμετρικής στατιστικής ανάλυσης για την επεξεργασία των δεδομένων. Στην ενότητα αυτή θα παρουσιάσουμε ορισμένες μεθόδους προβολής σε χώρους λιγότερων διαστάσεων, όπως η Ανάλυση Κυρίαρχων Συνιστωσών (Principal Components Analysis - PCA). Θα εξηγήσουμε επίσης τα διαγράμματα τύπου heatmap που συναντούνται συχνά στη βιβλιογραφία. Και οι δυο τεχνικές έχουν στόχο τον εντοπισμό συστάδων στα δεδομένα. Θα δούμε επίσης τις καμπύλες τύπου ROC (Receiver Operating Characteristic) που χρησιμοποιούνται για τη μέτρηση της ποιότητας διαχωρισμού ενός συστήματος δυο κλάσεων, καθώς μεταβάλλεται η τιμή ενός κατωφλίου. Αυτές οι μέθοδοι συναντώνται συχνά στις εργασίες μεταβολομικής και για αυτόν τον λόγο παρουσιάζονται σε αυτό το σημείο. Με άλλες, όχι και τόσο συνήθεις μεθόδους εξόρυξης δεδομένων στη μεταβολομική, θα ασχοληθούμε στο επόμενο κεφάλαιο.

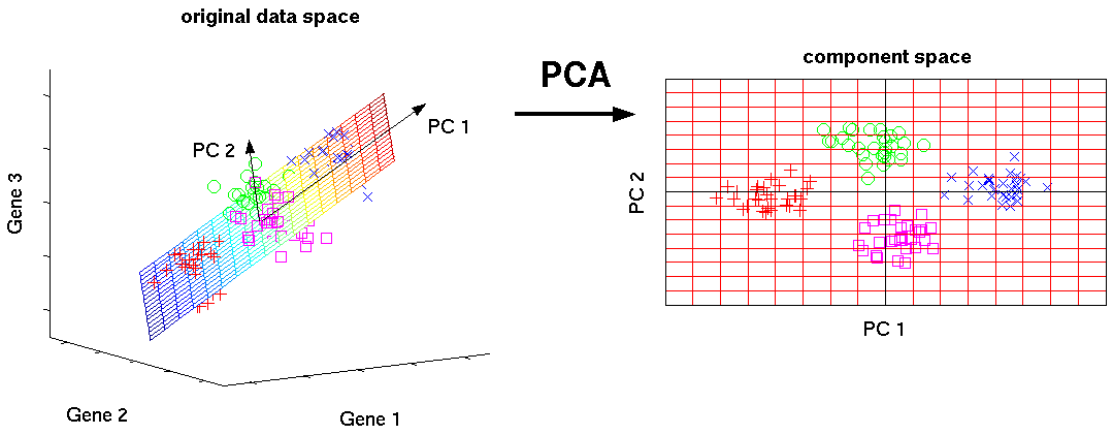
2.4.1 Μέθοδοι προβολής

Τα δεδομένα μεταβολομικής περιέχουν συνήθως δεκάδες έως χιλιάδες μεταβλητές. Δεν είναι εύκολο να φανταστούμε δεδομένα σε πολυδιάστατους χώρους, πόσο μάλλον να τα οργανώσουμε σε ομάδες. Για το λόγο αυτό χρησιμοποιούνται μέθοδοι που προβάλλουν τα δεδομένα σε χώρους δύο ή τριών διαστάσεων. Οι νέες μεταβλητές-διαστάσεις που δημιουργούνται έχουν διαφορετικό νόημα από τις προηγούμενες, το οποίο ίσως να είναι ευκολότερα αντιληπτό ή ίσως αυτό να έχει τελικά σημασία. Για παράδειγμα, αν σε ένα σύνολο δεδομένων έχουμε δυο διαστάσεις ημερομηνίας (π.χ. ημερομηνία γέννησης και ημερομηνία θανάτου), μπορούμε να τις προβάλλουμε σε μια διάσταση, αυτήν της ηλικίας, η οποία ίσως στο πρόβλημα που μελετάμε να σημαίνει πολύ περισσότερα πράγματα από ότι δυο ημερομηνίες. Ο μετασχηματισμός εδώ είναι προφανής: αφαιρούμε την ημερομηνία γέννησης από την ημερομηνία θανάτου, με τον κατάλληλο τρόπο. Αντιστοίχως, ο λόγος δυο μεταβλητών, όπως το βάρος ως προς το ύψος, μπορεί να είναι πιο χρήσιμος από ότι οι δυο μεταβλητές ξεχωριστά. Συνήθεις τέτοιοι μετασχηματισμοί παράγονται με γραμμικό συνδυασμό των μεταβλητών.

Η συνηθέστερα χρησιμοποιούμενη, ίσως, μέθοδος στις εργασίες μεταβολομικής είναι η Ανάλυση Κυρίαρχων Συνιστωσών (PCA). [1] Παρατηρήστε το σχήμα 2.4. Τα δεδομένα απεικονίζονται αρχικά σε ένα ορθογώνιο σύστημα τριών αξόνων. Με τη μέθοδο PCA, παράγουμε τόσους νέους άξονες, όσοι και οι αρχικοί, διατηρώντας την ορθογωνιότητα του συστήματος. Η αλλαγή του προσανατολισμού δεν αλλάζει τα δεδομένα ή τη διασπορά τους, παρά μόνο τον τρόπο απεικόνισης. Το ίδιο αποτέλεσμα θα είχαμε αν απλώς περιστρέφαμε το σχήμα στον χώρο.

Οι νέοι άξονες δημιουργούνται ως εξής:

1. Ο πρώτος άξονας επιλέγεται στην διεύθυνση όπου παρατηρείται η μέγιστη διασπορά στα δεδομένα. Στο σχήμα 2.4 φαίνεται ξεκάθαρα ότι η μέγιστη διασπορά βρίσκεται



Σχήμα 2.4: Εφαρμογή της μεθόδου PCA και απεικόνιση από 3D σε 2D για παράδειγμα δεδομένων. Μετά την εφαρμογή, είναι ευδιάκριτες οι διαφορετικές ομάδες. *(Πηγή: Matthias Scholz PhD. Thesis - άδεια CC BY)*

στη διεύθυνση του άξονα PC1. Ο άξονας αυτός προκύπτει ως γραμμικός συνδυασμός των αρχικών αξόνων, όπως και οι επόμενοι.

2. Ο δεύτερος άξονας επιλέγεται υποχρεωτικά κάθετος στον πρώτο, προς την κατεύθυνση όπου μεγιστοποιείται και πάλι η διασπορά. Αυτός είναι ο PC2.

3. Ο τρίτος άξονας (δεν απεικονίζεται) επιλέγεται κάθετος στους προηγούμενους, προς την κατεύθυνση μέγιστης διασποράς. Στη συγκεκριμένη περίπτωση είναι ο τελευταίος, αφού το αρχικό σύστημα ήταν 3D, και θα τον συμβολίζαμε PC3.

Η συνολική διασπορά για το σύστημα των τριών νέων αξόνων είναι ίδια με τη συνολική διασπορά για το σύστημα των τριών αρχικών αξόνων. Μπορούμε όμως, κρατώντας μονάχα τους δυο πρώτους άξονες, να παραστήσουμε ένα μεγάλο μέρος της διασποράς. Ας το δούμε διαφορετικά: το συγκεκριμένο 3D σχήμα μπορούμε να πούμε ότι μοιάζει σε επίπεδο σχήμα, με κάποιες μικρές «ανωμαλίες» στην επιφάνειά του. Κατά βάση όμως, μοιάζει να είναι ένα επίπεδο σχήμα, το οποίο θα μπορούσε να περιγραφεί αρκετά καλά από τους δυο πρώτους άξονες. Έτσι, περιστρέφοντας κατάλληλα το σχήμα, έχουμε ένα διάγραμμα δύο διαστάσεων που περιέχει σχεδόν αναλοίωτη (στην συγκεκριμένη περίπτωση), την πληροφορία που δίνει και το αντίστοιχο διάγραμμα τριών διαστάσεων. Όμως στο 2D σχήμα, είναι ευδιάκριτες οι διαφορετικές ομάδες των δεδομένων.

Η τεχνική αυτή μπορεί να εφαρμοστεί ώστε πάρα πολλές μεταβλητές να αντιστοιχηθούν τελικά σε μόνο 2 ή 3 και να είναι ευκολότερη η παρατήρηση και η επεξεργασία τους. Κατά την PCA προκύπτουν τόσες «συνιστώσες» ή «συστατικά» όσοι και οι κύριοι άξονες. Από ένα σημείο και μετά, ωστόσο, μειώνεται πολύ η πρόσθετη συνεισφορά στη συνολική διασπορά, με αποτέλεσμα να μην βελτιώνεται αισθητά η περιγραφή (ενώ ταυτόχρονα αυξάνεται η περιπλοκότητα του συστήματος). Μάλιστα, μπορεί οι πρόσθετες μεταβλητές να

περιγράφουν απλώς τον θόρυβο του συστήματος. Για το λόγο αυτό, βασιζόμαστε μονάχα σε ορισμένο πλήθος από τις πρώτες συνιστώσες, τις οποίες και ονομάζουμε «κύριες συνιστώσες» (principal components).

Με σχετικά παρόμοιο τρόπο λειτουργεί και η παλινδρόμηση μερικών ελαχίστων τετραγώνων (Partial Least Squares regression - PLS). [1] Βασική διαφορά είναι ότι, εκτός από το να προσπαθεί να καλύψει τη μέγιστη διακύμανση, προσπαθεί να βρει άξονες που να συσχετίζονται κατά το δυνατόν με κάποια κλάση. Στη βασική εκδοχή της, η μέθοδος αναφέρεται σε αριθμητικές κλάσεις. Όπως υποδηλώνει και το όνομά της, προορίζεται για παλινδρόμηση και όχι για κατηγοριοποίηση. Για κατηγορικές κλάσεις (δηλαδή διακριτές ομάδες και όχι ένα σύνολο τιμών) χρησιμοποιείται ευρέως η παραλλαγή PLS Discriminant Analysis (PLS-DA). Η μέθοδος αυτή μπορεί επίσης να κατατάσει τις (αρχικές) μεταβλητές σύμφωνα με τη «σημαντικότητά» τους στο νέο σύστημα απεικόνισης.

Αφού παραστήσουμε το σύστημα σε λιγότερες διαστάσεις, παρατηρούμε αν σχηματίζονται διακριτές ομάδες. Στο σημείο αυτό μπορούμε να παραστήσουμε στα σημεία των δεδομένων και την κλάση τους, αν είναι κλινικά γνωστή. Ο διαχωρισμός του σχήματος 2.4 πλησιάζει μια ιδανική κατάσταση. Στην πράξη, συναντάμε αρκετές φορές «ξένες προσμίξεις» σε ομάδες σημείων. Με αλγοριθμικό τρόπο μπορούμε να σχηματίσουμε διαχωριστικές γραμμές που να περικλείουν όλα τα πιθανά σημεία κάθε κλάσης με ένα βαθμό βεβαιότητας. Στη συνέχεια, μπορούμε να υπολογίσουμε ένα μέτρο επιτυχίας του διαχωρισμού. Π.χ. θα μπορούσαμε να υπολογίσουμε πόσα σημεία από κάθε κλάση βρέθηκαν μόνο στη σωστή περιοχή (δηλαδή αντιστοιχήθηκαν στη σωστή κατηγορία), ως προς το σύνολο των σημείων που ανήκουν σε αυτήν την κλάση. Αθροίζοντας τα επιμέρους κλάσματα μπορούμε να παράξουμε ένα μέτρο επιτυχίας. Το συγκεκριμένο μέτρο ωστόσο δεν λέει πάντοτε όλη την αλήθεια: αν έχουμε ένα σύνολο ατόμων που μόνο το 5% είναι ασθενή, μπορούμε, αντιστοιχίζοντας όλα τα σημεία στην υγιή ομάδα, να έχουμε 95% επιτυχία στον διαχωρισμό! Για το λόγο αυτό είναι δόκιμο να λαμβάνουμε υπ' όψιν και άλλα κριτήρια, όπως θα δούμε παρακάτω.

2.4.2 Διαγράμματα τύπου heatmap

Ορισμένες φορές είναι χρήσιμο να αναπαραστήσουμε τον πίνακα των δεδομένων με κάποια χρωματική κλίμακα. Αφού ομαδοποιήσουμε τις γραμμές του πίνακα που αντιστοιχούν σε δείγματα της ίδιας (γνωστής) κλάσης, χρωματίζουμε κάθε κελί του πίνακα σύμφωνα με την τιμή του σε σχέση με τα υπόλοιπα κελιά της ίδιας στήλης. Παρατηρώντας το συνολικό διάγραμμα, ενδεχομένως να μπορούμε να διακρίνουμε ουσίες με αρκετά διαφορετικές συγκεντρώσεις για διαφορετικές ομάδες. Ένα χαρακτηριστικό παράδειγμα φαίνεται στο άρθρο των Zacharias et al. [3] που εξετάζει το πρόβλημα που θα μελετήσουμε στο κεφάλαιο 5. Στο ίδιο διάγραμμα ενδέχεται να παρουσιάζονται μεμονωμένες γραμμές με έντονα διαφορετικές τιμές για πολλές στήλες. Ένα δείγμα που έχει δώσει μια τέτοια γραμμή είναι πιθανώς παραπλανητικό (outlier). Είναι λοιπόν μια μέθοδος που μπορεί να δώσει χρήσιμες πρώτες πληροφορίες και να καθοδηγήσει τις επόμενες κινήσεις μας.

Συχνά τα διαγράμματα αυτά συνοδεύονται και από δυο ιεραρχικά δέντρα. Το ένα από αυτά κατατάσσει τους διάφορους μεταβολίτες (γενικότερα, τις παρατηρούμενες μεταβλητές) σε επίπεδα μεταξύ τους συσχέτισης. Δυο μεταβολίτες που συνδέονται άμεσα, μέσα από

έναν και μόνο κόμβο, παρουσιάζουν μεγάλη συσχέτιση μεταξύ τους. Αυτοί, με τη σειρά τους, είναι λιγότερο συσχετισμένοι με τους μεταβολίτες με τους οποίους συνδέονται μέσα από δυο κόμβους του δενδροδιαγράμματος και ούτω καθ' εξής. Αντίστοιχη πληροφορία μπορεί να αντληθεί μέσω του correlation heatmap (δεν πρέπει να συγχέονται). Το άλλο δενδοδιάγραμμα κατατάσσει τα διάφορα δείγματα σε συστάδες. Έτσι, δυο δείγματα που συνδέονται μέσω ενός κόμβου βρίσκονται πιο «κοντά» μεταξύ τους από ότι δυο δείγματα που συνδέονται μέσω περισσότερων κόμβων.

2.4.3 Συσταδοποίηση

Τι σημαίνει όμως ότι δυο δείγματα βρίσκονται «κοντά» ή «μακριά» σε έναν πολυδιάστατο χώρο; Δυο σημεία σε ένα επίπεδο απέχουν, ως γνωστόν, όσο η τετραγωνική ρίζα του αθροίσματος των τετραγώνων των διαφορών των αντίστοιχων συντεταγμένων τους. Η Ευκλείδια αυτή απόσταση επεκτείνεται εύκολα σε περισσότερες διαστάσεις, προσθέτοντας απλώς τους αντίστοιχους όρους τετραγώνων διαφοράς συντεταγμένων:

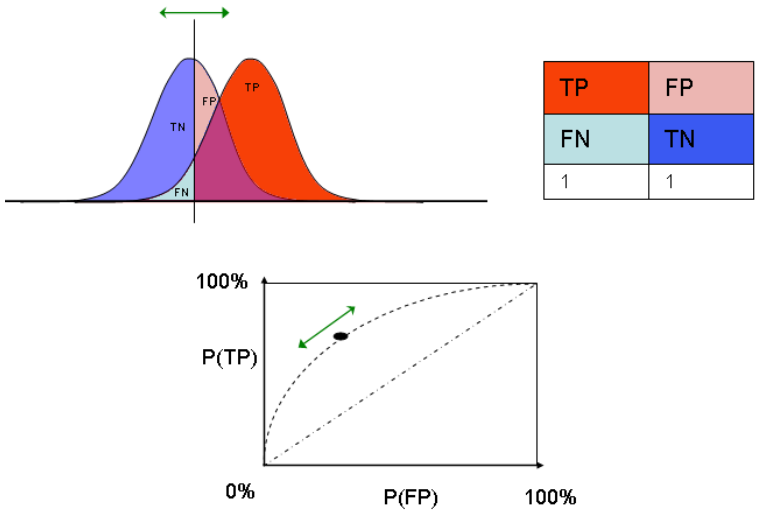
$$d(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \ldots + (x_{ik} - x_{jk})^2} \tag{2.1}$$

όπου k η μέγιστη διάσταση του χώρου. [1] Υπάρχουν και άλλου είδους αποστάσεις, όπως η απόσταση «οικοδομικού τετραγώνου» (city block distance ή Manhattan distance) ή η απόσταση Mahalanobis. Συχνά οι εκφράσεις αυτές αναφέρονται ως «μετρικές απόστασης» ή «μετρικές ομοιότητας».

Η απόσταση μεταξύ δυο δειγμάτων ενδιαφέρει στην συσταδοποίηση (clustering) των δεδομένων. Δυο δείγματα που βρίσκονται κοντά μεταξύ τους έχουν μεγαλύτερη πιθανότητα να βρεθούν στην ίδια συστάδα. Η συσταδοποίηση όμως είναι μια διαδικασία που μπορεί να γίνει «από πάνω προς τα κάτω» είτε «από κάτω προς τα πάνω». Μπορούμε να ξεκινήσουμε δηλαδή με μια συστάδα που να περιλαμβάνει όλα τα δείγματα και να αρχίσουμε να δημιουργούμε μικρότερες υπο-συστάδες χωρίζοντας ομάδες που φαίνονται να είναι διακριτές μεταξύ τους. Μπορούμε επίσης να ξεκινήσουμε με (τυχαία κατανεμημένα) κέντρα συστάδων και να αντιστοιχίζουμε κάθε δείγμα στην κοντινότερη για αυτό συστάδα. Κάθε φορά που προτίθεται ένα δείγμα σε μια συστάδα, το «κέντρο» της μπορεί να μετατοπίζεται και τελικά να έχουμε συγχωνεύσεις στοιβάδων. Περισσότερα για τους διάφορους αλγορίθμους συσταδοποίησης θα δούμε στο επόμενο κεφάλαιο.

2.4.4 Έλεγχος απόδοσης: καμπύλες ROC

Συχνά διαχωρίζουμε δύο ομάδες σύμφωνα με την τιμή ενός κριτηρίου. Το κριτήριο αυτό μπορεί να είναι η συγκέντρωση ενός μεταβολίτη ή ένα άλλο κριτήριο που να συνδυάζει περισσότερους μεταβολίτες. Στην ιδανική περίπτωση που οι δύο ομάδες είναι εμφανώς διαχωρισμένες μεταξύ τους, μια κατάλληλη τιμή για αυτό το κριτήριο μπορεί να δώσει τέλειο διαχωρισμό. Ωστόσο, όπως είπαμε, συχνά οι ομάδες έχουν μια κοινή «διεπιφάνεια». Η επιλογή της βέλτιστης τιμής τότε δεν είναι προφανής. Βελτιώνοντας την πρόβλεψη για τη μια ομάδα, χειροτερεύουμε την πρόβλεψη για την άλλη ομάδα. Όπως στον διαχωρισμό δυο



Σχήμα 2.5: Παράδειγμα καμπύλης ROC. Μετακινώντας το κατώφλι πάνω από το οποίο τα άτομα αντιστοιχίζονται στην κλάση "Positives" αλλάζει η επιτυχία πρόβλεψης και των δυο κλάσεων. *(Πηγή: wikimedia commons - άδεια CC BY)*

υγρών διαφορετικής πυκνότητας σε μια διαχωριστική χωάνη: για να αυξήσουμε την καθαρότητα του ενός, αναγκαστικά μειώνουμε την καθαρότητα του άλλου. Αν υπάρχει κάποιο εξωτερικά ορισμένο κριτήριο, όπως π.χ. η σωστή κατηγοριοποίηση όλων των ασθενών ατόμων, ή π.χ. το 99% αυτών, η λύση είναι προφανής. Αν ωστόσο θέλουμε να θέσουμε ένα όριο-κατώφλι που να διαχωρίζει βέλτιστα τα δεδομένα μας, τότε μπορούμε να χρησιμοποιήσουμε τις καμπύλες ROC (Receiver Operating Characteristic). Ο όρος προέρχεται από την αναγνώριση σημάτων, όπου χαρακτηρίζει την ανταλλαγή σημάτων ως προς την σωστή ή εσφαλμένη απόκριση του δέκτη σε συνθήκες θορύβου.

Οι καμπύλες αυτές έχουν τη μορφή του σχήματος 2.5. Για λίγα δεδομένα, όπως συνήθως έχουμε στη μεταβολομική, οι καμπύλες αυτές είναι κλιμακωτές. Στο εξής θα θεωρούμε ότι η κλάση "Positive" αντιστοιχεί σε ασθενή άτομα, ενώ η κλάση "Negative" σε υγιή. Στον κατακόρυφο άξονα απεικονίζεται το ποσοστό των ασθενών που κατηγοριοποιήθηκαν σωστά ως ασθενείς (True Positives - TP). Στον οριζόντιο άξονα απεικονίζεται το ποσοστό των υγιών που κατηγοριοποιήθηκαν λανθασμένα ως ασθενείς (False Positives - FP). Όπως θα περιμέναμε, για να αυξήσουμε το κλάσμα των ασθενών που κατηγοριοποιούνται σωστά, θα πρέπει να ανεχθούμε μια αύξηση και στο κλάσμα των υγιών ατόμων για τα οποία έχουμε «λανθασμένο συναγερμό».

Δυο χρήσιμα μεγέθη που μπορούμε να ορίσουμε είναι η «ευαισθησία» (sensitivity-Sn) και η «ευστοχία» (specificity-Sp) [13]:

$$Sn = \frac{TP}{TP + FN} \tag{2.2}$$

$$Sp = \frac{TN}{TN + FP} \tag{2.3}$$

Οι καμπύλες ROC συχνά εμφανίζονται έχοντας στον κατακόρυφο άξονα την ευαισθησία, Sn και στον οριζόντιο άξονα τον όρο $1 - Sp$. Για να κατασκευάσουμε μια τέτοια καμπύλη:

1. Ταξινομούμε τα διαθέσιμα άτομα, σύμφωνα με την τιμή που δίνουν για το κριτήριο που εξετάζουμε. Αν αυτό είναι η συγκέντρωση ενός μεταβολίτη, κατατάσσουμε τα άτομα σύμφωνα με τη συγκέντρωση που εμφανίζουν για αυτόν τον μεταβολίτη.

2. Έστω ότι χαρακτηρίζουμε ως ασθενή-positives τα άτομα που έχουν συγκέντρωση ίση ή μεγαλύτερη από το κατώφλι που θέτουμε. Δοκιμάζουμε το κατώφλι στη συγκέντρωση που αντιστοιχεί σε κάθε σημείο.

3. Για κάθε τιμή κατωφλίου (ή αντίστοιχο σημείο) υπολογίζουμε τα αντίστοιχα κλάσματα TP και FP και τα απεικονίζουμε με σημεία στο διάγραμμα.

4. Ενώνουμε τα σημεία αυτά με ευθύγραμμα τμήματα.

Ο «παράδεισος» του ερευνητή που ψάχνει ένα καλό μοντέλο διαχωρισμού βρίσκεται στην άνω αριστερή γωνία του διαγράμματος. Στο ιδανικό σημείο (0,1) θα είχαμε εντοπίσει σωστά όλα τα ασθενή άτομα, χωρίς να έχουμε κάνει λάθος σε κανένα υγιές. Έτσι, επιλέγουμε ως τιμή κατωφλίου, την τιμή που αντιστοιχεί στο σημείο της καμπύλης ROC που βρίσκεται πιο κοντά στην άνω αριστερή γωνία του διαγράμματος. Για να έχει νόημα ένα μοντέλο, πρέπει η καμπύλη του να βρίσκεται στην περιοχή πάνω από την διαγώνιο του διαγράμματος. Αν η καμπύλη βρίσκεται πάνω στην διαγώνιο, αυτό σημαίνει ότι το μοντέλο συμπεριφέρεται τυχαία.

Ως μέτρο της απόδοσης ενός μοντέλου χρησιμοποιούμε το εμβαδό της περιοχής κάτω από την καμπύλη ROC (Area Under Curve - AUC), το οποίο θέλουμε να είναι όσο το δυνατόν κοντά στη μονάδα. Αυτό όμως δεν είναι απόλυτο, κυρίως όταν θέλουμε να συγκρίνουμε δυο διαφορετικά μοντέλα. Για παράδειγμα, ενδέχεται οι καμπύλες των δυο μοντέλων να εναλλάσσονται ως προς το ποια βρίσκεται υψηλότερα σε επιμέρους περιοχές του διαγράμματος, αλλά να έχουν γενικώς το ίδιο AUC. Σε αυτήν την περίπτωση χρησιμοποιείται μόνο το εμβαδό της περιοχής ενδιαφέροντος (partial AUC). Αναλόγως λοιπόν με το αν επιθυμούμε υψηλότερη ευαισθησία ή υψηλότερη ευστοχία, συγκρίνουμε το εμβαδό στην αντίστοιχη περιοχή του διαγράμματος. Τέλος, μεγάλη σημασία έχει το γεγονός ότι το μέτρο αυτό δεν δίνει παραπλανητικά αποτελέσματα όταν υπάρχει ανισοκατανομή στις κλάσεις. [13]

2.4.5 Επιλογή μεταβολιτών

Αναφέραμε ήδη τρόπους μείωσης των μεταβλητών του συστήματος. Χρησιμοποιήσαμε νέες μεταβλητές, οι οποίες προέρχονταν από συνδυασμό των αρχικών. Στην πράξη ωστόσο,

χρειαζόμαστε απλά μοντέλα, τα οποία να μπορούν να εφαρμοστούν κλινικά. Μοντέλα που να βασίζονται σε ελάχιστους μεταβολίτες, αλλά και να δίνουν καλές προβλέψεις.

Αν έχουμε ήδη κατατάξει τους διαθέσιμους μεταβολίτες-μεταβλητές ως προς κάποια σειρά «σημαντικότητας» (βλ. μέθοδο PLS-DA), μπορούμε να εφαρμόσουμε την εξής μέθοδο (feature filtering): [13]

1. Επιλέγουμε διαδοχικά $1 \dots N$ μεταβολίτες, σύμφωνα με τη σειρά κατάταξής τους.
2. Κατασκευάζουμε ένα νέο μοντέλο χρησιμοποιώντας μόνο τις τρέχουσες μεταβλητές.
3. Ελέγχουμε την επιτυχία του μοντέλου (π.χ. με καμπύλες ROC).
4. Αν έχει επιτευχθεί η επιθυμητή επιτυχία σταματάμε, διαφορετικά προσθέτουμε επιπλέον μεταβλητές από τη λίστα και επαναλαμβάνουμε τη διαδικασία.

Παρότι ο αλγόριθμος αυτός είναι αρκετά απλός στη σύλληψή του, απαιτεί καλή γνώση του αλγορίθμου που χρησιμοποιείται για τη μοντελοποίηση, ενώ ενδέχεται να απαιτεί τροποποίηση της λίστας με τη «σημαντικότητα» των μεταβολιτών. Επίσης, δεν υπάρχει καμία θεωρητική εγγύηση ότι το βέλτιστο υποσύνολο θα περιλαμβάνει τις κορυφαίες μεταβλητές της λίστας. Στην ενότητα 3.7 θα παρουσιάσουμε και άλλες μεθόδους επιλογής μεταβλητών.

2.4.6 Εξερευνώντας τα μεταβολικά μονοπάτια

Η δημιουργία μοντέλων που βασίζονται σε λίγους μεταβολίτες δεν έχει σκοπιμότητα μονάχα μαθηματική ή κλινική. Ας μην ξεχνάμε ότι σκοπός της μεταβολομικής δεν είναι μόνο να φτιάξει ένα «μαύρο κουτί» που να προβλέπει επιτυχώς ασθένειες ή άλλα φαινόμενα. Σκοπός είναι και αυτή η μαθηματική περιγραφή του σύνθετου βιολογικού συστήματος που μελετάμε να δώσει πληροφορίες που θα καθοδηγήσουν την επιστημονική κοινότητα στην βαθύτερη κατανόηση των βιοχημικών διεργασιών. Έτσι, καταλήγοντας σε ένα κατά το δυνατόν μικρό, «κομψό» μοντέλο μπορούμε να κατανοήσουμε και τον τρόπο λειτουργίας του. Μέσα από αυτήν την οδό μπορεί να γίνει ένα μεταβολομικό μοντέλο αποδεκτό από τους κλινικούς επιστήμονες. Επίσης, μέσα από αυτήν την οδό μπορούν να σχεδιαστούν κατάλληλα φάρμακα για τις ασθένειες που ενδεχομένως μελετώνται. Πακέτα λογισμικού μεταβολομικής, όπως το MetaboAnalyst, μπορούν να αναλύσουν τα βιοχημικά μονοπάτια που συνδέονται με τους μεταβολίτες που επιλέχθηκαν και να βοηθήσουν στην εξαγωγή χρήσιμων συμπερασμάτων.

2.5 Προκλήσεις

Ο τομέας της μεταβολομικής συγκεντρώνει επιστήμονες πολλών διαφορετικών κλάδων. Πειραματιστές, κλινικοί επιστήμονες, φαρμακοποιοί, βιοχημικοί, στατιστικοί, πληροφορικοί και πολλοί άλλοι. Βασική πρόκληση είναι να δημιουργηθούν κοινοί κώδικες επικοινωνίας και κοινές, γενικώς αποδεκτές, πρακτικές σε ότι αφορά την έρευνα και τις δημοσιεύσεις. Για παράδειγμα, στο εισαγωγικό, ελεύθερης πρόσβασης άρθρο των Xia, Broadhurst et al. [13] επισημαίνονται μερικά συχνά προβλήματα και δίνονται ορισμένοι κανόνες «ορθής πρακτικής» ως προς την παραγωγή και τη δημοσίευση σχετικών ερευνών.

Μεγάλη είναι η ευκαιρία επίσης, οι διαφορετικοί κλάδοι να ενσωματώσουν και να μεταδώσουν τις γνώσεις που τους διαφοροποιούν, πάνω σε κοινά προβλήματα. Για παράδειγμα, κάτι που υποδεικνύεται και από το πνεύμα της εργασίας αυτής, είναι μια καλή ευκαιρία οι εργαστηριακοί επιστήμονες να επωφεληθούν από την ποικιλία ιδεών που μπορεί να προσφέρει η εξόρυξη δεδομένων και, ειδικότερα, η μηχανική μάθηση στην ανάλυση των δεδομένων που προκύπτουν. Είναι πρόκληση η δημιουργία μοντέλων με μεγαλύτερη επιτυχία, λιγότερες μεταβλητές και υψηλότερη ευρωστία, με μικρότερες υπολογιστικές απαιτήσεις. Θα ήταν ευχάριστη εξέλιξη αν κάποια μέρα ασθενείς μπορούσαν να χρησιμοποιούν μικρές φορητές συσκευές που θα κάνουν έγκαιρες διαγνώσεις και θα βασίζονται αποκλειστικά σε μεταβολομικά μοντέλα.

Άλλη μια πρόκληση είναι η διαχείριση των γνωστών και (κυρίως) των άγνωστων μεταβολιτών. Αντίθετα με το γονιδίωμα, το μεταβόλωμα του ανθρώπου δεν έχει χαρτογραφηθεί πλήρως ακόμη. Ειδικότερα στα μη στοχευμένα πειράματα προκύπτουν πολύ συχνά άγνωστες κορυφές οι οποίες φαίνεται να είναι από τα κρίσιμα κομμάτια του εκάστοτε «πάζλ». Χρειάζεται λοιπόν να γίνει αρκετή πειραματική δουλειά σε αυτήν την κατεύθυνση. Οι αντίστοιχες βάσεις δεδομένων πρέπει να εμπλουτιστούν, ενώ είναι σημαντικό η κοινότητα να δημοσιεύει κατά το δυνατόν τα δεδομένα που διαθέτει, ώστε να είναι δυνατή η σε βάθος ανάλυση κάθε προβλήματος, αλλά και να μπορούν να γίνονται συγκρίσεις μεταξύ αντίστοιχων προβλημάτων. Για παράδειγμα, συγκρίνοντας διαφορετικές μορφές μιας ασθένειας, τις οποίες μελετούν διαφορετικά εργαστήρια, μπορεί να στραφεί το ενδιαφέρον προς ορισμένους μεταβολίτες και να απαλειφθούν μεταβολίτες που εμφανώς δεν σχετίζονται με το ευρύτερο πρόβλημα. Η τεχνική αυτή ονομάζεται "μέτα-ανάλυση". [2]

Η μεγαλύτερη πάντως πρόκληση, ίσως είναι να δημιουργηθούν μοντέλα τόσο αξιόπιστα, που να χρησιμοποιηθούν ευρέως από την ιατρική κοινότητα. Αυτή τη στιγμή, η μόνη ίσως εκτεταμένη κλινική εφαρμογή βρίσκεται στον έλεγχο των νεογνών για έμφυτα προβλήματα μεταβολισμού. [13]

Κεφάλαιο 3

Εξόρυξη δεδομένων

Στο προηγούμενο κεφάλαιο είδαμε την τρέχουσα κατάσταση στη μεταβολομική, η οποία αποτελεί μονάχα ένα από τα πάρα πολλά πεδία εφαρμογής της εξόρυξης δεδομένων. Παρουσιάσαμε μερικά από τα βασικότερα εργαλεία ανάλυσης δεδομένων που χρησιμοποιούνται από τη διεθνή κοινότητα, παραλείποντας όμως να αναφερθούμε σε μεθόδους μηχανικής μάθησης. Τώρα, έχοντας αποκτήσει μια εικόνα της εφαρμογής που θα μελετήσουμε, μπορούμε να δούμε με ποιους τρόπους μπορούμε να μπούμε βαθύτερα στα δεδομένα, δημιουργώντας συνθετότερα μοντέλα με υψηλότερο βαθμό επιτυχίας στις προβλέψεις.

Η μηχανική μάθηση αναφέρεται στις τεχνικές που επιτρέπουν στους Η/Υ να «μαθαίνουν» από δεδομένα, χωρίς όμως να έχουν προγραμματιστεί συγκεκριμένα για αυτά. Δεν αναφερόμαστε σε εφαρμογή θεωρητικών μοντέλων, αλλά στην αλγοριθμική κατασκευή μοντέλων, σύμφωνα με τα πρότυπα που ανιχνεύονται στα δεδομένα. Έτσι, μπορούμε να χρησιμοποιήσουμε ένα σύνολο δεδομένων για να εκπαιδεύσουμε ένα μοντέλο που να περιγράφει καλά αυτό το σύνολο αλλά και να ανταποκρίνεται επιτυχώς σε εξωτερικά δεδομένα ελέγχου. Με την έννοια της καλής περιγραφής, εννοούμε την απόδοση σωστών τιμών σε κάποια μεταβλητή των δεδομένων την οποία ονομάζουμε κλάση. Στη συνέχεια, μπορούμε να χρησιμοποιήσουμε το μοντέλο για να αντιστοιχίσουμε νέα δείγματα στην αντιπροσωπευτικότερη για κάθε δείγμα κλάση.

Σε αυτό το κεφάλαιο θα δούμε τα διάφορα στάδια της εξόρυξης δεδομένων. Θα εξετάσουμε πώς δομούνται τα μοντέλα που προέρχονται από ορισμένους χαρακτηριστικούς αλγορίθμους μηχανικής μάθησης και θα δούμε πώς δημιουργούνται και πώς μπορούμε να συνδυάσουμε παραπάνω από ένα μοντέλα ή αλγορίθμους για τη δημιουργία καλύτερων μοντέλων. Θα δούμε επίσης πώς ελέγχουμε ένα μοντέλο ως προς την επιτυχία του και πώς μπορούμε να μειώσουμε την πολυπλοκότητά του. Τέλος, θα αναφέρουμε μερικούς τομείς εφαρμογής τεχνικών εξόρυξης δεδομένων.

3.1 Δομή δεδομένων

Ανάλογα με την εφαρμογή, μεταβάλλεται το πλήθος και η δομή των δεδομένων, καθώς και οι δυσκολίες. Τα προβλήματα μεταβολομικής είναι συνήθως μικρής έως μεσαίας κλίμακας. Οι μεταβλητές που μετρώνται είναι συνήθως μερικές δεκάδες ή εκατοντάδες, ενώ τα δείγ-

ματα δυστυχώς σπάνια είναι της τάξης των εκατοντάδων. Σε προβλήματα σχετικά με τον παγκόσμιο ιστό, τα δείγματα μπορεί να είναι πολύ πάνω από χιλιάδες και συχνά αναλύονται σε υπερυπολογιστικά συστήματα. Στην υπόλοιπη εργασία θα εστιάζουμε το ενδιαφέρον μας μόνο στη μεταβολομική.

Το πρόβλημα που θα μελετήσουμε στο κεφάλαιο 5 εμπεριέχει 106 δείγματα (instances) και 701 μεταβλητές (attributes). Τα δείγματα χωρίζονται σε δυο κλάσεις, ονομοιόμορφα κατανεμημένες (72 υγιείς άνθρωποι και 34 ασθενείς). Οι μεταβλητές αντιστοιχούν σε μικρά, συνεχόμενα, κομμάτια φασμάτων NMR (spectral bins). Το πρόβλημα που θα μελετήσουμε στο κεφάλαιο 6 εμπεριέχει 40 δείγματα και 38 μεταβλητές. Τα δείγματα είναι χωρισμένα σε 6 κλάσεις, σχετικά ομοιόμορφα κατανεμημένες (6+6+11+6+5+6 επίμυες). Οι μεταβλητές αντιστοιχούν σε χαρακτηριστικές κορυφές NMR συγκεκριμένων μεταβολιτών. Σε κανένα από τα δύο προβλήματα δεν υπήρχε πρόβλημα με άγνωστες τιμές. Και στις δύο περιπτώσεις, τα δεδομένα φιλοξενούνται σε έναν απλό πίνακα δύο διαστάσεων, όσο μεγάλες και αν είναι αυτές. Σε πιο περίπλοκα προβλήματα εξόρυξης δεδομένων μπορεί να απαιτούνται ειδικά σχεδιασμένες βάσεις δεδομένων, κάτι που δεν θα μας απασχολήσει στη συγκεκριμένη διπλωματική εργασία.

Αρχικά, τα δεδομένα πρέπει να συγκεντρωθούν σε έναν πίνακα κατάλληλης μορφής. Συνηθίζεται τα δείγματα να καταχωρούνται σε γραμμές και οι μεταβλητές του συστήματος σε στήλες, με την πρώτη γραμμή του πίνακα να περιέχει τους αντίστοιχους τίτλους. Ένας πλήρης πίνακας δεδομένων πρέπει να περιέχει την ταυτότητα, την κλάση και τις τιμές των μεταβλητών για κάθε δείγμα. Η σειρά μπορεί να διαφέρει ανάλογα με το λογισμικό που χρησιμοποιούμε, ωστόσο τέτοιες αλλαγές μπορούν εύκολα να γίνουν. Η ταυτότητα του δείγματος ενδέχεται να πρέπει να διαγραφεί πριν την τροφοδότηση σε κάποιο λογισμικό, ωστόσο τα δεδομένα θα πρέπει να δημοσιεύονται μαζί με τις ταυτότητες των δειγμάτων. Θα πρέπει επίσης να δημοσιεύεται η επεξεργασία στην οποία έχουν υποβληθεί τα δείγματα πριν καταχωρηθούν στον πίνακα. Καλό είναι τα δείγματα να είναι άμεσα συγκρίσιμα, π.χ. να έχουν ήδη γίνει οι κατάλληλες διορθώσεις σύμφωνα με τυχόν αραιώσεις κτλ. Το κάθε τι πρέπει να περιγράφεται με σαφήνεια και, αν οι μεταβλητές του συστήματος είναι μεταβολίτες, τα ονόματα αυτών πρέπει να καταγράφονται σύμφωνα με κάποιο κοινώς αποδεκτό πρότυπο. Επίσης, απαιτείται καλή οργάνωση ως προς τα αρχεία που χρησιμοποιούνται. Ένα λάθος σε αυτό το στάδιο μπορεί να σημαίνει στην καλύτερη περίπτωση επανάληψη μέρους ή ολόκληρης της ανάλυσης αργότερα.

3.2 Προεπεξεργασία

Πριν προχωρήσουμε στην ανάλυση των δεδομένων μπορούμε να κάνουμε κάποια προεπεξεργασία σε αυτά. Μια καλή προεπεξεργασία μπορεί να οδηγήσει σε πολύ καλύτερα μοντέλα στη συνέχεια. Αντιθέτως, μια κακή (ή καθόλου) προεπεξεργασία μπορεί να οδηγήσει σε προβληματικά και μη αποδεκτά μοντέλα. Αρχικά, τα δεδομένα πρέπει να συμπληρωθούν όπου υπάρχουν άγνωστες τιμές. Έπειτα, πρέπει να καθαριστούν από προβληματικά, παραπλανητικά δείγματα (outliers), από ακραίες τιμές (extreme values) και από θόρυβο. Στη συνέχεια, πρέπει όλες οι μεταβλητές να κλιμακωθούν σε ένα εύρος τιμών ώστε να διευκολύνονται οι υπολογισμοί και οι συγκρίσεις. Σε ορισμένες περιπτώσεις, μπορεί να γίνει και

αναπαράσταση σε κάποιο χώρο λιγότερων μεταβλητών, όπως αναφέραμε στην ενότητα 2.4.1 ή, με άλλους τρόπους, να μειωθούν τα προς ανάλυση δεδομένα για ταχύτερη επεξεργασία.

3.2.1 Καθαρισμός δεδομένων

Είναι πιθανό να συναντήσουμε ελλιπή δεδομένα λόγω τεχνικών δυσκολιών ή άλλων λόγων. Εάν σε κάποια δείγματα υπάρχουν άγνωστες τιμές, τότε αυτές μπορούν να αντικατασταθούν με τον μέσο όρο των αντίστοιχων μεταβλητών για όλα τα δείγματα ή για τα δείγματα της ίδιας κλάσης (αν αυτή είναι γνωστή). Μπορούν επίσης να αντικατασταθούν με την πιο πιθανή τιμή, σύμφωνα με κάποια παλινδρόμηση.

Είναι επίσης πιθανό να υπάρχουν παραπλανητικά ή ακραία δείγματα στα δεδομένα. Αυτά μπορούν να εντοπιστούν μέσω κάποιας τεχνικής παλινδρόμησης ή με χωρισμό συστάδων. Δεδομένα που βρίσκονται μακριά από τις δημιουργούμενες συστάδες θεωρούνται outliers. Αυτά γενικώς αφαιρούνται ή αλλιώς διορθώνονται και χρησιμοποιούνται. Δείγματα με ακραίες τιμές μπορεί να μην είναι outliers, αλλά να δημιουργούν εμπόδια στη δημιουργία καλών μοντέλων. Σε αυτήν την περίπτωση τα αφαιρούμε, καθορίζουμε τα όρια μέσα στα οποία έχει εκπαιδευτεί το μοντέλο και το εφαρμόζουμε μόνο σε δείγματα που ανήκουν μέσα σε αυτά τα όρια. Η περιοχή μέσα στην οποία μπορούμε να χρησιμοποιήσουμε ένα μοντέλο λέγεται Domain of Applicability και υπολογίζεται με μεθοδολογίες που δεν θα εξετάσουμε, ωστόσο αναπτύσσονται εκτενώς στη βιβλιογραφία. [17, 18]

Στην περίπτωση που παρουσιάζεται θόρυβος (τυχαία σφάλματα) στα δεδομένα, μπορούν να εφαρμοστούν μέθοδοι εξομάλυνσης (smoothing). Γενικώς, διαχωρίζουμε τα δεδομένα σε μικρές ομάδες, σύμφωνα με τις τιμές για κάποια μεταβλητή η οποία παρουσιάζει θόρυβο. Οι ομάδες αυτές μπορεί να είναι ίσου πλάτους (δηλαδή να περιλαμβάνουν ένα εύρος τιμών, ακατάλληλος τρόπος για ασύμμετρα δεδομένα) ή ίσου βάθους (δηλαδή να περιλαμβάνουν ένα πλήθος δειγμάτων). Σε κάθε ομάδα γίνεται εξομάλυνση σύμφωνα με τον μέσο όρο, τη διάμεσο ή τα όρια τιμών για αυτήν. [19]

3.2.2 Μείωση εξεταζόμενων δεδομένων

Σε ένα πρόβλημα εξόρυξης δεδομένων, ενδέχεται το πλήθος των μεταβλητών και των δειγμάτων να είναι τόσο υψηλό που να δυσκολεύει ή να καθυστερεί την ανάλυση. Στην ενότητα 2.4.1 παρουσιάσαμε μεθόδους, όπως την PCA, για αντιστοίχιση πολλών μεταβλητών σε λιγότερες. Είδαμε επίσης ότι μέθοδοι όπως η PLS-DA κατατάσσουν τις μεταβλητές ως προς τη «σημαντικότητά τους». Μεταβλητές που συνεισφέρουν ελάχιστα στη διακύμανση των δεδομένων θα μπορούσαν να εξεταστούν ως προς την απαλοιφή τους.

Για τη μείωση του όγκου των δεδομένων μπορεί να γίνει κατάλληλη δειγματοληψία, από όλες τις κλάσεις που εμφανίζονται (αν είναι γνωστές). Η δειγματοληψία μπορεί να γίνει με ή χωρίς επαναπόθεση (replacement). Αυτό καθορίζει τη δυνατότητα του δείγματος να επιλεγεί παραπάνω από μια φορές. Μπορούμε επίσης να οργανώσουμε κοντινά σημεία σε μικρές συστάδες, δουλεύοντας στη συνέχεια μόνο με κάποιο αντιπροσωπευτικό σημείο (π.χ. το κεντροειδές). Συνεχείς μεταβλητές μπορούν να διακριτοποιηθούν. Ακόμη, εάν τα δεδομένα που χρησιμοποιούμε εμπεριέχουν κείμενο, εικόνα ή ήχο, μπορούμε να εφαρμόσουμε και τεχνικές συμπίεσης.

3.3 Δομή εξαγόμενης πληροφορίας

Τα μοντέλα μπορεί να έχουν, μεταξύ άλλων, μια από τις εξής μορφές: [1]

Πίνακες Πρακτικά ένας τέτοιος πίνακας έχει την ίδια μορφή με τον πίνακα που φιλοξενεί τα ίδια τα δεδομένα και δεν προσφέρει ιδιαίτερες πληροφορίες. Εάν ένα δείγμα έχει τις ίδιες τιμές με κάποια καταχώριση του πίνακα, τότε θα έχει και την ίδια κλάση. Πίνακες τέτοιας μορφής είναι κατάλληλοι και για αριθμητικές κλάσεις. Σε αυτή την περίπτωση καλούνται πίνακες παλινδρόμησης. Η ουσιαστικότερη διαφορά που μπορεί να έχει ένας πίνακας ενός μοντέλου από τον αρχικό είναι να αναπαριστά τα δεδομένα με λιγότερες μεταβλητές, π.χ. μόνο όσες φαίνεται να επηρεάζουν τις κλάσεις.

Γραμμικά (ή μη) μοντέλα Για αριθμητικές κλάσεις, μπορούμε να ορίσουμε κάποιο γραμμικό μοντέλο παλινδρόμησης, όπως μια ευθεία ελαχίστων τετραγώνων ή, σε έναν πολυδιάστατο χώρο, έναν γραμμικό συνδυασμό των μεταβλητών που να προσαρμόζεται στην κατανομή των δεδομένων. Επίσης, μπορούμε να ορίσουμε ευθείες που να διαχωρίζουν δυο ομάδες δεδομένων. Κατ' επέκταση, αντί για ευθείες, μπορούμε να ορίσουμε μη γραμμικά μοντέλα, ενώ διαχωρισμός περισσότερων κλάσεων μπορεί να γίνει με συνδυασμό περισσότερων της μιας διαχωριστικών γραμμών.

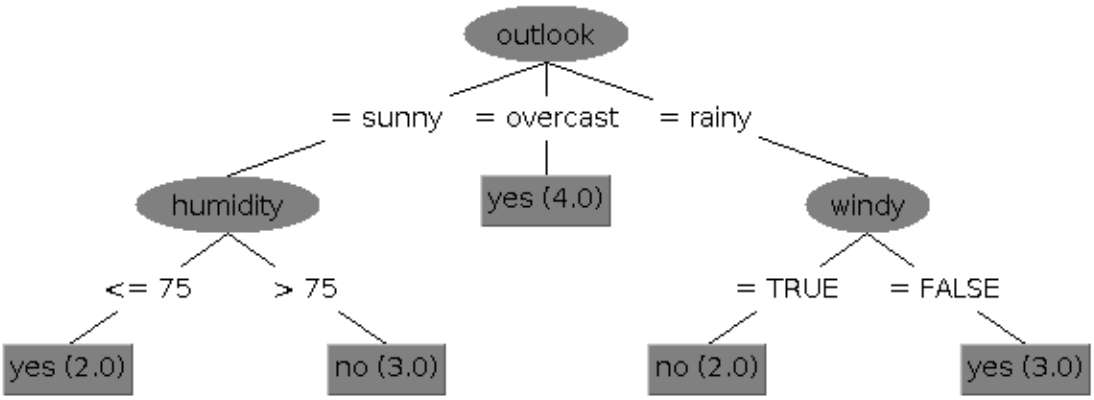
Δέντρα αποφάσεων Τα δέντρα αποφάσεων έχουν τη μορφή του σχήματος 3.1. Σε αυτό το δέντρο ελέγχεται η καταλληλότητα των καιρικών συνθηκών για ένα παιχνίδι (απροσδιόριστο ποιο). Στους κόμβους ενός δέντρου γίνεται έλεγχος της τιμής κάποιας μεταβλητής (ή κάποιας συνάρτησης των μεταβλητών). Για κατηγορικές μεταβλητές, συνήθως ακολουθούν τόσοι κλάδοι όσες και οι πιθανές τιμές της μεταβλητής. Για αριθμητικές μεταβλητές γίνεται έλεγχος ως προς κάποια σταθερά ή ως προς κάποια άλλη μεταβλητή. Μια μεταβλητή μπορεί να ελέγχεται σε πολλά σημεία του δέντρου. Τα "φύλλα" του δέντρου οδηγούν στην τιμή της κλάσης που προβλέπεται από αυτό. Δέντρα χρησιμοποιούνται κυρίως για κατηγορικές κλάσεις, χωρίς να αποκλείεται και η χρήση για αριθμητικές.

Κανόνες Ένα μοντέλο κανόνων περιέχει διαδοχικούς ελέγχους συνθηκών, οι οποίοι μπορεί να περιέχουν τους συνηθισμένους λογικούς τελεστές. Παρότι εκφράζουν με διαφορετικό τρόπο την ίδια πληροφορία με τα δέντρα, ορισμένες φορές μπορούν να την περιγράψουν σε λιγότερο χώρο, μπορούν να προστεθούν ευκολότερα νέοι έλεγχοι συνθηκών και κωδικοποιούνται ευκολότερα. Κανόνες μπορούν να χρησιμοποιούνται όχι μόνο για την πρόβλεψη της κλάσης, αλλά για τον συμπερασμό της τιμής οποιασδήποτε από τις μεταβλητές που συμμετέχουν (association rules). Τέλος, στη συνοπτικότητα των περιγραφών συμβάλει και η δυνατότητα χρήσης κανόνων με εξαιρέσεις.

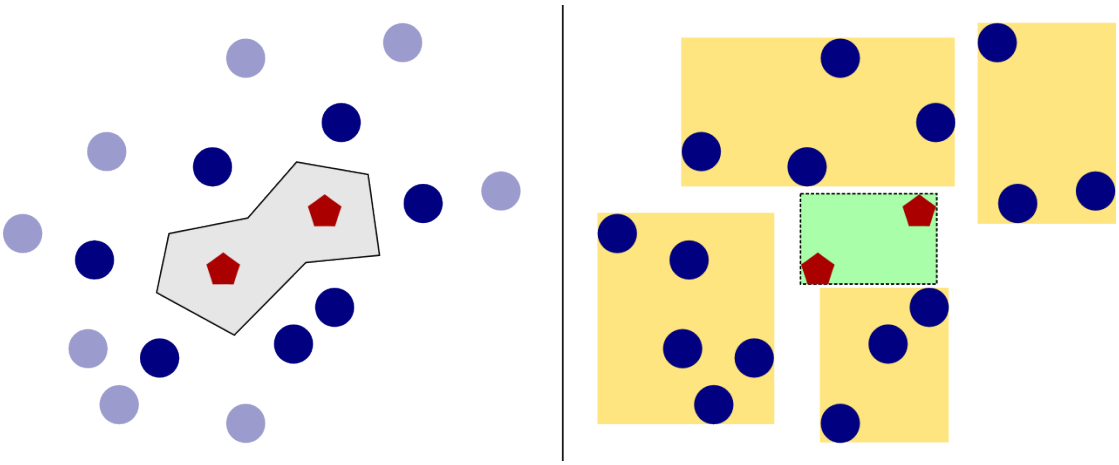
Μάθηση βασισμένη σε παραδείγματα Ορισμένοι αλγόριθμοι δεν δίνουν ένα ολοκληρωμένο μοντέλο το οποίο να απαντά από μόνο του για την κλάση ενός νέου δείγματος. Αντιθέτως, αναβάλλουν την κυρίως διαδικασία της μάθησης για τη στιγμή που το νέο αυτό δεδομένο θα τεθεί υπό έλεγχο. Τέτοιοι αλγόριθμοι δίνουν ως έξοδο χαρακτηριστικά σημεία των δεδομένων. Η ιδέα είναι ότι, όταν πρέπει να ελεγχθεί ένα

νέο δεδομένο, τότε θα αναζητηθεί το κοντινότερο (ή ένα πλήθος από κοντινότερα) σε αυτό γνωστό σημείο και θα του αποδοθεί η ίδια (ή επικρατέστερη) κλάση. Σε αυτή την περίπτωση χρησιμοποιούμε έναν αλγόριθμο εύρεσης του κοντινότερου γείτονα, ή των k κοντινότερων γειτόνων. Δεν είναι πάντοτε απαραίτητο να δίνονται όλα τα γνωστά σημεία. Κάποιες περιοχές όπου συγκεντρώνονται πολλά σημεία της ίδιας κλάσης μπορούν να αναπαρασταθούν από μερικά από τα σημεία της περιοχής. Μια άλλη προσέγγιση είναι να ορίσουμε διαχωριστικές γραμμές ανάμεσα σε περιοχές ομοειδών σημείων, σύμφωνα με τον αλγόριθμο του κοντινότερου γείτονα. Σε αυτή την περίπτωση, χρειάζεται να αναφερθούν μόνο τα σημεία με βάση τα οποία ορίζεται μια τέτοια διαχωριστική γραμμή. Τέλος, μπορούμε να φτιάξουμε ομάδες συγκεκριμένης γεωμετρίας που να περιέχουν μόνο σημεία μίας κλάσης. Π.χ. πολλά ορθογώνια σε έναν χώρο δύο μεταβλητών, τα οποία ισοδυναμούν με μια σύζευξη κανόνων για τις δυο μεταβλητές. Οι δυο ιδέες παρουσιάζονται στο σχήμα 3.2. Τέτοιες περιοχές μπορεί να βρίσκονται και η μια «μέσα» στην άλλη, με σαφή όμως διάκριση. Η εσωτερική περιοχή αποτελεί εξαίρεση για αυτήν που την περιβάλλει. Είναι σημαντικό να μην αλληλεπικαλύπτονται οι περιοχές, ώστε να είναι σίγουρο ότι ένα σημείο εμπίπτει πάντα μόνο σε μία περιοχή. Έτσι αντιμετωπίζεται το πρόβλημα που μπορεί να παρουσιάσουν οι κανόνες: για ένα σημείο να ισχύουν ταυτόχρονα δυο διαφορετικοί κανόνες.

Συσταδοποίηση Μπορούμε να χωρίσουμε τα δεδομένα που διαθέτουμε σε συστάδες. Αυτό μπορεί να γίνει είτε φτιάχνοντας υπερεπίπεδα στον πολυδιάστατο χώρο των μεταβλητών του συστήματος, είτε φτιάχνοντας κλειστά σχήματα (πχ ελλειψοειδή). Μια συστάδα περιέχει σημεία τα οποία βρίσκονται γενικώς πιο κοντά μεταξύ τους από ότι με άλλα σημεία. Οι συστάδες μπορεί να αλληλοκαλύπτονται και κάποια σημεία μπορεί να ανήκουν σε πάνω από μια συστάδες. Τέλος, μπορούμε να δημιουργήσουμε συστάδες μέσα σε άλλες συστάδες (ιεραρχική συσταδοποίηση). Μπορούμε έτσι να έχουμε πολλές μικρές ή λιγότερες αλλά μεγαλύτερες στοιβάδες.



Σχήμα 3.1: Παράδειγμα δέντρου απόφασης, που προκύπτει από την εφαρμογή του αλγορίθμου J48 στα δεδομένα με την ταυτότητα "weather data" του βιβλίου των Witten et al. [1] Στο δέντρο παρουσιάζονται τα στάδια λήψης απόφασης σχετικά με την εύνοια των καιρικών συνθηκών για ένα παιχνίδι (απροσδιόριστο ποιο).



Σχήμα 3.2: Παράδειγμα δομής μοντέλου μάθησης βασισμένης σε παραδείγματα. Στο αριστερό σχήμα διακρίνεται η διαχωριστική γραμμή που προκύπτει με τον αλγόριθμο κοντινότερου γείτονα. Στο δεξιό σχήμα διακρίνονται οι ορθογώνιες περιοχές με σημεία της ίδιας κλάσης. Τα δυο σχήματα είναι προσεγγιστικά.

3.4 Αλγόριθμοι μάθησης

Είδαμε πώς παριστάνεται γενικώς η «γνώση» που εξορύσσεται από δεδομένα. Διακρίναμε διάφορες κατηγορίες. Στον ίδιο διαχωρισμό θα βασιστούμε και για την παρουσίαση των τρόπων με τους οποίους μπορεί να δημιουργηθεί αυτή η γνώση. [1]

3.4.1 Απλοί κανόνες

Πολλά συστήματα στην πράξη υπακούουν σε πολύ απλούς κανόνες. Έτσι, αλγόριθμοι που παράγουν απλούς κανόνες δίνουν μερικές φορές πολύ καλά αποτελέσματα, παρά την απλότητά τους. Παράδειγμα αυτής της κατηγορίας είναι ο αλγόριθμος 1R (1-rule). Ο αλγόριθμος αυτός δημιουργεί ένα δέντρο απόφασης με έναν μόνο κόμβο, ως εξής: Για κάθε μεταβλητή του συστήματος ελέγχονται όλες οι πιθανές τιμές (για αριθμητικές μεταβλητές μπορεί να γίνει διαμερισμός). Σε κάθε τιμή αντιστοιχίζεται η πιο συχνή (για αυτήν την τιμή) κλάση. Στη συνέχεια ελέγχεται κάθε μεταβλητή ως προς την επιτυχία που δίνει, αν χρησιμοποιηθεί ως μοναδικός κόμβος στο δέντρο και επιλέγεται η μεταβλητή που δίνει την υψηλότερη επιτυχία (βλ. τον αλγόριθμο 1).

Αλγόριθμος 1 Αλγόριθμος 1R [1]

για κάθε μεταβλητή **επανάλαβε**
  για κάθε τιμή της μεταβλητής **επανάλαβε**
    Δημιούργησε έναν κανόνα ως εξής:
    μέτρησε πόσες φορές εμφανίζεται κάθε κλάση
    βρες την πιο συχνά εμφανιζόμενη κλάση
    αντιστοίχισε αυτήν την κλάση σε αυτήν την τιμή της μεταβλητής
  Υπολόγισε το σφάλμα των κανόνων
Επέλεξε τους κανόνες με το μικρότερο σφάλμα

3.4.2 Στατιστική μοντελοποίηση

Αντί να χρησιμοποιήσουμε μόνο μια μεταβλητή του συστήματος, μπορούμε να τις χρησιμοποιήσουμε όλες και να θεωρήσουμε ότι η κάθε μία συνεισφέρει εξίσου και ανεξάρτητα στην τελική απόφαση. Προς χάριν της παρουσίασης, θεωρούμε το (αυθαίρετο) σύνολο δεδομένων που περιγράφεται στον πίνακα 3.1. Στον πίνακα 3.2 υπολογίζουμε την κατανομή των τιμών κάθε μεταβλητής στις δύο κλάσεις. Έστω τώρα ότι εξετάζουμε ένα νέο σημείο, το οποίο έχει τις εξής τιμές: (A3,B1,C2,D1). Τότε, οι πιθανότητες για κάθε κλάση υπολογίζονται ως εξής:

$$\text{Πιθανότητα κλάσης P} = (2/3 \times 1/3 \times 1/2 \times 3/4) \times 3/6 = 0.0417 \tag{3.1}$$

$$\text{Πιθανότητα κλάσης N} = (1/3 \times 2/3 \times 1/2 \times 1/4) \times 3/6 = 0.0139 \tag{3.2}$$

Η πιθανότητα της κλάσης P είναι υψηλότερη αυτής της κλάσης N, έτσι το μοντέλο αντιστοιχίζει την κλάση P στο υπό εξέταση δείγμα. Αν κανονικοποιήσουμε αυτές τις τιμές έτσι ώστε να έχουν άθροισμα ίσο με τη μονάδα, είναι:

$$\text{Πιθανότητα κλάσης P} = \frac{0.0417}{0.0417 + 0.0139} = 75\% \tag{3.3}$$

$$\text{Πιθανότητα κλάσης N} = \frac{0.0139}{0.0417 + 0.0139} = 25\% \tag{3.4}$$

Η μέθοδος αυτή βασίζεται στο θεώρημα (ή τύπο) του Bayes, το οποίο εκφράζει την πιθανότητα μιας υπόθεσης H δεδομένου ενός γεγονότος E:

$$\Pr[H|E] = \frac{\Pr[E|H] \cdot \Pr[H]}{\Pr[E]} \tag{3.5}$$

Στην περίπτωσή μας έχουμε τέσσερα γεγονότα, θεωρούμενα ανεξάρτητα, τα οποία θα ονομάσουμε E_1 έως E_4 και αντιστοιχούν στην απόδοση μιας συγκεκριμένης τιμής για τις τέσσερις μεταβλητές του υποθετικού συστήματος. Αφού τα γεγονότα θεωρούνται ανεξάρτητα, μπορούμε να πολλαπλασιάσουμε τις πιθανότητές τους. Η υπόθεση που εξετάζουμε είναι να ανήκει το δείγμα μας στην κλάση P. Έτσι:

$$\Pr[\text{P}|E] = \frac{\Pr[E_1|\text{P}] \times \Pr[E_2|\text{P}] \times \Pr[E_3|\text{P}] \times \Pr[E_4|\text{P}] \times \Pr[\text{P}]}{\Pr[E]} \tag{3.6}$$

Λόγω της «αφελούς» υπόθεσης ανεξαρτησίας των δεδομένων, η μέθοδος ονομάζεται Naïve Bayes.

Όπως εύκολα παρατηρούμε, θα υπάρξει πρόβλημα εάν το υπό εξέταση δείγμα περιλαμβάνει π.χ. την τιμή A1, για την οποία η πιθανότητα εμφάνισης της κλάσης P είναι μηδενική. Στον πίνακα 3.2 βλέπουμε ότι η μεταβλητή A, η οποία έχει 3 πιθανές τιμές, έχει κατανομή (A1,A2,A3)=(0/3, 1/3, 2/3). Μπορούμε να διορθώσουμε αυτό το πρόβλημα προσθέτοντας έναν αριθμό στους αριθμητές και τον τριπλάσιό του (στη συγκεκριμένη περίπτωση, λόγω τριών τιμών) στους παρονομαστές. Εάν προσθέσουμε τη μονάδα (Laplace estimator) έχουμε: (A1,A2,A3)=(1/6, 2/6, 3/6).

Άγνωστες τιμές δεν αποτελούν πρόβλημα για τη μέθοδο. Αν έχουμε αριθμητικές παραμέτρους, θεωρούμε ότι ακολουθούν κάποια γνωστή (συνήθως κανονική) κατανομή. Στον αντίστοιχο του πίνακα 3.2, στις αριθμητικές μεταβλητές, παραθέτουμε απλώς τις τιμές. Αντί για τη συχνότητα εμφάνισης κάθε τιμής, χρησιμοποιούμε τον μέσο όρο και την τυπική απόκλιση της μεταβλητής. Η συνέχεια αναλύεται στη βιβλιογραφία. [1]

3.4.3 Δέντρα αποφάσεων

Ένα δέντρο κατασκευάζεται επιλέγοντας κατά σειρά κατάλληλους κόμβους ελέγχου για κάθε κλάδο. Ξεκινώντας από τον κορυφαίο κόμβο, δημιουργούμε τόσους κλάδους όσες και οι πιθανές τιμές της αντίστοιχης κατηγορικής μεταβλητής, ή όσα τα διαστήματα που επιθυμούμε, για αριθμητικές μεταβλητές. Κάθε κλάδος διαχωρίζεται περαιτέρω αν χρειαστεί. Το κρίσιμο ερώτημα είναι ποιες μεταβλητές θα επιλέξουμε να τοποθετήσουμε σε κάθε κόμβο

Πίνακας 3.1: Αυθαίρετα δεδομένα για την παρουσίαση των στατιστικών μοντέλων.

Μεταβλητή A	**Μεταβλητή B**	**Μεταβλητή C**	**Μεταβλητή D**	**Κλάση**
A1	B1	C1	D1	N
A1	B1	C1	D2	N
A2	B1	C1	D1	P
A3	B2	C1	D1	P
A3	B3	C2	D1	P
A3	B3	C2	D2	N

Πίνακας 3.2: Κατανομή τιμών κάθε μεταβλητής στις κλάσεις των αυθαίρετων δεδομένων. Και οι δύο κλάσεις εμφανίζονται με συχνότητα 3/6.

Μετ.A	P	N	Μετ.B	P	N	Μετ.C	P	N	Μετ.D	P	N
A1	0	2	B1	1	2	C1	2	2	D1	3	1
A2	1	0	B2	1	0	C2	1	1	D2	0	2
A3	2	1	B3	1	1						
A1	0/2	2/2	B1	1/3	2/3	C1	2/4	2/4	D1	3/4	1/4
A2	1/1	0/1	B2	1/1	0/1	C2	1/2	1/2	D2	0/2	2/2
A3	2/3	1/3	B3	1/2	1/2						

ελέγχου. Η τακτική που ακολουθούμε είναι να τοποθετούμε στον εκάστοτε κόμβο τη μεταβλητή η οποία θα προσφέρει το μεγαλύτερο «κέρδος πληροφορίας». Τι σημαίνει όμως αυτό;

Η «πληροφορία» είναι μια ποσότητα η οποία:

1. Έχει μηδενική τιμή όταν εμφανίζεται μόνο μία κλάση.
2. Μεγιστοποιείται όταν υπάρχει ομοιόμορφη κατανομή κλάσεων.
3. Μπορεί να υπολογιστεί σε διαδοχικά στάδια, όπως θα δούμε αμέσως παρακάτω.

Έστω ότι έχουμε σε έναν κλάδο ενός δέντρου π.χ. 9 δείγματα και έστω ότι υπάρχουν 3 διαφορετικές κλάσεις. Έστω ότι 2 δείγματα ανήκουν στην πρώτη κλάση, 3 στην δεύτερη, 4 στην τρίτη, δηλαδή η κατανομή είναι $[2, 3, 4]$. Η πληροφορία που υπολογίζεται με βάση αυτήν την κατανομή συμβολίζεται $\text{info}([2, 3, 4])$. Ωστόσο, μπορεί να υπολογιστεί πχ και ως εξής:

$$\text{info}([2, 3, 4]) = \text{info}([2, 7]) + \frac{7}{9} \cdot \text{info}([3, 4]) \tag{3.7}$$

δηλαδή μπορούμε να υπολογίσουμε ξεχωριστά την πληροφορία της κατανομής «ανήκει στην πρώτη κλάση ή ανήκει σε κάποια άλλη κλάση» και ξεχωριστά την πληροφορία της κατανομής «ανήκει στην δεύτερη κλάση ή ανήκει στην τρίτη κλάση» και να τις συνδυάσουμε ώστε να υπολογίσουμε την πληροφορία της πλήρους κατανομής. Η πληροφορία μπορεί να αντιστοιχηθεί σε εντροπία σε αυτήν την περίπτωση ως εξής:

$$\text{info}([2,3,4]) = \text{entropy}\left(\frac{2}{9}, \frac{3}{9}, \frac{4}{9}\right) \tag{3.8}$$

όπου π.χ. 2/9 το κλάσμα των δειγμάτων που ανήκουν στην πρώτη κλάση, ως προς το σύνολο. Χρησιμοποιώντας αυτά τα κλάσματα $(p_1, p_2, \ldots, p_n)$ υπολογίζουμε την εντροπία ως εξής:

$$\text{entropy}(p_1, p_2, \ldots, p_n) = -p_1 \cdot \log_2 p_1 - p_2 \cdot \log_2 p_2 \ldots - p_n \cdot \log_2 p_n \tag{3.9}$$

Όταν χρησιμοποιούνται λογάριθμοι με βάση 2 (όπως στην εξίσωση 3.9), τότε η εντροπία προκύπτει σε "bits". Σε όρους εντροπίας, ο υπολογισμός σε δύο στάδια που παρουσιάστηκε παραπάνω μπορεί να γραφεί ως εξής:

$$\text{entropy}(p, q, r) = \text{entropy}(p, q + r) \; + \; (q + r) \cdot \text{entropy}\left(\frac{q}{q + r}, \frac{r}{q + r}\right) \tag{3.10}$$

όπου $p + q + r = 1$.

Με αυτόν τον τρόπο υπολογίζουμε την εντροπία (και στη συνέχεια την πληροφορία) που δίνει κάθε κλάδος ενός κόμβου. Στη συνέχεια λαμβάνουμε τον σταθμισμένο μέσο όρο για όλους τους κλάδους ενός κόμβου. Την ίδια διαδικασία επαναλαμβάνουμε για κάθε μεταβλητή που διεκδικεί τον συγκεκριμένο κόμβο, καθώς και για τον κλάδο που οδηγεί σε αυτόν. Στη συνέχεια υπολογίζουμε το κέρδος πληροφορίας για κάθε μεταβλητή, αφαιρώντας την πληροφορία που αντιστοιχεί στην μεταβλητή από την πληροφορία που αντιστοιχεί στον κλάδο στον οποίο θα συνδεθεί. Η μεταβλητή που επιλέγεται για τον κόμβο είναι αυτή που προσφέρει το μεγαλύτερο κέρδος πληροφορίας.

3.4.4 Κανόνες κάλυψης

Τα δέντρα αποφάσεων μπορούν να μετατραπούν άμεσα στα αντίστοιχα σύνολα κανόνων. Αυτή η προσέγγιση ωστόσο δεν παράγει πάντα καλούς κανόνες. Για την κατασκευή ενός δέντρου, τα δεδομένα χωρίζονται σύμφωνα με τις μεταβλητές σε μικρότερες κλάσεις. Οι κανόνες κάλυψης κατασκευάζονται ξεκινώντας από τις επιμέρους κλάσεις, προσπαθώντας να καλύψουν όλα τα στοιχεία μιας κλάσης, χωρίς να καλύπτουν στοιχεία άλλων κλάσεων. Το τελευταίο στοιχείο δεν είναι και τόσο ακριβές. Στην πράξη, επιτρέπονται μικρές «προσμίξεις» από άλλες κλάσεις, έτσι ώστε να διατηρηθεί απλό το σύνολο των κανόνων και να αποφευχθεί η υπερπροσαρμογή (overfitting). Ωστόσο, αυτό εξαρτάται και από το πόσο καλά θέλουμε να περιγράψουμε μια συγκεκριμένη κλάση. Συνεχίζοντας τη σύγκριση με τα δέντρα, παρότι μοιάζουν με αυτά, οι κανόνες συχνά οδηγούν σε πιο σύντομες περιγραφές. Επίσης, σε προβλήματα με πολλές κλάσεις, ενώ τα δέντρα εξετάζουν συνεχώς όλες τις κλάσεις, κάθε κανόνας ασχολείται με μόνο μία κλάση, αγνοώντας τις υπόλοιπες.

Ένας κανόνας κάλυψης έχει τη μορφή:

```
Εάν συνθήκη1 [ΚΑΙ συνθήκη2 ΚΑΙ ... ΚΑΙ συνθήκηΝ] τότε κλάση
```

Για παράδειγμα, έστω ότι κατασκευάζουμε ένα μοντέλο που ξεχωρίζει είδη φρούτων. Τότε, ένας κανόνας για να περιγράφει το είδος «καρπούζι» θα μπορούσε να είναι:

```
Εάν βάρος > 1kg ΚΑΙ χρώμα = πράσινο τότε είδος = καρπούζι
```

Ένας τέτοιος κανόνας μπορεί να κατασκευαστεί ως εξής: Αρχικά ψάχνουμε μια συνθήκη που να καλύπτει όσο το δυνατόν περισσότερα δείγματα της εξεταζόμενης κλάσης. Δηλαδή ψάχνουμε έναν κανόνα της μορφής: `Εάν ? τότε είδος = καρπούζι`. Ελέγχουμε όλες τις πιθανές συνθήκες ως προς την επιτυχία τους. Π.χ. τέτοιες συνθήκες θα μπορούσαν να είναι `γεύση = γλυκιά` ή `βάρος > 1kg`. Φανταζόμαστε ότι, στο υποθετικό αυτό σύνολο, ο πρώτος κανόνας μπορεί να ισχύει π.χ. σε 16 δείγματα, από τα οποία στο είδος «καρπούζι» να ανήκουν μόνο 4 από αυτά. Αντιθέτως, φανταζόμαστε ότι ο δεύτερος κανόνας θα ισχύει π.χ. σε 7 δείγματα, από τα οποία στο είδος «καρπούζι» θα ανήκουν τα 5. Ο δεύτερος κανόνας λοιπόν περιγράφει καλύτερα το σύνολο, όμως και πάλι υπάρχουν κάποια «ξένα» δείγματα που εσφαλμένα έχουν αντιστοιχηθεί σε αυτήν την κλάση. Πριν προχωρήσουμε στο επόμενο βήμα, διαγράφουμε τα σωστά κατηγοριοποιημένα δείγματα από τα δεδομένα. Αναζητούμε στη συνέχεια μια δεύτερη συνθήκη, η οποία θα δίνει από μόνη της την υψηλότερη επιτυχία σε σχέση με τις υπόλοιπες υποψήφιες. Έστω ότι η συνθήκη αυτή υποδεικνύει το πράσινο χρώμα και ότι ο συνολικός κανόνας είναι τέλειος, δηλαδή δεν ισχύει για κανένα άλλο είδος-κλάση.

Αυτή είναι η ενδεικτική πορεία. Μερικές σημειώσεις: εάν δεν έχουν κατηγοριοποιηθεί κάποια δείγματα της κλάσης, φτιάχνουμε επιπλέον κανόνες. Εάν έχουμε να επιλέξουμε ανάμεσα σε δυο συνθήκες που δίνουν την ίδια επιτυχία, επιλέγουμε τυχαία. Εάν η επιτυχία για δυο κανόνες είναι π.χ. "1/2" και "3/6", τότε επιλέγουμε τον δεύτερο κανόνα, λόγω μεγαλύτερης καλυπτικότητας (3 δείγματα αντί για 1). Τέλος, εάν το επιθυμούμε, μπορούμε να ορίσουμε μια κλάση ως "προεπιλεγμένη". Για αυτήν τότε δεν χρειάζεται να κατασκευάσουμε κανόνα αυτής της μορφής. Εάν ένα δείγμα δεν βρεθεί στα όρια εφαρμογής κάποιου κανόνα, τότε θα αντιστοιχηθεί στην προεπιλεγμένη κλάση. Η μέθοδος που περιγράψαμε καλείται «μέθοδος PRISM» (βλ. αλγόριθμο 2).

Αλγόριθμος 2 Μέθοδος PRISM για κατασκευή κανόνων κάλυψης [1]

για κάθε κλάση C **επανάλαβε**
  Αρχικοποίησε το σύνολο E ώστε να περιλαμβάνει όλα τα δεδομένα.
  όσο το E περιέχει δείγματα της κλάσης C **επανάλαβε**
    Δημιούργησε έναν κανόνα R με κενό αριστερό μέλος, ο οποίος να προβλέπει την κλάση C.
    όσο ο R είναι τέλειος (ή δεν υπάρχουν άλλες μεταβλητές) **επανάλαβε**
      για κάθε μετ. A που δεν αναφέρεται στον R, και κάθε τιμή v **επανάλαβε**
        Εξέτασε την προσθήκη της $A = v$ στο αριστερό μέλος του R.
        Επέλεξε τα A και v ώστε να μεγιστοποιούν την ακρίβεια *p/total*.
        (σε τυχόν διλήμματα διάλεξε τη συνθήκη με το μεγαλύτερο p)
      Πρόσθεσε τη συνθήκη $A = v$ στον R.
    Αφαίρεσε από το E τα δείγματα που καλύπτονται από τον R.

Η μέθοδος μπορεί να βελτιωθεί ελέγχοντας, για κάθε κανόνα, αν μπορεί να δημιουργηθεί ένας καλύτερος αφαιρώντας κάποιες συνθήκες. Πράγματι, αυτό σε ορισμένες περιπτώσεις οδηγεί σε απλούστερους κανόνες με ταυτόχρονα υψηλότερη ακρίβεια. Η διαδικασία αυτή

ονομάζεται incremental reduced-error pruning. Αφού ολοκληρωθεί η δημιουργία ενός συνόλου κανόνων για μία κλάση, αυτό μπορεί να βελτιστοποιηθεί συνολικά. Για κάθε κανόνα παράγονται δύο νέοι κανόνες: ένας συνθετότερος (με προσθήκη συνθηκών στον αρχικό) και ένας νέος, ο οποίος παράγεται από το μηδέν. Στη συνέχεια, ο αρχικός κανόνας αντικαθίσταται από αυτόν (εκ των δύο) που έχει το μικρότερο «μήκος περιγραφής» (descriptive length). Η διαδικασία αυτή ονομάζεται RIPPER (repeated incremental pruning to produce error reduction) και υλοποιείται στη WEKA με το όνομα `JRip`.

3.4.5 Γραμμικά μοντέλα

Τα γραμμικά μοντέλα, παρά την «απλότητα» της γραμμικότητάς τους, πολλές φορές σε πραγματικά προβλήματα αποδίδουν αρκετά ικανοποιητικά. Με τέτοια μοντέλα μπορούμε να κάνουμε είτε παλινδρόμηση (regression) είτε ταξινόμηση (classification). Η παλινδρόμηση είναι ήδη αρκετά γνωστή και δεν θα ασχοληθούμε περισσότερο από το να αναφέρουμε πως ζητούμενος είναι ο προσδιορισμός των βαρών $w_1, w_2, \ldots, w_k$ στη σχέση:

$$x = w_0 + w_1 \cdot \alpha_1 + w_2 \cdot \alpha_2 + \ldots + w_k \cdot \alpha_k \tag{3.11}$$

όπου x είναι η κλάση και $\alpha_1, \alpha_2, \ldots, \alpha_k$ οι μεταβλητές του συστήματος. Ο προσδιορισμός αυτός μπορεί να γίνει ελαχιστοποιώντας την απόσταση μεταξύ προβλεπόμενων και πραγματικών κλάσεων.

Μπορούμε να επεκτείνουμε την ίδια ιδέα στην ταξινόμηση, χωρίζοντας τα σημεία των δεδομένων με μια ευθεία (ή επίπεδο, ή υπερεπίπεδο). Γενικώς, κάνουμε γραμμική παλινδρόμηση για κάθε κλάση, με βάση μια «συνάρτηση συμμετοχής» (membership function). Σε πρώτη προσέγγιση, η συνάρτηση αυτή μπορεί να δίνει τιμή 1 αν ένα δείγμα ανήκει στην υπό εξέταση κλάση, ή τιμή 0 αλλιώς. Σε αυτήν την περίπτωση, δημιουργούμε μια γραμμική σχέση για κάθε κλάση. Όταν εξετάζουμε ένα νέο δείγμα, υπολογίζουμε την τιμή κάθε σχέσης για αυτό και το ταξινομούμε στην κλάση για την οποία υπολογίζεται υψηλότερη τιμή. Αυτή η μέθοδος λέγεται «γραμμική παλινδρόμηση πολλαπλών αποκρίσεων» (multiresponse linear regression) και συχνά αποδίδει ικανοποιητικά.

Ο ορισμός της συνάρτησης συμμετοχής με αυτόν τον τρόπο είναι αρκετά απλοϊκός και παρουσιάζει κάποια προβλήματα. Οι γραμμικές σχέσεις που προκύπτουν μπορούν να παράγουν τιμές συμμετοχής έξω από το διάστημα $[0, 1]$, οπότε δεν προκύπτουν καλώς ορισμένες πιθανότητες. Επίσης, η παλινδρόμηση μερικών ελαχίστων τετραγώνων προϋποθέτει ότι τα σφάλματα είναι στατιστικά ανεξάρτητα και ότι ανήκουν σε κανονική κατανομή με την ίδια τυπική απόκλιση. Στην περίπτωσή μας ωστόσο αυτό δεν μπορεί να εφαρμοστεί, αφού το σφάλμα για κάθε σημείο θα είναι ή 0 ή 1. [1]

Για να αντιμετωπιστούν τέτοια προβλήματα, χρησιμοποιείται μια παραλλαγή της μεθόδου, η λογιστική παλινδρόμηση (Logistic Regression). Η συνάρτηση συμμετοχής που χρησιμοποιεί λαμβάνει υπ' όψιν της ότι όσο πιο μακριά από τη διαχωριστική γραμμή βρίσκεται ένα σημείο, τόσο υψηλότερη πιθανότητα έχει να ανήκει στην αντίστοιχη κλάση. Η πιθανότητα αυτή αναπαρίσταται από μια λογιστική συνάρτηση (με τη γνώριμη σιγμοειδή καμπύλη). Θεωρούμε ότι κάθε σημείο λαμβάνει μια «ετικέτα» l, η οποία σχετίζεται με την κλάση του. Η l μπορεί να πάρει τιμές $+1$ ή -1. Η πιθανότητα για $l = +1$ μπορεί να εκφραστεί

ως εξής:

$$P[l = +1 \,|\, \mathbf{x}] = \frac{1}{1 + \exp(-(ax + by + c))} \tag{3.12}$$

Θεωρώντας ότι τα σημεία είναι ανεξάρτητα, μπορούμε να γράψουμε:

$$\prod_{i=1}^{N} P[l_i \,|\, \mathbf{x}_i] = \prod_{i=1}^{N} \frac{1}{1 + \exp(-l_i(ax_i + by_i + c))} \tag{3.13}$$

Η σχέση 3.13 δίνει την «πιθανότητα των δεδομένων» (data likelihood). Επιθυμούμε να προσεγγίζει κατά το δυνατόν τη μονάδα. Έτσι, για να βρούμε τις παραμέτρους a,b,c της γραμμής, επιλύουμε το αντίστοιχο πρόβλημα μεγιστοποίησης. Συνήθως, δεν χρησιμοποιούμε τη σχέση 3.13, αλλά τη μετασχηματισμένη σχέση 3.14:

$$\log\Big(\prod_{i=1}^{N} P[l_i \,|\, \mathbf{x}_i]\Big) = \sum_{i=1}^{N} -\log(1 + \exp(-l_i(ax_i + by_i + c))) \tag{3.14}$$

ή, πολλαπλασιάζοντας με -1 ώστε να μετατρέψουμε το πρόβλημα μεγιστοποίησης σε πρόβλημα ελαχιστοποίησης, τη σχέση 3.15, όπου L η «συνάρτηση απώλειας» (loss function) ή «συνάρτηση κόστους» (cost function):

$$L = \sum_{i=1}^{N} \log(1 + \exp(-l_i(ax_i + by_i + c))) \tag{3.15}$$

Τελικά, ο προσδιορισμός της διαχωριστικής ευθείας (ή επιπέδου ή υπερεπιπέδου) μετασχηματίζεται στην επίλυση του προβλήματος ελαχιστοποίησης που περιγράφεται από τη σχέση 3.15, με μεταβλητές τις παραμέτρους της καμπύλης. [20]

3.4.6 Μάθηση βασισμένη σε παραδείγματα

Στη μάθηση που βασίζεται σε παραδείγματα, είδαμε ότι η κυρίως διαδικασία της «μάθησης» γίνεται τη στιγμή που χρειάζεται να αποφανθούμε για την κλάση ενός νέου δείγματος. Η γενική μέθοδος υπαγορεύει να βρούμε το κοντινότερο για το εξεταζόμενο δείγμα, σημείο των δεδομένων και να του αποδώσουμε την ίδια κλάση. Η απόσταση ορίζεται είτε ως η συνηθισμένη ευκλείδεια απόσταση, είτε ως η απόσταση οικοδομικών τετραγώνων (city-block) είτε με άλλους, καταλληλότερους κατά περίπτωση, τρόπους. Σημαντικό σε αυτή τη μέθοδο είναι οι διάφορες μεταβλητές των δεδομένων να έχουν κανονικοποιηθεί. Διαφορετικά, μεταβλητές με γενικώς υψηλότερες τιμές θα συνεισφέρουν περισσότερο στην απόσταση, χωρίς όμως αυτό να σχετίζεται με τη σημαντικότητά τους. Έτσι, κάθε μεταβλητή v_i, με μέγιστη και ελάχιστη τιμή στα διαθέσιμα δεδομένα $\max v_i$ και $\min v_i$ αντιστοιχίζεται σε μια κανονικοποιημένη a_i, η οποία παίρνει τιμές στο διάστημα $[0, 1]$, σύμφωνα με τη σχέση:

$$a_i = \frac{v_i - \min v_i}{\max v_i - \min v_i} \tag{3.16}$$

Εάν ελέγξουμε όλα τα διαθέσιμα σημεία για να βρούμε το κοντινότερο, τότε ο χρόνος που απαιτείται εξαρτάται γραμμικά από το πλήθος των δεδομένων (πολυπλοκότητα $O(N)$). Η διαδικασία μπορεί να συντομευθεί σημαντικά, δομώντας κατάλληλα τα δεδομένα. Για ένα σύστημα k διαστάσεων μπορούμε να φτιάξουμε ένα kD-tree ως εξής: Διαλέγουμε τον άξονα κατά τον οποίο υπάρχει η μεγαλύτερη διασπορά των δεδομένων. Σε αυτόν τον άξονα, επιλέγουμε ένα σημείο των δεδομένων που να βρίσκεται στη διάμεσο ή κοντά στη μέση τιμή. Δημιουργούμε ένα διαχωριστικό υπερεπίπεδο κάθετο σε αυτόν τον άξονα, στο οποίο ανήκει το επιλεγμένο σημείο. Έτσι, έχουμε έναν κόμβο, από τον οποίο εξέρχονται ένας κλάδος για κάθε υποπεριοχή. Ο κόμβος συμβολίζεται από το σημείο και τον άξονα ως προς τον οποίο γίνεται ο διαχωρισμός. Στη συνέχεια, κάθε υποπεριοχή χωρίζεται περαιτέρω, κατά προτίμηση έτσι ώστε να σχηματίζονται υπερτετράγωνα. Επισημαίνεται ότι τα δέντρα που κατασκευάζονται σε αυτήν την περίπτωση δεν εμπεριέχουν την πληροφορία της κλάσης για κάθε σημείο και έτσι είναι διαφορετικά από τα δέντρα αποφάσεων που αναλύσαμε προηγουμένως.

Έχοντας δομήσει έτσι τα δεδομένα, η αναζήτηση του κοντινότερου γείτονα περιορίζεται σε κάποιες μόνο υποπεριοχές. Αρχικά, εντοπίζουμε την υποπεριοχή στην οποία ανήκει το σημείο που εξετάζουμε, καθώς και ένα δεδομένο σημείο που βρίσκεται σε αυτήν την περιοχή. Αν υπάρχει κοντινότερος γείτονας, τότε αυτός θα βρίσκεται στο εσωτερικό ενός κύκλου (για δυο διαστάσεις) με κέντρο το εξεταζόμενο σημείο και ακτίνα την απόστασή του από το δεδομένο σημείο που βρίσκεται στην ίδια περιοχή. Ο κύκλος θα τέμνει κάποιες γειτονικές περιοχές. Οι περιοχές αυτές μπορούν να αναζητηθούν εύκολα ακολουθώντας το δέντρο από την περιοχή που ανήκει το εξεταζόμενο σημείο, προς τη ρίζα του. Έτσι, χρειάζεται να ελέγξουμε μόνο τις περιοχές η διαχωριστική γραμμή των οποίων τέμνει τον κύκλο, ενώ μπορούμε να αποκλείσουμε ολόκληρους κλάδους του δέντρου. Αυτό είναι αρκετά εύκολο από τη στιγμή που οι διαχωριστικές γραμμές είναι κάθετες στους άξονες. Η πολυπλοκότητα αυτού του συστήματος είναι $O(\log_2 N)$, αν το δέντρο είναι ισορροπημένο (well balanced). Εκτός από τη μείωση του υπολογιστικού χρόνου, σημαντικό πλεονέκτημα αυτής της δομής δεδομένων είναι ότι μπορούν εύκολα να προστεθούν νέα δεδομένα στον δειγματοχώρο: εντοπίζουμε την υποπεριοχή στην οποία ανήκει το σημείο, το τοποθετούμε σε αυτήν και, αν δεν είναι το μοναδικό, ορίζουμε μια νέα διαχωριστική γραμμή.

Ένα πρόβλημα που παρατηρείται σε αυτή τη μέθοδο είναι η περιοχή έρευνας που ορίζεται (δηλαδή ο κύκλος που αναφέραμε στην περίπτωση 2D προβλημάτων) να επικαλύπτεται με μεγάλο πλήθος περιοχών. Στις δύο διαστάσεις, μπορούμε να φανταστούμε τον κύκλο να τέμνει πολλές γειτονικές περιοχές αλλά να εμπεριέχει μόνο τις γωνίες τους. Για να αντιμετωπιστεί αυτό το πρόβλημα, το οποίο αυξάνει τον υπολογιστικό χρόνο, χρησιμοποιείται μια παραλλαγή της μεθόδου kD-tree, η μέθοδος "ball tree". Σύμφωνα με αυτήν, δεν ορίζονται ορθογώνιες περιοχές, αλλά κυκλικές περιοχές (υπερσφαίρες) που διαχωρίζουν σταδιακά το σύνολο των δεδομένων σε γειτονικά. [1]

3.4.7 Συσταδοποίηση

Η συσταδοποίηση στοχεύει στο διαχωρισμό των δεδομένων σε ομάδες, χωρίς να γνωρίζουμε την κλάση τους. Η ευρύτερη κατηγορία μεθόδων που δεν χρησιμοποιούν την κλάση για το διαχωρισμό ονομάζεται μη καθοδηγούμενη μάθηση (unsupervised learning). Ο χωρισμός

σε ομάδες-συστάδες (clusters) μπορεί να γίνει με διάφορους τρόπους. Ένας τρόπος είναι να ορίσουμε εξ' αρχής ένα πλήθος συστάδων και να αντιστοιχίσουμε κάθε δείγμα στην κοντινότερή του συστάδα. Ένας άλλος τρόπος είναι να ξεκινήσουμε χωρίζοντας δύο ομάδες και στη συνέχεια να τις αναπτύξουμε περαιτέρω (ιεραρχική συσταδοποίηση).

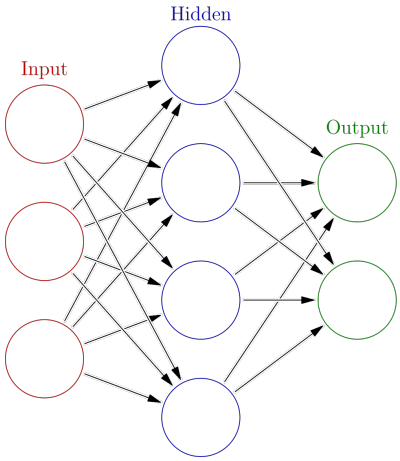
Η κλασσική μέθοδος συσταδοποίησης λέγεται "k-means". Σύμφωνα με αυτήν, ορίζουμε εξαρχής ότι θα χωρίσουμε k διαφορετικές ομάδες. Στη συνέχεια, επιλέγουμε k αρχικά σημεία (seeds) τα οποία θεωρούμε ως κέντρα των συστάδων. Αντιστοιχίζουμε κάθε δείγμα στη συστάδα που ορίζεται από το κοντινότερο στο δείγμα κέντρο. Έπειτα, υπολογίζουμε το κεντροειδές ή το μέσο (mean) των σημείων της ίδια συστάδας και επαναλαμβάνουμε τη διαδικασία, χρησιμοποιώντας τώρα ως κέντρο κάθε συστάδας το κεντροειδές της. Επαναλαμβάνουμε έως ότου να επιλέγονται τα ίδια σημεία σε κάθε συστάδα ή αλλιώς να μην μετακινούνται άλλο τα κέντρα των συστάδων.

Η λύση που υπολογίζεται με αυτή τη μέθοδο δεν είναι το παγκόσμιο άριστο, αλλά μονάχα ένα τοπικό. Αλλάζοντας ελαφρώς την αρχική τοποθέτηση των σημείων μπορεί να λάβουμε αρκετά διαφορετική τελική λύση. Έτσι, συχνά επαναλαμβάνουμε τη διαδικασία αρκετές φορές, με διαφορετικά αρχικά σημεία, και επιλέγουμε την καλύτερη λύση. Για να κάνουμε μια πιο εύστοχη επιλογή των αρχικών σημείων, μπορούμε να χρησιμοποιήσουμε την παραλλαγή "k-means++". Σύμφωνα με αυτήν, το πρώτο αρχικό σημείο επιλέγεται τυχαία. Το δεύτερο επιλέγεται με πιθανότητα ανάλογη της απόστασής του από το πρώτο. Το τρίτο με πιθανότητα ανάλογη της απόστασής του από τα άλλα δύο και ούτω καθ' εξής. Έτσι, μπορούμε να έχουμε και ταχύτερη σύγκλιση. Για μείωση του απαιτούμενου χρόνου μπορούμε και εδώ να εφαρμόσουμε τις μεθόδους kD-tree και ball tree που είδαμε στη «μάθηση που βασίζεται σε παραδείγματα». Τέλος, βασικό ερώτημα είναι το πόσες συστάδες πρέπει να ορίσουμε σε ένα συγκεκριμένο πρόβλημα. Για να αποφανθούμε, μπορούμε να ξεκινήσουμε με δύο μεγάλες συστάδες ($k = 2$) δημιουργώντας περισσότερες και κρίνοντας αν κάθε συστάδα αν αξίζει να αναπτυχθεί περαιτέρω. [1]

3.4.8 Νευρωνικά Δίκτυα

Τα τεχνητά νευρωνικά δίκτυα (Artificial Neural Networks) είναι εμπνευσμένα από τη δομή των νευρώνων βιολογικών οργανισμών. Ένας νευρώνας λαμβάνει ένα σήμα εισόδου και αποκρίνεται σε αυτό δίνοντας ένα σήμα εξόδου, πχ πολλαπλασιάζοντας το σήμα εισόδου με μια σταθερά. Πολλοί νευρώνες συνδυάζονται και μαζί δίνουν ένα σήμα εξόδου το οποίο εξαρτάται από όλες τις εισόδους. Μια απλή αλλά πολύ διαδεδομένη μορφή νευρωνικού δικτύου, της κατηγορίας feedforward (δηλαδή δεν παρουσιάζει κλειστούς βρόχους στη δομή του) είναι το Multilayer Perceptron. Στο σχήμα 3.3 φαίνεται η γενική δομή ενός τέτοιου δικτύου με ένα κρυφό επίπεδο. Για να κατασκευαστεί ένα τέτοιο δίκτυο πρέπει να καθοριστούν η δομή και οι συντελεστές-βάρη των νευρώνων.

Ένας απλός νευρώνας λειτουργεί όπως ένας γραμμικός classifier, άρα δημιουργεί ένα διαχωριστικό υπερεπίπεδο. Πολλαπλασιάζει την είσοδο με έναν συντελεστή-βάρος και προσθέτει μια σταθερά, η οποία καλείται bias. Η είσοδος μπορεί να είναι τιμές πολλών παραμέτρων, οπότε ο πολλαπλασιασμός γίνεται με ένα διάνυσμα βαρών. Το διάνυσμα αυτό προσδιορίζεται ως εξής: σε κάθε βήμα ελέγχονται όλα τα σημεία των δεδομένων. Όταν βρεθεί ένα σημείο που να μην ταξινομείται σωστά, τότε όλα τα βάρη ενημερώνονται ώστε



Σχήμα 3.3: Δομή ενός feedforward νευρωνικού δικτύου, όπως το Multilayer Perceptron. *(Πηγή: Wikimedia Commons - άδεια CC BY-SA)*

να ταξινομείται σωστά αυτό το σημείο. Πιο συγκεκριμένα, αν έχουμε ένα πρόβλημα δύο κλάσεων και θεωρήσουμε μια από αυτές ως κύρια, τότε: αν το σημείο που δεν ταξινομείται σωστά ανήκει στην κύρια κλάση, τότε στο διάνυσμα των παραμέτρων (συντελεστών βαρύτητας) προσθέτουμε το διάνυσμα των συντεταγμένων του σημείου, διαφορετικά το αφαιρούμε. Ο αλγόριθμος αυτός ονομάζεται αλγόριθμος perceptron (βλ. αλγόριθμο 3). Μια παρόμοια μέθοδος, για δεδομένα με δυαδικές μεταβλητές, είναι η μέθοδος Winnow. [1]

Αλγόριθμος 3 Αλγόριθμος Perceptron για την εκπαίδευση ενός νευρώνα [1]

Θέσε όλους τους συντελεστές βαρύτητας ίσους με το μηδέν.
επανάλαβε
    για κάθε δείγμα I στα δεδομένα εκπαίδευσης **επανάλαβε**
        εάν το I δεν ταξινομείται σωστά από το perceptron **τότε**
            εάν το I ανήκει στην κύρια κλάση **τότε**
                πρόσθεσέ το στο διάνυσμα συντελεστών βαρύτητας.
            αλλιώς
                αφαίρεσέ το από το διάνυσμα συντελεστών βαρύτητας.
μέχρι όλα τα δείγματα στα δεδομένα εκπαίδευσης να ταξινομηθούν σωστά

Ένα νευρωνικό δίκτυο αποτελείται από πολλούς νευρώνες και μπορεί να αντιμετωπίσει μη γραμμικά προβλήματα. Κάθε νευρώνας του επιπέδου εισόδου λαμβάνει ένα σήμα εισόδου (τις τιμές των παραμέτρων-συντεταγμένων που αντιστοιχούν σε κάθε σημείο) και δίνει ένα σήμα εξόδου. Οι νευρώνες κάθε κρυφού επιπέδου λαμβάνουν σήματα μόνο από τους νευρώνες εισόδου ή τους νευρώνες του προηγούμενου κρυφού επιπέδου και όχι από το περιβάλλον, ωστόσο λειτουργούν με τον ίδιο τρόπο. Οι νευρώνες του επιπέδου εξόδου δίνουν ως έξοδο το τελικό σήμα (ένα ή περισσότερα). Κάθε νευρώνας λαμβάνει ως είσοδο τον γραμμικό συνδυασμό των σημάτων εξόδου των προηγούμενων νευρώνων. Το συνολικό

σήμα εισόδου χρησιμοποιείται ως όρισμα στη «συνάρτηση ενεργοποίησης» του νευρώνα, η οποία παράγει την έξοδο ή «ενεργοποίησή» του. Σε μια πρώτη προσέγγιση, μια συνάρτηση ενεργοποίησης είναι πρακτικά μια βηματική συνάρτηση που δίνει μοναδιαία έξοδο πάνω από ένα κατώφλι ή μηδενική αλλιώς. Ωστόσο, επειδή είναι χρήσιμο οι συναρτήσεις ενεργοποίησης να είναι παραγωγίσιμες, στη θέση των βηματικών συναρτήσεων χρησιμοποιούνται σιγμοειδείς συναρτήσεις με παρόμοια μορφή.

Για τον υπολογισμό των βαρών κάθε νευρώνα εφαρμόζεται συνήθως μια διαδικασία που ονομάζεται "πίσω διάδοση" (back propagation). Ο αλγόριθμος αυτός βασίζεται στην ιδέα ότι το σφάλμα εξόδου κάθε νευρώνα προκύπτει από τα σφάλματα των νευρώνων που τον τροφοδοτούν και άρα μπορούμε να το μοιράσουμε σε αυτούς (βλ. τον αλγόριθμο 4).

Αλγόριθμος 4 Back-propagation για εκπαίδευση νευρωνικών δικτύων [21]

Δεδομένα: σύνολο εκπαίδευσης με διάνυσμα εισόδου $\mathbf{x}$ και διάνυσμα εξόδου $\mathbf{y}$, ένα δίκτυο με L επίπεδα, βάρη $W_{j,i}$ και συνάρτηση ενεργοποίησης κάθε κόμβου g.

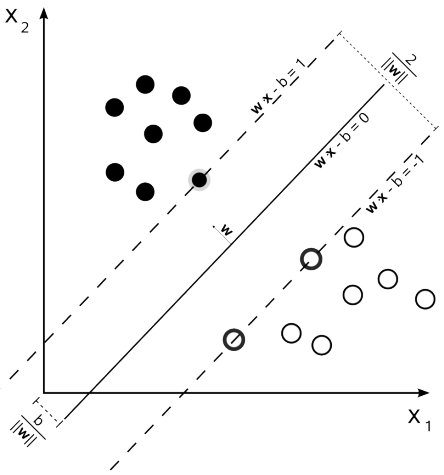
επανάλαβε
  για κάθε δείγμα e στα δεδομένα εκπαίδευσης **επανάλαβε**

    για κάθε κόμβο j στο επίπεδο εισόδου **επανάλαβε**
      $a_j \leftarrow x_j[e]$
    για $l = 2$ έως L **επανάλαβε**
      $in_i \leftarrow \sum_j W_{j,i} a_j$
      $a_i \leftarrow g(in_i)$

    για κάθε κόμβο i στο επίπεδο εξόδου **επανάλαβε**
      $\Delta_i \leftarrow g'(in_i) \times (y_i[e] - a_i)$
    για $l = L - 1$ έως 1 **επανάλαβε**
      για κάθε κόμβο j στο επίπεδο l **επανάλαβε**
        $\Delta_j \leftarrow g'(in_j) \sum_i W_{j,i} \Delta_i$
        για κάθε κόμβο i στο επίπεδο $l + 1$ **επανάλαβε**
          $W_{j,i} \leftarrow W_{j,i} + \alpha \times a_j \times \Delta_i$

μέχρι να ικανοποιηθεί κάποιο κριτήριο τερματισμού.

Ένα νευρωνικό δίκτυο είναι τελικά μια περίπλοκη μη γραμμική συνάρτηση, η οποία δημιουργείται από σύνθεση των συναρτήσεων ενεργοποίησης των κόμβων. Στα δίκτυα που παρουσιάσαμε τα σήματα τροφοδοτούνται μονάχα προς τα εμπρός, ξεκινώντας από το επίπεδο εισόδου. Δεν εμφανίζεται σε κανένα σημείο ανατροφοδότηση ή είσοδος σε ενδιάμεσο σημείο. Υπάρχουν ακόμα και άλλες δομές νευρωνικών δικτύων [21], όπως και πολλοί άλλοι αλγόριθμοι εκπαίδευσης νευρωνικών δικτύων, μεταξύ των οποίων και αλγόριθμοι που έχουν αναπτυχθεί από τη μονάδα Αυτόματης Ρύθμισης και Πληροφορικής της Σχολής Χημικών Μηχανικών ΕΜΠ [22, 23].



Σχήμα 3.4: Παράδειγμα διαχωρισμού με SVM (2D, 2 κλάσεις, γραμμικώς διαχωριζόμενα δεδομένα)*(Πηγή: Wikimedia commons - άδεια public domain)*

3.4.9 Μηχανές Διανυσμάτων Υποστήριξης (SVM)

Οι αλγόριθμοι τύπου Support Vector Machines (SVM) είναι μια σχετικά πρόσφατη προσέγγιση, που δίνει μερικές φορές καλύτερα αποτελέσματα από άλλους, παλαιότερους αλγορίθμους. Η κεντρική ιδέα είναι η δημιουργία μιας διαχωριστικής ευθείας γραμμής (γενικότερα υπερεπιπέδου) η οποία να έχει τη μεγαλύτερη δυνατή απόσταση από τις προς διαχωρισμό κλάσεις. Στην απλούστερη περίπτωση, έχουμε δύο κλάσεις και τα δείγματα αυτών είναι τελείως γραμμικώς διαχωριζόμενα. Δηλαδή έχουμε δύο ευδιάκριτες ομάδες σημείων που μπορούν να χωριστούν με π.χ. μια ευθεία γραμμή (σε 2D προβλήματα). Επεκτείνοντας, μπορούμε να ορίσουμε μια ανοχή σε σφάλματα, ώστε αν γενικώς τα δείγματα είναι γραμμικώς διαχωριζόμενα, αλλά κάποια σημεία ταξινομούνται λανθασμένα, να είναι δυνατή η χρήση γραμμικών μεθόδων, αποδεχόμενοι όμως ένα μικρό σφάλμα. Σημαντική επέκταση αυτών των αλγορίθμων είναι η αντιμετώπιση μη-γραμμικώς διαχωριζομένων κλάσεων, μέσω αντιστοίχισης του χώρου των δειγμάτων (πχ 2D) σε έναν χώρο περισσότερων διαστάσεων, όπου είναι δυνατός ο γραμμικός διαχωρισμός. Η υπόθεση για δύο κλάσεις μπορεί να ικανοποιηθεί και για προβλήματα περισσότερων κλάσεων, με κατάλληλους τρόπους, ενώ μπορούν να γίνουν και κατάλληλες μετατροπές για επιτυχή αντιμετώπιση προβλημάτων με έντονα μη ισοπληθείς κλάσεις. Μια πολύ καλή εισαγωγή στους αλγορίθμους SVM γίνεται στο review του Ovidiu Ivanciuc [24].

Ας δούμε πρώτα την περίπτωση που έχουμε **δύο γραμμικώς διαχωριζόμενες κλάσεις**, τις οποίες συμβολίζουμε με +1 και -1. Ένα παράδειγμα φαίνεται στο σχήμα 3.4. Για να προσδιορίσουμε το διαχωριστικό υπερεπίπεδο, αρκεί να υπολογίσουμε τους συντελεστές της εξίσωσης που το εκφράζει:

$$\{\mathbf{x} \in S | \mathbf{w} \cdot \mathbf{x} + b = 0\}, \mathbf{w} \in S, b \in \mathbb{R} \tag{3.17}$$

δηλαδή τους συντελεστές $\mathbf{w}$ των διαστάσεων $\mathbf{x}$ και τον σταθερό όρο b. Η κλάση ενός

σημείου συμπεραίνεται από την "πλευρά" του υπερεπιπέδου στην οποία αυτό βρίσκεται. Έτσι, εδώ η κλάση ενός νέου σημείου $\mathbf{x_k}$ θα είναι:

$$\text{class}(\mathbf{x_k}) = \{+1 \text{ εάν } \mathbf{w} \cdot \mathbf{x} + b > 0 \text{ ή } -1 \text{ εάν } \mathbf{w} \cdot \mathbf{x} + b < 0\} \tag{3.18}$$

Το διαχωριστικό που θέλουμε να κατασκευάσουμε πρέπει να είναι τέτοιο ώστε να ισχύουν ταυτόχρονα:

$$\mathbf{w} \cdot \mathbf{x}_i + b > +1 \text{ εάν } y_i = +1 \tag{3.19}$$

$$\mathbf{w} \cdot \mathbf{x}_i + b < -1 \text{ εάν } y_i = -1 \tag{3.20}$$

ή, κομψότερα:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geqslant 0 \tag{3.21}$$

Η απόσταση ενός σημείου $\mathbf{x}$ από το διαχωριστικό υπερεπίπεδο (το οποίο ορίζεται από τα $\mathbf{w}$ και b) είναι:

$$d(\mathbf{x}; \mathbf{w}, b) = \frac{|\mathbf{w} \cdot \mathbf{x} + b|}{||\mathbf{w}||} \tag{3.22}$$

ενώ η απόσταση του διαχωριστικού από την αρχή των αξόνων είναι:

$$d_{\min} = \frac{|b|}{||\mathbf{w}||} \tag{3.23}$$

Για τα υπερεπίπεδα-ευθείες στα οποία ισχύουν ακριβώς οι ισότητες, όπου συναντάμε τα πρώτα σημεία κάθε κλάσης, οι αποστάσεις από την αρχή των αξόνων είναι:

$$d_{-1} = \frac{|-1-b|}{||\mathbf{w}||} \text{ και } d_{+1} = \frac{|+1-b|}{||\mathbf{w}||} \tag{3.24}$$

έτσι, η απόσταση μεταξύ των δύο αυτών υπερεπιπέδων-ευθειών είναι $2/||\mathbf{w}||$ Η απόσταση αυτή καλείται περιθώριο (margin). Τελικά, το πρόβλημα κατασκευής ενός υπερεπιπέδου με τη μέγιστη απόσταση από τα δεδομένα μεταφράζεται στη μεγιστοποίηση του περιθωρίου, ή αλλιώς στην ελαχιστοποίηση του $||\mathbf{w}||$. Στο ίδιο αποτέλεσμα οδηγεί επίσης η ελαχιστοποίηση του $||\mathbf{w}||^2/2$. Τελικά, το **πρόβλημα βελτιστοποίησης** που πρέπει να επιλυθεί είναι το:

$$\begin{aligned} &\text{ελαχιστοποίησε την } f(\mathbf{x}) = \tfrac{||\mathbf{w}||^2}{2} \\ &\text{με τους περιορισμούς } g_i(\mathbf{x}) = y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geqslant 0, i = 1, \dots, m \end{aligned} \tag{3.25}$$

Σε πραγματικά προβλήματα, συνήθως δεν παρατηρείται τέλειος γραμμικός διαχωρισμός. Δηλαδή, ενώ φαίνεται κατά κύριο λόγο τα δεδομένα να διαχωρίζονται από π.χ. μια ευθεία γραμμή, κάποια σημεία καταλήγουν στη λάθος πλευρά, για κάθε δυνατή ευθεία. Σε τέτοιες περιπτώσεις, ο παραπάνω αλγόριθμος δεν μπορεί να δώσει λύση, μπορεί όμως να τροποποιηθεί, ώστε να δημιουργηθεί η διαχωριστική ευθεία που θα δίνει το ελάχιστο δυνατό σφάλμα ταξινόμησης.

Για να αντιμετωπιστεί το πρόβλημα, εισάγεται μια μεταβλητή ποινής, η οποία συμβολίζεται με ξ. Η μεταβλητή αυτή παίρνει μηδενική τιμή για σημεία τα οποία ταξινομούνται σωστά και τιμή η οποία αυξάνεται καθώς αυξάνει η απόσταση από το διαχωριστικό, για τα σημεία που δεν ταξινομούνται σωστά:

$$\xi_i(\mathbf{w}, b) = \begin{cases} 0 & \text{αν } y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geqslant +1 \\ 1 - y_i(\mathbf{w} \cdot \mathbf{x}_i + b) & \text{αν } y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \leqslant +1 \end{cases} \tag{3.26}$$

Σε αυτήν την περίπτωση, οι περιορισμοί του προβλήματος διαμορφώνονται ως εξής:

$$\begin{cases} \mathbf{w} \cdot \mathbf{x}_i + b \geqslant +1 - \xi_i & \text{αν } y_i = +1 \\ \mathbf{w} \cdot \mathbf{x}_i + b \leqslant -1 + \xi_i & \text{αν } y_i = -1 \\ \xi_i \geqslant 0, \forall i \end{cases} \tag{3.27}$$

ή αλλιώς:

$$\begin{cases} y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geqslant +1 - \xi_i, & i = 1, \dots, m \\ \xi_i \geqslant 0, & i = 1, \dots, m \end{cases} \tag{3.28}$$

Παρατηρείται το εξής: από τη μια, πρέπει να μεγιστοποιηθεί το εύρος, καθώς αυτό υπαγορεύει η αρχή της μεθόδου. Από την άλλη, πρέπει να ελαχιστοποιηθεί το σφάλμα ταξινόμησης, το οποίο θα είναι τόσο μεγαλύτερο, όσο μεγαλύτερο είναι το εύρος. Για να αντιμετωπιστεί αυτό, προσθέτουμε έναν όρο ποινής στην αντικειμενική συνάρτηση, η οποία τελικά γίνεται:

$$\frac{\|\mathbf{w}\|^2}{2} + C \cdot \left(\sum_{i=1}^{m} \xi_i \right)^k \tag{3.29}$$

όπου C είναι μια παράμετρος που καθορίζει το μέγεθος της ποινής και μπορεί να είναι διαφορετική για κάθε κλάση. Συνήθως προτιμάται η μορφή με $k = 1$, λόγω απλοποιήσεων που προκύπτουν στην επίλυση του προβλήματος βελτιστοποίησης. Έτσι, το προς επίλυση πρόβλημα διαμορφώνεται ως εξής:

$$\begin{aligned} &\text{ελαχιστοποίησε την } \tfrac{\|\mathbf{w}\|^2}{2} + C \cdot \textstyle\sum_{i=1}^{m} \xi_i \\ &\text{με τους περιορισμούς } \begin{cases} y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geqslant +1 - \xi_i, & i = 1, \dots, m \\ \xi_i \geqslant 0, & i = 1, \dots, m \end{cases} \end{aligned} \tag{3.30}$$

Σε ορισμένες περιπτώσεις, τα δεδομένα δεν μπορούν να περιγραφούν ικανοποιητικά από έναν απλό γραμμικό διαχωρισμό. Όμως, **με κατάλληλο μετασχηματισμό σε ένα χώρο υψηλότερης διάστασης, μπορούν να γίνουν γραμμικώς διαχωριζόμενα**. Ένας τέτοιος μετασχηματισμός γράφεται:

$$\mathbf{x} = (x_1, x_2, \dots, x_n) \to \varphi(\mathbf{x}) = (\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \dots, \varphi_h(\mathbf{x})) \tag{3.31}$$

και το μετασχηματισμένο σύνολο δεδομένων είναι:

$$\varphi(T) = \{(\varphi(\mathbf{x}_1), y_1), (\varphi(\mathbf{x}_2), y_2), \dots, (\varphi(\mathbf{x}_m), y_m)\} \tag{3.32}$$

Για να συμπεράνουμε την κλάση ενός σημείου $\mathbf{x}_k$ χρησιμοποιούμε, αντί για το ίδιο το σημείο, την απεικόνισή του στον νέο χώρο:

$$\text{class}(\mathbf{x}_k) = \text{sign}[\mathbf{w} \cdot \varphi(\mathbf{x}_k) + b] = \text{sign}\Big(\sum_{i=1}^{m} \lambda_i y_i \varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_k) + b \Big) \tag{3.33}$$

όπου λ_i είναι πολλαπλασιαστές Lagrange που χρησιμοποιούνται (όπως και στη γραμμική μέθοδο) για την επίλυση του προβλήματος βελτιστοποίησης. Παρατηρούμε ότι για να συμπεράνουμε την κλάση, είναι απαραίτητο να υπολογίσουμε το εσωτερικό γινόμενο $\varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_k)$ για όλα τα διανύσματα υποστήριξης $\mathbf{x}_i$. Ένα ειδικό είδος συναρτήσεων, οι οποίες καλούνται kernels, επιτρέπουν τον υπολογισμό αυτού του εσωτερικού γινομένου στον αρχικό χώρο (χαμηλότερης διάστασης). Για παράδειγμα, έστω ο αρχικός χώρος 2 διαστάσεων $\mathbf{x} = (x_1, x_2)$ και ο εξής μετασχηματισμός σε έναν χώρο 6 διαστάσεων:

$$\varphi(\mathbf{x}) = \Big(1, \sqrt{2}x_1, \sqrt{2}x_2, {x_1}^2, {x_2}^2, \sqrt{2}x_1x_2\Big) \tag{3.34}$$

το εσωτερικό γινόμενο μεταξύ δύο σημείων στον μεγαλύτερης διάστασης χώρο μπορεί να υπολογιστεί χρησιμοποιώντας μονάχα τα σημεία στον αρχικό χώρο, ως εξής:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j) = (1 + \mathbf{x}_i \cdot \mathbf{x}_j)^2 \tag{3.35}$$

Το παραπάνω είναι ένα πολυωνυμικό kernel 2ου βαθμού. Υπάρχουν και άλλα είδη kernel functions. Η πιο απλή μορφή, η οποία χρησιμοποιείται μονάχα για τον έλεγχο της μη-γραμμικότητας και τη σύγκριση με άλλα kernels είναι η τετραγωνική:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j \tag{3.36}$$

Η γενικότερη μορφή των πολυωνυμικών kernels (βαθμού d) είναι:

$$K(\mathbf{x}_i, \mathbf{x}_j) = (1 + \mathbf{x}_i \cdot \mathbf{x}_j)^d \tag{3.37}$$

Παρότι τα πολυωνυμικά kernels είναι απλά και αρκετά αποτελεσματικά, υπάρχει ο κίνδυνος του overfitting για υψηλές τιμές του βαθμού d. Πολύ συνηθισμένα είναι επίσης τα Radial Basis Function kernels, κυρίως στην γκαουσιανή τους μορφή:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\Big(- \frac{||x_i - x_j||^2}{2\sigma^2} \Big) = \exp\Big(- \gamma ||x_i - x_j||^2 \Big) \tag{3.38}$$

Η παράμετρος σ (ή, αντιστοίχως, η γ) καθορίζει το σχήμα του διαχωριστικού υπερεπιπέδου και επηρεάζει το πλήθος των σημείων που χρησιμοποιούνται ως διανύσματα υποστήριξης. Το βέλτιστο για την τιμή της στην πράξη καθορίζεται με μια διαδικασία cross-validation. Μια παρόμοια μορφή RBF kernel είναι η εκθετική, που διαφέρει μόνο στην απουσία του εκθέτη στον αριθμητή. Συνηθισμένος τύπος είναι επίσης τα σιγμοειδή kernels:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(a\mathbf{x}_i \cdot \mathbf{x}_j + b) \tag{3.39}$$

Τελικά, το πρόβλημα των μη-γραμμικώς διαχωριζόμενων δεδομένων, αντιμετωπίζεται ομοίως με την απλούστερη περίπτωση όπου υπάρχει γραμμικός διαχωρισμός, αντικαθιστώντας τα

σημεία $\mathbf{x}$ με τα μετασχηματισμένα $\varphi(\mathbf{x})$ και το εσωτερικό γινόμενο $\varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j)$ με μια συνάρτηση kernel $K(\mathbf{x}_i, \mathbf{x}_j)$.

Μια άλλη προσέγγιση των αλγορίθμων SVM είναι η **ν-SVM**. Εδώ, η παράμετρος ποινής C (η οποία μπορεί να λάβει τιμές στο $[0, +\infty)$) αντικαθίσταται από μια παράμετρο ν, η οποία λαμβάνει τιμές στο διάστημα $[0, 1]$. Η παράμετρος αυτή καθορίζει το κλάσμα των σημείων του δειγματοχώρου τα οποία είναι διανύσματα υποστήριξης και βρίσκονται στη λάθος πλευρά του διαχωριστικού υπερεπιπέδου. Το πρόβλημα βελτιστοποίησης που προκύπτει για το ν-SVM είναι:

$$\begin{aligned} &\text{ελαχιστοποίησε την } \frac{\|w\|^2}{2} - \nu\rho + \frac{1}{2}\sum_{i=1}^{m}\xi_i \\ &\text{με τους περιορισμούς } \begin{cases} y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geqslant \rho - \xi_i, & i = 1, \dots, m \\ \xi_i \geqslant 0, & i = 1, \dots, m \end{cases} \end{aligned} \tag{3.40}$$

Εάν ένας ν-SVM classifier οδηγεί σε $\rho > 0$, τότε ένας C-SVM classifier με $C = 1/m\rho$ έχει την ίδια συνάρτηση απόφασης.

Σε όλη την ανάλυση παρουσιάστηκαν προβλήματα δύο κλάσεων. Εάν οι κλάσεις είναι περισσότερες, τότε το πρόβλημα μπορεί να αντιμετωπιστεί σχεδόν άμεσα με τους αλγορίθμους δύο κλάσεων. Μια προσέγγιση για ένα k-κλάσεων πρόβλημα είναι να εξετάσουμε k τω πλήθος μοντέλα, όπου σε κάθε ένα εξετάζουμε τα σημεία της μιας κλάσης, ως προς τα σημεία όλων των υπολοίπων κλάσεων μαζί. Μια άλλη προσέγγιση είναι να εξετάσουμε κάθε κλάση ως προς κάθε μια άλλη κλάση ξεχωριστά και να αποδώσουμε στο προς εξέταση δείγμα την πιο συχνή τιμή κλάσης. Στη βιβλιογραφία συναντάει κανείς και περισσότερο εξελιγμένες μεθόδους [24].

3.5 Συναινετική μάθηση

Οι σημαντικές αποφάσεις συχνά δεν αφήνονται στην κρίση ενός ανθρώπου, αλλά λαμβάνονται από κάποια επιτροπή. Ας φανταστούμε ένα δύσκολο ιατρικό περιστατικό και μια ομάδα γιατρών η οποία έχει αναλάβει να το εξετάσει. Οι γιατροί που συμμετέχουν στην ομάδα μπορεί να έχουν, ο καθένας μόνος του, διαφορετικό ποσοστό επιτυχημένων αποφάσεων. Ενδεχομένως μάλιστα ο κάθε γιατρός να αντιμετωπίζει καλύτερα συγκεκριμένες περιπτώσεις. Στόχος της ομάδας είναι να συνδυάσει τις διαφορετικές διαγνώσεις ώστε να αποφανθεί όσο το δυνατόν καλύτερα για την περίπτωση.

Αντιστοίχως στη μηχανική μάθηση, μπορούμε να συνδυάσουμε διαφορετικά μοντέλα ώστε να δημιουργήσουμε ένα καλύτερο μοντέλο. Τα μοντέλα που θα χρησιμοποιήσουμε μπορεί να προέρχονται από διαφορετικούς αλγορίθμους μάθησης ή από τον ίδιο, χρησιμοποιώντας διαφορετικά σύνολα εκπαίδευσης. Σε προβλήματα ταξινόμησης, η κλάση μπορεί να προκύπτει από «ψηφοφορία» μεταξύ των μοντέλων και κάθε μοντέλο μπορεί να συμμετέχει ισότιμα ή με κάποια βαρύτητα στην τελική απόφαση. Αντιστοίχως, σε προβλήματα παλινδρόμησης, μπορεί να θεωρηθεί ως έξοδος η μέση τιμή (σταθμισμένη ή μη) των εξόδων κάθε μοντέλου. Τελικά, συνδυάζοντας μοντέλα με χαμηλότερη επιτυχία σε σχέση με κάποιο άλλο, μπορούμε να επιτύχουμε υψηλότερη επιτυχία από αυτό, κάτι που παρατηρήθηκε και κατά την επεξεργασία του Προβλήματος 1.

Συνήθεις μέθοδοι που χρησιμοποιούνται είναι οι bagging, boosting και stacking. Στη μέθοδο bagging, συνδυάζονται μοντέλα ίδιου τύπου και το αποτέλεσμα προκύπτει από απλή ψηφοφορία. Στη μέθοδο boosting, συνδυάζονται πάλι μοντέλα ίδιου τύπου, αλλά το τελικό αποτέλεσμα προκύπτει από σταθμισμένη ψηφοφορία: κάθε μοντέλο λαμβάνει έναν συντελεστή βαρύτητας σχετικό με την επιτυχία του. Στη μέθοδο stacking, συνδυάζονται μοντέλα διαφορετικού είδους (δηλαδή μοντέλα που έχουν προκύψει από διαφορετικούς αλγορίθμους). Το τελικό αποτέλεσμα προκύπτει πάλι από ψηφοφορία, ωστόσο χρειάζεται τα διαφορετικά μοντέλα να έχουν αντίστοιχους βαθμούς επιτυχίας. Για τη χρήση συντελεστών βαρύτητας, η μέθοδος επεκτείνεται, προσπαθώντας να κατανοήσει ποιοι αλγόριθμοι είναι οι πιο αξιόπιστοι. [1]

3.6 Έλεγχος παραγόμενου μοντέλου

Αφού εκπαιδευτεί ένα μοντέλο, πρέπει να μετρηθεί η ικανότητά του να περιγράφει σωστά το υπό εξέταση σύστημα, δηλαδή η ικανότητά του να γενικεύει τη γνώση που αποκτά από τα δεδομένα εκπαίδευσης. Δεν μπορούμε όμως να αξιολογήσουμε ένα μοντέλο χρησιμοποιώντας ως κριτήριο την επιτυχία πρόβλεψης της κλάσης των δεδομένων πάνω στα οποία έχει εκπαιδευτεί, καθώς αυτό δεν δίνει καμία πληροφορία ως προς το πώς θα συμπεριφερθεί το μοντέλο εκεί που τελικά χρειάζεται, δηλαδή στο να αποφανθεί για νέα, άγνωστα δείγματα (συνήθως ένα κλάσμα των συνολικών δεδομένων, το οποίο δεν χρησιμοποιείται στη φάση της εκπαίδευσης).

Με ένα περίπλοκο μοντέλο, μπορούμε να επιτύχουμε 100% επιτυχία στα δεδομένα εκπαίδευσης, χωρίς όμως αυτό το μοντέλο να έχει καμία αξία γενίκευσης. Ας σκεφτούμε τι θα γινόταν αν σε π.χ. 5 σχεδόν συνευθειακά σημεία στον $\mathbb{R}^2$ προσαρμόζαμε ένα πολυώνυμο 6ου βαθμού. Παρότι θα επιτυγχάναμε 100% επιτυχία για τα 5 σημεία μας, νέα σημεία δεν θα προβλέπονταν σωστά. Ο κίνδυνος της υπερπροσαρμογής (overfitting) είναι υπαρκτός, ωστόσο υπάρχει τρόπος να αντιμετωπιστεί. Συνήθως, το σύνολο δεδομένων χωρίζεται σε N ισοπληθή υποσύνολα, με τυχαίο τρόπο. Η διαδικασία εκπαίδευσης και αξιολόγησης διανύει N επαναλήψεις και ονομάζεται N-fold cross-validation. Σε κάθε επανάληψη, ένα από τα υποσύνολα (διαφορετικό κάθε φορά) χρησιμοποιείται μονάχα για την αξιολόγηση, ενώ τα υπόλοιπα $N-1$ για την εκπαίδευση και σταδιακά σχηματίζονται τα συγκεντρωτικά αποτελέσματα με τις προβλέψεις για κάθε σημείο. Συχνά χρησιμοποιείται 10-fold cross-validation. Η διαδικασία αυτή μειώνει αρκετά τον κίνδυνο υπερπροσαρμογής και αξιοποιεί όλα τα διαθέσιμα δεδομένα, τα οποία σε ορισμένα προβλήματα μπορεί να είναι λίγα.

Αν στην παράμετρο N δωσουμε τιμή ίση με το πλήθος των διαθέσιμων δεδομένων, τότε έχουμε τη μέθοδο Leave-One-Out Cross-Validation. Κάθε σημείο δοκιμάζεται σε ένα μοντέλο που έχει δημιουργηθεί χρησιμοποιώντας σχεδόν όλα τα διαθέσιμα δεδομένα (ωστόσο αυτό απαιτεί την εκπαίδευση N μοντέλων). Αυτή είναι επίσης μια ντετερμινιστική διαδικασία, καθώς δεν υπεισέρχεται τυχαιότητα στον τρόπο με τον οποίο δημιουργούνται τα υποσύνολα. Μια ακόμα διαδικασία είναι η Bootstrap, με την οποία τα δεδομένα στα οποία γίνεται η εκπαίδευση επιλέγονται με επαναπόθεση, δηλαδή είναι πιθανό κάποια δεδομένα να χρησιμοποιηθούν πάνω από μία φορές. [1]

Ανεξάρτητα από τον τρόπο χωρισμού των δεδομένων, ένα σημαντικό θέμα είναι και το

μέτρο με το οποίο γίνεται η αξιολόγηση. Όπως αναφέραμε και στην ενότητα 2.4.4, σε ένα σύνολο δεδομένων με ανομοιόμορφη κατανομή κλάσεων, όπου πχ το 95% των δειγμάτων ανήκει στην κλάση Α και το 5% στην κλάση Β, ένας κανόνας που κατατάσσει όλα τα δείγματα στην κλάση Α θα έχει 95% συνολική ακρίβεια, αλλά προφανώς δε θα έχει καμία χρησιμότητα. Για αυτόν τον λόγο είναι σημαντικό να εξετάζουμε και την ικανότητα ενός μοντέλου να προβλέπει κάθε κλάση ξεχωριστά. Αυτό μπορεί να γίνει παρατηρώντας τον "confusion matrix" που προκύπτει κατά τον έλεγχο ενός μοντέλου. Πολλά παραδείγματα εμφανίζονται στην ανάλυση των προβλημάτων (βλ. πχ τον πίνακα 6.2 στην σελίδα 80). Κάθε γραμμή του πίνακα αντιστοιχεί σε μια από τις γνωστές κλάσεις. Κάθε στήλη αντιστοιχεί σε μια από τις προβλεπόμενες κλάσεις. Τα στοιχεία $a_{i,j}$ του πίνακα αντιστοιχούν στο πλήθος των σημείων της i κλάσης που έχουν αποδοθεί στην κλάση j. Ένα καλό μοντέλο πρέπει να δίνει υψηλές τιμές στα στοιχεία της διαγωνίου και μικρές στα υπόλοιπα. Ένας κανόνας που κατατάσσει όλα τα στοιχεία σε μία κλάση, όπως στο τελευταίο παράδειγμα, θα έχει τιμές μόνο σε μία στήλη του πίνακα. Έχοντας διαθέσιμο τον confusion matrix είναι δυνατός ο υπολογισμός οποιουδήποτε επιθυμητού άλλου μέτρου αξιολόγησης όπως η ευστοχία (specificity) και η ευαισθησία (sensitivity) που αναφέραμε στην ενότητα 2.4.4.

3.7 Επιλογή μεταβλητών

Σε ορισμένα προβλήματα μπορεί να έχουμε στη διάθεσή μας ένα μεγάλο πλήθος μετρούμενων μεταβλητών, χωρίς συνήθως πολλές από αυτές να παρέχουν σημαντική πληροφορία και χωρίς πάντα να έχουμε αντιστοίχως μεγάλο πλήθος δεδομένων-δειγμάτων. Παρότι μπορούμε να κατασκευάσουμε μοντέλα χρησιμοποιώντας όλες τις μεταβλητές, είναι καλό να χρησιμοποιήσουμε τις ελάχιστες δυνατές. Ενσωματώνοντας μεταβλητές που δεν σχετίζονται με την έξοδο, προσθέτουμε περισσότερο θόρυβο στα δεδομένα και δυσχαιρένουμε την εύρεση των πραγματικών συσχετίσεων μεταξύ εισόδου και εξόδου. Επίσης παράγουμε πολυπλοκότερα μοντέλα, κάτι που μπορεί να οδηγήσει σε αναξιοπιστία (βλ. υπερπροσαρμογή) και αύξηση των παραμέτρων που πρέπει να προσδιοριστούν. Ακόμα, αυξάνει την πιθανότητα εμφάνισης αλληλοσυσχετιζόμενων μεταβλητών εισόδου.

Στόχος λοιπόν είναι να μειώσουμε τις μεταβλητές που χρησιμοποιούμε σε ένα μοντέλο. Αυτό συνήθως αντιμετωπίζεται ως ένα πρόβλημα βελτιστοποίησης με στόχο την ελαχιστοποίηση του σφάλματος, μεταβάλλοντας το πλήθος των μεταβλητών. Ένα πρόβλημα επιλογής μεταβλητών αποτελείται από τα εξής συστατικά:

1. έναν αλγόριθμο έρευνας
2. μια αντικειμενική συνάρτηση
3. έναν αλγόριθμο μοντελοποίησης της συσχέτισης μεταξύ των μεταβλητών εισόδου και εξόδου

Συνηθισμένοι αλγόριθμοι έρευνας σε προβλήματα επιλογής μεταβλητών είναι οι Γενετικοί Αλγόριθμοι και η Προσομοιωμένη Ανόπτηση (δεν θα ασχοληθούμε). Στόχος του προβλήματος επιλογής μεταβλητών είναι τόσο η μείωση των μεταβλητών (απλότητα) όσο και η

μείωση του σφάλματος (ακρίβεια). Οι δύο συχνά αντικρουόμενοι στόχοι μπορεί να αντιμετωπιστούν μαζί σε μία αντικειμενική συνάρτηση ή ως ξεχωριστά προβλήματα. Τέλος, ο αλγόριθμος μοντελοποίησης που θα χρησιμοποιηθεί πρέπει να είναι ακριβής αλλά και γρήγορος, αφού θα εφαρμοστεί πολλές φορές. [22]

Στο Πρόβλημα 2 εφαρμόζουμε μια διαδικασία επιλογής μεταβλητών όπου χρησιμοποιούμε έναν Γενετικό Αλγόριθμο για να υπολογίσουμε τα καλύτερα μοντέλα 1 μεταβλητής, 2 μεταβλητών, ..., N μεταβλητών και τελικά επιλέγουμε κάποιο που να δίνει ικανοποιητικά μικρό πλήθος μεταβλητών ταυτόχρονα με ικανοποιητικά υψηλή ακρίβεια. Αυτή η μέθοδος ονομάζεται Subset Selection και ανήκει σε μια οικογένεια μεθόδων που ονομάζεται "wrapper method" η οποία έχει τις εξής παραλλαγές: [13]

Forward Selection Ξεκινώντας χωρίς καμία μεταβλητή, προσθέτουμε σε κάθε βήμα τη μεταβλητή που βελτιώνει περισσότερο την επιτυχία του μοντέλου.

Backward Elimination Ξεκινώντας με όλες τις διαθέσιμες μεταβλητές, αφαιρούμε σε κάθε βήμα τη μεταβλητή που μειώνει λιγότερο την επιτυχία του μοντέλου.

Stepwise Selection Συνδυασμός των δυο παραπάνω. Ξεκινώντας χωρίς καμία μεταβλητή, προσθέτουμε σε κάθε βήμα τη μεταβλητή που βελτιώνει περισσότερο την επιτυχία του μοντέλου. Μετά από κάθε προσθήκη, ελέγχονται οι μεταβλητές που έχουν επιλεγεί και αφαιρούνται όσες δεν συνεισφέρουν πλέον σημαντικά στην επιτυχία του μοντέλου.

Subset Selection Σταδιακά δημιουργείται το καλύτερο μοντέλο μιας μεταβλητής, το καλύτερο μοντέλο δύο μεταβλητών, κτλ, μέχρι τη δημιουργία του καλύτερου μοντέλου N μεταβλητών. Οι μεταβλητές ελέγχονται είτε λαμβάνοντας όλους τους πιθανούς συνδυασμούς, είτε χρησιμοποιώντας κάποια ευρετική μέθοδο.

3.7.1 Γενετικοί αλγόριθμοι

Στη φύση παρατηρείται εξέλιξη των ειδών, κατά την οποία επικρατούν οργανισμοί με χαρακτηριστικά που ευνοούν την επιβίωσή τους. Η αναπαραγωγή με δύο γονείς οδηγεί σε απογόνους που φέρουν χαρακτηριστικά και των δύο. Άτομα που έχουν «καλύτερα» χαρακτηριστικά έχουν υψηλότερη πιθανότητα να επιβιώσουν και να δώσουν απογόνους. Μάλιστα, ένας οργανισμός με καλύτερα χαρακτηριστικά θα δώσει και περισσότερους απογόνους. Ταυτόχρονα, ένα μικρό κομμάτι του γενετικού υλικού μεταλλάσσεται μέσα στον χρόνο με τυχαίο τρόπο, οδηγώντας ορισμένες φορές στην εμφάνιση νέων χαρακτηριστικών. Τελικά, τα «καλά» χαρακτηριστικά, δηλαδή αυτά που δίνουν στους οργανισμούς την ικανότητα να επιβιώνουν και να δίνουν απογόνους, διατηρούνται από γενιά σε γενιά και τελικά επικρατούν.

Αντιστοίχως λειτουργεί η Γενετική Έρευνα. Ξεκινώντας με έναν πληθυσμό διαφορετικών λύσεων, δημιουργούνται με εξελικτικό τρόπο νέες λύσεις, εφαρμόζοντας τις ίδιες ιδέες της επιλογής, της διασταύρωσης και της μετάλλαξης. Οι λύσεις αναπαρίστανται ως διανύσματα που καλούνται «χρωμοσώματα» και κάθε στοιχείο του διανύσματος λαμβάνει μια δυαδική, ακέραια ή συνεχή τιμή, ανάλογα με το πρόβλημα. Π.χ., στο πρόβλημα επιλογής

αντικειμένων, κάθε στοιχείο έχει τιμή 0 ή 1, που αντιπροσωπεύει το αν το αντίστοιχο αντικείμενο θα επιλεγεί ή όχι. Από την άλλη, στο πρόβλημα του περιοδεύοντος πωλητή, κάθε στοιχείο έχει μια ακέραια τιμή που δείχνει μια πόλη, ενώ όλο το χρωμόσωμα δείχνει τη σειρά των πόλεων. Ο αλγόριθμος 5 αποτελεί μια τυπική περίπτωση γενετικού αλγορίθμου. [25]

Αλγόριθμος 5 Γενικό σχήμα ενός Γενετικού Αλγορίθμου [25]

```
Δημιούργησε έναν αρχικό πληθυσμό λύσεων
Υπολόγισε την καταλληλότητα κάθε ατόμου από τον πληθυσμό αυτό
επανάλαβε
    επανάλαβε
        Επέλεξε δύο άτομα από τον πληθυσμό για αναπαραγωγή
        Συνδύασε τους γεννήτορες για την παραγωγή δύο απογόνων
        Υπολόγισε την καταλληλότητα των απογόνων
        Εισήγαγε τους απογόνους στο νέο πληθυσμό λύσεων
    μέχρι να συμπληρωθεί μια πλήρης λύση
μέχρι να ικανοποιηθεί κάποιο κριτήριο τερματισμού
```

Η διαδικασία της επιλογής βασίζεται σε μια συνάρτηση καταλληλότητας. Αυτή μπορεί να είναι π.χ. το κόστος ή το όφελος κάθε λύσης και δείχνει την ικανότητα επιβίωσης του «φαινότυπου» που προκύπτει από το αντίστοιχο «χρωμόσωμα». Μια συνάρτηση καταλληλότητας πρέπει να είναι ομαλή και απλή, έτσι ώστε χρωμοσώματα παρόμοιας καταλληλότητας να είναι παρόμοια. Αφού σχηματιστεί ένας πληθυσμός και αξιολογηθούν τα χρωμοσώματα, η επιλογή μπορεί να γίνει με δύο τρόπους. Ο πρώτος τρόπος είναι σύμφωνα με την καταλληλότητα κάθε χρωμοσώματος. Για παράδειγμα, στη μέθοδο επιλογής τροχού ρουλέτας, κάθε ένα από τα n άτομα (με καταλληλότητα f_i) απεικονίζεται ως ένα ευθύγραμμο τμήμα μήκους $f_i / \sum_{j=1}^{n} f_j$ και όλα τα ευθύγραμμα τμήματα τοποθετούνται δίπλα-δίπλα, σχηματίζονται ένα διάστημα $[0, 1]$. Για να επιλεχθεί ένα χρωμόσωμα, παράγεται ένας τυχαίος αριθμός στο διάστημα $[0, 1]$ και επιλέγεται αυτό στο διάστημα του οποίου βρίσκεται ο αριθμός. Έτσι, τα καταλληλότερα επιλέγονται πολλές φορές, ενώ τα λιγότερο κατάλληλα μπορεί να μην επιλεγούν και καμία φορά. Με παρόμοιο τρόπο λειτουργεί και η παγκόσμια στοχαστική δειγματοληψία, με τη διαφορά ότι, μετά την κατασκευή του «τροχού», η επιλογή δεν γίνεται με τυχαίους αριθμούς, αλλά με συγκεκριμένους, οι οποίοι αντιστοιχούν στα άκρα επιθυμητού πλήθους υποδιαστημάτων του $[0, 1]$. Ο δεύτερος τρόπος επιλογής είναι η άμεση (ή έμμεση) επαναποτύπωση της καταλληλότητας. Η μέθοδος αυτή έχει στόχο τη μεροληπτική επιλογή γεννητόρων και μπορεί να γίνει π.χ. με πρόσθεση ή αφαίρεση από τη συνάρτηση καταλληλότητας ενός σταθερού αριθμού, με αποτέλεσμα την κλιμάκωση της καταλληλότητας, έτσι ώστε να ευνοούνται περισσότερο οι υψηλότερες ή οι χαμηλότερες τιμές αυτής. Ακόμα, υπάρχει η μέθοδος της (πιθανοτικής) επιλογής πρωταθλήματος. Σε αυτήν, επιλέγονται τυχαία κάποια άτομα και επιλέγεται το καταλληλότερο, ενώ η διαδικασία επαναλαμβάνεται μέχρι να επιλεγεί το επιθυμητό πλήθος γεννητόρων. Στην πιθανοτική εκδοχή της μεθόδου, το καταλληλότερο δεν είναι σίγουρο ότι θα επιλεχθεί, αλλά υπεισέρχεται σε αυτό μια πιθανότητα.

Αφού επιλεχθούν οι γεννήτορες, πρέπει να συνδυαστούν ώστε να δώσουν απογόνους.

Στο γενικό σχήμα, δύο γεννήτορες δίνουν ισάριθμους απογόνους [25], ωστόσο είναι διαδεδομένη και μια διαδικασία σύμφωνα με την οποία δύο γεννήτορες δίνουν μόνο έναν απόγονο [21]. Η διασταύρωση των γεννητόρων γίνεται ως εξής: επιλέγεται τυχαία ένα σημείο στο οποίο τα δύο χρωμοσώματα κόβονται (ίδιο και για τα δύο). Έτσι προκύπτουν δύο τμήματα «κεφαλής» και δύο τμήματα «ουράς». Ο πρώτος απόγονος προκύπτει από την κεφαλή του Α γεννήτορα και την ουρά του Β, ενώ ο δεύτερος από την κεφαλή του Β και την ουρά του Α. Αν το σημείο κοπής βρεθεί στο πρώτο ή στο τελευταίο γονίδιο, τότε απλώς τα χρωμοσώματα μεταφέρονται στην επόμενη γενιά. Στη συνέχεια, ακολουθεί η μετάλλαξη των απογόνων. Κάθε γονίδιο αλλάζει τιμή σύμφωνα με μια μικρή ανεξάρτητη πιθανότητα και αυτό βοηθάει τον αλγόριθμο να ξεφύγει από τοπικά ακρότατα. Οι γενετικοί αλγόριθμοι τερματίζονται συνήθως μετά από ένα προκαθορισμένο πλήθος γενεών ή εάν τα γονίδια έχουν συγκλίνει, δηλαδή εάν παρουσιαστούν στον πληθυσμό μόνο παρόμοια χρωμοσώματα.

Σε πιο εξελιγμένους γενετικούς αλγορίθμους, γενετικό υλικό μπορεί να ανταλλαχθεί ή να τροποποιηθεί και με άλλους τρόπους. Μια θεώρηση αντιμετωπίζει τα χρωμοσώματα ως κλειστούς βρόχους και κόβει το καθένα σε πάνω από ένα σημεία (τελεστής ανταλλαγής υλικού πολλών σημείων). Μια άλλη θεώρηση παράγει απογόνους το κάθε γονίδιο των οποίων αντιμετωπίζεται ανεξάρτητα και προέρχεται από τον έναν ή τον άλλο γεννήτορα. Ακόμη, μπορεί να επιβληθεί «καρυωτυπική» εξέλιξη, δηλαδή να αλλάξει η διάταξη των γονιδίων, πχ αντιστρέφοντας ένα κομμάτι του γονιδίου. Στη φύση ωστόσο, η καρυωτυπική εξέλιξη είναι αρκετά πιο σπάνια από την γενοτυπική. Σε ορισμένα προβλήματα επίσης μπορεί να θεωρηθεί ότι παρουσιάζονται εξαρτήσεις ανάμεσα στα γονίδια. Έτσι, μια αλλαγή σε ένα γονίδιο μπορεί να δίνει πάντα την ίδια αλλαγή στην καταλληλότητα του χρωμοσώματος, ή η αλλαγή αυτή μπορεί να εξαρτάται από τις τιμές άλλων γονιδίων (επίσταση). [25]

3.7.2 Επιλογή μεταβλητών με γενετική έρευνα

Οι γενετικοί αλγόριθμοι μπορούν να χρησιμοποιηθούν, μεταξύ άλλων, και σε προβλήματα επιλογής μεταβλητών. Στην περίπτωση αυτή, κάθε θέση-γονίδιο αντιστοιχεί σε μία από τις μεταβλητές. Τα γονίδια λαμβάνουν δυαδικές τιμές (0 ή 1), αναπαριστώντας τη συμπερίληψη ή μη της αντίστοιχης μεταβλητής στο μοντέλο. Κάθε χρωμόσωμα πολλαπλασιάζεται στοιχείο προς στοιχείο με το διάνυσμα των μεταβλητών και έτσι προκύπτει το διάνυσμα των επιλεγμένων μεταβλητών, το οποίο τροφοδοτείται στον αλγόριθμο μηχανικής μάθησης της επιλογής μας. Όταν ο γενετικός αλγόριθμος συγκλίνει ή τερματιστεί, η κυρίαρχη τιμή κάθε γονιδίου μας υποδεικνύει τη χρήση ή μη της εκάστοτε μεταβλητής.

Στην εργασία αυτή εφαρμόστηκε η εξής σκέψη: οι γενετικοί αλγόριθμοι δεν καταλήγουν πάντα στην ίδια λύση, συνεπώς απαιτείται να εκτελεστούν πολλές φορές. Στις λύσεις που προκύπτουν, κάποιες μεταβλητές τείνουν να εμφανίζονται πιο συχνά από άλλες. Κατατάσσονται έτσι όλες οι μεταβλητές σύμφωνα με την συχνότητα επιλογής τους. Στη συνέχεια, ξεκινώντας από την πιο συχνά εμφανιζόμενη μεταβλητή, προσθέτουμε διαδοχικά μεταβλητές στο μοντέλο, καταγράφοντας την ακρίβειά του. Έτσι έχουμε τελικά στη διάθεσή μας έναν πίνακα ακρίβειας σε σχέση με το πλήθος των μεταβλητών, ιδιαίτερα χρήσιμο για να επιλέξουμε ένα ικανοποιητικά απλό και ταυτόχρονα ακριβές μοντέλο.

3.8 Ενδεικτικές εφαρμογές

Η εξόρυξη δεδομένων βρίσκει πλειάδα άλλων εφαρμογών πέρα από τη μεταβολομική, όπως στο marketing (ανάλυση ζήτησης και καλύτερη προώθηση), στην αξιολόγηση δανείων (θα μπορέσει ο υποψήφιος δανειζόμενος να επιστρέψει το δάνειο;), στην πρόρρηση ιδιοτήτων χημικών ουσιών (τι συμπεριφορά είχαν άλλες ουσίες με συγγενική δομή;), στην αναγνώριση εικόνων (ποια είναι τα χαρακτηριστικά που ταυτοποιούν π.χ. μια πετρελαιοκηλίδα σε μια δορυφορική εικόνα ή ένα πρόσωπο σε μια φωτογραφία;), στην πρόβλεψη φόρτου σε ηλεκτρικά ή άλλα δίκτυα, και αλλού. Πολύ μεγάλο ενδιαφέρον παρουσιάζεται τα τελευταία χρόνια και στην εφαρμογή μεθόδων εξόρυξης δεδομένων σε διαδικτυακές εφαρμογές. Χαρακτηριστικό παράδειγμα είναι οι μηχανές αναζήτησης που «μαντεύουν» τι σκέφτεται ο χρήστης και «προλαβαίνουν» τις επιθυμίες του, όπως και τα φίλτρα ανεπιθύμητης αλληλογραφίας. Συχνά πλέον, εταιρείες που θέλουν να δημιουργήσουν ή να βελτιώσουν διαδικτυακές εφαρμογές ζητάνε πλέον επιστήμονες και μηχανικούς που να έχουν γνώσεις μηχανικής μάθησης και εξόρυξης δεδομένων.

Κεφάλαιο 4

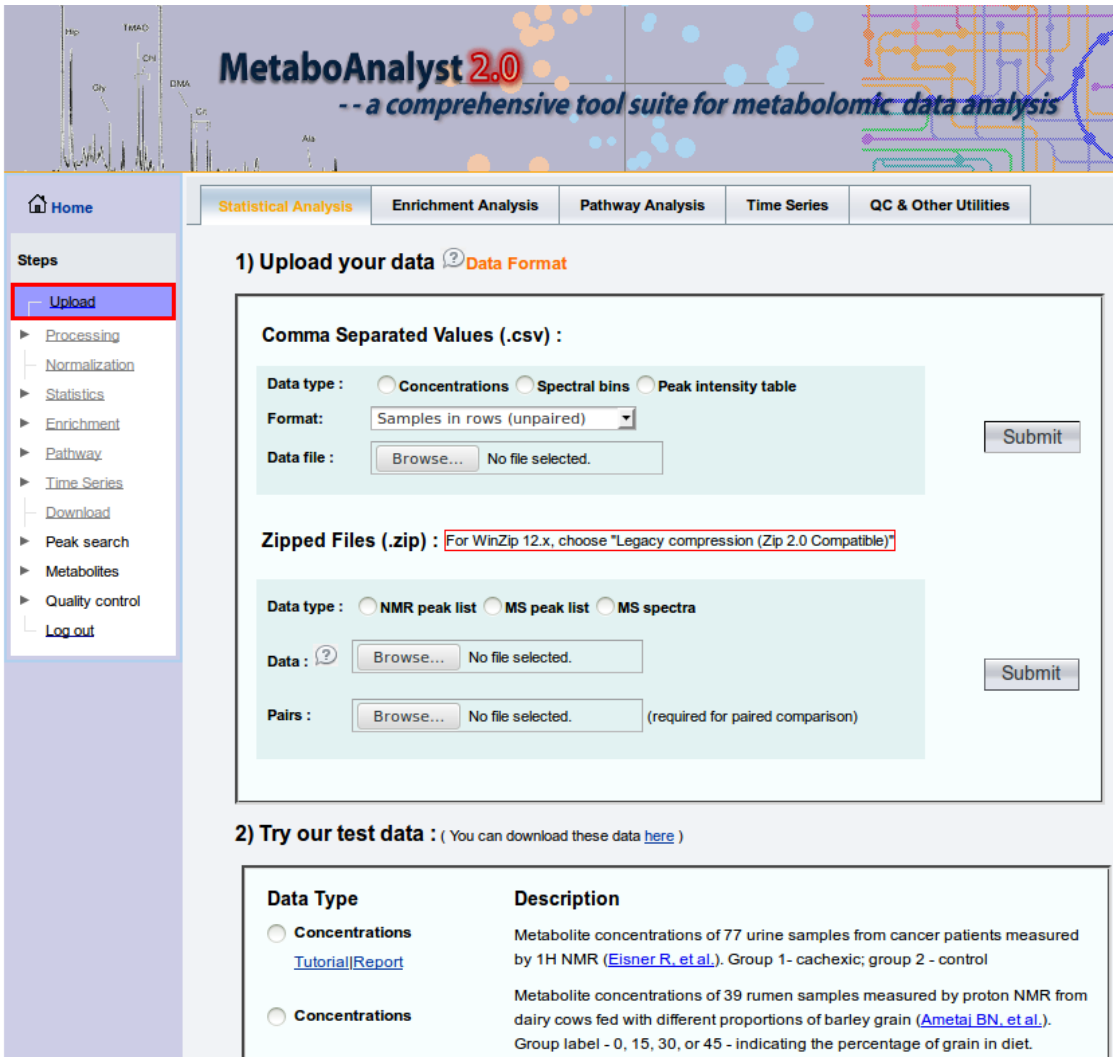
Λογισμικό

Για την ανάλυση των δεδομένων μας χρησιμοποιήθηκαν κυρίως ήδη διαθέσιμα πακέτα λογισμικού. Το εργαστήριο που μας παραχώρησε τα δεδομένα του δεύτερου προβλήματος χρησιμοποιεί ήδη τη σουίτα online εργαλείων MetaboAnalyst, η οποία εξειδικεύεται σε προβλήματα ανάλυσης μεταβολομικών δεδομένων. Η σουίτα WEKA είναι γενικότερης χρήσης και παρέχει μεγάλη ποικιλία διαφορετικών αλγορίθμων. Επιλέχθηκε ακριβώς λόγω αυτής της ποικιλίας, ώστε να είναι δυνατή η σύγκριση πολλών διαφορετικών μεθόδων μάθησης. Συνεισέφερε επίσης στην επιλογή το γεγονός ότι είχε χρησιμοποιηθεί ξανά στο παρελθόν από το εργαστήριό μας, καθώς και το ότι είναι ελεύθερο λογισμικό. Τόσο ως κομμάτι της WEKA όσο και ως αυτόνομο εργαλείο χρησιμοποιήθηκε και η υλοποίηση μηχανών διανυσμάτων υποστήριξης LibSVM, λόγω των καλών αποτελεσμάτων που συναντώνται στη βιβλιογραφία και της εύκολης σύνδεσής της με τη WEKA. Τέλος, αναπτύχθηκε κώδικας GNU Octave/MATLAB™ ο οποίος υλοποιεί έναν γενετικό αλγόριθμο, χρησιμοποιώντας τη LibSVM για την παραγωγή μοντέλων.

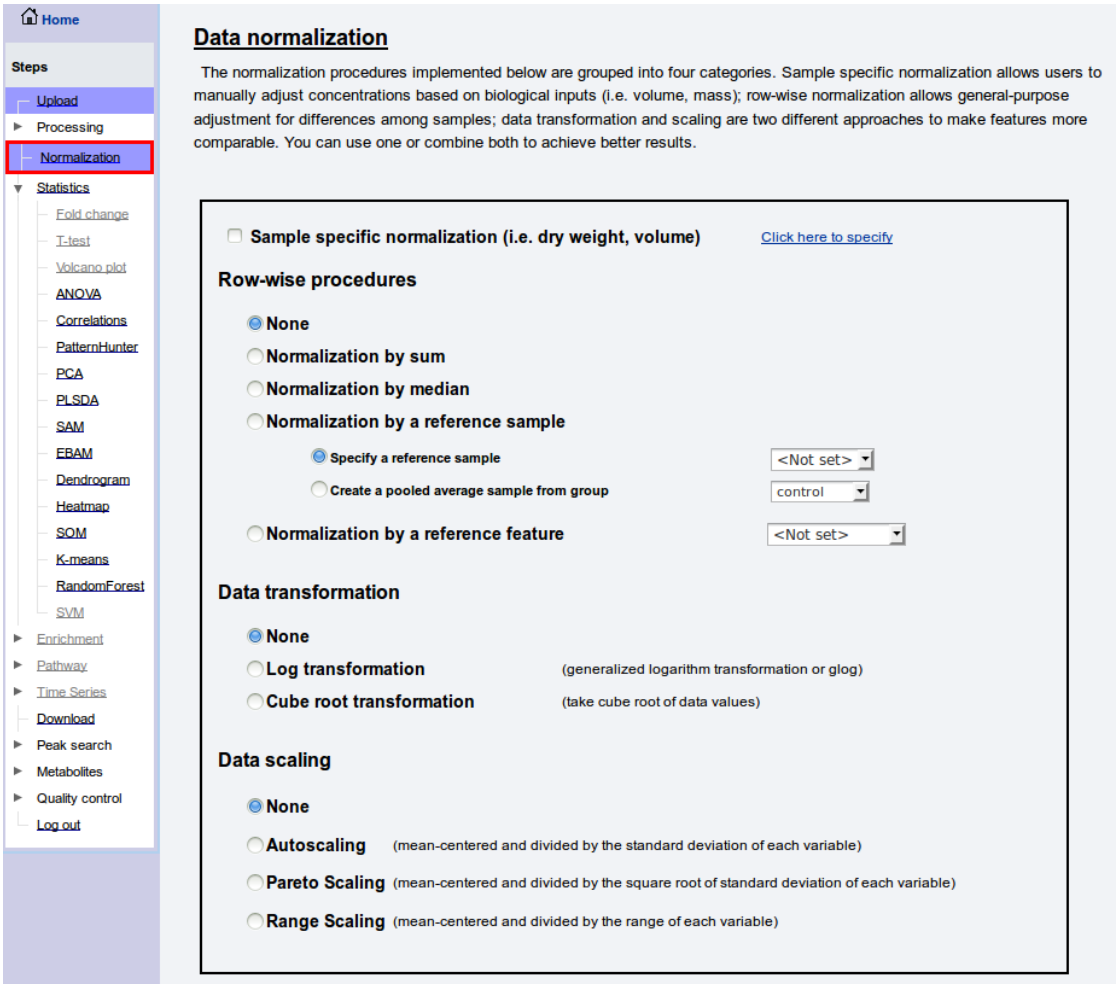
4.1 MetaboAnalyst

Το metaboanalyst.ca αποτελεί μια συλλογή εργαλείων για την ανάλυση μεταβολομικών δεδομένων. Οποιοσδήποτε μπορεί να συνδεθεί σε αυτό και να υποβάλει δεδομένα προς ανάλυση. Δεν απαιτείται κάποιου είδους εγγραφή και τα δεδομένα παραμένουν ιδιωτικά, ενώ μπορεί κανείς (εναλλακτικά) να κατεβάσει τα απαραίτητα αρχεία κώδικα και να το χρησιμοποιήσει τοπικά. Δημιουργήθηκε από τον Jeff Xia (University of Alberta), χρησιμοποιώντας κυρίως Java και R, ενώ χρηματοδοτείται από το "The Metabolomics Innovation Centre (TMIC)" του Καναδά. Στην ιστοσελίδα του MetaboAnalyst υπάρχει εκτενής βιβλιογραφία σχετικά με τη χρήση και τις δυνατότητές του. [7, 8, 13]

Στο σχήμα 4.1 φαίνεται η αρχική σελίδα του MetaboAnalyst. Διακρίνεται η φόρμα όπου μπορεί κανείς να υποβάλει τα δικά του δεδομένα ή να χρησιμοποιήσει κάποια που παρέχονται ως παραδείγματα. Στα αριστερά διακρίνεται το μενού με τα στάδια ανάλυσης. Τα δεδομένα μπορεί να αντιστοιχούν σε συγκεντρώσεις ουσιών ή φάσματα NMR/MS. Δέχεται απλά αρχεία CSV όπου η πρώτη στήλη αντιστοιχεί στην ταυτότητα των δειγμάτων (με τίτλο Sample), η δεύτερη στην κλάση (Label) και οι υπόλοιπες στις υπό ανάλυση ουσίες (στην



Σχήμα 4.1: Η αρχική σελίδα του MetaboAnalyst 2.0.



Σχήμα 4.2: MetaboAnalyst - Επιλογές προεπεξεργασίας των δεδομένων.

περίπτωση συγκεντρώσεων). Αμέσως μετά την υποβολή των δεδομένων ακολουθεί έλεγχος των δεδομένων ως προς κάποια βασικά δομικά χαρακτηριστικά (μορφή, άγνωστες τιμές, διαστάσεις και πλήθος κλάσεων,...) ενώ δίνεται η δυνατότητα εκτίμησης αγνώστων τιμών.

Για την προεπεξεργασία των δεδομένων παρέχονται μέθοδοι κανονικοποίησης ως προς το άθροισμα ή τη διάμεσό τους, ή σύμφωνα με ένα δείγμα ή συστατικό αναφοράς. Επιτρέπονται επίσης διορθώσεις για κάθε δείγμα ξεχωριστά. Θα πρέπει να προσεχθεί εάν έχει γίνει ήδη κάποια αντίστοιχη προεπεξεργασία. Παρέχονται επίσης μέθοδοι μετασχηματισμού (λογαριθμικός, κυβική ρίζα) καθώς και μέθοδοι scaling (Autoscaling, Pareto Scaling, Range Scaling). Μετά την εφαρμογή, ακολουθεί γραφική επισκόπηση του αποτελέσματος της προεπεξεργασίας, όπου φαίνεται (δειγματοληπτικά), η κατανομή τιμών στα διάφορα χαρακτηριστικά-συστατικά (βλ. σχήμα 6.1).

Στην κυρίως στατιστική επεξεργασία, παρέχονται οι εξής δυνατότητες:

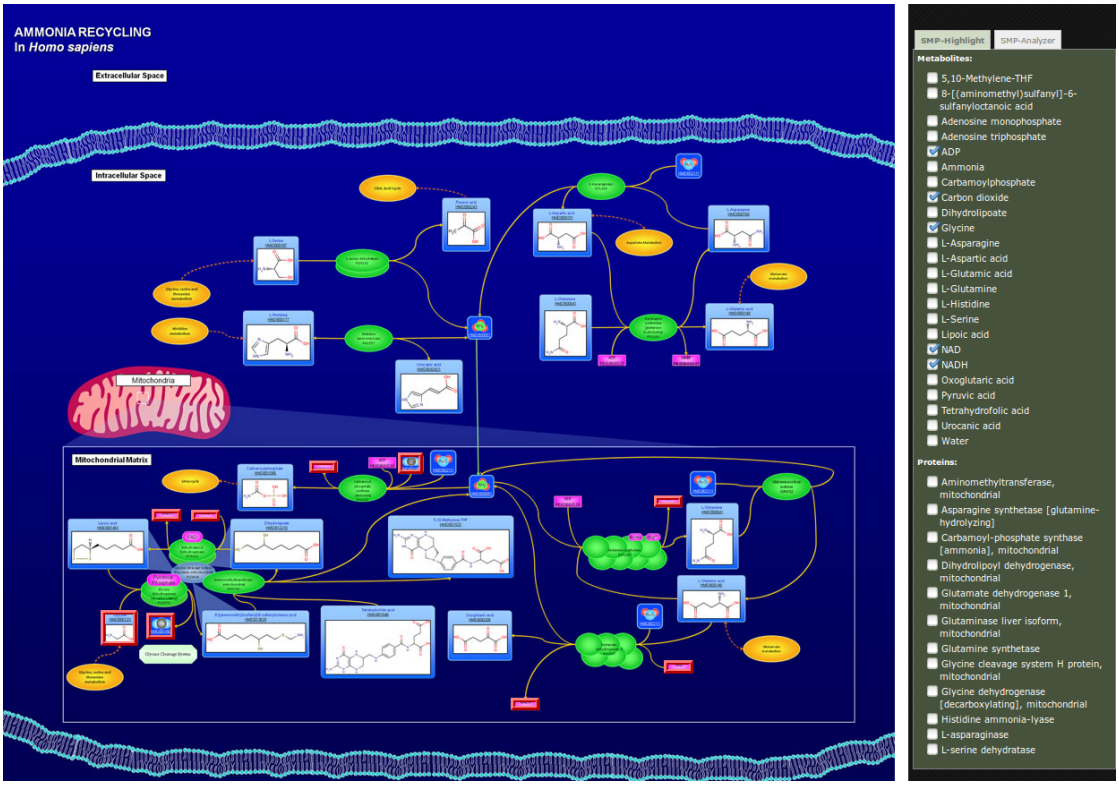
- Μονοπαραμετρική ανάλυση
  - Fold Change Analysis, t-Tests, Volcano plot (μόνο για δύο κλάσεις)
  - One-way Analysis of Variance (ANOVA)
  - Correlation Analysis & Pattern Searching
- Πολυπαραμετρική ανάλυση
  - Principal Component Analysis (PCA)
  - Partial Least Squares - Discriminant Analysis (PLS-DA)
- Αναγνώριση σημαντικών χαρακτηριστικών
  - Significance Analysis of Microarray (and Metabolites) (SAM)
  - Empirical Bayesian Analysis of Microarray (and Metabolites) (EBAM)
- Ανάλυση συστάδων
  - Hierarchical Clustering - Dendrogram & Heatmap
  - Partitional Clustering - K-Means & Self Organizing Map (SOM)
- Ταξινόμηση και επιλογή χαρακτηριστικών
  - Random Forest
  - Support Vector Machine (SVM) (μόνο για δύο κλάσεις)

Οι μέθοδοι PCA και PLS-DA δίνουν άμεση εικόνα για την ποιότητα του διαχωρισμού μεταξύ των κλάσεων και είναι από τα βασικότερα διαγράμματα που παρουσιάζονται σε σχετικές εργασίες. Εδώ ο χρήστης μπορεί να επιλέξει το πλήθος των principal components με τα οποία θέλει να εργαστεί, να τα προβάλει όλα μαζί σε ένα χάρτη και στη συνέχεια να εστιάσει σε 2D ή 3D διαγράμματα. Παραδείγματα τέτοιων διαγραμμάτων φαίνονται στα σχήματα 6.3 έως 6.8 (σελ. 74).

Τα διαγράμματα τύπου Heatmap, στην ανάλυση συστάδων, είναι επίσης ένας κλασσικός τρόπος για την εποπτική ανίχνευση τυχόν διαχωρισμών, ο οποίος σε ορισμένα προβλήματα δίνει ξεκάθαρα αποτελέσματα [3]. Ένα παράδειγμα Heatmap για τα δεδομένα του Προβλήματος 2 μπορείτε να βρείτε στη σελίδα 78.

Από αλγορίθμους μη εποπτευόμενης μάθησης παρέχονται ελάχιστες επιλογές (Random Forest και SVM, μόνο για δύο κλάσεις). Η υλοποίηση του αλγορίθμου Random Forest παρέχει έναν confusion matrix καθώς και μια κατάταξη των μεταβλητών (χαρακτηριστικών) του συστήματος ως προς τη σημασία τους στην ακρίβεια του μοντέλου, καθώς και με ποιες κλάσεις αυτές συνδέονται περισσότερο. Ακόμα, παρέχει ένα διάγραμμα από το οποίο μπορούν να ανιχνευθούν πιθανά outliers.

Το MetaboAnalyst δεν απευθύνεται τόσο σε ερευνητές που ενδιαφέρονται για τις μεθόδους ανάλυσης και μάθησης, αλλά περισσότερο σε επιστήμονες βιολογικών, φαρμακευτικών και ιατρικών πεδίων. Παρότι δεν δίνει τη δυνατότητα σημαντικής παρέμβασης στα



Σχήμα 4.3: MetaboAnalyst - Μεταβολικά μονοπάτια. Οι επιλεγμένοι στα δεξιά μεταβολίτες εμφανίζονται σε κόκκινο πλαίσιο στο σχήμα. Πατώντας σε κάθε στάδιο, ο χρήστης μπορεί να λάβει περισσότερες πληροφορίες.

εργαλεία, απλοποιεί σημαντικά τις αναλύσεις δεδομένων που χρησιμοποιούνται συχνά στη Μεταβολομική. Η αξία του ωστόσο ενισχύεται από επιπλέον δυνατότητες που παρέχει στην ανάλυση μεταβολικών μονοπατιών. Παράγει εντυπωσιακά διαδραστικά σχήματα, μέσα στα οποία μπορεί κανείς να εντοπίσει μεταβολίτες που παρουσιάζουν ενδιαφέρον στο υπό εξέταση πρόβλημα, να δει σε ποια σημεία εμφανίζονται, και να ανακαλύψει συσχετίσεις με άλλους μεταβολίτες και μονοπάτια. Ένα παράδειγμα φαίνεται στο σχήμα 4.3.

Αφού ολοκληρωθεί η ανάλυση, ο χρήστης μπορεί να κατεβάσει όλα τα διαγράμματα που έχουν παραχθεί στην επιθυμητή ανάλυση, καθώς και άλλα βοηθητικά αρχεία. Βασικότερο όλων είναι μια αυτόματη αναφορά, η οποία παρουσιάζει συνοπτικά τις μεθόδους, τις παραμέτρους και τα αποτελέσματα.

4.2 WEKA

Η WEKA παρέχει μια μεγάλη συλλογή αλγορίθμων εξόρυξης δεδομένων, τόσο στο κομμάτι της προεπεξεργασίας, όσο και στο κυρίως κομμάτι της μηχανικής μάθησης. Αναπτύσσεται

σε Java από το Machine Learning Group του University of Waikato και είναι διαθέσιμη ελεύθερα για όλες τις βασικές πλατφόρμες [9]. Το πολύ κατατοπιστικό βιβλίο "Data Mining: Practical Machine Learning Tools and Techniques" των Witten κ.α. [1] εξηγεί όλες τις μεθόδους που παρέχονται στη WEKA και αποτελεί έναν πολύ καλό οδηγό χρήσης της.

Τα δεδομένα μπορούν να τροφοδοτηθούν σε δύο μορφές αρχείων. Η βασική είναι η μορφή `ARFF`. Ένα τέτοιο αρχείο περιέχει, εκτός από τα δεδομένα, πληροφορίες για τις μεταβλητές του συστήματος. Ένα απόσπασμα από το αρχείο δεδομένων του Προβλήματος 1, σε μορφή `ARFF` είναι:

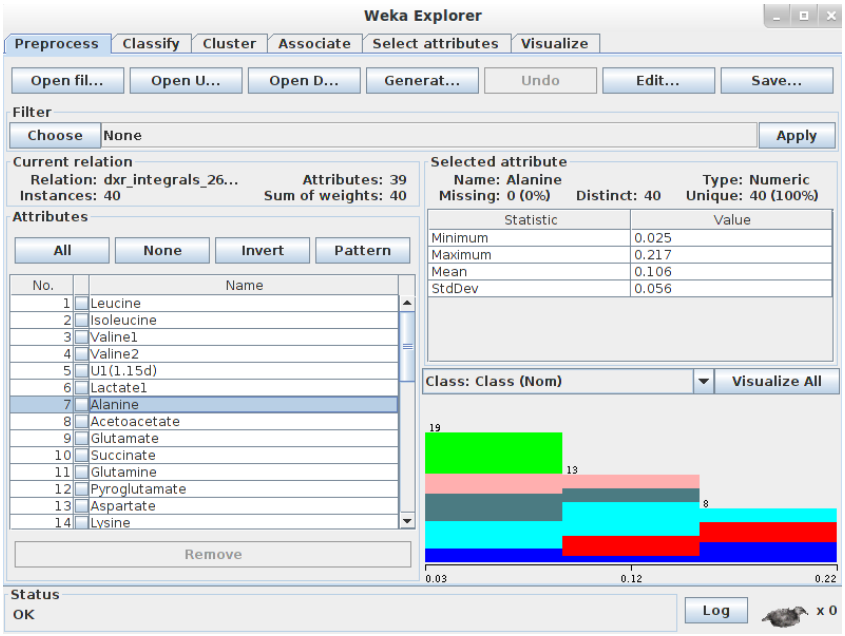
```
@relation DATA3

@attribute patient {AKI_8_24_01_110722,AKI_8_24_03_110722, ... }
@attribute kidney_state {'Biopsy kidney normal','Acute Kidney Injury'}
@attribute 1 numeric
@attribute 2 numeric
...
@attribute 701 numeric

@data
AKI_8_24_01_110722,'Biopsy kidney normal',-0.005178,-0.004944, ...
AKI_8_24_03_110722,'Biopsy kidney normal',-0.00492,-0.006055, ...
...
```

Η γραμμή `@relation` δίνει ένα όνομα στα δεδομένα και ενσωματώνει πληροφορίες για τυχόν προεπεξεργασία των δειγμάτων που έχει γίνει με τη WEKA (δεν φαίνεται στο παράδειγμα). Ως κλάση θεωρείται αυτομάτως η τελευταία μεταβλητή (παρότι αυτό μπορεί να τροποποιηθεί), οπότε μια αναδιάταξη των μεταβλητών στο στάδιο της προεπεξεργασίας θα ήταν πρακτική και μια τέτοια επέμβαση θα αναγραφόταν σε αυτήν την γραμμή. Οι γραμμές `@attribute` καθορίζουν το όνομα και το είδος των τιμών κάθε μεταβλητής, οι οποίες μπορεί να είναι κατηγορικές, αριθμητικές ή ημερομηνίας-ώρας (κατά ISO 8601). Στην περίπτωση κατηγορικών μεταβλητών, πρέπει να καθοριστούν οι επιτρεπτές τιμές, όπως στο παράδειγμα. Στη συνέχεια ακολουθεί η γραμμή `@data` και από κάτω οι τιμές των μεταβλητών για κάθε δείγμα σε ξεχωριστή γραμμή, χωρισμένες με κόμμα. Άγνωστες τιμές αντικαθίστανται με `?`. Εκτός από `ARFF`, η WEKA μπορεί να χειριστεί και αρχεία `CSV`, δίνοντας τη δυνατότητα μετατροπής τους σε μορφή `ARFF`. [1]

Το βασικό περιβάλλον της WEKA είναι ο `Explorer`. Μέσω αυτού είναι προσβάσιμες όλες οι μέθοδοι που παρέχονται. Ο χρήστης έχει επίσης τη δυνατότητα χειρισμού μέσω τριών ακόμα εναλλακτικών. Το περιβάλλον `Experimenter` επιτρέπει δοκιμές μεγάλης κλίμακας, με μαζικές δοκιμές διαφορετικών αλγορίθμων, παραμέτρων και δεδομένων. Το περιβάλλον `Knowledge Flow` δίνει τη δυνατότητα χειρισμού των μεθόδων υπό μορφή διαγράμματος ροής, θυμίζοντας πακέτα επεξεργασίας σημάτων. Τέλος, είναι δυνατή η χρήση της WEKA και σε περιβάλλον εντολών (`CLI`).

Ο `Explorer` διαθέτει τις εξής καρτέλες:

Preprocess για την προεπεξεργασία των δεδομένων.



Σχήμα 4.4: WEKA Explorer - Καρτέλα Preprocess. Στη δεξιά στήλη βρίσκονται οι μεταβλητές του συστήματος και στα αριστερά το ιστόγραμμα της επιλεγμένης στα δεδομένα, σε σχέση και με την κλάση τους.

Classify για εποπτευόμενη μάθηση.

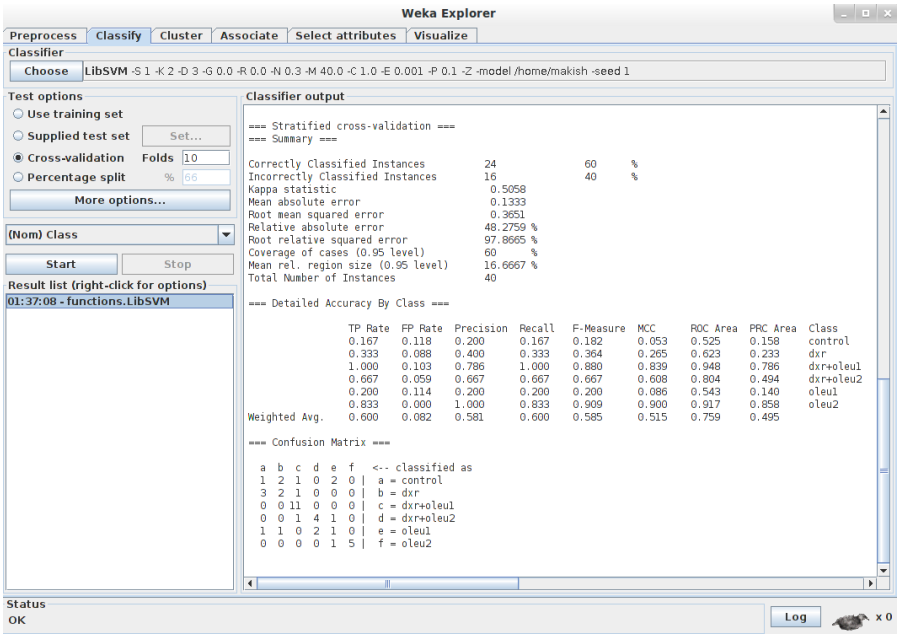
Cluster για μη εποπτευόμενη μάθηση.

Associate για την μάθηση κανόνων συσχέτισης.

Select attributes για την επιλογή μεταβλητών.

Visualize για την αναπαράσταση των δεδομένων.

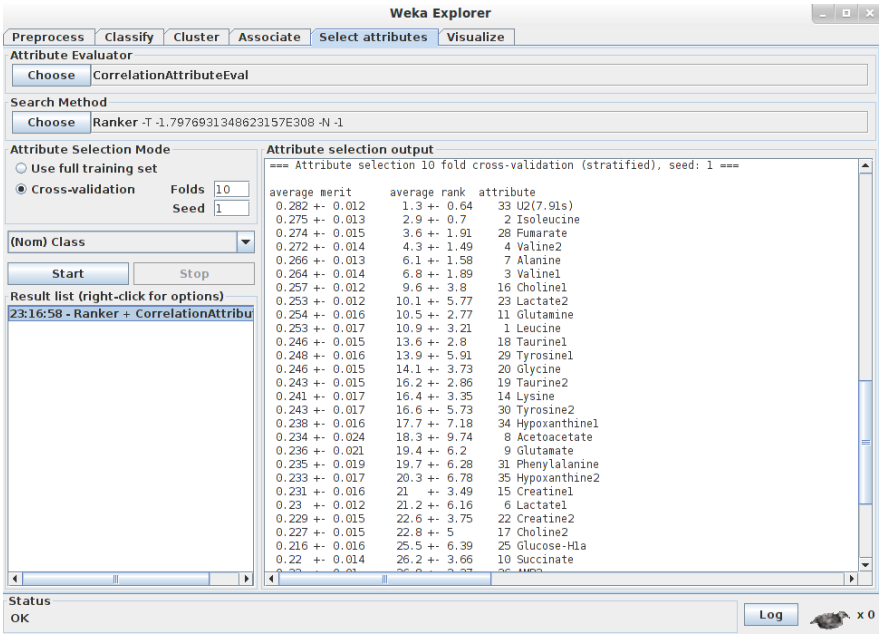
Στο σχήμα 4.4 διακρίνεται η καρτέλα της προεπεξεργασίας, για τα δεδομένα του Προβλήματος 2. Ενδεικτικά, παρέχονται λειτουργίες όπως διαγραφή ή διόρθωση δεδομένων, επισκόπηση της κατανομής τους σε ιστόγραμμα και εφαρμογή διαφόρων εποπτευόμενων ή μη φίλτρων (κανονικοποίηση, PCA, δειγματοληψία, αντιμετώπιση αγνώστων ή ακραίων τιμών και πολλά ακόμα). Για παράδειγμα, η ταυτότητα των δειγμάτων μπορεί να επηρεάσει τους αλγορίθμους μάθησης, οπότε είναι καλό να αφαιρεθεί. Αυτό μπορεί να γίνει επιλέγοντας τη σχετική μεταβλητή στην αριστερή λίστα και πατώντας `Remove`. Επιλέγοντας κάποια μεταβλητή μπορούμε επίσης να δούμε την κατανομή της στα δείγματα, σε σχέση με την κλάση τους. Σε κάποια (συνήθως μικρά) συστήματα, αυτό μπορεί να οδηγήσει σε απλούς κανόνες διαχωρισμού των κλάσεων. Η εφαρμογή ενός «φίλτρου» μπορεί να γίνει επιλέγοντάς το από το κουμπί `Choose`, ρυθμίζοντάς το πατώντας πάνω στη γραμμή με το όνομα και τις παραμέτρους του και πατώντας `Apply`.



Σχήμα 4.5: WEKA Explorer - Καρτέλα Classify. Αποτελέσματα ελέγχου ενός SVM μοντέλου για τα δεδομένα του Προβλήματος 2.

Η καρτέλα **Classify** φαίνεται στο σχήμα 4.5 και είναι αυτή που μας απασχόλησε περισσότερο στα πλαίσια αυτής της εργασίας. Στην κορυφή της οθόνης επιλέγεται ο αλγόριθμος μάθησης και ρυθμίζεται πατώντας στη γραμμή στην οποία εμφανίζεται το όνομα και οι παράμετροί του. Μια αρκετά πλήρη λίστα των παρεχόμενων αλγορίθμων μπορείτε να βρείτε στον πίνακα 6.3 της σελίδας 82. Παρέχονται ακόμα meta-learners για συλλογική μάθηση, όπως voting, bagging, stacking, boost και άλλοι. Από κάτω, στην αριστερή στήλη, επιλέγουμε τη μέθοδο ελέγχου του μοντέλου και ακριβώς από κάτω τη μεταβλητή η οποία αντιστοιχεί στην κλάση. Πατώντας το **start** ξεκινάει η διαδικασία εκπαίδευσης και ελέγχου και εμφανίζονται σταδιακά τα αποτελέσματα σε μορφή κειμένου στα δεξιά. Στο μενού **More options** μπορούμε να τροποποιήσουμε τις πληροφορίες που εμφανίζονται, όπως π.χ. να προσθέσουμε στα αποτελέσματα τις προβλέψεις για κάθε δείγμα σε κάθε υποσύνολο ελέγχου. Οι χρόνοι εκτέλεσης για το προβλήματα που εξετάσαμε κυμαίνονται από κλάσματα του δευτερολέπτου έως λίγα δευτερόλεπτα σε σύγχρονο προσωπικό Η/Υ.

Την ίδια μορφή έχει και η καρτέλα **Cluster** που επιτρέπει το χωρισμό των δεδομένων σε στοιβάδες, παραλείποντας μεταβλητές όπως η κλάση κάθε δείγματος. Παρόμοια αλλά απλούστερη μορφή έχει και η καρτέλα **Associate** για την εξαγωγή κανόνων συσχέτισης μεταξύ των μεταβλητών, χωρίς να παρέχονται εργαλεία ελέγχου αυτών των κανόνων. Η καρτέλα **Select Attributes** που φαίνεται στο σχήμα 4.6 δίνει τη δυνατότητα για την επιλογή μεταβλητών με διάφορες μεθόδους αξιολόγησης και έρευνας. Τέλος, η καρτέλα **Visualize** δίνει τη δυνατότητα δισδιάστατης απεικόνισης των δεδομένων σε διαγράμματα όπως αυτό του σχήματος 4.7 για εποπτική παρατήρηση των δεδομένων.



Σχήμα 4.6: WEKA Explorer - Καρτέλα Select Attributes. Επιλογή μεταβλητών για τα δεδομένα του Προβλήματος 2.

Σχήμα 4.7: WEKA Explorer - Καρτέλα Visualize. 2D αναπαράσταση των δεδομένων του Προβλήματος 2 για δύο από τις επιλεγμένες μεταβλητές.

4.3 LibSVM

Η LibSVM [10] είναι μια υλοποίηση αλγορίθμων Μηχανών Διανυσμάτων Υποστήριξης (βλ. ενότητα 3.4.9), τόσο για ταξινόμηση (αλγόριθμοι C-SVC, nu-SVC), για παλινδρόμηση (epsilon-SVR, nu-SVR) και για πρόβλεψη κατανομής (one-class SVM). Πολύ σημαντικό χαρακτηριστικό της σε σχέση με άλλες υλοποιήσεις είναι ότι μπορεί να διαχειριστεί προβλήματα πολλών κλάσεων. Αναπτύσσεται σε C/C++ από τους Chih-Chung Chang και Chih-Jen Lin και διατίθεται ελεύθερα για όλες τις βασικές πλατφόρμες. Μπορεί να λειτουργήσει αυτόνομα, σε συνεργασία με Octave/MATLAB™, με WEKA καθώς και με Java, R, Python, Ruby, SciLab και πολλά άλλα περιβάλλοντα.

Στη WEKA μπορεί να ενσωματωθεί χρησιμοποιώντας τον wrapper των EL-Manzalawy κ.α. [11]. Η διαδικασία απλοποιείται αν χρησιμοποιηθεί η έκδοση 3.8 της WEKA (αυτή τη στιγμή βρίσκεται υπό ανάπτυξη και συναντάται ως 3.7.10) η οποία διαθέτει package manager. Από το βασικό παράθυρο της WEKA (`gui chooser`): `Tools > Package manager > LibSVM > Install`. Στη συνέχεια μπορεί να βρεθεί ανάμεσα στους άλλους αλγορίθμους, στην κατηγορία `functions`.

Για Octave/MATLAB™ παρέχονται εγγενώς αρχεία μορφής `.mex` τα οποία μπορούν να μεταγλωτιστούν με την εντολή `make octave` σε Octave ή με την εντολή `make` σε MATLAB, αφού πρώτα έχει επιλεγεί το directory που περιέχει τα `.mex` αρχεία. Σε MATLAB σε Windows ενδέχεται να πρέπει πρώτα να επιλεγεί ένας mex compiler. Αυτό μπορεί να γίνει με την εντολή `mex -setup`. Περισσότερες πληροφορίες για την εγκατάσταση μπορούν να αντληθούν από το README αρχείο στον υποφάκελο matlab της LibSVM.

Τα αρχεία που χρησιμοποιούνται από τη LibSVM πρέπει να περιέχουν τα δεδομένα στην εξής μορφή:

```
<label> <index1>:<value1> <index2>:<value2> ...
```

όπου `label` είναι ένας ακέραιος που υποδεικνύει την κλάση του δείγματος, `<index>` ο δείκτης της κάθε μεταβλητής και `<value>` η τιμή της. Η μορφή του αρχείου δεδομένων μπορεί να ελεγχθεί με το παρεχόμενο εργαλείο `checkdata.py`.

Παρέχονται τρία βασικά προγράμματα, τα `svm-scale`, `svm-train` και `svm-predict`. Το `svm-scale` αναλαμβάνει την κλιμάκωση των δεδομένων στο επιθυμητό εύρος (προεπιλεγμένο είναι το $[-1, 1]$), επιστρέφει ένα αρχείο της ίδιας μορφής, με τα τροποποιημένα δεδομένα και (στην έκδοση 3.17) καλείται ως εξής:

```
svm-scale [options] data_filename
```

με επιλογές:

```
-l lower : x scaling lower limit (default -1)
-u upper : x scaling upper limit (default +1)
-y y_lower y_upper : y scaling limits (default: no y scaling)
```

```
-s save_filename : save scaling parameters to save_filename
-r restore_filename : restore scaling parameters from restore_filename
```

Το `svm-train` αναλαμβάνει την εκπαίδευση ενός μοντέλου SVM (προαιρετικά με cross-validation), καλείται ως εξής:

```
 svm-train [options] training_set_file [model_file]

-s svm_type : set type of SVM (default 0)
  0 -- C-SVC (multi-class classification)
  1 -- nu-SVC (multi-class classification)
  2 -- one-class SVM
  3 -- epsilon-SVR (regression)
  4 -- nu-SVR (regression)

-t kernel_type : set type of kernel function (default 2)
  0 -- linear: u'*v
  1 -- polynomial: (gamma*u'*v + coef0)^degree
  2 -- radial basis function: exp(-gamma*|u-v|^2)
  3 -- sigmoid: tanh(gamma*u'*v + coef0)
  4 -- precomputed kernel (kernel values in training_set_file)

-d degree : degree in kernel function (default 3)
-g gamma : gamma in kernel function (default 1/num_features)
-r coef0 : coef0 in kernel function (default 0)

-c cost : the parameter C of C-SVC, epsilon-SVR, and nu-SVR (default 1)
-n nu : the parameter nu of nu-SVC, one-class SVM, and nu-SVR (default 0.5)
-p epsilon : the epsilon in loss function of epsilon-SVR (default 0.1)

-m cachesize : cache memory size in MB (default 100)
-e epsilon : tolerance of termination criterion (default 0.001)
-h shrinking : whether to use the shrinking heuristics, 0 or 1 (default 1)
-b probability_estimates : whether to train a SVC or SVR model
   for probability estimates, 0 or 1 (default 0)
-wi weight : the parameter C of class i to weight*C, for C-SVC (default 1)
-v n: n-fold cross validation mode
-q : quiet mode (no outputs)
```

και επιστρέφει ένα αρχείο με το παραγόμενο μοντέλο και την ακρίβεια του μοντέλου με cross validation, εάν έχει οριστεί η παράμετρος `-v`.

Το `svm-predict` ελέγχει ένα μοντέλο με ένα σύνολο ελέγχου και επιστρέφει ένα αρχείο με τις προβλέψεις για κάθε δείγμα. Καλείται ως εξής:

```
svm-predict [options] test_file model_file output_file
```

με μοναδική επιλογή την:

```
-b probability_estimates: whether to predict probability
   estimates, 0 or 1 (default 0); for one-class SVM only 0 is supported
```

Στη WEKA, οι παράμετροι που χρησιμοποιούνται συνδέονται με τις παραμέτρους της svm-train και της svm-predict όπως φαίνεται στον πίνακα 4.1. Η διαδικασία του cross validation σε αυτή την περίπτωση διαχειρίζεται από την WEKA.

Πίνακας 4.1: Αντιστοιχία παραμέτρων WLSVM - LibSVM

WLSVM	-S	-K	-D	-G	-R	-N	-M	-C	-E	-P	-H	-B
LibSVM	-s	-t	-d	-g	-r	-n	-m	-c	-e	-p	-h 0	-b

Στο Octave/MATLAB™, παρέχονται οι εξής συναρτήσεις: `libsvmread`, `libsvmwrite`, `svmtrain`, `svmpredict`. Η `libsvmread` διαβάζει ένα αρχείο της μορφής που περιγράψαμε και αντιστοιχίζει τα δεδομένα σε δύο πίνακες: ένα διάνυσμα για τις κλάσεις των δειγμάτων και έναν δισδιάστατο πίνακα για τις τιμές των μεταβλητών τους. Καλείται ως εξής:

```
[label_vector, instance_matrix] = libsvmread('data.txt');
```

Η `libsvmwrite` δημιουργεί ένα τέτοιο αρχείο:

```
libsvmwrite('data.txt', label_vector, instance_matrix)
```

Η `svmtrain` καλείται ως εξής:

```
model = svmtrain(training_label_vector, training_instance_matrix
                [, 'libsvm_options']);
```

όπου `libsvm_options` ένα character string ίδιας μορφής με τις επιλογές του αυτόνομου προγράμματος `svm-train`. Εάν χρησιμοποιηθεί cross validation, τότε αυτή επιστρέφει απλώς μια τιμή: την ακρίβεια του μοντέλου. Διαφορετικά, επιστρέφει ένα μοντέλο το οποίο μπορεί να χρησιμοποιηθεί σε μελλοντικές προβλέψεις. Αντιστοίχως, η `svmpredict` μπορεί να κληθεί με τους εξής τρόπους:

```
[predicted_label, accuracy, decision_values/prob_estimates] =
   svmpredict(testing_label_vector, testing_instance_matrix, model
              [, 'libsvm_options']);

[predicted_label] = svmpredict(testing_label_vector,
                   testing_instance_matrix, model [, 'libsvm_options']);
```

Εκτός από τα αρχεία README που συνοδεύουν τη LibSVM, στην ιστοσελίδα της υπάρχει ένας πολύ χρήσιμος εισαγωγικός οδηγός χρήσης με παραδείγματα, συνηθισμένα λάθη και ειδικές περιπτώσεις [26]. Υπάρχουν επίσης αρκετά δημοσιευμένα datasets για δοκιμή.

4.4 Κώδικας επιλογής μεταβλητών

Για την μείωση των μεταβλητών του συστήματος αναπτύχθηκε ένας κώδικας σε GNU Octave (συμβατός με MATLAB) που υλοποιεί έναν γενετικό αλγόριθμο, χρησιμοποιώντας την LibSVM για την εκπαίδευση SVM μοντέλων. Μπορείτε να τον βρείτε στο παράρτημα Β΄. Το κεντρικό script, στο οποίο φαίνεται η δομή του γενετικού αλγορίθμου, είναι το `GeneticLibSVM`. Οι διαδικασίες της μάθησης, της επιλογής χρωμοσωμάτων και του συνδυασμού τους έχουν υλοποιηθεί ως ξεχωριστές συναρτήσεις ώστε να είναι ευκολότερα τροποποιήσιμες. Ολόκληρος ο αλγόριθμος επαναλαμβάνεται πολλές φορές με το script `logGeneticLibSVM`.

4.4.1 Βασικό script GeneticLibSVM

Ως είσοδοι απαιτούνται:

- Το σύνολο των (μετασχηματισμένων με την `svm-scale`) δεδομένων με `nInst` δείγματα και `nAttr` μεταβλητές, το οποίο αποθηκεύεται στο διάνυσμα `Labels` και στον πίνακα `Instances`.
- Το character string με τις παραμέτρους της `svmtrain`, το οποίο αποθηκεύεται στη μεταβλητή `svmconfig`.
- Οι παράμετροι του γενετικού αλγορίθμου: πλήθος των χρωμοσωμάτων στον πληθυσμό (`nChrom`), μέγιστο πλήθος γενεών το οποίο χρησιμοποιείται ως κριτήριο τερματισμού (`maxGenerations`), πιθανότητα μετάλλαξης κάθε γονιδίου (`mutProb`) και πιθανότητα εμφάνισης της τιμής 1 για κάθε γονίδιο του αρχικού πληθυσμού (`initGeneProb`).

Αφού καθαριστεί το workspace από μεταβλητές προηγούμενων επαναλήψεων του αλγορίθμου, δημιουργείται ένας τυχαίος αρχικός πληθυσμός (πίνακας `Chromosomes`). Σε όλες τις γενεές το πλήθος τους διατηρείται σταθερό. Αφού δημιουργηθεί ο αρχικός πληθυσμός, κάθε χρωμόσωμα πολλαπλασιάζεται στοιχείο-στοιχείο με κάθε γραμμή του πίνακα δειγμάτων και δημιουργείται ο πίνακας `LocalInstances`, ο οποίος διαφέρει από τον γενικό πίνακα των δειγμάτων ως προς το ότι έχει μηδενικές στήλες στις μεταβλητές που δεν έχουν επιλεγεί από το υπό εξέταση χρωμόσωμα. Με αυτόν εκπαιδεύεται ένα SVM μοντέλο και η ακρίβειά του καταχωρείται στην δεύτερη στήλη του πίνακα `ChromScore`, στη γραμμή που αντιστοιχεί στο χρωμόσωμα από το οποίο προήλθε. Η πρώτη στήλη φιλοξενεί τις ταυτότητες των χρωμοσωμάτων.

Αφού σχηματιστεί και ελεγχθεί ο αρχικός πληθυσμός, ξεκινάει ο κύριος βρόχος του αλγορίθμου, ο οποίος τερματίζεται όταν συμπληρωθούν `maxGenerations` επαναλήψεις. Αρχικά, με τη συνάρτηση `ruletteConstruct` κατασκευάζεται ο «τροχός ρουλέτας» για την

επιλογή των χρωμοσωμάτων προς αναπαραγωγή. Πρέπει να επιλεχθούν συνολικά `nChrom` χρωμοσώματα, δύο κάθε φορά, τα οποία θα δώσουν ισάριθμους απογόνους. Το πλήθος των απογόνων που έχουν παραχθεί από την τρέχουσα γενιά καταχωρείται στην μεταβλητή `counter`. Στη συνέχεια, επιλέγονται δύο χρωμοσώματα (τα `i1` και `i2`) με την συνάρτηση `ruletteSelect`. Επιλέγεται ένα τυχαίο σημείο κοπής (`point`) και τα χρωμοσώματα αναπαράγονται με την συνάρτηση `reproduce`. Μετά την αναπαραγωγή, λαμβάνει χώρα η μετάλλαξη των χρωμοσωμάτων. Κάθε γονίδιο κάθε ενός από τους δύο απογόνους αλλάζει τιμή (από 0 σε 1 ή το αντίστροφο) με πιθανότητα `nMut`. Οι απόγονοι ελέγχονται για την ακρίβεια του μοντέλου που δίνουν και αυτή αποθηκεύεται στον πίνακα `newChromScore`. Η πρώτη στήλη του περιέχει την ταυτότητα κάθε χρωμοσώματος και η δεύτερη την ακρίβεια του μοντέλου που προκύπτει από αυτό. Οι απόγονοι μπαίνουν έπειτα στον πίνακα της νέας γενιάς (`newGeneration`) είτε αρχικοποιώντας τον (αν είναι οι πρώτοι απόγονοι που παράγονται) είτε επεκτείνοντάς τον.

Μετά την ολοκλήρωση της παραγωγής μιας νέας γενιάς, η «νέα» γενιά γίνεται τρέχουσα και η προηγούμενη χάνεται. Όταν παραχθούν `maxGenerations` γενεές ο κύριος βρόχος σταματάει και επιστρέφονται ο τρέχων πληθυσμός χρωμοσωμάτων με τα αντίστοιχα scores. Στη συνέχεια, υπολογίζονται οι επικρατούσες τιμές των γονιδίων και δημιουργείται ένα χρωμόσωμα με αυτές (`genome`). Με αυτό δημιουργείται ένα μοντέλο και η ακρίβεια αυτού αποθηκεύεται στη μεταβλητή `genomeScore`. Το script επιστρέφει επίσης το πλήθος των μεταβλητών που έχουν επιλεγεί στη μεταβλητή `numAttributesUsed`.

4.4.2 Συναρτήσεις

Το GeneticLibSVM script χρησιμοποιεί τις εξής συναρτήσεις:

svmTrainAndScore για την εκπαίδευση και επαλήθευση του μοντέλου. Στην υλοποίηση αυτή απλώς καλείται η συνάρτηση svmtrain με τα ίδια ορίσματα, ωστόσο μπορεί εύκολα να τροποποιηθεί.

ruletteConstruct για την κατασκευή του «τροχού ρουλέτας», δηλαδή την αντιστοίχιση των χρωμοσωμάτων σε διαδοχικά υποδιαστήματα του $[0, 1]$, σύμφωνα με την ακρίβεια των μοντέλων που παράγουν.

ruletteSelect για την επιλογή χρωμοσωμάτων δεδομένου ενός «τροχού ρουλέτας».

reproduce για τη σταυρωτή αναπαραγωγή των χρωμοσωμάτων.

Η `ruletteConstruct` δέχεται ως μοναδικό όρισμα τον πίνακα `ChromScore` με την ακρίβεια (score) του μοντέλου που παράγει κάθε χρωμόσωμα. Στη συνέχεια υπολογίζει το άθροισμα των scores (`ScoreSum`) και διαιρεί το καθένα με το άθροισμά τους, δημιουργώντας έτσι τον πίνακα `ChromScoreNorm`, ο οποίος στη θέση του κάθε score έχει το κλασματικό score. Οι γραμμές αυτού του πίνακα αναδιατάσσονται κατά φθίνουσα σειρά ακρίβειας. Έπειτα, αθροίζονται σταδιακά τα scores και για κάθε i χρωμόσωμα του πίνακα `ChromScoreNorm` υπολογίζεται το σωρευτικό (accumulated) score για τα χρωμοσώματα 1 έως i.

Η `ruletteSelect` δέχεται ως όρισμα την έξοδο της `ruletteConstruct`. Παράγει έναν τυχαίο αριθμό `R` και επιλέγει το πρώτο χρωμόσωμα το οποίο έχει σωρευτικό score πάνω από `R`, επιστρέφοντας την ταυτότητά του στη scalar μεταβλητή `selectedChrom`.

Η `reproduce` δέχεται ως ορίσματα το σημείο κοπής των χρωμοσωμάτων (`point`), τις ταυτότητες των δύο χρωμοσωμάτων που θα αναπαραχθούν (`i1` και `i2`) και τον πίνακα με τον τρέχοντα πληθυσμό χρωμοσωμάτων (`Chromosomes`). Επιστρέφει έναν $2 \times$ `nAttr` πίνακα με τους δύο απογόνους που προέκυψαν από την διασταύρωση.

4.4.3 Πολλαπλές επανεκκινήσεις και καταγραφή

Ένας γενετικός αλγόριθμος δεν παράγει πάντα ακριβώς την ίδια λύση, λόγω της τυχαιότητας που υπεισέρχεται στην κατασκευή του αρχικού πληθυσμού και στην επιλογή των χρωμοσωμάτων. Για το λόγο αυτό, συνήθως εκτελείται πολλές φορές, ξεκινώντας από διαφορετικό κάθε φορά πληθυσμό, ώστε να ελεγχθεί η σύγκλισή του. Η διαδικασία αυτή γίνεται με το script `logGeneticLibSVM`, το οποίο δέχεται ως είσοδο μονάχα το επιθυμητό πλήθος επανεκκινήσεων (μεταβλητή `logBuffer`).

Το βασικό script εκτελείται `logBuffer` φορές και καταγράφονται οι μεταβλητές `genome`, `genomeScore`, `numAttributesUsed`. Στη συνέχεια, υπολογίζεται πόσο συχνά εμφανίζεται το κάθε γονίδιο, δηλαδή πόσες φορές από τις `logBuffer` εμφάνισε την τιμή 1. Οι συχνότητες αυτές αποθηκεύονται στον πίνακα `logGenomeFreq`, η πρώτη στήλη του οποίου φιλοξενεί τη θέση-ταυτότητα κάθε μεταβλητής και η δεύτερη στήλη τη συχνότητα εμφάνισής της. Οι γραμμές του πίνακα αναδιατάσσονται κατά φθίνουσα σειρά συχνότητας.

Το script καταγραφής εμφανίζει αποτελέσματα για το πλήθος των μεταβλητών που επιλέχθηκαν σε κάθε εκτέλεση του αλγορίθμου, το μέσο πλήθος αυτών, το `genome` για κάθε εκτέλεση, το score του, ένα μέσο score καθώς και τον πίνακα με τις συχνότητες επιλογής των μεταβλητών.

4.4.4 Έλεγχος της ακρίβειας για διαφορετικό πλήθος μεταβλητών

Το script logGeneticLibSVM παρήγαγε έναν πίνακα με τις μεταβλητές του συστήματος, διαταγμένες κατά φθίνουσα συχνότητα εμφάνισης. Μπορούμε να υποθέσουμε ότι όσο συχνότερα επιλέγεται μια μεταβλητή, τόσο μεγαλύτερη σημασία θα έχει για την κατασκευή καλών μοντέλων. Σε ένα πρόβλημα επιλογής μεταβλητών στόχος είναι, εκτός από την κατασκευή ενός μοντέλου με υψηλή ακρίβεια, η επιλογή των ελάχιστων δυνατών μεταβλητών. Το script `testScript` ξεκινάει με μόνο την πιο συχνά εμφανιζόμενη μεταβλητή και σταδιακά χτίζει και ελέγχει μοντέλα με τις 2, τις 3, έως τις `maxTestAttr` μεταβλητές. Στη συνέχεια, εμφανίζονται τα αποτελέσματα κάθε δοκιμής και ο χρήστης μπορεί να επιλέξει το συνδυασμό πλήθους μεταβλητών και ακρίβειας που τον ικανοποιεί περισσότερο. Αξίζει να σημειωθεί ότι μπορεί να παρατηρηθεί ολικό μέγιστο ακρίβειας σε ενδιάμεση τιμή πλήθους μεταβλητών, κάτι που ενισχύει τα κίνητρα για την επιλογή ενός μοντέλου με λιγότερες από τις διαθέσιμες μεταβλητές.

Κεφάλαιο 5

Πρόβλημα 1: AKI

Το πρώτο πρόβλημα που εξετάζεται προέρχεται από δημοσιευμένη εργασία των Zacharias et al. (2012) [3]. Η φυσική εικόνα του προβλήματος είναι η εξής: έπειτα από χειρουργικές επεμβάσεις στην καρδιά, ενδέχεται να προκληθεί οξεία νεφρική βλάβη (Acute Kidney Injury) σε ορισμένους ασθενείς. Θα ήταν ιδιαίτερα σημαντικό αν μπορούσαμε να προλάβουμε την σοβαρή αυτή παρενέργεια στα πρώτα της στάδια, ώστε να αντιμετωπιστεί κατάλληλα και από ότι φαίνεται στη σχετική εργασία, η Μεταβολομική μπορεί να βοηθήσει ιδιαίτερα σε αυτό, καθώς εντοπίζονται ξεκάθαρες αλλαγές στον μεταβολισμό των ατόμων που τελικά παρουσιάζουν AKI.

5.1 Δεδομένα

Στην σχετική εργασία αναλύθηκαν με NMR δείγματα ούρων από 106 ανθρώπους, από τους οποίους οι 34 εμφάνισαν AKI (στο εξής θα αναφέρονται ως «ασθενείς») και οι 72 παρέμειναν υγιείς. Δείγματα ούρων ελήφθησαν την ημέρα πριν από την επέμβαση, καθώς και 4 και 24 ώρες μετά από αυτήν. Στην περίπτωσή μας θα ασχοληθούμε με τα δείγματα 24 ώρες μετά την επέμβαση, τα οποία αναφέρεται ότι έδωσαν την καλύτερη συνολική επιτυχία. Τα δείγματα που ελήφθησαν 4 ώρες μετά είχαν πολύ υψηλές συγκεντρώσεις σε πρόσθετες ουσίες που χρησιμοποιήθηκαν κατά τη διάρκεια της επέμβασης και έδιναν χαμηλότερη επιτυχία στις προβλέψεις. Τα δεδομένα είναι ελεύθερα διαθέσιμα στη βάση δεδομένων MetaboLights [4] και αποτελούνται από κανονικοποιημένες τιμές (Quantile Normalization) 701 χαρακτηριστικών (περιοχές του φάσματος NMR) για τους 106 ασθενείς. Επιπλέον, το σχετικό αρχείο περιέχει τον κωδικό κάθε δείγματος και την κλάση του (`Biopsy kidney normal` ή `Acute Kidney Injury`). Στη δημοσιευμένη εργασία, έγινε ταξινόμηση με SVM και για τα δείγματα 24h μετά την επέμβαση επετεύχθη συνολική επιτυχία 76%.

5.2 Ανάλυση με μεμονωμένους αλγορίθμους

Το αρχείο δεδομένων μετατράπηκε από την παρεχόμενη μορφή στην κατάλληλη μορφή `arff` που δέχεται η WEKA. Τα δεδομένα ανακατεύθηκαν και επειδή, όπως παρατηρήθηκε αργό-

τερα, ορισμένοι αλγόριθμοι επηρεάζονται από την ταυτότητα κάθε δείγματος, αυτή αφαιρέθηκε. Στη συνέχεια, εξετάστηκαν όλοι οι διαθέσιμοι αλγόριθμοι ταξινόμησης, έπειτα από προσεκτική επιλογή των παραμέτρων του καθενός ώστε να μεγιστοποιείται η συνολική επιτυχία. Σε περιπτώσεις που για διαφορετικές παραμέτρους μπορούσε να επιτευχθεί σχεδόν η ίδια συνολική επιτυχία με διαφορετική κατανομή επιτυχίας στις κλάσεις, επιλέχθηκε το σύνολο παραμέτρων που προέβλεπε καλύτερα την κλάση `AKI`. Σε κάθε περίπτωση χρησιμοποιήθηκε 10-fold Cross Validation. Οι χρόνοι εκτέλεσης κυμαίνονταν από κλάσματα του δευτερολέπτου έως μερικά δευτερόλεπτα ή ελάχιστα λεπτά για τους περισσότερους αλγορίθμους. Ο αλγόριθμος MultilayerPerceptron, για τις υψηλότερες τιμές της παραμέτρου «εποχών» `N` χρειάστηκε χρόνο της τάξης δεκάδων λεπτών. Σημειώνεται ότι, μεταξύ άλλων λόγω της χρήσης cross-validation, η εκπαίδευση θα μπορούσε να επιταχυνθεί σημαντικά εάν οι απαραίτητες μέθοδοι υλοποιούνταν έτσι ώστε να εκμεταλλεύονται διαθέσιμες υποδομές παράλληλης επεξεργασίας.

Τα αποτελέσματα προβλέψεων για τα μοντέλα που εκπαιδεύτηκαν φαίνονται συγκεντρωτικά στον πίνακα 5.1. Λόγω του ότι στο πρόβλημα εμφανίζονται μονάχα δύο κλάσεις και του ότι κανένας από τους αλγορίθμους δεν αφήνει μη ταξινομημένα δεδομένα, είναι εύκολο να παραχθούν όλοι οι confusion matrices μονάχα από αυτόν τον πίνακα. Ένα παράδειγμα φαίνεται στον πίνακα 5.2. Στον πίνακα αυτόν, όπως και σε όλους τους confusion matrices που εμφανίζονται σε αυτήν την εργασία, οι γραμμές αντιστοιχούν στις «πραγματικές κλάσεις» και οι στήλες στις «προβλεπόμενες κλάσεις». Έτσι, στον πίνακα 5.2 65 δείγματα της κλάσης `normal` έχουν αποδοθεί σωστά σε αυτήν, ενώ 7 δεν αναγνωρίστηκαν σωστά ως `normal` αλλά αποδόθηκαν εσφαλμένα στην κλάση `AKI`. Το άθροισμα των στοιχείων της γραμμής πρέπει να ισούται με το σύνολο των δεδομένων της κλάσης για τα οποία έγινε ταξινόμηση. Σε ειδικές περιπτώσεις, δεδομένα για τα οποία ένα μοντέλο δεν μπορεί να αποφανθεί με βεβαιότητα, μπορεί συνειδητά να αφεθούν αταξινόμητα, κάτι που βελτιώνει την ακρίβεια του μοντέλου, αλλά δεν δίνει απαντήσεις για ορισμένα δείγματα. Ένας τέτοιος πίνακας διαβάζεται και κατά στήλες: π.χ. συνολικά στην κλάση `AKI` αποδόθηκαν 26+7=33 δείγματα. Από αυτά όμως μόνο τα 26 ανήκουν όντως σε αυτήν. Άρα, η απόφαση ότι ένα νέο δείγμα ανήκει στην κλάση `AKI` θα έχει πιθανότητα 26/33=79% να είναι σωστή.

Πίνακας 5.1: Επιτυχία αλγορίθμων για το σύνολο και επιτυχία ως προς τις κλάσεις. a:`Biopsy kidney normal`, b:`Acute Kidney Injury`

Αλγόριθμος-Μοντέλο	**Επιτυχία**	**a(72)**	**a(%)**	**b(34)**	**b(%)**
Bayes					
BayesNet	75.5%	56	78%	24	71%
NaiveBayes	74.5%	55	76%	24	71%
NaiveBayesUpdateable	68.9%	55	76%	18	53%
Functions					
LibSVM	80.2%	63	88%	22	65%
Logistic	80.2%	61	85%	24	71%
MultilayerPerceptron (N=20)	71.7%	55	76%	21	62%
MultilayerPerceptron (N=100)	75.5%	58	81%	22	65%
MultilayerPerceptron (N=500)	76.4%	59	82%	22	65%
SGD	80.2%	61	85%	24	71%
SimpleLogistic	80.2%	63	88%	22	65%
SMO	77.4%	59	82%	23	68%
VotedPerceptron	77.4%	61	85%	21	62%
Lazy					
Ibk	71.7%	61	85%	15	44%
Kstar	*67.9%*	*72*	*100%*	*0*	*0%*
LWL (DecisionStump)	78.3%	53	74%	30	88%
Rules					
DecisionTable	81.1%	65	90%	21	62%
JRip	85.8%	65	90%	26	76%
OneR	66.0%	53	74%	17	50%
PART	78.3%	62	86%	21	62%
ZeroR	*67.9%*	*72*	*100%*	*0*	*0%*
Trees					
DecisionStump	79.2%	53	74%	31	91%
J48	81.1%	63	88%	23	68%
LMT	79.2%	62	86%	22	65%
RandomForest	77.4%	64	89%	18	53%
RandomTree	*72.6%*	*67*	*93%*	*10*	*29%*
REPTree	80.2%	58	81%	27	79%
Αλγόριθμος-Μοντέλο	**Επιτυχία**	**a(72)**	**a(%)**	**b(34)**	**b(%)**

Πίνακας 5.2: JRip: Confusion Matrix

	a	**b**	
a: Biopsy kidney normal	**65**	72-65=7	0.903
b: Acute Kidney Injury	34-26=8	**26**	0.765
Επιτυχία			85.8%

5.3 Ανάλυση με συνδυασμό αλγορίθμων

Έχοντας στη διάθεσή μας μια εικόνα για τη γενική συμπεριφορά των διαφόρων αλγορίθμων στο συγκεκριμένο πρόβλημα, μπορούμε να τους συνδυάσουμε, με στόχο τελικά να λάβουμε καλύτερα αποτελέσματα. Παρότι είναι διαθέσιμοι πολλοί meta-classifiers στη WEKA που μπορούν να συνδυάσουν αλγορίθμους με διάφορους τρόπους (βλ. ενότητα 6.3.2), στο πρόβλημα αυτό η διαδικασία του συμψηφισμού των αποτελεσμάτων των αλγορίθμων έγινε σε κατάλληλο λογιστικό φύλλο το οποίο δημιουργήθηκε στα πλαίσια της εργασίας.

Έπειτα από την εκτέλεση ενός αλγορίθμου, τα αποτελέσματα που εμφανίζονται στη WEKA αποθηκεύονται σε ένα αρχείο κειμένου, το οποίο μορφοποιείται κατάλληλα και στη συνέχεια εισάγεται ως νέο φύλλο στο LibreOffice Calc. Από τα αποτελέσματα ενδιαφέρουν μονάχα οι προβλέψεις ανά δείγμα. Μπορεί να έχουμε αφαιρέσει τις ταυτότητες των δειγμάτων, ωστόσο έχει ελεγχθεί, χρησιμοποιώντας ολόκληρα τα δεδομένα, ότι ο χωρισμός των δεδομένων κατά το cross-validation γίνεται κάθε φορά με τον ίδιο ακριβώς τρόπο. Στα αποτελέσματα ενδιαφέρει επίσης να εμφανίζεται η πιθανότητα για κάθε κλάση (αν υπολογίζεται), κάτι που μπορεί να ενεργοποιηθεί ως εξής: `More options... > Output predictions > [plain text] > (κλικ στο [plain text]) > outputDistribution`.

Αφού εισαχθούν τα αποτελέσματα των επιθυμητών μοντέλων στο βιβλίο εργασίας, επιλέγεται ποια από αυτά θα χρησιμοποιηθούν, αλλάζοντας μια δυαδική τιμή για το καθένα. Στη συνέχεια υπολογίζεται ο μέσος όρος της πιθανότητας. Οι δύο κλάσεις έχουν οριστεί ως 0 (`normal`) και 1 (`AKI`). Αν ο μέσος όρος είναι υψηλότερος από ένα κατώφλι (π.χ. 60%), τότε αντιστοιχίζεται στην κλάση 1. Αν είναι χαμηλότερος από ένα άλλο κατώφλι (π.χ. 40%), τότε αντιστοιχίζεται στην κλάση 0. Αν βρίσκεται ενδιάμεσα, τότε δεν λαμβάνεται καμία απόφαση. Αυτό μπορεί να αποφευχθεί θέτοντας τα δύο κατώφλια ίσα με 50%. Τα λάθη στις προβλέψεις αθροίζονται και υπολογίζεται η συνολική επιτυχία. Χρησιμοποιώντας τα παρεχόμενα εργαλεία βελτιστοποίησης (solver) μπορούμε να εντοπίσουμε τον συνδυασμό μοντέλων που μεγιστοποιεί τη συνολική ακρίβεια αλλάζοντας τις αντίστοιχες δυαδικές μεταβλητές. Μερικά ενδιαφέροντα παραδείγματα, με τιμή 50% και για τα δύο κατώφλια:

- με DecisionStump + REPTree + SMO λαμβάνουμε 84.0% (υψηλότερη)
- με REPTree + LibSVM λαμβάνουμε 84.9% (υψηλότερη)
- με REPTree + SMO λαμβάνουμε 77.4% (ίση με του SMO)
- με DecisionStump + JRip + REPTree + LibSVM λαμβάνουμε 86.8% (υψηλότερη)
- με DecisionStump + JRip + REPTree + SMO λαμβάνουμε 87.7% (υψηλότερη)

Όπως φαίνεται από τα παραδείγματα, μπορούμε να επιτύχουμε αρκετά υψηλότερη ακρίβεια με ένα συλλογικό μοντέλο σε σχέση με την ακρίβεια που επιτυγχάνει κάθε ένα από τα συμμετέχοντα μοντέλα. Παρατηρήστε επίσης ότι η προσθήκη του ίδιου αλγορίθμου δεν αυξάνει πάντοτε την ακρίβεια. Ο confusion matrix για το τελευταίο παράδειγμα φαίνεται στον πίνακα 5.3.

Πίνακας 5.3: Consensus: Confusion Matrix. Συνδυάζονται τα μοντέλα που προέκυψαν από τους DecisionStump, JRip, REPTree και SMO.

	a	**b**	
a: Biopsy kidney normal	**65**	7	0.903
b: Acute Kidney Injury	6	**28**	0.824
Επιτυχία			87.7%

5.4 Συζήτηση

Όπως φαίνεται στον πίνακα 5.1 σε αυτό το πρόβλημα υπάρχει έντονος ανταγωνισμός για τη θέση του «καλύτερου» αλγορίθμου. Επισημαίνεται ότι τα νούμερα που παρουσιάζονται αφορούν το συγκεκριμένο μοντέλο που παράχθηκε με συγκεκριμένες παραμέτρους από έναν αλγόριθμο και εφαρμόζεται σε αυτό το σύνολο δεδομένων με 10-fold cross-validation. Δεν θα πρέπει να ερμηνευτεί ως γενικό μέτρο αξιολόγησης των αλγορίθμων. Αυτό γίνεται εύκολα αντιληπτό συγκρίνοντας τα αποτελέσματα με αυτά που φαίνονται στον πίνακα 6.3 (σελ. 82) για το Πρόβλημα 2, όπου τα καταλληλότερα μοντέλα προκύπτουν από διαφορετικούς αλγορίθμους από ότι εδώ. Επίσης, είναι πιθανό ορισμένοι αλγόριθμοι να μπορούν να δώσουν μοντέλα με υψηλότερη ακρίβεια από αυτή που παρουσιάζεται εδώ, αν κανείς εμβαθύνει περισσότερο σε αυτούς.

Ως προς τη συνολική ακρίβεια, καλύτερα μοντέλα για αυτό το πρόβλημα φαίνεται να δίνει ο αλγόριθμος `JRip`. Μάλιστα, δίνει και την υψηλότερη ακρίβεια στην πρόβλεψη της κλάσης `normal`, μαζί με τον αλγόριθμο `DecisionTable` (και οι δύο στην κατηγορία `Rules`). Σημειώστε πως τα μοντέλα των αλγορίθμων που εμφανίζονται με πλάγιες γλυφές παρουσιάζουν ένα σημαντικό πρόβλημα: δεν προβλέπουν καθόλου, ή προβλέπουν ελάχιστα, τη μία από τις δύο κλάσεις, ενώ προβλέπουν φαινομενικά πολύ καλά την άλλη. Αν στεκόμασταν στη συνολική ακρίβεια ή στην ακρίβεια της κλάσης που προβλέπεται καλύτερα θα βγάζαμε λανθασμένα συμπεράσματα. Τα μοντέλα αυτά φαίνεται να επηρεάζονται κυρίως από την κατανομή των κλάσεων, δίνοντας «εύκολες» αποφάσεις με υψηλή συνολική ακρίβεια, οι οποίες όμως δεν έχουν κάποια αξία. Μάλιστα, ο αλγόριθμος `ZeroR` βασίζεται ακριβώς στην πλειοψηφούσα κλάση. Μεγάλο ενδιαφέρον παρουσιάζει επίσης ο αλγόριθμος `DecisionStump` καθώς δίνει (χωρίς καμία παράμετρο) την καλύτερη ακρίβεια για την κλάση `AKI`, δίνοντας ταυτόχρονα καλές τιμές για την ακρίβεια της κλάσης `normal` και για την συνολική ακρίβεια.

Σε αυτό το πρόβλημα τα δεδομένα είναι αρκετά και οι διαθέσιμες μεταβλητές πολύ περισσότερες. Επίσης, όπως φαίνεται και οπτικά στα φάσματα NMR [3], υπάρχουν σημαντικές διαφορές ανάμεσα στις δύο κλάσεις. Επομένως, μπορούν να διαχωριστούν χωρίς να χρειάζεται να εκπαιδευτούν ιδιαίτερα περίπλοκα μοντέλα. Εξ' άλλου, τα απλούστερα μοντέλα τείνουν πολλές φορές να συμπεριφέρονται καλύτερα στην πράξη, όταν και τα συστήματα που μελετώνται είναι σχετικά απλά. Παρότι στο (αρκετά πιο περίπλοκο) Πρόβλημα 2 θα δούμε ότι ο αλγόριθμος LibSVM παράγει με διαφορά καλύτερα μοντέλα από τους υπόλοιπους αλγορίθμους, οι οποίοι δεν αποδίδουν τόσο καλά όσο εδώ, στη συγκεκριμένη περίπτωση βρίσκεται αρκετά ψηλά αλλά όχι στην πρώτη θέση. Ειδικά ως προς την κλάση `AKI`, συμπεριφέρεται με παρόμοιο τρόπο με τους υπόλοιπους. Παρόλα αυτά, όλοι οι αλγόριθμοι της

κατηγορίας `Functions` δίνουν αρκετά καλά αποτελέσματα, χωρίς να παρουσιάζουν ακραία συμπεριφορά όπως πχ ο `RandomTree`.

Δεν πρέπει ωστόσο να εστιάζουμε μονάχα σε ένα ή δύο «καλύτερα» μοντέλα. Όπως είδαμε, μπορούμε συνδυάζοντας μοντέλα να δημιουργήσουμε ένα συναινετικό μοντέλο με ακρίβεια υψηλότερη από αυτή κάθε μοντέλου που συμμετέχει ή και υψηλότερη από άλλα μοντέλα τα οποία δεν συμμετέχουν. Αυτό είναι σημαντικό, μεταξύ άλλων, καθώς ενδέχεται ένα σύνολο από απλούστερα μοντέλα να εκπαιδεύεται πολύ πιο γρήγορα από ένα πιο περίπλοκο μοντέλο με υψηλότερη ακρίβεια.

Κεφάλαιο 6

Πρόβλημα 2: DXR

Το δεύτερο πρόβλημα που εξετάζεται προέρχεται από υπό μελέτη δεδομένα της Φαρμακευτικής Σχολής του Εθνικού Καποδιστριακού Πανεπιστημίου Αθηνών, της ερευνητικής ομάδας του καθηγητή Εμμ. Μικρού, τα οποία μας παραχωρήθηκαν ευγενικά. Η φυσική εικόνα του προβλήματος είναι η εξής: η αδριαμυκίνη (DXR) θεωρείται ένα από τα πιο αποτελεσματικά αντινεοπλασματικά φάρμακα εναντίον πολλών μορφών καρκίνου όπως λευχαιμία, σάρκωμα, όγκος του μαστού, των πνευμώνων και των ωοθηκών. Η χορήγησή της όμως επιφέρει αρκετές παρενέργειες, όπως μυελοτοξικότητα, ναυτία, εμετό, διάρροια, αλωπεκία και, το κυριότερο, καρδιοτοξικότητα. Στη σχετική διατριβή του κ. Κ. Ιωαννίδη [5], στην οποία μελετώνται τα ίδια δεδομένα, εξετάζεται αν η συγχορήγηση ολευρωπαΐνης (συστατικό των φύλλων της ελιάς) μπορεί να αναστείλει την καρδιοτοξικότητα που επιφέρει η αδριαμυκίνη, ενώ ενδιαφέρει επίσης και η επίδραση της ολευρωπαΐνης σε υγιή άτομα.

6.1 Δεδομένα

Τα δεδομένα μεταβολομικής που εξετάζονται αποτελούν φάσματα πυρηνικού μαγνητικού συντονισμού (NMR) υδρογόνου από εκχυλίσματα ιστού καρδιάς 40 επιμύων οι οποίοι χωρίζονται σε 6 γνωστές κλάσεις. Η κατανομή των ατόμων-δειγμάτων σε κλάσεις παρουσιάζεται στον πίνακα 6.1. Για κάθε άτομο διατίθενται εμβαδά 38 κορυφών NMR και η κλάση στην οποία αυτό ανήκει. Το σύνολο των δεδομένων προέρχεται από καθαρισμό εκτενέστερου συνόλου, από το οποίο έχουν αφαιρεθεί τα ακραία δείγματα (outliers) και έχουν προσαρμοστεί οι τιμές των εμβαδών σύμφωνα με την κορυφή του χρησιμοποιούμενου προτύπου (TSP). Για την ανάλυση χρησιμοποιήθηκαν τα καθαρισμένα δεδομένα, όπως δόθηκαν από το Τμήμα Φαρμακευτικής.

Η κλάση `control` περιέχει άτομα στα οποία δεν έχει χορηγηθεί καμία δραστική ουσία. Η κλάση `dxr` περιέχει άτομα που έχουν λάβει αδριαμυκίνη (DXR). Οι κλάσεις `oleu1` και `oleu2` περιέχουν άτομα που έχουν λάβει ολευρωπαΐνη σε δυο διαφορετικές δόσεις (χαμηλότερη και υψηλότερη, αντιστοίχως, με σχέση 1:2). Οι κλάσεις `dxr+oleu1` και `dxr+oleu2` περιέχουν άτομα που έχουν λάβει συνδυασμό των αντίστοιχων δραστικών ουσιών. Το βασικό ερώτημα είναι αν οι διαφορετικές κλάσεις διαχωρίζονται μεταξύ τους. Στην προσέγγισή μας, στόχος είναι η διάκριση των ατόμων που ανήκουν στις 6 κλάσεις με στόχο την αντιστοίχιση νέων,

Πίνακας 6.1: Κατανομή δειγμάτων σε κλάσεις

Κλάση	**Πλήθος ατόμων**
control	6
dxr	6
dxr+oleu1	11
dxr+oleu2	6
oleu1	5
oleu2	6
Σύνολο	40

άγνωστων ατόμων-δειγμάτων στις αντιπροσωπευτικότερες για αυτά κλάσεις.

6.2 Ανάλυση με το MetaboAnalyst

Για μια πρώτη επεξεργασία των δεδομένων χρησιμοποιήθηκε το online εργαλείο Metaboanalyst [7,8] και εξετάστηκε, με διάφορους τρόπους, η δυνατότητα διαχωρισμού των δεδομένων σε ομάδες.

Ως τύπος δεδομένων επιλέχθηκε `Concentrations`, τα δεδομένα κλιμακώθηκαν με την τεχνική `Pareto Scaling`, ενώ δεν εφαρμόστηκε κανονικοποίηση. Το αποτέλεσμα φαίνεται στο σχήμα 6.1. Δεν υπήρχαν άγνωστες τιμές. Υπενθυμίζεται ότι τα δεδομένα που χρησιμοποιούνται έχουν ήδη υποστεί «καθαρισμό» από outliers και από -για άλλους λόγους- προβληματικά δείγματα.

Ακολούθησε στατιστική επεξεργασία (μεταξύ άλλων ανάλυση συσχέτισης, PCA, PLS-DA). Στο σχήμα 6.2 φαίνεται η ιεραρχική εξάρτηση των διαφόρων ουσιών. Ως μετρική απόστασης έχει χρησιμοποιηθεί το `Pearson r`.

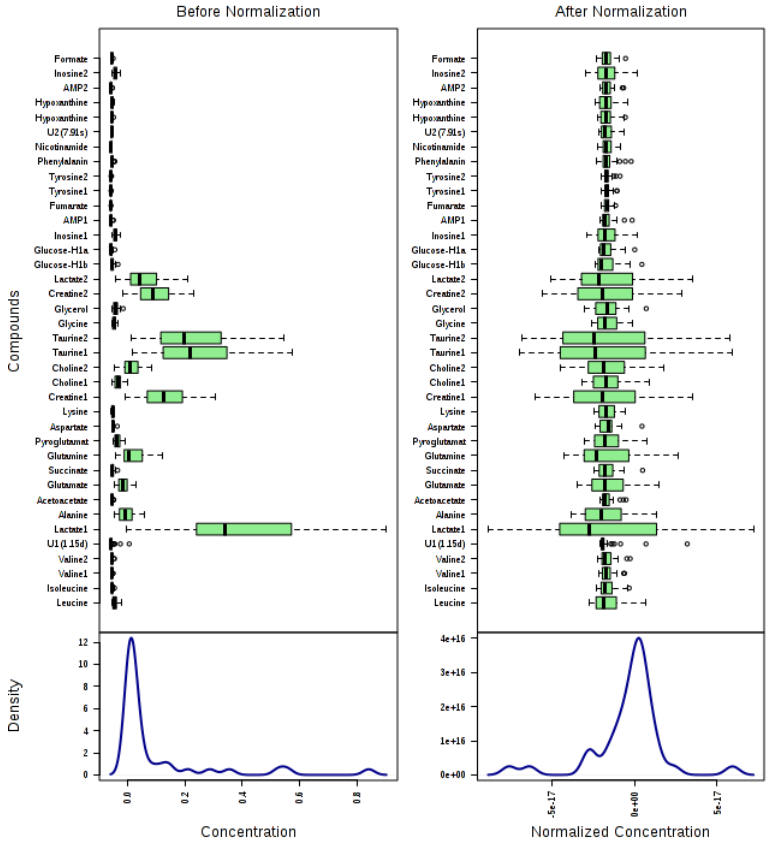
Η Ανάλυση Κυρίαρχων Συνιστωσών (Principal Components Analysis - PCA) για τα 5 κυριότερα συστατικά-κλειδιά φαίνεται στο σχήμα 6.3. Δεν φαίνεται να επιτυγχάνεται κάποιος αξιοσημείωτος διαχωρισμός για καμία κλάση. Ενδεικτικά, παρατίθενται μεγεθυμένα τα 2D διαγράμματα για κάποια Principal Components. Αντίστοιχη εικόνα παρουσιάζεται και στα υπόλοιπα διαγράμματα.

Αντίστοιχη εικόνα παρουσιάζεται και με την ανάλυση μερικών ελαχίστων τετραγώνων (Partial Least Squares - Discriminant Analysis ή PLS-DA). Η συνοπτική εικόνα φαίνεται στο σχήμα 6.6. Ενδεικτικά, δίνονται τα 2D score plots για τα συστατικά 3-1 (σχήμα 6.7) και 5-4 (σχήμα 6.8). Εκτός από έναν σχετικό διαχωρισμό των κλάσεων `control` και `oleu2` δεν διακρίνεται κάποιος άλλος ικανοποιητικός διαχωρισμός. Στο σχήμα 6.9 παρουσιάζεται ενδεικτικά η κατάταξη των 15 κυριότερων μεταβολιτών για την PLS-DA (component 1).

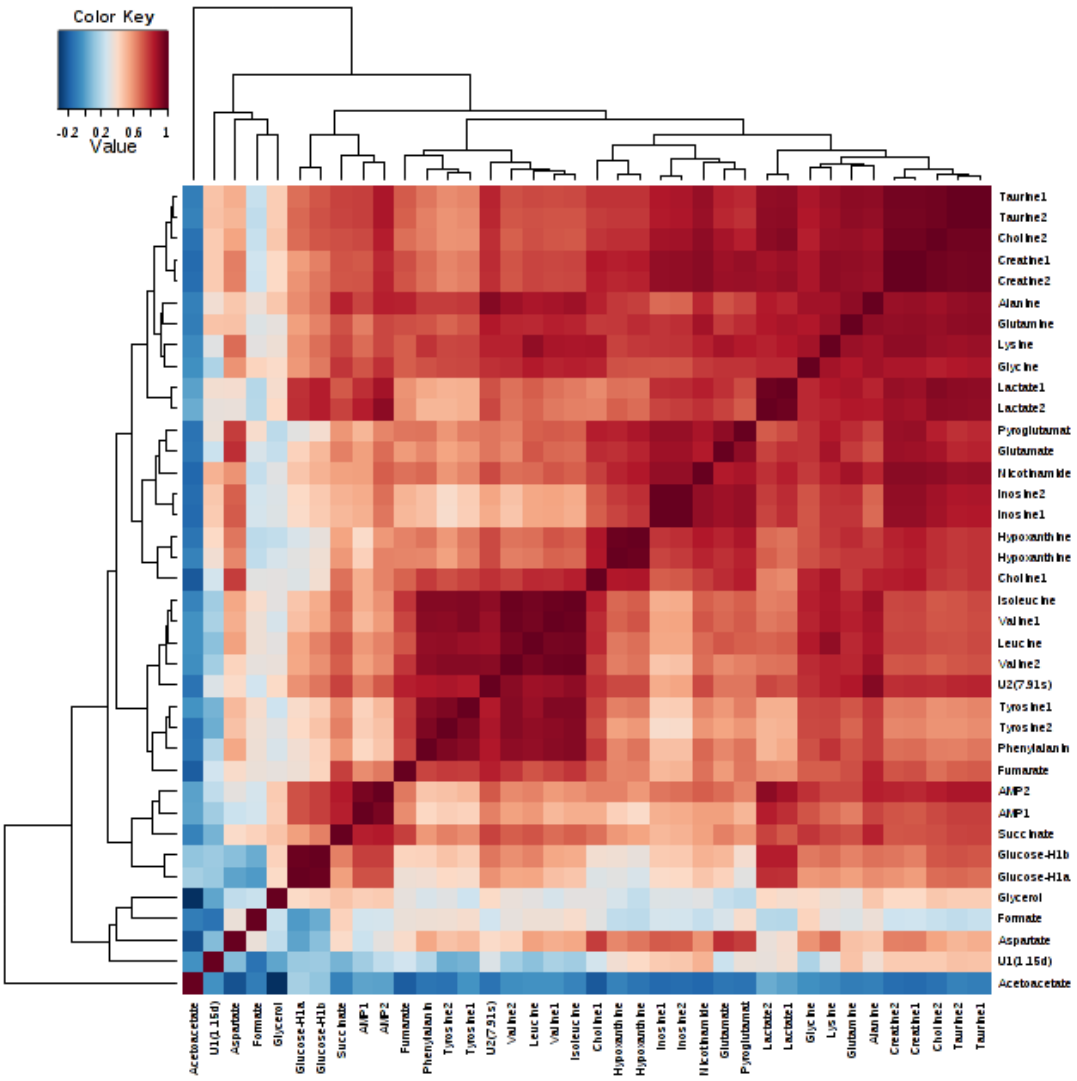
Στο σχήμα 6.10 παρουσιάζεται ένα heatmap όπου απεικονίζονται πιθανές διαφοροποιήσεις συγκεντρώσεων των μεταβολιτών για τις διαφορετικές κλάσεις. Δεν διακρίνονται σαφείς διαχωριστικές γραμμές μεταξύ των διαφορετικών κλάσεων. Η απόσταση υπολογίστηκε κατά Pearson και χρησιμοποιήθηκε ο αλγόριθμος συσταδοποίησης (clustering) `Ward`.

Αντίστοιχα με το σχήμα 6.9, στο σχήμα 6.11 φαίνεται μια κατάταξη των μεταβολιτών ως προς τη σημασία τους για ταξινόμηση με τον παρεχόμενο αλγόριθμο `Random Forest`.

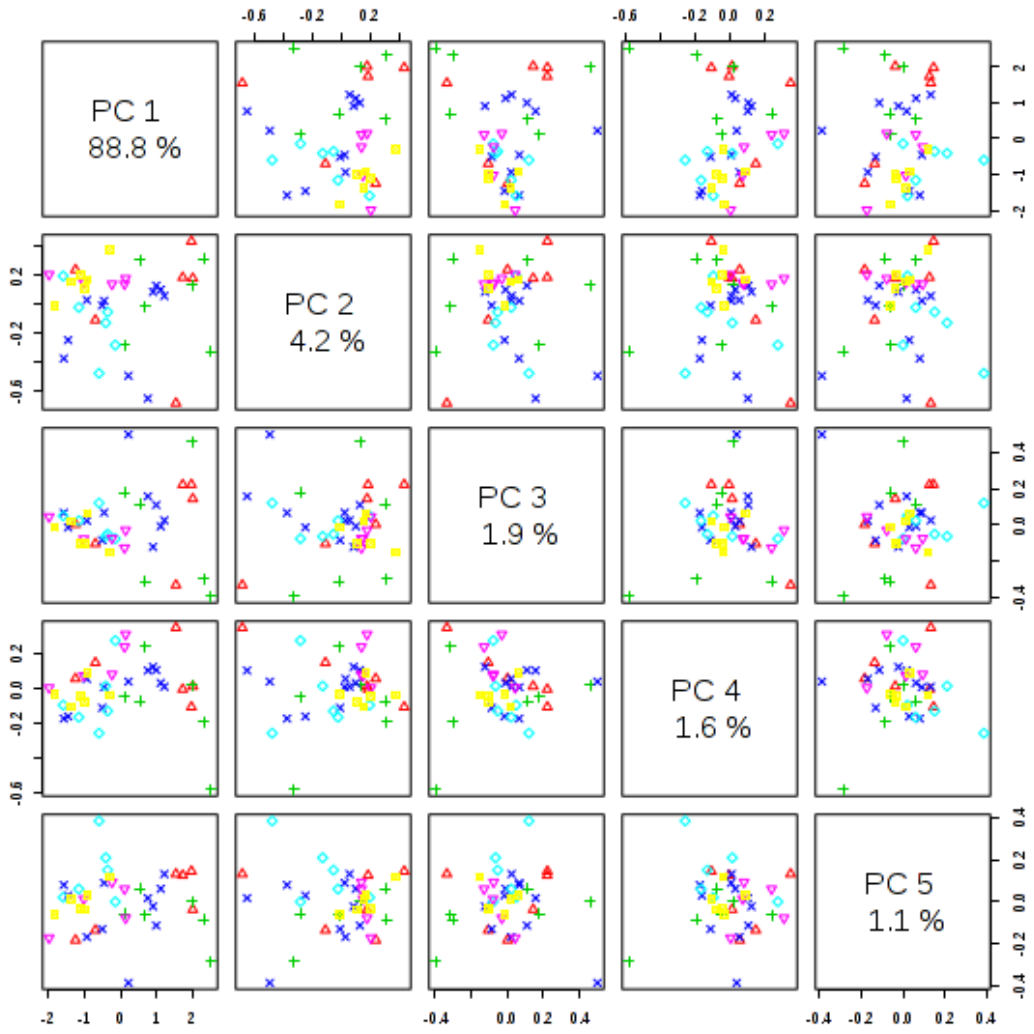
Στο σχήμα 6.12 φαίνεται η απόδοση του αλγορίθμου Random Forest για την ταξινόμηση των δεδομένων. Είναι ξεκάθαρη η διαφορά στην επιτυχία της ταξινόμησης ως προς την κλάση `dxr+oleu1` και ως προς την κλάση `oleu2` σε σχέση με τις υπόλοιπες και η αδυναμία πρόβλεψης της κλάσης `oleu1`. Αυτό φαίνεται και στον σχετικό πίνακα - confusion matrix (πίνακας 6.2). Συνολικά, η επιτυχία του μοντέλου ανέρχεται στο 47.5% (True Positives ως προς το σύνολο).



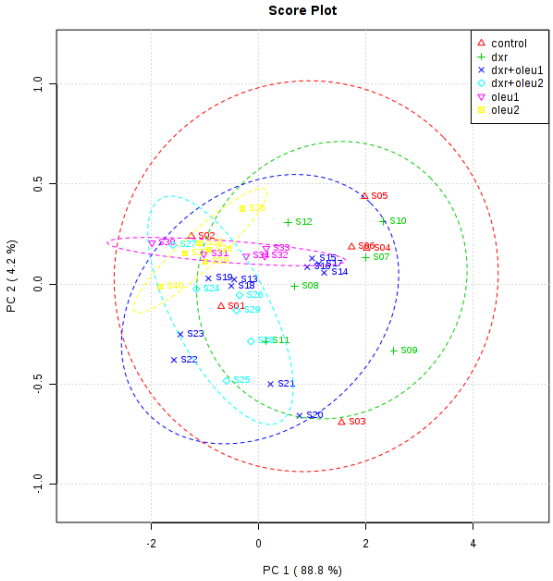
Σχήμα 6.1: Τα δεδομένα πριν και μετά την εφαρμογή Pareto Scaling



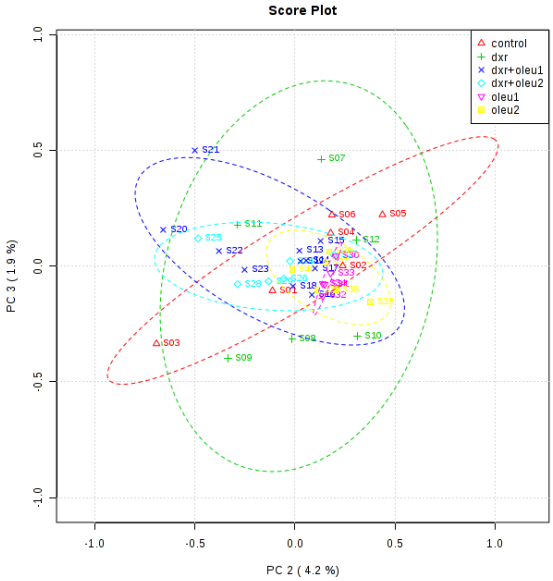
Σχήμα 6.2: Correlation heatmap



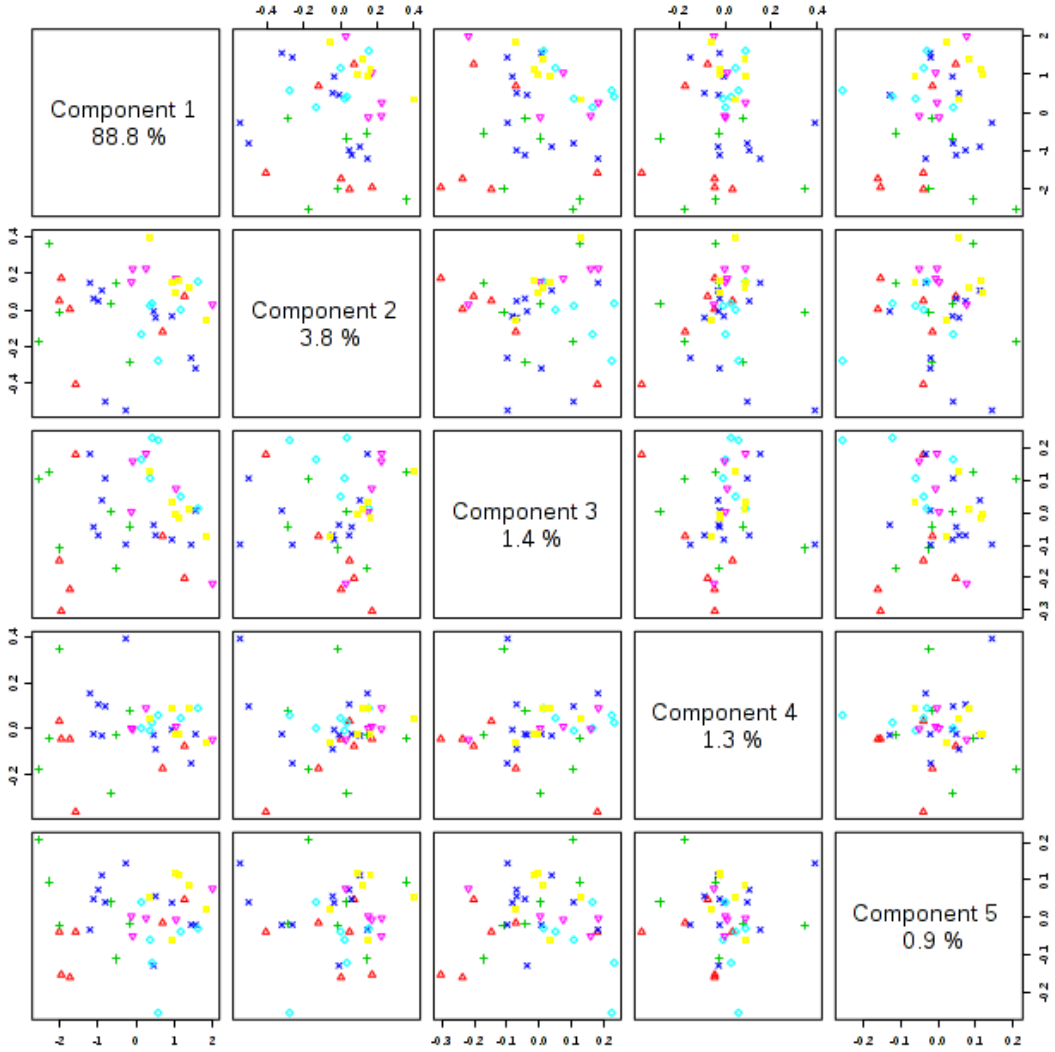
Σχήμα 6.3: PCA για τα 5 κυριότερα Principal Components



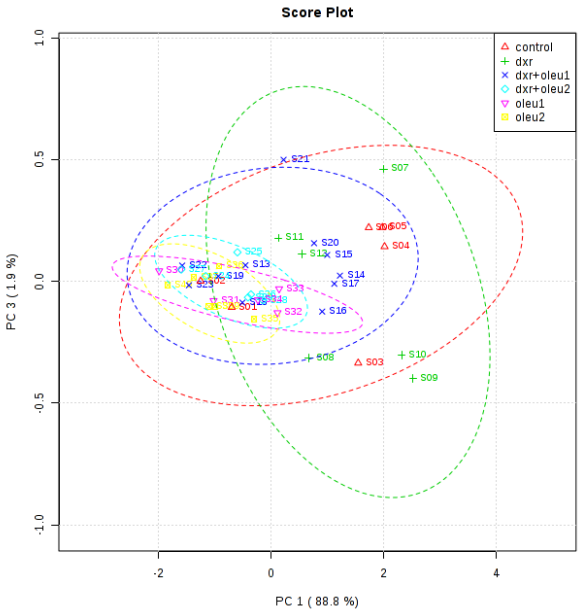
Σχήμα 6.4: PCA: 2D score plot για τα Principal Components 1 και 2 (διάστημα εμπιστοσύνης 95%) - Δεν διακρίνεται κάποιος ικανοποιητικός διαχωρισμός κλάσεων. Η συγχορήγηση ολευρωπαΐνης μαζί με την αδριαμυκίνη φαίνεται να μετακινεί τα σημεία αριστερότερα.



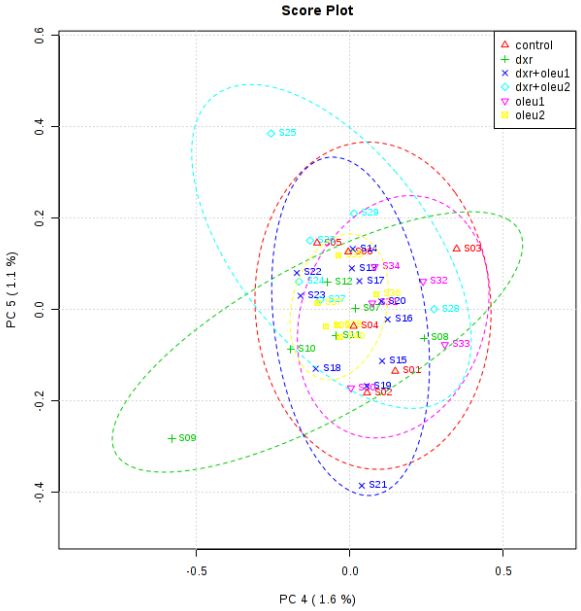
Σχήμα 6.5: PCA: 2D score plot για τα Principal Components 2 και 3 (διάστημα εμπιστοσύνης 95%) - Δεν διακρίνεται κάποιος ικανοποιητικός διαχωρισμός κλάσεων. Η συγχορήγηση ολευρωπαΐνης μαζί με την αδριαμυκίνη φαίνεται να μετακινεί τα σημεία προς μια κεντρική περιοχή όπου μοιάζουν να συγκεντρώνονται τα μη-dxr σημεία.



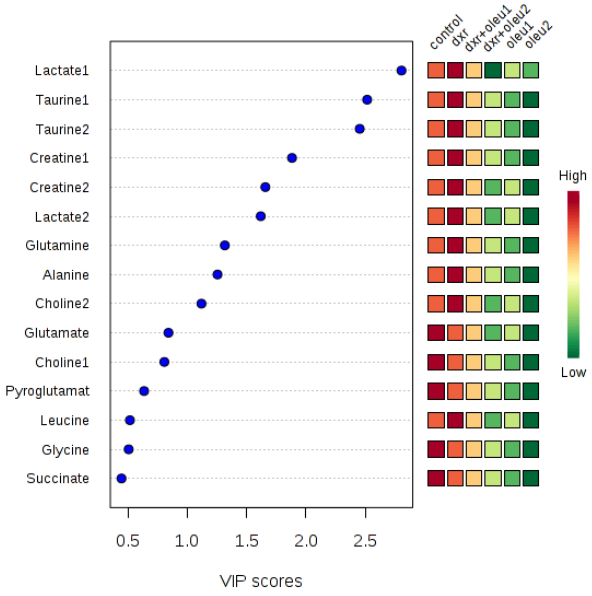
Σχήμα 6.6: PLS-DA για τα 5 κυριότερα συστατικά



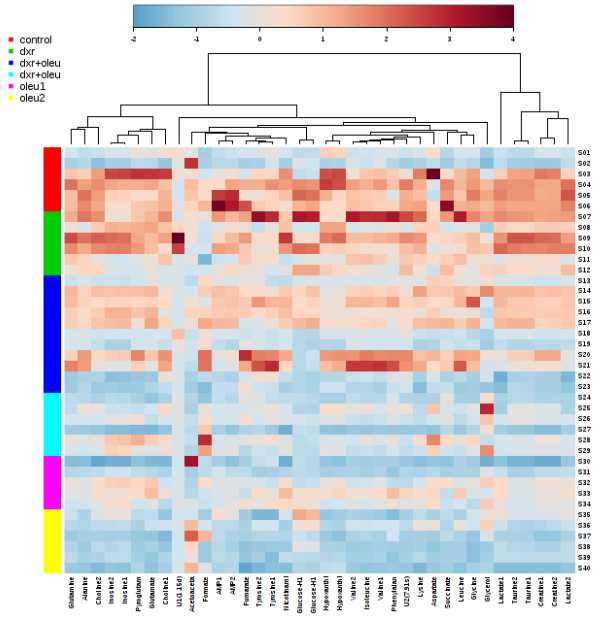
Σχήμα 6.7: PLS-DA: 2D score plot για τα Components 3 και 1 (διάστημα εμπιστοσύνης 95%) - Δεν διακρίνεται κάποιος ικανοποιητικός διαχωρισμός.



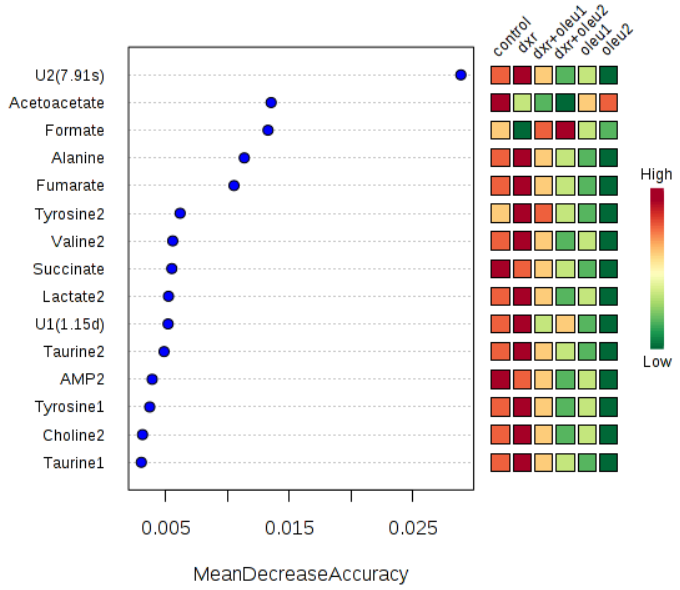
Σχήμα 6.8: PLS-DA: 2D score plot για τα Components 5 και 4 (διάστημα εμπιστοσύνης 95%) - Δεν διακρίνεται κάποιος ικανοποιητικός διαχωρισμός. Η συγχορήγηση ολευρωπαΐνης μαζί με αδριαμυκίνη φαίνεται να μετακινεί τα σημεία προς μια περιοχή μη-dxr σημείων.



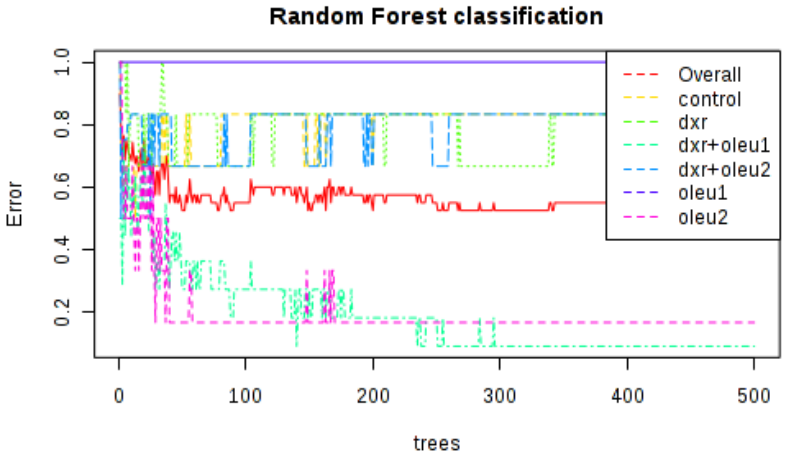
Σχήμα 6.9: PLS-DA: Variable Importance in Projection score (component 1)



Σχήμα 6.10: Concentration Heatmap: Σχετικές συγκεντρώσεις των μεταβολιτών στα δείγματα των διαφορετικών κλάσεων.



Σχήμα 6.11: Random Forest - Variable Importance



Σχήμα 6.12: Random Forest - classification: Διακρίνεται η επιτυχία στον διαχωρισμό της κλάσης dxr+oleu1 σε σχέση με τις υπόλοιπες.

Πίνακας 6.2: Random Forest: Confusion Matrix (Metaboanalyst)

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**1**	3	0	1	0	1	0.270
b: dxr	1	**2**	3	0	0	0	0.330
c: dxr+oleu1	0	0	**10**	1	0	0	0.910
d: dxr+oleu2	0	0	3	**1**	1	1	0.270
e: oleu1	0	0	2	1	**0**	2	0.000
f: oleu2	0	0	0	0	1	**5**	0.830
Επιτυχία							47.5%

6.3 Ανάλυση με τη WEKA

Η σουίτα `WEKA` [9] παρέχει εργαλεία data mining γενικής χρήσης, μεταξύ των οποίων κλασσικούς αλγορίθμους μηχανικής μάθησης. Παρέχονται τόσο εργαλεία προεπεξεργασίας (π.χ. PCA) όσο και αλγόριθμοι ταξινόμησης διαφόρων κατηγοριών, τους οποίους δοκιμάσαμε για την περίπτωσή μας. Η ιδιαιτεροτητα των δεδομένων που εξετάζονται να είναι λίγα και χωρισμένα σε 6 κλάσεις (αντί για 2 που ήταν το πλήθος των κλάσεων στο προηγούμενο πρόβλημα) οδηγεί σε μειωμένη επιτυχία των προσφερόμενων αλγορίθμων. Συνδυάζοντας τα αποτελέσματα διαφορετικών μοντέλων με τεχνικές όπως συμψηφισμό αποτελεσμάτων (voting) μπορούμε να επιτύχουμε καλύτερα αποτελέσματα. Ιδιαίτερο ενδιαφέρον παρουσιάζουν οι αλγόριθμοι τύπου Support Vector Machines (SVM) και ειδικότερα η υλοποίηση `LibSVM` [10], λόγω της σχετικά υψηλότερης επιτυχίας τους στην ταξινόμηση. Σε κάθε περίπτωση, χρησιμοποιείται 10-fold Cross Validation. Αξίζει να προσεχθεί ότι, λόγω του μικρού πλήθους των δεδομένων, μεταβολή στην επιτυχία κάποιου αλγορίθμου κατά ένα άτομο, επιφέρει μεταβολή στη συνολική επιτυχία του αλγορίθμου κατά 2.5%.

6.3.1 Αλγόριθμοι χωρίς συνδυασμό

Εξετάστηκαν όλοι οι διαθέσιμοι στη `WEKA` αλγόριθμοι, έπειτα από προσεχτική επιλογή των παραμέτρων κάθε αλγορίθμου, ώστε να συγκριθεί η επιτυχία τους στο συγκεκριμένο πρόβλημα. Η επιτυχία των μοντέλων που προέκυψαν φαίνεται στον πίνακα 6.3. Επισημαίνεται ότι η φύση του προβλήματος καθιστά αδύνατη την εφαρμογή ορισμένων αλγορίθμων και για το λόγο αυτό δεν εμφανίζονται. Παρατίθενται οι πίνακες (confusion matrices) με τα αποτελέσματα μερικών από τους αλγορίθμους με τη μεγαλύτερη επιτυχία. Παρατηρείται ότι η κλάση `dxr+oleu1` προβλέπεται και εδώ με αρκετά μεγάλη επιτυχία, ενώ η κλάση `oleu1` δεν προβλέπεται καθόλου από μερικά μοντέλα. Γυρνώντας στον πίνακα 6.1, διακρίνεται μια ανισοκατανομή των κλάσεων. Η κλάση dxr+oleu1 περιέχει τα περισσότερα δείγματα (11), ενώ η κλάση `oleu1` τα λιγότερα (5). Σε ορισμένους αλγορίθμους (όπως στον `LibSVM`) η επιτυχία μπορεί να βελτιωθεί με αντιστοίχιση κατάλληλων βαρών στις κλάσεις, ώστε να διορθώνεται η ανισοκατανομή.

6.3.2 Συνδυασμός αλγορίθμων

Οι meta-classifiers παρέχουν τρόπους βελτίωσης της επιτυχίας των διαθέσιμων αλγορίθμων. Π.χ. παρέχουν τρόπους συνδυασμού διαφορετικών μοντέλων, κατά τρόπο αντίστοιχο με το συνδυασμό πολλών ειδικών για την εξαγωγή μιας «καλύτερης» απόφασης. Συγκεκριμένα, ο `Vote` meta-classifier της `WEKA` συνδυάζει τις προβλέψεις μεμονωμένων αλγορίθμων για κάθε άτομο. Συνδυάζοντας τους αλγορίθμους `Logistic` (50%), `LWL-BayesNet` (50%), `J48` (50%) με τις ίδιες παραμέτρους, χρησιμοποιώντας τον κανόνα συνδυασμού `Majority Voting`, επιτυγχάνεται επιτυχία 52.5%. Επιτυγχάνεται σωστή πρόβλεψη λοιπόν για 1 επιπλέον περίπτωση. Και πάλι δεν προβλέπεται καθόλου η κλάση `oleu1` (βλ. πίνακα 6.12), καθώς δεν προβλέπεται επιτυχώς ούτε στα μοντέλα που χρησιμοποιούνται. Συνδυάζοντας κάποια μοντέλα δεν παρατηρείται καμία βελτίωση, ενώ σε πολλές περιπτώσεις τα αποτελέσματα είναι χειρότερα από αυτά που δίνει ένα από τα συνδυαζόμενα μοντέλα.

6.3.3 Σταδιακός διαχωρισμός κλάσεων

Παρατηρήθηκε ότι κάποια μοντέλα διακρίνουν με μεγάλη επιτυχία κάποιες κλάσεις (π.χ. την `dxr+oleu1`), ενώ αποτυγχάνουν στο διαχωρισμό των υπολοίπων. Εξετάζεται λοιπόν η εξής ιδέα: εάν η απόφαση ενός μοντέλου δείξει π.χ. την κλάση `dxr+oleu1`, τότε αυτό το συμπέρασμα είναι αποδεκτό. Εάν, ωστόσο, επιλεγεί κάποια από τις υπόλοιπες κλάσεις, τότε τα νέα δεδομένα τροφοδοτούνται σε ένα διαφορετικό μοντέλο, το οποίο έχει εκπαιδευτεί με βάση μονάχα τα δεδομένα των υπολοίπων κλάσεων. Εξαιρώντας, ωστόσο, τα δεδομένα της κλάσης `dxr+oleu1`, δεν παρατηρείται ιδιαίτερα καλύτερος διαχωρισμός. Μάλιστα οι αλγόριθμοι που δεν προέβλεπαν την κλάση `oleu1` εξακολουθούν να προβλέπουν μόνο μία, δύο ή (κυρίως) καμία από τις 5 περιπτώσεις. Ως προς την συνολική τους επιτυχία, κάποιοι αλγόριθμοι συμπεριφέρονται αισθητά καλύτερα (π.χ. Bayes, LMT, RandomTree, REPTree, OneR), ενώ άλλοι χειρότερα (π.χ. LibSVM, IBk, KStar). Σε γενικές γραμμές, τα αποτελέσματα δεν είναι πολύ διαφορετικά.

Πίνακας 6.3: Επιτυχία αλγορίθμων για το σύνολο και επιτυχία ως προς τις κλάσεις. a: control, b: dxr, c:dxr+oleu1, d:dxr+oleu2, e:oleu1, f:oleu2.

Αλγόριθμος-Μοντέλο	**Επιτυχία**	**a(6)**	**b(6)**	**c(11)**	**d(6)**	**e(5)**	**f(6)**
Bayes							
BayesNet	30.0%	0	1	5	1	0	5
NaiveBayes	30.0%	1	3	1	3	0	4
NaiveBayesUpdateable	30.0%	1	3	1	3	0	4
Functions							
LibSVM	60.0%	1	2	11	4	1	5
Logistic	50.0%	1	2	8	4	1	4
MultilayerPerceptron	45.0%	1	2	9	2	1	3
SimpleLogistic	40.0%	1	3	6	2	0	4
SMO	47.5%	1	2	9	2	0	5
Lazy							
Ibk	57.5%	3	1	10	3	1	5
Kstar	47.5%	3	1	9	3	1	2
LWL (BayesNet)	47.5%	3	3	6	2	0	5
LWL (NaiveBayes)	40.0%	2	3	4	3	0	4
LWL DecisionStump	37.5%	0	0	8	1	0	6
Rules							
DecisionTable	35.0%	0	0	4	4	0	6
Jrip	30.0%	0	2	5	0	0	5
OneR	22.5%	2	2	5	0	0	0
PART	52.5%	2	4	6	4	0	5
ZeroR	27.5%	0	0	11	0	0	0
Trees							
DecisionStump	25.0%	0	0	8	0	0	2
J48	50.0%	3	3	5	4	0	5
LMT	40.0%	2	2	5	2	1	4
RandomForest (trees=10)	40.0%	2	1	7	3	0	3
RandomForest (trees=500)	42.5%	1	1	7	3	0	5
RandomTree	30.0%	2	1	5	1	0	3
REPTree	27.5%	0	0	6	3	0	2
Αλγόριθμος-Μοντέλο	**Επιτυχία**	**a(6)**	**b(6)**	**c(11)**	**d(6)**	**e(5)**	**f(6)**

Πίνακας 6.4: LibSVM: Confusion Matrix

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**1**	2	1	0	2	0	0.167
b: dxr	3	**2**	1	0	0	0	0.333
c: dxr+oleu1	0	0	**11**	0	0	0	1.000
d: dxr+oleu2	0	0	1	**4**	1	0	0.667
e: oleu1	1	1	0	2	**1**	0	0.200
f: oleu2	0	0	0	0	1	**5**	0.833
Επιτυχία							60.0%

Πίνακας 6.5: IBk: Confusion Matrix

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**3**	1	1	0	0	1	0.500
b: dxr	2	**1**	1	0	2	0	0.167
c: dxr+oleu1	0	0	**10**	1	0	0	0.909
d: dxr+oleu2	0	0	2	**3**	0	1	0.500
e: oleu1	0	0	0	3	**1**	1	0.200
f: oleu2	0	0	0	0	1	**5**	0.833
Επιτυχία							57.5%

Πίνακας 6.6: PART: Confusion Matrix

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**2**	1	1	0	1	1	0.333
b: dxr	2	**4**	0	0	0	0	0.667
c: dxr+oleu1	1	3	**6**	0	1	0	0.545
d: dxr+oleu2	0	0	1	**4**	1	0	0.667
e: oleu1	0	1	1	2	**0**	1	0.000
f: oleu2	0	0	0	1	0	**5**	0.833
Επιτυχία							52.5%

Πίνακας 6.7: Logistic: Confusion Matrix

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**1**	2	1	0	2	0	0.167
b: dxr	3	**2**	1	0	0	0	0.333
c: dxr+oleu1	0	1	**8**	1	1	0	0.727
d: dxr+oleu2	0	0	1	**4**	1	0	0.667
e: oleu1	1	1	0	2	**1**	0	0.200
f: oleu2	0	0	0	0	2	**4**	0.667
Επιτυχία							50.0%

Πίνακας 6.8: J48: Confusion Matrix

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**3**	1	0	0	1	1	0.500
b: dxr	2	**3**	1	0	0	0	0.500
c: dxr+oleu1	2	4	**5**	0	0	0	0.455
d: dxr+oleu2	0	0	1	**4**	0	1	0.667
e: oleu1	0	1	1	2	**0**	1	0.000
f: oleu2	0	0	0	1	0	**5**	0.833
Επιτυχία							50%

Πίνακας 6.9: SMO: Confusion Matrix

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**1**	2	2	0	0	1	0.167
b: dxr	2	**2**	2	0	0	0	0.333
c: dxr+oleu1	0	0	**9**	2	0	0	0.818
d: dxr+oleu2	0	0	3	**2**	0	1	0.333
e: oleu1	0	0	3	1	**0**	1	0.000
f: oleu2	0	0	0	0	1	**5**	0.833
Επιτυχία							47.5%

Πίνακας 6.10: MultilayerPerceptron: Confusion Matrix

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**1**	2	1	0	1	1	0.167
b: dxr	2	**2**	2	0	0	0	0.333
c: dxr+oleu1	0	1	**9**	1	0	0	0.818
d: dxr+oleu2	0	0	1	**2**	2	1	0.333
e: oleu1	1	1	0	2	**1**	0	0.200
f: oleu2	0	0	0	1	2	**3**	0.500
Επιτυχία							45.0%

Πίνακας 6.11: RandomForest (500 trees): Confusion Matrix (WEKA)

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**1**	3	0	1	0	1	0.167
b: dxr	3	**1**	2	0	0	0	0.167
c: dxr+oleu1	0	2	**7**	1	0	1	0.636
d: dxr+oleu2	0	0	2	**3**	0	1	0.500
e: oleu1	0	0	2	2	**0**	1	0.000
f: oleu2	1	0	0	0	0	**5**	0.833
Επιτυχία							42.5%

Πίνακας 6.12: Vote (Logistic, LWL-BayesNet, J48): Confusion Matrix

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**2**	2	0	0	2	0	0.333
b: dxr	2	**3**	1	0	0	0	0.500
c: dxr+oleu1	1	3	**7**	0	0	0	0.636
d: dxr+oleu2	1	0	1	**4**	1	0	0.667
e: oleu1	0	1	1	2	**0**	1	0.000
f: oleu2	1	0	0	0	0	**5**	0.833
Επιτυχία							52.5%

Πίνακας 6.13: Επιτυχία αλγορίθμων (χωρίς την κλάση dxr+oleu1) για το σύνολο και επιτυχία ως προς τις κλάσεις. a: control, b: dxr, d:dxr+oleu2, e:oleu1, f:oleu2.

Αλγόριθμος-Μοντέλο	**Επιτυχία**	**a(6)**	**b(6)**	**d(6)**	**e(5)**	**f(6)**
Bayes						
BayesNet	34.5%	1	3	2	0	4
NaiveBayes	44.8%	2	2	4	1	4
NaiveBayesUpdateable	44.8%	2	2	4	1	4
Functions						
LibSVM	55.2%	2	2	5	1	6
Logistic	44.8%	1	2	4	1	5
MultilayerPerceptron	51.7%	2	2	5	1	5
SimpleLogistic	51.7%	1	3	4	2	5
SMO	48.3%	2	2	4	0	6
Lazy						
Ibk	37.9%	3	2	3	1	2
Kstar	37.9%	3	2	3	1	2
LWL (BayesNet)	34.5%	1	1	3	0	5
LWL (NaiveBayes)	44.8%	2	3	3	1	4
LWL DecisionStump	41.4%	0	2	5	0	5
Rules						
DecisionTable	31.0%	1	1	2	0	5
Jrip	31.0%	2	1	2	0	4
OneR	44.8%	1	4	4	0	4
PART	37.9%	2	1	4	0	4
ZeroR	0%	0	0	0	0	0
Trees						
DecisionStump	17.2%	0	0	5	0	0
J48	48.3%	3	3	4	0	4
LMT	51.7%	1	3	4	2	5
RandomForest (trees=10)	48.3%	2	3	5	1	3
RandomForest (trees=500)	37.9%	0	2	4	0	5
RandomTree	48.3%	2	6	3	1	2
REPTree	44.8%	3	1	3	0	6
Αλγόριθμος-Μοντέλο	**Επιτυχία**	**a(6)**	**b(6)**	**d(6)**	**e(5)**	**f(6)**

6.4 Επιλογή μεταβλητών

Στους αλγορίθμους που παρουσιάστηκαν παραπάνω τροφοδοτήθηκε ολόκληρος ο πίνακας ατόμων (instances) - χαρακτηριστικών (attributes). Ενδέχεται ωστόσο, να είναι δυνατόν να παραχθεί αποτέλεσμα ταξινόμησης αντίστοιχης ή και καλύτερης ποιότητας, χρησιμοποιώντας λιγότερες μεταβλητές-χαρακτηριστικά. Έτσι, μπορούμε να εστιάσουμε την προσοχή μας σε αυτές και, σε ορισμένες περιπτώσεις, να δημιουργήσουμε μοντέλα που θα χρειάζονται δεδομένα από απλούστερες, οικονομικότερες ή γρηγορότερες μεθόδους ανάλυσης. Για την επιλογή των μεταβλητών υλοποιήθηκε ένας γενετικός αλγόριθμος στο GNU Octave (συμβατός με MATLAB). Στην υλοποίηση αυτή χρησιμοποιείται ο αλγόριθμος μηχανικής μάθησης LibSVM για την ταξινόμηση, καθώς έδωσε τα καλύτερα αποτελέσματα στις παραπάνω δοκιμές.

Χρησιμοποιώντας τα 12 χαρακτηριστικά που παρουσιάζονται στον πίνακα 6.14 επετεύχθη επιτυχία 60% (TP: 24/40), μεγαλύτερη της επιτυχίας που επετεύχθη χρησιμοποιώντας και τα 38 διαθέσιμα χαρακτηριστικά (55%) με τις ίδιες παραμέτρους, στο ίδιο περιβάλλον (LibSVM μέσω Octave). Ενδιαφέρον είναι επίσης ότι χρησιμοποιώντας μόνο τα πρώτα 2 χαρακτηριστικά από αυτά, η επιτυχία ανέρχεται στο 47.5% (TP: 19/40). Στον πίνακα 6.15 φαίνεται ο confusion matrix που προκύπτει από τον LibSVM μέσω WEKA, για 12 χρησιμοποιούμενα χαρακτηριστικά και στον πίνακα 6.16 για 2 χαρακτηριστικά. Η σειρά με την οποία εμφανίζονται τα χαρακτηριστικά στον πίνακα 6.14 δηλώνει την σειρά "προτίμησης" των αντίστοιχων γονιδίων στον τελικό γονότυπο για πολλές επανεκκινήσεις του αλγορίθμου. Δηλαδή, κάθε φορά που εκτελείται ο γενετικός αλγόριθμος, ξεκινώντας από τυχαίο κάθε φορά αρχικό πληθυσμό, συγκλίνει σε κάποιες τιμές για κάθε γονίδιο. Κρατώντας την επικρατέστερη τιμή (εκδήλωση ή μη του αντίστοιχου χαρακτηριστικού) για κάθε γονίδιο, προκύπτει το εξελικτικά καλύτερο χρωμόσωμα. Κάθε φορά επιλέγονται διαφορετικά γονίδια, κάποια όμως από αυτά έχουν την τάση να επιλέγονται συχνότερα. Λόγω της φύσης των γενετικών αλγορίθμων, ενδέχεται να μπορούν να επιτευχθούν ακόμα καλύτερα αποτελέσματα, πιθανότατα με κάποιες διαφορές στα επιλεγόμενα χαρακτηριστικά. Κάποιος που θα ήθελε να εστιάσει σε αυτό το σημείο, θα μπορούσε να βελτιστοποιήσει τις παραμέτρους του γενετικού αλγορίθμου.

Πίνακας 6.14: Επιλεγμένα χαρακτηριστικά (attributes)

α/α	**ID**	**Όνομα χαρακτηριστικού**	**Συχνότητα επιλογής**
1	21	Glycerol	0.83
2	33	U2(7.91s)	0.73
3	38	Formate	0.73
4	7	Alanine	0.70
5	25	Glucose-H1a	0.70
6	15	Creatine1	0.67
7	23	Lactate2	0.67
8	13	Aspartate	0.63
9	3	Valine1	0.60
10	4	Valine2	0.60
11	6	Lactate1	0.60
12	8	Acetoacetate	0.60

Πίνακας 6.15: LibSVM με 12 attributes: Confusion Matrix (WEKA)

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**3**	0	2	0	1	0	0.500
b: dxr	1	**2**	2	0	1	0	0.333
c: dxr+oleu1	1	0	**10**	0	0	0	0.909
d: dxr+oleu2	0	0	1	**3**	0	2	0.500
e: oleu1	0	0	2	1	**1**	1	0.200
f: oleu2	0	0	0	0	1	**5**	0.833
Επιτυχία							60.0%

Πίνακας 6.16: LibSVM με 2 attributes: Confusion Matrix (WEKA)

	a	**b**	**c**	**d**	**e**	**f**	**TP rate**
a: control	**2**	1	2	0	0	1	0.333
b: dxr	2	**0**	4	0	0	0	0.000
c: dxr+oleu1	0	2	**8**	1	0	0	0.727
d: dxr+oleu2	0	0	2	**2**	1	1	0.333
e: oleu1	0	0	1	2	**0**	2	0.000
f: oleu2	0	0	0	1	0	**5**	0.833
Επιτυχία							42.5%

6.5 Συζήτηση

6.5.1 Διαγράμματα PCA/PLS-DA/Heatmap

Από τα διαγράμματα PCA και PLS-DA δεν βλέπουμε κάποιο ιδιαίτερα σαφή διαχωρισμό κλάσεων. Μπορούμε ωστόσο να παρατηρήσουμε κάποια στοιχεία. Στο σχήμα 6.4 παρατηρούμε ότι η συγχορήγηση ολευρωπαΐνης μαζί με την αδριαμυκίνη μετακινεί τα σημεία (αριστερότερα) στο διάγραμμα, άρα επηρεάζεται το μεταβολικό προφίλ. Τα σημεία των κλάσεων `oleu1` και `oleu2` βρίσκονται αρκετά κοντά μεταξύ τους, κάτι που θα δικαιολογήσει παρακάτω την αδυναμία διάκρισής τους. Δυστυχώς, τα σημεία της κλάσης `control` είναι λίγα και αρκετά διασπαρμένα ώστε να βγάλουμε κάποιο ασφαλές συμπέρασμα σύγκρισης, ωστόσο αναμένουμε να βρίσκονται κοντά στα σημεία των κλάσεων `oleu1`/`oleu2`. Στο σχήμα 6.5 μπορούμε να παρατηρήσουμε ότι υπάρχει μια κεντρική περιοχή στην οποία συγκεντρώνονται τα σημεία των ομάδων που δεν έχουν λάβει αδριαμυκίνη, ενώ τα σημεία της κλάσης `dxr` βρίσκονται προς το εξωτερικό αυτής της περιοχής. Αυτή η συμπεριφορά θα παρατηρούνταν διαυγέστερα αν αγνοούσαμε το σημείο `S03`. Η συγχορήγηση ολευρωπαΐνης φαίνεται να μετακινεί τα σημεία σταδιακά προς την κεντρική περιοχή. Στο σχήμα 6.7 παρατηρούμε και πάλι ότι η συγχορήγηση ολευρωπαΐνης μαζί με την αδριαμυκίνη μετακινεί τα σημεία προς μια περιοχή του διαγράμματος όπου συγκεντρώνονται τα σημεία των κλάσεων που δεν έχουν λάβει αδριαμυκίνη. Και πάλι, ωστόσο, τα σημεία της κλάσης `control` δεν βοηθάνε ιδιαίτερα για την εξαγωγή κάποιου σαφούς συμπεράσματος.

Στο σχήμα 6.10 μπορούμε να παρατηρήσουμε ότι στην κλάση `dxr` εμφανίζονται γενικώς υψηλότερες συγκεντρώσεις μεταβολιτών σε σχέση με τις υπόλοιπες κλάσεις. Αυτό μπορεί να ερμηνευτεί από την καρδιοτοξική επίπτωση της αδριαμυκίνης, λόγω της οποίας τα κύτταρα αλλάζουν το μεταβολισμό τους. Συγχορηγώντας ολευρωπαΐνη, παρατηρούμε οι συγκεντρώσεις σταδιακά να μειώνονται και θα μπορούσαμε να εκλάβουμε αυτήν την παρατήρηση ως ένδειξη ότι η ολευρωπαΐνη βοηθάει στην αναστολή της καρδιοτοξικής δράσης. Τα δείγματα της ομάδας `control` θα περιμέναμε να παρουσιάζουν μικρότερες συγκεντρώσεις, ωστόσο έχει φανεί και από τα διαγράμματα PCA ότι δεν μπορούν να μας οδηγήσουν σε ασφαλή συμπεράσματα. Τα δείγματα των κλάσεων `oleu1` και `oleu2` παρουσιάζουν ξεκάθαρα χαμηλότερες συγκεντρώσεις.

6.5.2 Αλγόριθμος Random Forest - MetaboAnalyst

Στον πίνακα 6.2 παρατίθενται τα αποτελέσματα ταξινόμησης με τον αλγόριθμο Random Forest στο MetaboAnalyst. Καλύτερα από όλες προβλέπεται η κλάση `dxr+oleu1`, για την οποία διατίθενται σχεδόν τα διπλάσια δείγματα από τις υπόλοιπες, ενώ τα χειρότερα αποτελέσματα παρουσιάζονται για την κλάση `oleu1` η οποία διαθέτει τα λιγότερα δείγματα. Θα περιμέναμε, αν είχαμε περισσότερα δεδομένα και ισοκατανεμημένες κλάσεις, να είχαμε και καλύτερα αποτελέσματα, παρότι δεν αρκεί αυτή η παρατήρηση για να στηρίξει τον ισχυρισμό. Ας δούμε τώρα ποιες κλάσεις συγχέονται από τον αλγόριθμο, σε σχέση και με τα διαγράμματα PCA/PLS-DA και το Concentration Heatmap. Η κλάση `control` συγχέεται με την κλάση `dxr`. Είδαμε και στο σχήμα 6.10 ότι στα περισσότερα `control` παρουσιάζονται υψηλές συγκεντρώσεις, όπως και στα περισσότερα `dxr`. Η κλάση `dxr` συγχέεται τόσο

με την κλάση control όσο και με την κλάση dxr+oleu1. Δηλαδή δεν μπορούμε να διαχωρίσουμε αρκετά καλά τα άτομα που έχουν λάβει μόνο αδριαμυκίνη, από τα άτομα στα οποία έχει συγχορηγηθεί και μικρή δόση ολευρωπαΐνης. Οι κλάσεις dxr+oleu1 και dxr+oleu2 συγχέονται κυρίως μεταξύ τους, όπως θα περιμέναμε, ενώ η υψηλή δόση ολευρωπαΐνης δείχνει να οδηγεί σε ομοιότητες με τα άτομα που έχουν λάβει μόνο ολευρωπαΐνη. Είναι θετικό το γεγονός ότι δεν συγχέονται με άτομα που έχουν λάβει μόνο dxr. Η κλάση oleu1 συγχέεται με την κλάση oleu2. Πράγματι, είδαμε ότι σε όλα τα προηγούμενα διαγράμματα, τα σημεία των δύο κλάσεων βρίσκονταν αρκετά κοντά μεταξύ τους. Η κλάση oleu2 δείχνει να διατηρεί μια δυνατή ταυτότητα, καθώς καταφέρνει να διακριθεί ικανοποιητικά ακόμα και από την συγγενή της oleu1. Ο πίνακας αυτός, όπως και οι επόμενοι που θα δούμε, δείχνει ότι συνθετότερες μέθοδοι μηχανικής μάθησης μπορούν να τα καταφέρουν καλύτερα από άλλες απλούστερες στατιστικές μεθόδους. Στο σχήμα 6.12 φαίνεται με άλλη μορφή το σφάλμα ταξινόμησης που είδαμε και στον πίνακα 6.2. Παρατηρούμε επίσης ότι για μικρό πληθυσμό επιλεγμένων δέντρων το σφάλμα είναι υψηλό και διακυμαίνεται εύκολα, ενώ για υψηλό αριθμό δέντρων (πάνω από 300) μειώνεται και σταθεροποιείται. Αυτό εξ' άλλου είναι ένα αναμενόμενο αποτέλεσμα. Στα σχήματα 6.2 και 6.11 θα επανέλθουμε στην επιλογή μεταβλητών.

6.5.3 Αλγόριθμοι στη WEKA

Ας προσπαθήσουμε να συγκρίνουμε σε πρώτη φάση τις κατηγορίες αλγορίθμων, με βάση τον πίνακα 6.3. Καλύτερα αποτελέσματα φαίνεται να δίνουν οι κατηγορίες Functions και Lazy και χειρότερα οι απλούστεροι Bayes και Rules. Καλύτερα αποτελέσματα δίνει ο αλγόριθμος LibSVM, ακολουθούμενος από τον IBk. Οι αλγόριθμοι τύπου Bayes προβλέπουν καλύτερα την κλάση oleu2, ενώ αποτυγχάνουν πλήρως να προβλέψουν την κλάση oleu1. Δεν συμπεριφέρονται ικανοποιητικά ούτε στην πολυπληθή κλάση dxr+oleu1. Στην κατηγορία Functions βλέπουμε ότι οι αλγόριθμοι αποδίδουν αρκετά καλά στην πρόβλεψη της κλάσης drx+oleu1 και της oleu2 αλλά δεν δίνουν καθόλου ικανοποιητικά αποτελέσματα στην (διασκορπισμένη) control και στην oleu1. Αντίστοιχη κατανομή εμφανίζεται στην κατηγορία Lazy, οι οποίοι προβλέπουν ελαφρώς καλύτερα την κλάση Control, σε βάρος της κλάσης dxr. Στην κατηγορία Rules τα αποτελέσματα δεν είναι καθόλου ενθαρρυντικά. Ορισμένοι αλγόριθμοι, όπως ο Decision Table δίνουν καλά αποτελέσματα ως προς την κλάση oleu2, ενώ ευχάριστη έκπληξη αποτελεί ο αλγόριθμος PART ο οποίος δίνει μια σχετικά ομοιόμορφη κατανομή ως προς την επιτυχία που παρουσιάζει για κάθε κλάση, αδυνατώντας όμως και αυτός να προβλέψει την oleu1. Στην κατηγορία Trees διακρίνεται και πάλι σημαντική επιτυχία στην πρόβλεψη της πολυπληθούς dxr+oleu1, ωστόσο κάποιοι αλγόριθμοι δίνουν πολύ μικρότερη συνολική ακρίβεια από άλλους. Κοινή δυσκολία εμφανίζεται στην πρόβλεψη της κλάσης dxr.

Στη συνέχεια, ας εξετάσουμε τον confusion matrix του μοντέλου που προέκυψε από την LibSVM (πίνακας 6.4), η οποία έδωσε την υψηλότερη συνολική ακρίβεια. Παρατηρούμε ότι και πάλι, σύγχυση των κλάσεων παρατηρείται κυρίως με γειτονικές τους. Εξαίρεση αποτελεί η κλάση control η οποία φαίνεται να αντιμετωπίζεται με σχεδόν τυχαίο τρόπο, ωστόσο τα δεδομένα είναι λίγα για να κρίνουμε. Παρόμοια συμπεριφέρεται και η κλάση oleu1, ωστόσο κανένα από τα μοντέλα που εκπαιδεύσαμε δεν κατάφερε να επιτύχει καλύτερο αποτέλεσμα.

Παρατηρήστε ότι στην κλάση `dxr+oleu1` το μοντέλο αυτό έχει 100% ευαισθησία (προβλέπονται και οι 11 στις 11 περιπτώσεις) και 79% ευστοχία (από τις 14 περιπτώσεις που αντιστοιχούνται στην κλάση, οι 11 ανήκουν όντως σε αυτήν), ενώ στην κλάση `oleu2` έχει 100% ευστοχία (ως `oleu2` προβλέπονται μόνο άτομα της σωστής κλάσης) και 83% ευαισθησία. Αντίστοιχη κατανομή φαίνεται να παρατηρείται και στους πίνακες 6.5 έως 6.11. Οι κλάσεις `control` και `oleu1` αντιμετωπίζονται σχεδόν τυχαία, ενώ στις κλάσεις `dxr+oleu1` και `oleu2` σημειώνται γενικώς η υψηλότερη επιτυχία. Ενδιαφέρον παρουσιάζει ο PART, ο οποίος προβλέπει αρκετά καλά την «δύσκολη» κλάση `dxr`.

Συνδυάζοντας μοντέλα (πίνακας 6.12) παρατηρούμε ότι είναι δυνατόν (όχι όμως κανόνας) να επιτύχουμε υψηλότερη ακρίβεια από αυτή των μοντέλων που συνδυάζουμε. Και πάλι, ωστόσο, κλάσεις που δεν προβλέπονται από κανένα μοντέλο δεν μπορούν να προβλεφθούν.

Στην εκδοχή όπου εξετάζονται μονάχα οι 5 κλάσεις του προβλήματος μπορούμε να παρατηρήσουμε ότι αλγόριθμοι που βάσιζαν την συνολική επιτυχία τους στην πολύ καλή πρόβλεψη της κλάσης `dxr+oleu1` παρουσιάζουν γενικώς χαμηλότερη συνολική ακρίβεια μετά την αφαίρεσή της καθώς, στους περισσότερους αλγορίθμους, η ακρίβεια πρόβλεψης των υπόλοιπων κλάσεων δεν φαίνεται να βελτιώνεται σημαντικά.

6.5.4 Επιλογή Μεταβλητών

Στην ενότητα 6.4 φαίνονται τα αποτελέσματα της επιλογής μεταβλητών. Παρήχθησαν αρκετά ενθαρρυντικά αποτελέσματα με μόλις 12 ή ακόμα και με μόλις 2 επιλεγμένες μεταβλητές, από σύνολο 38 μεταβλητών στο σύστημα. Μάλιστα, με 12 μεταβλητές παρήχθησαν καλύτερα αποτελέσματα από ότι χρησιμοποιώντας το σύνολο των μεταβλητών. Αντιπαραβάλλοντας τον πίνακα 6.14 με το σχήμα 6.11 διαπιστώνουμε ότι επιλέγονται σε μεγάλο βαθμό οι ίδιες μεταβλητές με τις δύο μεθόδους. Αυτό ενισχύει το συμπέρασμα ότι ο αλγόριθμος που αναπτύχθηκε αποδίδει σε γενικές γραμμές καλά. Ένας παρατηρητικός αναγνώστης θα μπορούσε να προσέξει ότι και στο σχήμα 4.6 παρουσιάζεται αντίστοιχη εικόνα, με τα παρεχόμενα εργαλεία της WEKA. Μπορούμε όμως ακόμα να παρατηρήσουμε, ότι οι περισσότερες επιλεγμένες μεταβλητές εμφανίζονται σε μια περιοχή πολύ υψηλής συσχέτισης στο correlation heatmap (σχήμα 6.2). Όμως, αυτό δεν ισχύει για τις δύο πρώτες επιλεγμένες μεταβλητές με τις οποίες επιτυγχάνεται 42.5% επιτυχία.

Κεφάλαιο 7

Συμπεράσματα - Προτάσεις για μελλοντική έρευνα

7.1 Συμπεράσματα

Εξετάσαμε δύο προβλήματα, με αρκετά διαφορετικά χαρακτηριστικά και δυσκολίες. Στο πρώτο πρόβλημα είχαμε πολλά δεδομένα, εύκολα διαχωρίσιμα, μονάχα δύο κλάσεις. Στο δεύτερο πρόβλημα είχαμε λίγα δεδομένα, δύσκολα διαχωρίσιμα, κατανεμημένα σε έξι κλάσεις. Εξετάσαμε πολλούς διαφορετικούς αλγορίθμους και στα δύο προβλήματα και παρατηρήσαμε ότι ο ίδιος αλγόριμθος μπορεί να έχει διαφορετική απόδοση σε κάθε περίπτωση. Στο Πρόβλημα 1 σχεδόν όλα τα μοντέλα έδωσαν πολύ καλά αποτελέσματα, ωστόσο κάποια απλούστερα μοντέλα κανόνων ή δέντρων ήταν πιο αποτελεσματικά σε σχέση με πιο σύνθετα μοντέλα νευρωνικών δικτύων ή SVM. Η εικόνα ήταν αρκετά διαφορετική στο Πρόβλημα 2, όπου τα περισσότερα απλά μοντέλα δεν κατάφεραν να ανταποκριθούν ικανοποιητικά. Τα μοντέλα τύπου SVM έδωσαν εκεί την καλύτερη συνολική ακρίβεια προβλέψεων. Συνεπώς, επιβεβαιώνεται και εδώ ότι πρέπει κανείς να αναζητά το κατάλληλο μοντέλο που δύναται να γενικεύσει σωστά το υπό εξέταση σύστημα. Δεν θα πρέπει το μοντέλο αυτό ούτε να είναι καταχρηστικά περίπλοκο, ούτε και αφελώς απλό. Υπάρχει πάντοτε ο κίνδυνος της υπερπροσαρμογής, ο οποίος αντιμετωπίστηκε στα πρόωρα στάδια εκπόνησης αυτής της εργασίας, καθώς επίσης και ο κίνδυνος της παραπλάνησης από πολύ ανομοιόμορφα κατανεμημένα δεδομένα. Χρήσιμα εργαλεία σε τέτοιες περιπτώσεις είναι το cross-validation και οι confusion matrices.

Στην περιπλοκότητα του συστήματος έρχεται να συνεισφέρει και το πλήθος των μεταβλητών. Όπως είδαμε στην ενότητα 6.4 είναι δυνατόν όχι μόνο να πετύχουμε πολύ καλά αποτελέσματα με ελάχιστες μεταβλητές, αλλά και να σημειώσουμε επιτυχία υψηλότερη από ότι θα παρατηρούσαμε χρησιμοποιώντας όλες τις μεταβλητές που έχουμε διαθέσιμες.

Υψηλότερη επιτυχία μπορεί να επιτευχθεί ακόμη με συνδυασμό περισσότερων του ενός μοντέλων με μια απλή ψηφοφορία. Είδαμε και στα δύο προβλήματα πώς από μοντέλα χαμηλότερης ακρίβειας είναι δυνατόν να επιτευχθεί ακρίβεια υψηλότερη των μοντέλων που συμμετέχουν στην ψηφοφορία. Η WEKA διαθέτει αρκετά σχετικά εργαλεία, ωστόσο είναι μια διαδικασία που έχει προσελκύσει το ενδιαφέρον των ερευνητών, με αποτέλεσμα να

παρουσιάζονται και νέοι αλγόριθμοι συναινετικής μοντελοποίησης.

Ως προς τη Μεταβολομική, φάνηκε ότι υπάρχουν αρκετές δυνατότητες βελτίωσης του διαχωρισμού δεδομένων σε σχέση με τη μέχρι τώρα πεπατημένη των κλασσικών στατιστικών μεθόδων. Το MetaboAnalyst παρέχει κάποιους βασικούς αλγορίθμους μηχανικής μάθησης ανάμεσα στα υπόλοιπα εργαλεία μεταβολομικής ανάλυσης. Η μεγάλη όμως ποικιλία διαθέσιμων εργαλείων data mining και οι προοπτικές βελτίωσής τους, δίνουν περισσότερες δυνατότητες παραμετροποίησης και ανάπτυξης περισσότερο αποτελεσματικών μοντέλων. Μάλιστα, υπάρχει επαρκής ποικιλία σχετικού λογισμικού το οποίο διατίθεται ελεύθερα για κάθε είδους χρήση, κάτι που δίνει σημαντικές βάσεις για ευκολότερη ανάπτυξη π.χ. server-side υπηρεσιών εξόρυξης δεδομένων σε μεταβολομικά προβλήματα.

7.2 Προτάσεις για μελλοντική έρευνα

Σκοπός της παρούσας εργασίας, ως πρώτης σχετικής διπλωματικής εργασίας στην εργαστηριακή μονάδα Αυτόματης Ρύθμισης και Πληροφορικής της Σχολής μας ήταν να διερευνήσει τις δυνατότητες εφαρμογής κλασσικών μεθόδων μηχανικής μάθησης στη Μεταβολομική. Είναι εξ' αρχής αναμενόμενο λοιπόν να αφήνει πολλά ανοιχτά σημεία, δίνοντας όμως ένα σημαντικό μέρος της απαραίτητης προεργασίας ώστε να μπορεί κάποιος ευκολότερα να ερευνήσει στο εξής το σχετικό πεδίο.

Σημαντική δουλειά μπορεί να γίνει στην σχολαστικότερη εφαρμογή συγκεκριμένων αλγορίθμων ή στην ανάπτυξη νέων, πρωτότυπων ή τροποποιημένων μεθόδων. Αρκετά από τα μοντέλα που εκπαιδεύτηκαν ενδέχεται να μπορούν να βελτιωθούν σε μικρό ή μεγαλύτερο βαθμό. Μπορούν επίσης να ελεγχθούν διαφορετικές μέθοδοι προεπεξεργασίας σε συνδυασμό με τους κυρίως αλγορίθμους μάθησης. Σημαντικό και αιχμηρό είναι επίσης το κομμάτι των meta-classifiers, για την επέκταση των δυνατοτήτων των κλασσικών αλγορίθμων. Ο συμψηφισμός που εξετάσαμε είναι μια από τις πιο απλές περιπτώσεις τέτοιων μεθόδων.

Εξαιρετικό ενδιαφέρον παρουσιάζει για τη μεταβολομική αυτή τη στιγμή η επιλογή μεταβλητών. Ένα μεγάλο πλήθος εργασιών ασχολείται με τον προσδιορισμό των ελάχιστων χαρακτηριστικών μεταβολιτών που μπορούν να δώσουν όλη την απαραίτητη πληροφορία για μια ασθένεια ή τον τρόπο δράσης ενός φαρμάκου. Είναι μεγάλη πρόκληση η δημιουργία απλών, αποδοτικών και κυρίως αξιόπιστων μοντέλων τα οποία θα μπορούν, με πολύ απλές χημικές αναλύσεις πεδίου να δίνουν σίγουρα συμπεράσματα ως προς την εξέλιξη της υγείας ενός ανθρώπου. Για παράδειγμα, στο κοντινό μέλλον θα μπορούσε κάτι τέτοιο να ενσωματωθεί σε μια μικρή ηλεκτρονική συσκευή. Φανταστείτε π.χ. ένα έξυπνο ρολόι το οποίο να παρέχει σε πραγματικό χρόνο πληροφορίες για το «μέλλον» ασθενών με καρδιακή ανεπάρκεια (ή άλλες παθήσεις) που χρειάζεται να αθλούνται και να προλαβαίνει εγκαίρως δυσάρεστες καταστάσεις.

Ο αλγόριθμος επιλογής μεταβλητών που αναπτύχθηκε μπορεί να βελτιωθεί προς διάφορες κατευθύνσεις. Σημαντική δουλειά μπορεί να γίνει ως προς την βελτιστοποίηση της ταχύτητας σύγκλισής του καθώς και της επίτευξης πιο ευσταθών λύσεων. Εκτός από τη γενετική έρευνα, θα μπορούσε κάποιος να ασχοληθεί και με άλλους αλγορίθμους αναζήτησης όπως η προσομοιωμένη ανόπτηση, αλγόριθμοι κοντινότερου γείτονα ή άλλες στοχαστικές μέθοδοι, όπου ήδη υπάρχει αρκετό σχετικό έργο στο ΕΜΠ.

Ακόμα, για να γίνουν πιο αποδοτικοί οι αλγόριθμοι μάθησης αλλά και επιλογής μεταβλητών, θα μπορούσε κάποιος να ασχοληθεί με την παράλληλη υλοποίηση διαφόρων τμημάτων της τυπικής διαδικασίας εξόρυξης δεδομένων. Το cross validation ή η αναπαραγωγή χρωμοσωμάτων σε έναν γενετικό αλγόριθμο είναι αρκετά απλά σημεία από τα οποία μπορεί να ξεκινήσει κάποιος την έρευνά του.

Η πληθώρα διαθέσιμων αλγόριθμων μηχανικής μάθησης και εξόρυξης δεδομένων συνδυασμένη με την διαφορετική απόδοση σε διάφορα προβλήματα διαμορφώνει ένα ακόμη ερώτημα που αξίζει να διερευνηθεί: την ανάπτυξη δηλαδή ενός μοντέλου που να επιλέγει τον κατάλληλο αλγόριθμο και να βελτιστοποιεί τις παραμέτρους του ανάλογα με τα χαρακτηριστικά του προβλήματος (όγκος δεδομένων, αριθμός κλάσεων, χαρακτηριστικά πίνακα δεδομένων κλπ).

Παράρτημα

Παράρτημα Α΄ Πληροφορίες για το λογισμικό που χρησιμοποιήθηκε.

Παράρτημα Β΄ Αρχεία που συνθέτουν τον κώδικα που δημιουργήθηκε για την επιλογή μεταβλητών στο Πρόβλημα 2: DXR.

Παράρτημα Γ΄ Σημειώσεις επί της Βιβλιογραφίας.

Παράρτημα Α΄

Λογισμικό

Για την εκπόνηση αυτής της εργασίας, από τους υπολογισμούς μέχρι τη συγγραφή και την παρουσίαση, χρησιμοποιήθηκαν τα παρακάτω πακέτα λογιμικού:

MetaboAnalyst 2 Online σουίτα εργαλείων για επεξεργασία δεδομένων μεταβολομικής ελεύθερης πρόσβασης [7, 8]. Διατίθεται με άδεια χρήσης GNU GPL στη διεύθυνση `http://www.metaboanalyst.ca/`. Στην ίδια διεύθυνση μπορούν να βρεθούν και αρκετά σχετικά tutorials.

WEKA 3.7.9 Σουίτα εργαλείων data mining γενικής χρήσης. [9] Η χρήση της περιγράφεται εκτενώς στο βιβλίο των Witten, Frank και Hall. [1] Διατίθεται με άδεια χρήσης GNU GPL στη διεύθυνση `http://www.cs.waikato.ac.nz/ml/weka/`.

LibSVM 3.17 Αυτόνομο λογισμικό μηχανικής μάθησης που υλοποιεί αλγορίθμους Support Vector Machines. [10] Μπορεί να συνεργαστεί με πολλά περιβάλλοντα, μεταξύ άλλων με τη WEKA [11] και το Octave/MATLAB™. Η χρήση του περιγράφεται εκτενώς στη συνοδευτική βιβλιογραφία. Χρήσιμο βοήθημα είναι ο οδηγός των Chih-Wei Hsu κ.α. [26]. Διατίθεται με άδεια χρήσης "modified BSD license" στη διεύθυνση `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`.

GNU Octave 3.6.4 Ολοκληρωμένο πακέτο ελεύθερου λογισμικού για μαθηματικά. Συμβατό σε μεγάλο βαθμό με το εμπορικό πακέτο MATLAB™. Διατίθεται με άδεια χρήσης GNU GPL στη διεύθυνση `http://www.gnu.org/software/octave/`.

TeX Live 2012 Διανομή πακέτων (Xe)(La)TeX. Το TeX είναι ένα σύστημα προετοιμασίας εγγράφων που χρησιμοποιείται ευρέως στην ακαδημαϊκή κοινότητα. Διατίθεται ελεύθερα (άδειες LPPL και GPL) στη διεύθυνση `http://www.tug.org/texlive/`.

Kile 2.1.3 Ολοκληρωμένο περιβάλλον συγγραφής κώδικα TeX. Διατίθεται με άδεια χρήσης GNU GPL στη διεύθυνση `http://kile.sourceforge.net/`.

LibreOffice 4.0 Calc Πρόγραμμα επεξεργασίας λογιστικών φύλλων. Συμβατό σε μεγάλο βαθμό με το εμπορικό πακέτο Microsoft Office Excel™. Διατίθεται με άδεια χρήσης GNU GPL στη διεύθυνση `http://www.libreoffice.org/`.

Inkscape 0.48 Πρόγραμμα επεξεργασίας διανυσματικών γραφικών. Διατίθεται με άδεια χρήσης GNU GPL στη διεύθυνση `http://inkscape.org/`.

Linux Mint 15 Διανομή του λειτουργικού συστήματος Linux. Διατίθεται με άδεια χρήσης GNU GPL κ.α. στη διεύθυνση `http://www.linuxmint.com/`.

Όλα τα παραπάνω ανήκουν στην ευρύτερη κατηγορία ελεύθερου λογισμικού - λογισμικού ανοιχτού κώδικα και είναι διαθέσιμα για όλες τις βασικές πλατφόρμες λειτουργικών συστημάτων. Ο κώδικας που δημιουργήθηκε στα πλαίσια αυτής της εργασίας παρουσιάζεται στο παράρτημα Β΄.

Παράρτημα Β΄

Κώδικες

Για την επιλογή μεταβλητών στο δεύτερο πρόβλημα (DXR) δημιουργήθηκε κώδικας που υλοποιεί έναν γενετικό αλγόριθμο. Εδώ θα βρείτε τα αρχεία που συνθέτουν αυτόν τον κώδικα καθώς και έναν πίνακα με τις μεταβλητές που εμφανίζονται σε αυτόν. Όλα είναι αρχεία του GNU Octave, με κατάληξη `.m`, συμβατά με το εμπορικό πακέτο MATLAB™ και διατίθενται υπό άδεια χρήσης GNU GPLv3. Η δομή και η χρήση του εξηγούνται στην ενότητα 3.7 (σελ. 59).

Β΄.1 Βασικό script GeneticLibSVM

```
% Prepare the workspace (preserve any variables used as log)
clear -x log*

%%% Parameters %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Number of chromosomes for each generation (an even number)
nChrom = 10;
% Probability for a gene to have value "1" intitially
initGeneProb = 0.2;
% Probability for mutation of each gene
mutProb = 0.05;
% Maximum number of generations after the initial
maxGenerations = 100;

% Input data (scaled with svm-scale)
[Labels, Instances] = libsvmread('../dxr.data.scale');
nInst = size(Instances,1);
nAttr = size(Instances,2);

% Configuration for the svmtrain function
% For WEKA wrapper we used: -S 1 -K 2 -D 3 -G 0.0 -R 0.0 -N 0.3
    -M 40.0 -C 1.0 -E 0.001 -P 0.1 -Z
svmconfig = '-s 1 -t 2 -d 3 -g 0.0 -r 0.0 -n 0.3 -m 40.0 -c 1.0
    -e 0.001 -p 0.1 -v 10 -b 1 -q';

%%% Initial population %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Create a random initial population of chromosomes
Chromosomes = round(rand(nChrom,nAttr));

for i=1:nChrom
        for j=1:nAttr
                a=rand(1);
                if ( a>initGeneProb )
                        Chromosomes(i,j) = 0;
                else
                        Chromosomes(i,j) = 1;
                end
        end
end


% Test the corresponding models
```

```
40 for i=1:nChrom
41         for j=1:nInst
42                 LocalInstances(j,:) = Instances(j,:) .*
                       Chromosomes(i,:);
43         end
44         ChromScore(i,1) = i;
45         ChromScore(i,2) = svmTrainAndScore(LocalInstances,
              Labels, svmconfig);
46 end
47
48 %%% Main loop %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
49 for generation=1:maxGenerations
50
51         % Create the rulette for selection (one time)
52         % Accumulated Normalized Score for Chromosomes - an (
              nChrom,2) array
53         ChromScoreNormAcc = ruletteConstruct(ChromScore);
54
55         counter = 0;
56         while counter <= nChrom-2
57                 % counter: how many new chromosomes
58                 % have been produced?
59
60                 % Select two chromosomes
61                 i1 = ruletteSelect(ChromScoreNormAcc);
62                 i2 = ruletteSelect(ChromScoreNormAcc);
63
64                 % Set a random point for crossover
65                 point = floor(rand(1)*nAttr)+1;
66                 % Reproduce the selected chromosomes
67                 newChrom = reproduce(point,i1,i2,Chromosomes);
68
69                 % Mutations
70                 for i=1:2
71                         for j=1:nAttr
72                                 % Toggle the value of a gene
73                                 % with a probability mutProb
74                                 if ( rand(1) < mutProb )
75                                         if (newChrom(i,j) == 0)
76                                             newChrom(i,j) = 1;
77                                         else
78                                             newChrom(i,j) = 0;
79                                         end % end if newChrom
```

```
80                                  end % end if rand(1)
81                          end % end for j
82                  end % end for i
83
84                  % Test the new chromosomes and store the scores
85                  counter = counter + 1;
86                  newChromScore(counter,1) = counter;
87                  for j=1:nInst
88                          LocalInstances(j,:) = Instances(j,:) .*
                                newChrom(1,:);
89                  end
90                  newChromScore(counter,2) = svmTrainAndScore(
                       LocalInstances, Labels, svmconfig);
91
92                  counter = counter + 1;
93                  newChromScore(counter,1) = counter;
94                  for j=1:nInst
95                          LocalInstances(j,:) = Instances(j,:) .*
                                newChrom(2,:);
96                  end
97                  newChromScore(counter,2) = svmTrainAndScore(
                       LocalInstances, Labels, svmconfig);
98
99                  % Add new chromosomes to the new generation
100                 if ( counter == 2 )
101                         newGeneration = newChrom;
102                 else
103                         newGeneration = [ newGeneration ;
                               newChrom ];
104                 end
105
106                 % Clear the (temporary) newChrom
107                 clear newChrom
108         end
109
110         % Replace the old generation with the new
111         Chromosomes = newGeneration;
112         ChromScore = newChromScore;
113
114         % Clear the new generation
115         clear newGeneration
116         clear newChromScore
117
```

```
118 end
119
120 % Output the last generation and scores
121 [ ChromScore, Chromosomes ];
122
123 % Output dominant values for each gene
124 for j=1:nAttr
125         genome(j) = round(mean(Chromosomes(:,j)));
126 end
127 numAttributesUsed = sum(genome);
128 genome = [ 1:nAttr ; genome ]
129
130 % Test the dominant model
131 for j=1:nInst
132         LocalInstances(j,:) = Instances(j,:) .* genome(2,:);
133 end
134 genomeScore = svmTrainAndScore(LocalInstances, Labels,
       svmconfig)
135
136 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
137 %%% Developed by Gerasimos Chourdakis - 2013, 2014 %%%%%%%%
138 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Β΄.2 Συνάρτηση svmTrainAndScore

```
function score = svmTrainAndScore(Instances, Labels, svmconfig)

        score = svmtrain(Labels, Instances, svmconfig);

end
```

Β΄.3 Συνάρτηση ruletteConstruct

```
function ChromScoreNormAcc = ruletteConstruct(ChromScore)

        nChrom = size(ChromScore,1);

        % Calculate the sum of the scores
        ScoreSum = sum(ChromScore(:,2));
        % Normalize the scores between 0 and 1
        ChromScoreNorm = [ ChromScore(:,1) , ChromScore(:,2) ./
            ScoreSum ];
        % Sort ChromScoreNorm rows
        % by 2nd column (score), descending
        ChromScoreNorm = sortrows(ChromScoreNorm,-2);
        % Accumulated score
        clear ChromScoreNormAcc
        for i=1:nChrom
                ChromScoreNormAcc(i,1) = ChromScoreNorm(i,1);
                if (i==1)
                        ChromScoreNormAcc(i,2) = ChromScoreNorm
                           (i,2);
                else
                        ChromScoreNormAcc(i,2) =
                           ChromScoreNormAcc(i-1,2) +
                           ChromScoreNorm(i,2);
                end
        end

end
```

Β′.4 Συνάρτηση ruletteSelect

```
function selectedChrom = ruletteSelect(ChromScoreNormAcc)

        nChrom = size(ChromScoreNormAcc,1);

        % Generate a random number between 0 and 1
        R = rand(1);
        % Select the first chromosome with score greater than R
        for i=1:nChrom
                if ( ChromScoreNormAcc(i,2) > R )
                        selectedChrom = ChromScoreNormAcc(i,1);
                        break
                end
        end

end
```

Β′.5 Συνάρτηση reproduce

```
function newChrom = reproduce(point, i1, i2, Chromosomes)

        nAttr = size(Chromosomes,2);

        newChrom(1,:) = [ Chromosomes(i1,1:point) Chromosomes(
           i2,point+1:nAttr) ];
        newChrom(2,:) = [ Chromosomes(i2,1:point) Chromosomes(
           i1,point+1:nAttr) ];

end
```

Β΄.6 Script πολλαπλών επαναλήψεων logGeneticLibSVM

```
% How many times to repeat GeneticLibSVM script?
logBuffer = 30;

% Repeat GeneticLibSVM script
for logCounter = 1:logBuffer

        GeneticLibSVM;

        if ( logCounter == 1 )
                logGenome = genome;
                logGenomeScore = genomeScore;
                logNumAttributesUsed = numAttributesUsed;
        else
                logGenome = [logGenome ; genome(2,:)];
                logGenomeScore = [logGenomeScore;genomeScore];
                logNumAttributesUsed = [logNumAttributesUsed ;
                   numAttributesUsed];
        end

end

% Calculate the frequency of appearance for each attribute
for i = 1:nAttr
        logGenomeFreq(i,1) = i;
        logGenomeFreq(i,2) = mean(logGenome(2:logBuffer+1,i));
end
logGenomeFreq = sortrows(logGenomeFreq,-2);

% Output
meanAttributesUsed = round(mean(logNumAttributesUsed))
logNumAttributesUsed;

logGenome;

meanGenomeScore = mean(logGenomeScore)
logGenomeScore

logGenomeFreq
```

Β΄.7 Script δοκιμής επιλεγμένων μεταβλητών

```
maxTestAttr = 12

for k=1:maxTestAttr
        TestChrom = zeros(1,nAttr);
        for i=1:k
                TestChrom(1,logGenomeFreq(i,1)) = 1;
        end
        TestChrom;
        for j=1:nInst
                LocalInstances(j,:) = Instances(j,:) .*
                   TestChrom(1,:);
        end
        k
        score = svmTrainAndScore(LocalInstances, Labels,
           svmconfig)
end
```

Β′.8 Μεταβλητές

Πίνακας Β′.1: Μεταβλητές στον κώδικα γενετικής έρευνας

Όνομα	**Διαστάσεις**	**Περιγραφή**
Είσοδοι		
Instances	nInst × nAttr	Πίν. με τις τιμές μεταβλητών κάθε δείγματος
Labels	nInst × 1	Διάνυσμα με την κλάση κάθε δείγματος
nChrom	scalar	Πλήθος χρωμοσωμάτων σε κάθε γενιά
initGeneProb	scalar	Πιθ. ένα γονίδιο να πάρει αρχικά τιμή 1
mutProb	scalar	Πιθ. ένα γονίδιο να μεταλλαχθεί
maxGenerations	scalar	Μέγιστο πλήθος γενεών (κριτήριο τερμ.)
svmconfig	character	Επιλογές της svm-train
GeneticLibSVM		
nInst	scalar	Πλήθος δειγμάτων
nAttr	scalar	Πλήθος μεταβλητών-χαρακτηριστικών
Chromosomes	nChrom × nAttr	Χρωμοσώματα τρέχουσας γενιάς
LocalInstances	nInst × nAttr	Πίν. Instances με κάποιες μηδενικές στήλες
ChromScore	nChrom × 2	Επιτυχία μοντέλου κάθε χρωμ. 1:id, 2:score
+ NormAcc	nChrom × 2	Θέση στον «τροχό» για κάθε χρωμόσωμα
i1, i2	scalar	Ταυτότητες επιλεγμένων χρωμοσωμάτων
point	scalar	Γονίδιο μετά το οποίο θα κοπούν
newChrom	2 × nAttr	Οι απόγονοι της τρέχουσας διασταύρωσης
newChromScore	nChrom × 2	Ακρίβεια μοντέλων νέων χρωμοσωμάτων
newGeneration	nChrom × nAttr	Νέα γενιά χρωμοσωμάτων
Συναρτήσεων		
ScoreSum	scalar	Άθροισμα των score κάθε χρωμοσώματος
ChromScoreNorm	nChrom × 2	Κλάσμα κάθε score προς το άθροισμα
+ Acc	nChrom × 2	Σωρευτικό score - θέση στον «τροχό»
selectedChrome	scalar	Ταυτότητα επιλεγμένου χρωμοσώματος
Έξοδοι		
genome	1 × nAttr	Χρωμόσωμα με τα επικρατέστερα γονίδια
genomeScore	scalar	Ακρίβεια του μοντέλου που δίνει το genome
numAttributesUsed	scalar	Πλήθος γονιδίων με τιμή 1
Καταγραφής		
log<name>		Μεταβλητή καταγραφής της <name>
logBuffer	scalar	Πλήθος επαναλήψεων του γενετικού αλγ.
logGenomeFreq	nAttr × 2	Συχνότητα εμφάνισης κάθε γονιδίου
mean<name>	scalar	Μέση τιμή της αντίστοιχης <name>
maxTestAttr	scalar	Μέγιστο επιθυμητό πλήθος μεταβλητών
TestChrom	1 × nAttr	Χρωμόσωμα με k γονίδια με τιμή 1

Παράρτημα Γ′

Σημειώσεις επί της βιβλιογραφίας

Σε αυτό το παράρτημα θα βρείτε σχόλια για τις βιβλιογραφικές πηγές που χρησιμοποιήθηκαν σε αυτήν την εργασία, τα οποία ίσως φανούν χρήσιμα σε όσους ξεκινούν να ασχοληθούν με αντίστοιχα θέματα.

- Το βιβλίο των Witten κ.α. [1] αποτελεί ένα πλήρες πρακτικό εγχειρίδιο για την εξόρυξη δεδομένων, το οποίο μπορεί άνετα να μελετήσει κάποιος χωρίς προηγούμενη γνώση του αντικειμένου. Αποτελεί παράλληλα έναν πολύ καλό οδηγό για τη WEKA, από τους δημιουργούς της.

- Το άρθρο των Patti κ.α. [2] είναι ένα γενικό εισαγωγικό άρθρο στη Μεταβολομική, το οποίο δίνει το γενικό περίγραμμα της πορείας ενός μεταβολομικού πειράματος, εξηγεί τις διαφορές μεταξύ στοχευμένων και μη στοχευμένων πειραμάτων και παρουσιάζει μερικές από τις προκλήσεις στον τομέα.

- Στο άρθρο των Zacharias κ.α. [3] μελετάται το Πρόβλημα 1 που εξετάσαμε (AKI) και είναι μια εργασία στην οποία εμφανίζονται όλα τα στάδια μιας μεταβολομικής ανάλυσης που συζητήθηκαν εδώ.

- Η βάση δεδομένων Metabolights [4] περιέχει μεγάλη πληθώρα δημοσιευμένων δεδομένων από εργασίες μεταβολομικής (μεταξύ άλλων και της [3]) και είναι μια καλή πηγή για εργαστήρια που δεν διαθέτουν δικά τους δεδομένα.

- Στη διδακτορική διατριβή του Κ. Ιωαννίδη [5] και στο προς δημοσίευση άρθρο των Andreadou κ.α. [6] μελετάται το Πρόβλημα 2 που εξετάσαμε (DXR), κυρίως από την πλευρά των μηχανισμών επίδρασης. Παρουσιάζονται αναλυτικά η πειραματική διαδικασία και στατιστικές αναλύσεις.

- Τα άρθρα [7--11] παρουσιάζουν τα πακέτα λογισμικού που χρησιμοποιήσαμε (MetaboAnalyst, WEKA, LibSVM, WLSVM wrapper).

- Το άρθρο του Nicholson (1999) [12] εισάγει για πρώτη φορά την έννοια της Μεταβολομικής.

- Το άρθρο των Xia κ.α. (2013) [13] είναι ένα πολύ καλό review για τις μεθόδους που εφαρμόζονται αυτή τη στιγμή στην ανάλυση μεταβολομικών δεδομένων για την επιλογή βιοδεικτών, με μεγάλη έμφαση στις καμπύλες ROC. Δίνονται οδηγίες για την παρουσίαση αποτελεσμάτων μεταβολομικών εργασιών και επισημαίνονται συχνά λάθη.

- Το βιβλίο Οργανικής Χημείας του McMurry [14] και το βιβλίο Φυσικοχημείας του Atkins [15] αφιερώνουν από ένα κεφάλαιο το καθένα στην παρουσίαση της μεθόδου NMR. Το σύγγραμμα του McMurry δίνει μια πολύ καλή εισαγωγική ματιά και το βιβλίο του Atkins επεκτείνει, μπαίνοντας βαθύτερα στο φαινόμενο του πυρηνικού μαγνητικού συντονισμού.

- Το εκτενές άρθρο των Lindon κ.α. [16] εξετάζει μια μεγάλη ποικιλία μεθόδων μηχανικής μάθησης για την εύρεση προτύπων σε φάσματα NMR και δίνει σχετικές εφαρμογές.

- Τα δύο άρθρα των Netzeva κ.α. [17] και Stanforth κ.α. [18] παρουσιάζουν μεθόδους προσδιορισμού του Domain of Applicability σε διαφορετικού είδους προβλήματα μάθησης ((Q)SAR) ωστόσο δίνουν καλές ιδέες που θα μπορούσαν να εφαρμοστούν και σε προβλήματα μεταβολομικής.

- Οι σημειώσεις από τις διαλέξεις των Μεγαλοοικονόμου, Μακρή από το Πανεπιστήμιο Πατρών [19], οι οποίες δίνονται ελεύθερα σε ηλεκτρονική μορφή, είναι μια πολύ καλή εισαγωγική πηγή στις μεθόδους εξόρυξης δεδομένων και μάλιστα στην ελληνική γλώσσα. Παρομοίως, πολύ χρήσιμες για εισαγωγή είναι και οι σημειώσεις από το University of Central Florida [20].

- Το εκτενές βιβλίο των Russel & Norvig [21] είναι μια πολύ καλή πηγή για μεθόδους μηχανικής μάθησης καθώς και επίλυσης προβλημάτων βελτιστοποίησης, η οποία είναι διαθέσιμη και στην ελληνική γλώσσα. Αναφέρεται μεταξύ άλλων στα Νευρωνικά Δίκτυα και (πολύ σύντομα) στους Γενετικούς Αλγορίθμους.

- Στη διδακτορική διατριβή και στο σχετικό άρθρο του Α. Αλεξανδρίδη [22, 23] παρουσιάζεται μια σύνδεση Νευρωνικών Δικτύων RBF με Γενετική Έρευνα καθώς και με Προσομοιωμένη Ανόπτηση, για την επιλογή μεταβλητών και τελικά για την εφαρμογή τους σε συστήματα Αυτόματης Ρύθμισης. Στη διατριβή παρουσιάζονται επίσης τρεις αλγόριθμοι για την εκπαίδευση Νευρωνικών Δικτύων RBF.

- Το εκτενές review του Ovidiu Ivanciuc [24] παρουσιάζει με ιδιαίτερα αναλυτικό αλλά και πρακτικό τρόπο τις διάφορες μεθοδολογίες Μηχανών Διανυσμάτων Υποστήριξης, δίνοντας έπειτα κάποιες εφαρμογές στη Χημεία. Συστήνεται ως πολύ καλή αρχή στα SVM.

- Το βιβλίο του Χ. Κυρανούδη [25] δίνει, μεταξύ άλλων μεθόδων έρευνας, μια πολύ κατατοπιστική εισαγωγή στη Γενετική Έρευνα, μαζί με αριθμητικά παραδείγματα από κλασικά προβλήματα Εφοδιαστικής Αλυσίδας, στην ελληνική γλώσσα.
- Ο οδηγός των Hsu κ.α. [26] παρουσιάζει τη χρήση της LibSVM για συνήθεις εφαρμογές και επισημαίνει συχνά λάθη. Προτείνεται για ανάγνωση πριν από τη χρήση της LibSVM.

Βιβλιογραφία

[1] Witten, I. H., Frank, E. & Mark, H. A. *DATA MINING: Practical Machine Learning Tools and Techniques* (Morgan Kaufmann, 2011), 3rd edn.

[2] Patti, G. J., Yanes, O. & Siuzdak, G. Metabolomics: the apogee of the omics trilogy. *Nature Reviews Molecular Cell Biology* **13**, 263–269 (2012).

[3] Zacharias, H. U., Schleym *et al.* Analysis of human urine reveals metabolic changes related to the development of acute kidney injury following cardiac surgery. *Metabolomics* **9**, 697–707 (2013).

[4] Haug, K. *et al.* Metabolights – an open-access general-purpose repository for metabolomics studies and associated meta-data. *Nucl. Acids Res.* (2013). Database available at `http://www.ebi.ac.uk/metabolights/`.

[5] Ιωαννίδης, Κ. Ε. *Επίδραση της ολευρωπαΐνης στην χρόνια καρδιακή ανεπάρκεια που προκαλεί η χορήγηση της αδριαμυκίνης. Μελέτη του μηχανισμού δράσης.* Ph.D. thesis, Τμήμα Φαρμακευτικής ΕΚΠΑ, Αθήνα (2012).

[6] Andreadou, I., Mikros, E., Ioannidis, K. *et al.* Oleuropein prevents chronic doxorubicin-induced cardiomyopathy by suppressing oxidative stress and inflammation and interfering with signaling molecules and cardiomyocyte energy metabolism. Under review.

[7] Xia, J., Psychogios, N., Young, N. & Wishart, D. S. MetaboAnalyst: a web server for metabolomic data analysis and interpretation. *Nucl. Acids Res.* **37**, 652–660 (2009). Software available at `http://www.metaboanalyst.ca/`.

[8] Xia, J., Mandal, R., Sinelnikov, I., Broadhurst, D. & Wishart, D. S. MetaboAnalyst 2.0 - a comprehensive server for metabolomic data analysis. *Nucl. Acids Res.* (2012). Software available at `http://www.metaboanalyst.ca/`.

[9] Hall, M. *et al.* The WEKA Data Mining software: An update. *SIGKDD Explorations* **11** (2009). Software available at `http://www.cs.waikato.ac.nz/ml/weka/`.

[10] Chang, C. & Lin, C. LIBSVM: A library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology* **2**, 27:1–27:27 (2011). Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[11] EL-Manzalawy, Y. & Honavar, V. *WLSVM: Integrating LibSVM into Weka Environment* (2005). Software available at `http://www.cs.iastate.edu/~yasser/wlsvm`.

[12] Nicholson, J. K., Lindon, J. C. & Holmes, E. Metabonomics: understanding the metabolic responses of living systems to pathophysiological stimuli via multivariate statistical analysis of biological nmr spectroscopic data. *Xenobiotica* **29**, 1181–1189 (1999).

[13] Xia, J., Broadhurst, D. I., Wilson, M. & Wishart, D. S. Translational biomarker discovery in clinical metabolomics: an introductory tutorial. *Metabolomics* **9**, 280–299 (2013).

[14] McMurry, J. *Οργανική Χημεία - Τόμος Ι* (Πανεπιστημιακές Εκδόσεις Κρήτης, 2009). Μετάφραση από την αγγλική έκδοση, 1996.

[15] Atkins, P. W. *Φυσικοχημεία - Τόμος ΙΙ* (Πανεπιστημιακές Εκδόσεις Κρήτης, 2009). Μετάφραση από την 5η αγγλική έκδοση, 1989.

[16] Lindon, J. C., Holmes, E. & Nicholson, J. K. Pattern recognition methods and applications in biomedical magnetic resonance. *Progress in Nuclear Magnetic Resonance Spectroscopy* **39**, 1–40 (2001).

[17] Netzeva, T. I., Worth, A. P. *et al.* Current status of methods for defining the applicability domain of (Quantitative) Structure-Activity Relationships. *Altern Lab Anim.* **33**, 155–73 (2005).

[18] Stanforth, R. W., Kolossov, E. & Mirkin, B. A measure of domain of applicability for QSAR modelling based on intelligent K-Means clustering. *QSAR Comb. Sci.* **26**, 837–844 (2007).

[19] Μεγαλοοικονόμου, Β. & Μακρής, Χ. Εξόρυξη Δεδομένων και Αλγόριθμοι Μάθησης - σημειώσεις διαλέξεων 2012-13. Διαθέσιμες στη διεύθυνση `http://mmlabold.ceid.upatras.gr/courses/data_mining/`.

[20] Tappen, M. *Linear Classification* (2010). Lecture for CAP5415 - Computer Vision. Available online at `http://www.cs.ucf.edu/~mtappen/cap5415/lecs/lec6.pdf`.

[21] Russel, S. & Norvig, P. *Τεχνητή Νοημοσύνη: Μια σύγχρονη προσέγγιση* (Κλειδάριθμος, 2004). Μετάφραση από τη 2η αμερικάνικη έκδοση, 2003.

[22] Αλεξανδρίδης, Α. *Ανάπτυξη αλγορίθμων εκπαίδευσης Νευρωνικών Δικτύων για μοντελοποίηση και ψηφιακή Αυτόματη Ρύθμιση μη γραμμικών δυναμικών συστημάτων*. Ph.D. thesis, Σχολή Χημικών Μηχανικών ΕΜΠ, Αθήνα (2003).

[23] Alexandridis, A., Patrinos, P., Sarimveis, H. & Tsekouras, G. A two-stage evolutionary algorithm for variable selection in the development of RBF neural network models. *Chemometrics and Intelligent Laboratory Systems* **75**, 149–162 (2005).

[24] Ivanciuc, O. Applications of support vector machines in chemistry. *Reviews in Computational Chemistry* **23**, 291–400 (2007).

[25] Κυρανούδης, Χ. *Μηχανική Συστημάτων Εφοδιαστικής Διαχείρισης* (Εθνικό Μετσόβιο Πολυτεχνείο, 2005).

[26] Hsu, C., Chang, C. & Lin, C. *A Practical Guide to Support Vector Classification* (2010). Available online at `http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf`.