ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΟΜΕΑΣ ΜΑΘΗΜΑΤΙΚΩΝ

Διπλωματική Εργασία

Μέθοδοι ταξινόμησης για δεδομένα υψηλών διαστάσεων και εφαρμογή σε πείραμα μελέτης πελατειακών σχέσεων (Customer Relationship Management - CRM)

Classification methods in High Dimensional Data Analysis with application in Customer Relationship Management (CRM)

ΧΑΤΖΗΘΕΟΔΩΡΙΔΗ ΣΟΦΙΑ

Επιβλέπων: Κουκουβίνος Χρήστος, Καθηγητής Ε.Μ.Π

Αθήνα, 2015

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Μέθοδοι ταξινόμησης για δεδομένα υψηλών διαστάσεων και εφαρμογή σε πείραμα μελέτης πελατειακών σχέσεων (Customer Relationship Management - CRM)

ΧΑΤΖΗΘΕΟΔΩΡΙΔΗ ΣΟΦΙΑ

Αθήνα, 2015

Ευχαριστίες

Η εκπόνηση της παρούσας διπλωματικής εργασίας πραγματοποιήθηκε υπό την επίβλεψη του Καθηγητή του Ε.Μ.Π κ. Χρήστου Κουκουβίνου, τον οποίο ευχαριστώ θερμά για την δυνατότητα που μου έδωσε να ασχοληθώ με ένα θέμα που ανήκει στα ερευνητικά μου ενδιαφέροντα.

Ιδιαίτερες ευχαριστίες θα ήθελα να εκφράσω στους υποψήφιους διδάκτορες Χριστίνα Παρπούλα και Ανδρουλάκη Μάνο, για την πολύτιμη βοήθεια τους και το συνεχές ενδιαφέρον τους κατά την διάρκεια εκπόνησης της διπλωματικής μου εργασίας.

Παράλληλα, θα ήθελα να εκφράσω την αγάπη μου και την ευγνωμοσύνη μου στους γονείς μου, που με την διαρκή τους υποστήριξη βοήθησαν στην επιτυχή ολοκλήρωση των σπουδών μου.

Εν κατακλείδι, δεν θα μπορούσα να μην ευχαριστήσω τους φίλους και συμφοιτητές μου για την αμέριστη βοήθεια τους και την συμπαράστασή τους.

Χατζηθεοδωρίδη Σοφία
Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών
Αθήνα,2015

Περίληψη

Η εκρηκτική ανάπτυξη της τεχνολογίας, και πιο συγκεκριμένα της πληροφορικής, τις τελευταίες δεκαετίες έχει καταστήσει εύκολη και 'οικονομική' την συσσώρευση τεράστιου όγκου πληροφορίας σχεδόν σε όλους τους τομείς της ανθρώπινης δραστηριότητας. Ωστόσο, οι βάσεις δεδομένων που προκύπτουν από οικονομικές και επιστημονικές δραστηριότητες έχουν πυροδοτήσει νέες εξελίξεις στον τομέα της στατιστικής, καθώς είναι αδύνατον να αναλυθούν με τις κλασσικές μεθόδους στατιστικής συμπερασματολογίας (π.χ μέθοδος ελαχίστων τετραγώνων). Πιο συγκεριμένα, το πρόβλημα της στατιστικής μοντελοποίησης και του εντοπισμού των σημαντικών μεταβλητών σε υψηλών διαστάσεων σύνολα δεδομένων έχει οδηγήσει στην διαδικασία Εξόρυξης Δεδομένων (Data Mining). Η Εξόρυξη Δεδομένων αποτελείται από μια σειρά τεχνικών που βασίζονται σε αλγορίθμους,αναλυτικές και αριθμητικές μεθόδους που επιτρέπουν την παραγωγή μοντέλων πρόβλεψης με αρκετά μικρό σφάλμα. Στην παρούσα διπλωματική θα ασχοληθούμε με τα Δέντρα Αποφάσεων (Decision Trees), τα Τεχνητά Νευρωνικά Δίκτυα (Neural Nets) και τα Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines).

Στο πρώτο κεφάλαιο, θα κάνουμε μια εισαγωγή στις έννοιες του Data Mining παρουσιάζοντας τις βασικές κατηγορίες του και θα προχωρήσουμε σε ανάλυση της διαδικασίας KDD (Knowledge Discovery in Databases). Στη συνέχεια θα γίνει μια σύντομη περιγραφή του προβλήματος της ταξινόμησης αλλά και παρουσίαση των μεθόδων που θα αναλυθούν στα επόμενα κεφάλαια.

Στο δεύτερο κεφάλαιο, αναλύθηκαν τα Δέντρα Ταξινόμησης ως τεχνική εξόρυξης γνώσης. Ειδικότερα μελετήσαμε τους αλγορίθμους των CHAID, CART, C4.5 και QUEST και αναλύσαμε τα χαρακτηριστικά τους, που βασίζονται οι διαφορές τους ενω στο τέλος παρουσιάζονται κάποιες εφαρμογές τους σε παλαιότερα σετ δεδομένων.

Το τρίτο κεφάλαιο αναφέρεται στα Νευρωνικά Δίκτυα, επικεντρώνοντας στα Single Layer Perceptrons και στα Multi Layer Perceptrons (MLPs), ενώ στο τέταρτο κεφάλαιο αναλύονται οι Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines - SVMs). Για τους παραπάνω μη-παραμετρικούς ταξινομητές παρουσιάζεται η κεντρική τους ιδέα καθώς επίσης και το μαθηματικό τους υπόβαθρο με ταυτόχρονη παράθεση αποτελεσμάτων από παλαιότερες έρευνες.

Το πέμπτο κεφάλαιο αφορά την αξιολόγηση των όλων των παραπάνω ταξινομητών με την χρήση μεθόδων όπως η πολλαπλή επικύρωση (Cross-validation) και οι καμπύλες ROC (ειδικότερα το εμβαδόν κάτω από την καμπύλη ROC). Επιπλέον συζητάμε τους όρους ευαισθησίας και ειδικότητας καθώς και την απόδοση των

μοντέλων που παρουσιάσαμε.

Το έκτο και τελευταίο κεφάλαιο της παρούσας διπλωματικής αποτελεί την πρακτική εφαρμογή των παραπάνω μεθόδων σε πραγματικά δεδομένα, όπως αυτά παρουσιάστηκαν στο ετήσιο διαγωνισμό για Data Mining KDD Cup το 2009. Τα δεδομένα αυτά παρασχέθηκαν από την Γαλλική Εταιρεία Κινητής Τηλεφωνίας Orange και αφορούν 100.000 πελάτες της εταιρείας. Μέσω των 230 μεταβλητών που αποτελούν την 'πληροφορία' για κάθε πελάτη έχουμε τρεις διαφορετικούς στόχους. Πρώτον, να προβλέψουμε την πιθανότητα κάποιος πελάτης να αλλάξει πάροχο τηλεφωνίας, δεύτερον να αγοράσει καινούρια προιόντα και υπηρεσίες και τρίτον αν θα ανταποκριθεί στην προβολή καινούριου διαφημιστικού υλικού. Πέραν των προαναφερθέντων, η εργασία μας θα επικεντρωθεί στην ανάδειξη των καταλληλότερων ταξινομητών από αυτούς που παρουσιάσαμε για μια ανάλυση δεδομένων μεγάλων διαστάσεων όπως αυτή που θα εκτελέσουμε προκειμένου να έχουμε όσο το δυνατόν ακριβέστερες προβλέψεις. Για την ανάλυση των δεδομένων χρησιμοποιήσαμε τα λογισμικά SPSS 22 και το πακέτο R.

Abstract

The exponential growth of technology, especially in computer science, in the last decade, has made the accumulation of vast amounts of information on almost every human activity, not only cheap but also easy to access. However, the databases that have occured from scientific and economic activities have triggered new development in the field of statistics, since they cannot be analyzed with conventional ways of statistical conclusion (for example least squares method). Furthermore, the problem of statistical modelling and locating key variables in high dimensional datasets has led to the development of Data Mining. Data Mining consists of a series of "techniques" that are based on algorithms, analytical and numerical methods that allow the construction of prediction models with minimal error. In this thesis, we will focus on Decision Trees, Neural Nets and Support Vector Machines.

In the first chapter,the basic principles of Data Mining are introduced along with the analysis of the KDD (Knowledge Discovery in Databases) procedure, followed by a short discussion of the classification problem in addition to the introduction of the methods that will be analyzed in the following chapters.

In the second chapter, Decision Trees are analyzed as a method of Data Mining. Furthermore, the algorithms of CHAID, CART, C4.5 & QUEST were studied and their key features were analyzed, their differences were underlined and their implemention in older data sets was presented.

In the third chapter, Neural Nets were studied, focusing on Single Layer Perceptrons and Multi Layer Perceptrons (MLPs), while on the fourth chapter Support Vector Machines were introduced. The main aspect for the above non-parametrical classifiers was explained along with their mathematical background while in the same time their application in older studies was presented.

In the fifth chapter,it was discussed how a model produced by the above classifiers can be assessed, referring to methods such as Cross-Validation and ROC curves (especially the area under the curve). Additionally, the sensitivity and specificity measures were discussed along with their role in the evaluation process.

The sixth and final chapter of the current thesis is consisted of the practical application of such models in real data sets , as those were introduced in the annual competition of Data Mining KDD Cup for 2009. These datasets were provided from the French Telecommunications Company Orange and involved 100.000 customers of the company. The information for each customer is con-

sisted of 230 variables and three response variables. The chances of a customer changing service provider was predicted (appetency) , along with the chances of one purchasing new products and services (churn) and lastly, the customer reaction to new advertisment material (up - selling) are our three response variables.

Moreover, this thesis will focus on revealing which of the above classifiers fits the best to the above dataset in order to have the most reliable predictions. For the data analysis SPSS 22 & R package software were used.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Λίστα Αλγορίθμων

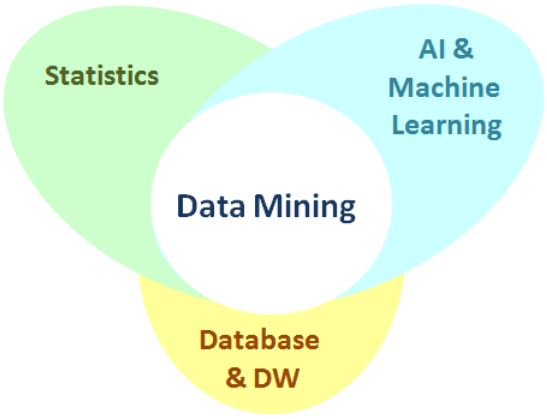
Κεφάλαιο 1

Εισαγωγή

1.1 Τι είναι η Εξόρυξη Δεδομένων (Data Mining)

Με τις αλματώδεις εξελίξεις στον τομέα της πληροφορικής, οι βάσεις δεδομένων πολύ υψηλών διαστάσεων είναι πλέον ο κανόνας και όχι η εξαίρεση. Σε συνδυασμό με τα τελευταία εργαλεία αυτοματοποιημένης συλλογής πληροφοριών, η ψηφιακή αποθήκευση έχει περάσει σε μια νέα εποχή αφού μπορούμε πια με ελάχιστο κόστος να συλλέξουμε όγκο πληροφορίας της τάξεως των terabytes. Ουδέν κακόν όμως αμιγές καλού! Είναι εύκολα αντιληπτό ότι η κατανόηση αυτής της πληροφορίας από πλευράς μας κινείται αντιστρόφως ανάλογα με την αύξηση του όγκου της. Αυτό είναι αποτέλεσμα του γεγονότος ότι οι μέθοδοι της κλασσικής συμπερασματολογίας αποδεικνύονται ανεπαρκείς αφού ο αριθμός των προς εκτίμηση παραμέτρων είναι πολύ μεγαλύτερος από των αριθμό των δεδομένων άρα ο οποιοσδήποτε υπολογισμός καθίσταται αδύνατος.

Αυτή η πρόκληση αποτέλεσε το έναυσμα για την δημιουργία ενός νέου επιστημονικού πεδίου γνωστό ως Εξόρυξη Δεδομένων (Data Mining), το οποίο έχει ως αντικείμενο του την εύρεση 'έξυπνων' μεθόδων που θα μετατρέπουν τα δεδομένα σε χρήσιμες πληροφορίες. Σε ένα τυπικό πρόβλημα data mining, έχουμε ένα σύνολο δεδομένων εκπαίδευσης (training set) για το οποίο γνωρίζουμε τις τιμές των μεταβλητών και του αποτελέσματος και προσπαθούμε να κατασκευάσουμε ένα μοντέλο πρόβλεψης. Το μοντέλο αυτό θα χρησιμοποιηθεί στη συνέχεια για την πρόβλεψη του αποτελέσματος παρόμοιων συνόλων δεδομένων προς εξέταση (test set), για τα οποία είναι γνωστές οι τιμές των μεταβλητών αλλά όχι του αποτελέσματος. Αν και το Data Mining είναι ακόμα ένας εξελισσόμενος κλάδος έχει καταφέρει να κερδίσει το ενδιαφέρον της επιστημονικής κοινότητας, με το MIT Technology Review να την κατατάσσει στις 10 πιο ανερχόμενες τεχνολογίες που θα αλλάξουν τον κόσμο. Ο λόγος για τον οποίο είναι τόσο δημοφιλές δεν έγκειται αποκλειστικά στην χρησιμότητα του αλλά και στο γεγονός ότι συνδυάζει την στατιστική ανάλυση με την μηχανική εκμάθηση, την οποία μέχρι τώρα δεν είχαμε συναντήσει στην ανάλυση μικρότερων συνόλων δεδομένων . Αυτοί οι δύο τομε-



Σχήμα 1.1: Το data mining ως σύνθεση τριών διαφορετικών επιστημονικών τομέων

ίς αποτελούν τους πυλώνες του data mining και χωρίς αυτούς δεν θα μπορούσε να μας προσφέρει ασφαλή αποτελέσματα. Αναλυτικότερα, τα πλεονεκτήματα που προσφέρουν είναι τα εξής:

- **Στατιστική**: μαθηματική ερμηνεία των αποτελεσμάτων με την χρήση τεστ υποθέσεων (hypothesis tests)
- **Μηχανική Εκμάθηση**: με την χρήση κατάλληλων αλγορίθμων εκμάθησης αποκτούμε μια λειτουργική περιγραφή του αποτελέσματος του εκάστοτε προβλήματος. Σε αντίθεση με τον σχετικά περιορισμένο ρόλο της στατιστικής, οι αλγόριθμοι εκμάθησης καλούνται να επιλύσουν και μια σειρά από πολλά προβλήματα όπως: η περαίωση της συνολικής ανάλυσης σε αποδεκτό χρόνο ανάλογα με το μέγεθος των δεδομένων, η ανίχνευση του 'θορύβου' των δεδομένων, ο υπολογισμός πιθανών σφαλμάτων.

Συνεπώς βλέπουμε ότι το data mining δεν είναι παρά το σημείο τομής τριών διαφορετικών κλάδων επιστημών. Ωστόσο, η εμπλοκή αυτών των κλάδων έχει προκαλέσει μια σύγχυση ως προς το τι θα ορίσουμε τελικά ως data mining. Ο ορισμός που αποδεχόμαστε σε αυτή την φάση είναι ο κάτωθι:

> *'Εξόρυξη Δεδομένων είναι η ανάλυση, συνήθως τεράστιων, παρατηρούμενων συνόλων δεδομένων, έτσι ώστε να βρεθούν μη παρατηρηθείσες σχέσεις και να συνοψιστούν τα δεδομένα με καινοφανείς τρόπους, οι οποίοι να είναι κατανοητοί και χρήσιμοι στον κάτοχο των δεδομένων.'*

Ο σκοπός του data mining είναι η εξαγωγή πληροφορίας από μια βάση δεδομένων. Για να καταφέρουμε κάτι τέτοιο έχουμε στην διάθεσή μας τις παρακάτω κατηγορίες μεθόδων :

1. **Μέθοδοι με επίβλεψη (supervised methods)**: Αποτελούνται από αλγορίθμους που χρησιμοποιούνται στην ταξινόμηση των μεταβλητών και την πρόβλεψη της απόκρισης. Στόχος των συγκεκριμένων τεχνικών είναι η μοντελοποίηση της μεταβλητής απόκρισης με βάση τις επεξηγηματικές μεταβλητές. Παραδείγματα τέτοιων τεχνικών είναι τα Δέντρα Ταξινόμησης (Classification Trees) και τα Νευρωνικά Δίκτυα (Neural Nets) με τα οποία θα ασχοληθούμε αναλυτικότερα στα επόμενα κεφάλαια.
2. **Μέθοδοι χωρίς επίβλεψη (unsupervised methods):** Αποτελούνται από αλγορίθμους για τους οποίους δεν υπάρχει μεταβλητή απόκρισης για να προβλεφθεί ή ταξινομηθεί. Ακριβέστερα, επικεντρώνονται στην εξερεύνηση της γενικότερης δομής των μεταβλητών και στην εύρεση πιθανών εγγενών τους σχέσεων. Τέτοιες μέθοδοι είναι οι k-means και τα Kohonen Networks.

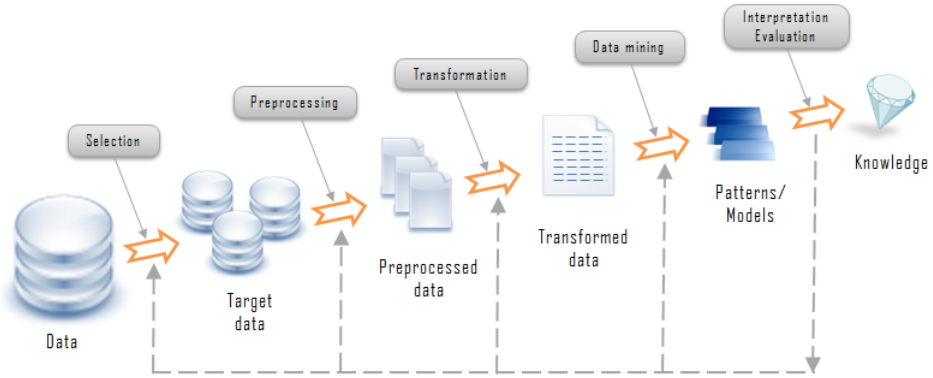
1.2 Η διαδικασία KDD

Όπως αναφέραμε και παραπάνω, μια βάση δεδομένων δεν έχει νόημα ύπαρξης αν δεν μπορεί να μετασχηματιστεί σε γνώση. Η διαδικασία απόκτησης γνώσης από μια βάση δεδομένων KDD (Knowledge Discovery in Databases) κερδίζει όλο και περισσότερο έδαφος τα τελευταία χρόνια. Η KDD είναι μια συγκροτημένη διαδικασία εντοπισμού μοναδικών προτύπων ακριβείας σε μεγάλες και περίπλοκες βάσεις δεδομένων με τελικό στόχο τα πρότυπα αυτά να μας εξασφαλίζουν την μεγαλύτερη δυνατή κατανόηση του εκάστοτε προβλήματος. Η γνώση που προκύπτει απο την διαδικασία αυτή κατηγοριοποιείται ανάλογα με τους στόχους που έχουμε σε κάθε πρόβλημα. Οι έξι βασικές εργασίες που καλείται κάποιος να πραγματοποιήσει ανάλογα με τον στόχο που έχει θέσει είναι οι εξής:

- **Ταξινόμηση (classification)**: εξέταση ενός νέου αντικειμένου και τοποθέτηση του ανάλογα με τις ιδιότητές του σε προκαθορισμένες κλάσεις δεδομένων.
- **Ομαδοποίηση ή Συσταδοποίηση (clustering)**: η κατάτμηση του συνόλου των δεδομένων σε συστάδες και καθορισμός των αντικειμένων που βρίσκονται μέσα σε κάθε συστάδα. Ουσιαστικά πρόκειται για την ανάδειξη ομάδων με όμοια στοιχεία.
- **Πρόβλεψη (prediction)**: μια καινούρια εγγραφή αξιολογείται με βάση κάποιες προβλεπόμενες μελλοντικές τιμές ή τάσεις.
- **Περιγραφή και οπτικοποίηση (description and visualization)**: διερευνητικό ή οπτικό data mining.

Τα βήματα της επεξεργασίας των δεδομένων που ακολουθεί η KDD διαδικασία είναι αυτά που παρουσιάζονται στο (Εικόνα 1.2) και συνοψίζονται ως εξής:

1. **Ανάπτυξη και κατανόηση του πεδίου της εφαρμογής**, συμπεριλαμβανομένης ενδεχομένως και παλαιότερης γνώσης σχετικά με το πρόβλημα, καθώς επίσης και των στόχων των χρηστών.



Σχήμα 1.2: Διαδικασία KDD

2. **Δημιουργία του στοχευμένου συνόλου δεδομένων (target data)** το οποίο θα χρησιμοποιηθεί για την περαιτέρω ανάλυση. Σε αυτό το σημείο είναι πολύ κρίσιμο να γίνει μια σωστή επιλογή των μεταβλητών που θα χρησιμοποιηθούν διότι το παραμικρό λάθος θα επηρεάσει αρνητικά την απόδοση του μοντέλου πρόβλεψης.

3. **Καθαρισμός και επεξεργασία των δεδομένων (data cleaning)**. Σε αυτό το σημείο απομακρύνονται μεταβλητές με μεγάλο ποσοστό ελλιπών παρατηρήσεων ή οποιαδήποτε άλλη αιτία 'θορύβου'.

4. **Μείωση της ποσότητας των υπό εξέταση μεταβλητών (data reduction)** για την εύρεση της καλύτερη αντιπροσώπευσης των δεδομένων. Ειδικά όταν πρόκειται να χρησιμοποιηθούν μέθοδοι με επίβλεψη όπως εδώ, γίνεται επιπλέον διαχωρισμός των μεταβλητών σε δεδομένα εκπαίδευσης (training set), δεδομένα επαλήθευσης (quiz-set validation) και δεδομένα ελέγχου (test dataset).

5. **Επιλογή της εργασίας εξόρυξης γνώσης (data mining)** που θα χρησιμοποιήσουμε για τις ανάγκες του προβλήματος (π.χ ταξινόμηση, πρόβλεψη, ομαδοποίηση κτλ).

6. **Επιλογή των τεχνικών εξόρυξης γνώσης** (π,χ νευρωνικά δίκτυα, λογιστική παλινδρόμηση κτλ) και των αλγορίθμων εκμάθησης που θα χρησιμοποιήσουμε για την αναζήτηση προτύπων στα δεδομένα.

7. **Ερμηνεία των προτύπων που ανακαλύφθηκαν και αξιολόγηση** της αποτελεσματικότητάς τους όσον αφορά την γνώση που θέλουμε με βάση μια σειρά κριτηρίων. Σε αυτό το βήμα, εάν καταλήξουμε σε σημαντικό ποσοστό λάθους στα αποτελέσματά μας θα πρέπει να επιστρέψουμε σε προηγούμενα βήματα για να υπάρξει μια περαιτέρω επεξεργασία των δεδομένων.

8. **Παρουσίαση της γνώσης που έχει εξαχθεί** και ενσωμάτωσή της στο σύστημα. Τέλος, με την χρήση τεχνικών αντιπροσώπευσης αυτής, η πληροφορία εξάγεται ευκρινώς στο χρήστη.

Θα περίμενε κανείς ότι η χρήση της εξόρυξης δεδομένων προορίζεται αυστηρά για μαθηματικούς και οικονομικούς σκοπούς. Κάτι τέτοιο ομώς είναι πολύ μακριά από την πραγματικότητα. Μπορεί η βιομηχανία να αποτελεί την κύρια πηγή τεράστιων βάσεων δεδομένων αλλά το data mining αποδεικνύεται εξαιρετικό εργαλείο και για πληθώρα διαφορετικών τομέων όπως η βιολογία, η χημεία, το μάρκετινγκ, η ιατρική και γενικότερα οποιαδήποτε διαδικασία λήψης αποφάσεων η οποία έχει να κάνει με βάση δεδομένων. Παραθέτουμε κάποια χαρακτηριστικά παραδείγματα:

- **Βιολογία,Χημεία,Ιατρική και άλλες επιστήμες**:

  1. Τα πειράματα γενετικής μελετούν χιλιάδες γονίδια σε ένα δείγμα ιστού.Αυτό επιτρέπει σε ένα ερευνητή να βρει ποια γονίδια είναι ιδιαίτερα δραστήρια σε μια ομάδα πειραμάτων σε σχέση με μια άλλη.
  2. Έλεγχος διαχωρισμού του φυσικού αερίου από το πετρέλαιο με την βοήθεια κανόνων που ρυθμίζουν τις παραμέτρους της διαδικασίας.
  3. Εκτίμηση πιθανότητας υποτροπιασμού του καρκίνου.
  4. Στην αστρονομία, κατασκευάστηκε ένα σύστημα αυτοματοποιημένης καταχώρησης ουράνιων αντικειμένων τα οποία δεν είναι ορατά με το ανθρώπινο μάτι.

- **Διαχείριση ρίσκου και ανάλυση αγοράς:**

  1. *Consumer Relation Management*: μελέτη της συμπεριφοράς του πελάτη.
  2. *Target management*
  3. *Market Basket Analysis*: Το γνωστό πείραμα 'Diapers and Beers'. Έχοντας παρατηρήσει ότι οι καταναλωτές που αγοράζουν πάνες αγοράζουν και μπύρα, επέτρεψε στο κατάστημα για το οποίο έγινε το πείραμα να τοποθετήσει κοντά τα δύο προιόντα. Επιπλέον, τοποθετώντας ανάμεσα τους πατατάκια κατάφερε να αυξήσει τις πωλήσεις και στα τρία είδη.

Βασική προυπόθεση για την λήψη ολοκληρωμένων αποφάσεων είναι τα δεδομένα να έχουν οργανωθεί με συνέπεια (data warehousing). Ειδικότερα, όταν το data mining που γίνεται αφορά παλαιότερα δεδομένα που χρησιμοποιούνται για πρόβλεψη μελλοντικών τάσεων είναι σχεδόν επιτακτικό οι data warehouses να περιέχουν ακριβή ιστορική καταγραφή των δεδομένων.

1.3 Κανόνες για σωστή επεξεργασία των δεδομένων

Κλείνοντας αυτό το κεφάλαιο, θεωρούμε ότι θα ήταν σωστό να παραθέσουμε κάποιους 'χρυσούς' κανόνες, οι οποίοι έχουν προκύψει από πειραματισμό, που πρέπει να έχει κανείς υπόψιν του όταν επεξεργάζεται δεδομένα υψηλών διαστάσεων.

Η πιθανότητα να ανιχνεύσουμε τις πραγματικές επιδράσεις με μεγάλη ακρίβεια είναι πολλές φορές τρομακτικά μικρή

Δεν θα πρέπει να μας ξενίσει το γεγονός ότι στα προβλήματα που θα κληθούμε να αντιμετωπίσουμε δεν θα μπορούμε να προσεγγίσουμε πάντα με ικανοποιητική ακρίβεια την λύση, καθώς ο όγκος των δεδομένων,η ποιότητα τους καθώς και ένα ευρύ φάσμα πιθανών υπολογιστικών αδυναμιών, λειτουργούν ως ανασταλτικοί παράγοντες. Χαρακτηριστικό παράδειγμα είναι αυτό που παρουσιάζεται στην διατριβή του Miller (2010) και αφορά πείραμα γενετικής με 6.319 αρχικές μεταβλητές. Παίρνοντας τα 90%- διαστήματα εμπιστοσύνης για την κατάταξη των 14 πιο σημαντικών μεταβλητών βλέπουμε ότι είναι αρκετά ευρεία, με αποτέλεσμα κατά την επανάληψη του πειράματος να προκύψει αλλαγή της κατάταξης των μεταβλητών ως προς την σημαντικότητα τους στο μοντέλο.

Η υπόθεση της σποραδικότητας του μοντέλου είναι σχεδόν αντίθετη με την πραγματικότητα αλλά είναι πάντα χρήσιμη

Ως σποραδικό μοντέλο ορίζουμε εκείνο το οποίο συμπεριλαμβάνει σχετικά λίγες μεταβλητές σε σχέση με τις αρχικώς διαθέσιμες. Οι ποινικοποιημένες μέθοδοι, οι οποίες τείνουν να υπολογίζουν 'αραιά' μοντέλα, είναι μια ανερχόμενη τάση ειδικά για τα γραμμικά μοντέλα. Το σκεπτικό με το οποίο λειτουργούν είναι ότι ακόμα και αν η πραγματική κατάσταση δεν αντικατοπτρίζεται από ένα 'αραιό' μοντέλο , η πιθανότητα να ενσωματωθούν σε ένα μοντέλο όλες οι μεταβλητές είναι πολύ μικρή και επομένως ένα μοντέλο το οποίο θα συμπεριλαμβάνει μόνο τις μεταβλητές με την μεγαλύτερη επίδραση θα έχει καλύτερη απόδοση.

Σωστή επικύρωση των αποτελεσμάτων

Από την μέχρι τώρα εμπειρία μας στην στατιστική θεωρούμε ότι η φυσική απόληξη της ανάλυσης ενός συνόλου δεδομένων, αφού έχουμε εντοπίσει το κατάλληλο μοντέλο, είναι η επικύρωση του μοντέλου. Αυτό όμως δεν ισχύει για δεδομένα υψηλών διαστάσεων καθώς θα οδηγούσε σε εσφαλμένα αποτελέσματα. Η συνήθης διαδικασία επεξεργασίας δεδομένων επιτάσσει ότι μετά τον εντοπισμό των σημαντικότερων μεταβλητών ακολουθεί η επιλογή του κατάλληλου μοντέλου. Στην περίπτωση των δεδομένων υψηλών διαστάσεων όμως, αν η επιλογή των μεταβλητών με την μεγαλυτερη επίδραση γίνει από ολόκληρο το σύνολο των δεδομένων, ακόμα και αν το μοντέλο τελικά επικυρωθεί, θα έχει γίνει υπερπροσαρμογή (overfitting). Για τον λόγο αυτό λοιπόν, σύμφωνα με την πρόταση του Stone (1974) η χρήση δύο στρωμάτων διασταυρωμένης επικύρωσης συνήθως μας προσφέρει ασφαλέστερα αποτελέσματα.

Δεν υπάρχει μια και μοναδική προσέγγιση που να επιλύει όλα τα προβλήματα υψηλών διαστάσεων

Αν ψάξει κανείς στην πλούσια εγχώρια και διεθνή βιβλιογραφία που αφορά πειράματα ανάλυσης δεδομένων, εύκολα θα διαπιστώσει ότι δεν υπάρχει μια 'συνταγή' που να δουλεύει εξίσου καλά για όλα τα πιθανά σύνολα δεδομένων. Αυτό οδηγεί στην κατάρριψη της αντίληψης των 'καλύτερων' και 'χειρότερων' μεθόδων στα προβλήματα που θα μας απασχολήσουν και ταυτόχρονα αποτελεί ώθηση για νέους ερευνητές να ψάξουν αποδοτικές μεθόδους για πολλά πιθανά σενάρια.

Κεφάλαιο 2

Δέντρα Αποφάσεων (Decision Trees)

2.1 Εισαγωγή στο πρόβλημα της ταξινόμησης

Η ταξινόμηση είναι μια από τις βασικότερες τεχνικές του data mining. Πρόκειται για μέθοδο με επίβλεψη (Supervised method) αφού οι κλάσεις ταξινόμησης και το πραγματικό αποτέλεσμα είναι ήδη γνωστά. Τυπικά παραδείγματα ταξινόμησης είναι ο εντοπισμός spam emails με βάση τον τίτλο τους και το περιεχόμενό τους, η κατηγοριοποίηση κυττάρων ως κακοήθη ή καλοήθη με βάση τα αποτελέσματα της μαγγνητικής τομογραφίας κ.α. Η κεντρική της ιδέα είναι η εξής: ένα νέο άγνωστο αντικείμενο εξετάζεται με βάση τις ιδιότητες του και κατηγοριοποιείται σε ένα σύνολο προκαθορισμένων κλάσεων. Συνεπώς η ταξινόμηση δεδομένων είναι μια διαδικασία η οποία βρίσκει τις κοινές ιδιότητες σε ένα σύνολο υποδειγμάτων σε μια βάση δεδομένων και ταξινομεί αυτά τα αντικείμενα σε διαφορετικές κλάσεις σύμφωνα με ένα μοντέλο ταξινόμησης.

2.1.1 Το μαθηματικό πρόβλημα της ταξινόμησης

Στην παραπάνω παράγραφο αναφερθήκαμε στο μοντέλο ταξινόμησης με το οποίο κατηγοριοποιούμε τα νέα αντικείμενα. Η κατασκευή του μοντέλου γίνεται από μια δειγματική βάση δεδομένων $E = \{t_1, t_2, ..., t_n\}$ όπου $t_1, t_2, .., t_n$ είναι πλειάδες της μορφής $t_{i_1}, t_{i_2}, ..., t_{i_p}$ και καλούνται στοιχεία ή εγγραφές ή παραδείγματα. Τα στοιχεία t_{i_k} με $k = 1, ..., p$ είναι τιμές (αριθμητικές ή διακριτές) οι οποίες αναφέρονται σε χαρακτηριστικά (features) $X_1, X_2, .., X_p$. Επομένως ένα διάνυσμα t_i είναι ένα διάνυσμα χαρακτηριστικών (features vector). Κάθε εγγραφή αποτελείται από το ίδιο σύνολο πολλαπλών χαρακτηριστικών και έχει μια γνωστή ετικέτα (label). Το σύνολο των κλάσεων το συμβολίζουμε ως $C = \{C_1, C_2, ..., C_m\}$.

Ο στόχος της ταξινόμησης είναι να αναλύσει τα δεδομένα του συνόλου εκπαίδευσης και να αναπτύξει μια περιγραφή/μοντέλο για κάθε κλάση χρησιμοποιώντας

τα χαρακτηριστικά που είναι διαθέσιμα στα δεδομένα. Στα μαθηματικά αυτό συνίσταται στην εύρεση μιας συνάρτησης αντιστοίχισης:

$$f : E \to C$$

όπου κάθε t_i αντιστοιχεί σε μια κλάση C_j. Η απεικόνιση αυτή ονομάζεται μοντέλο. Έτσι μια κλάση C_j ορίζεται ως το σύνολο των εγγραφών που κατατάσσονται σε αυτή:

$$C_j = \left\{ \frac{t_i}{f\left(t_i\right)} = C_j, 1 \leq i \leq n, t_i \in D \right\}.$$

Οι κλάσεις αναφέρονται και αυτές με την σειρά τους σε ένα χαρακτηριστικό X_f που ονομάζεται χαρακτηριστικό στόχου (target feature). Πιο συγκεκριμένα, οι κλάσεις αντιστοιχούν στις διαφορετικές τιμές που μπορεί να πάρει το χαρακτηριστικό στόχου. Οι περιγραφές των κλάσεων που προκύπτουν χρησιμοποιούνται στη συνέχεια για να ταξινομήσουν μελλοντικά δεδομένα (test set) στη βάση δεδομένων ή για να αναπτύξουν μια καλύτερη περιγραφή, τις οποίες ονομάζουμε 'κανόνες ταξινόμησης', για κάθε κλάση στη βάση δεδομένων. Συνοψίζοντας συνεπώς, καταλήγουμε στο συμπέρασμα ότι με την ταξινόμηση διαμερίζουμε την βάση δεδομένων E σε κλάσεις ισοδυναμίας και ότι το πρόβλημα της πρόβλεψης ταυτίζεται με το πρόβλημα της ταξινόμησης αλλά με άπειρο αριθμό κλάσεων. Οι μέθοδοι ταξινόμησης που θα αναπτύξουμε σε αυτή την εργασία είναι τα Δέντρα Αποφάσεων, τα Νευρωνικά Δίκτυα και οι Μηχανές Διανυσμάτων Υποστήριξης.

2.2 Δέντρα Αποφάσεων (Decision Trees)

2.2.1 Ορισμός και λειτουργία δέντρων αποφάσεων

Το πρώτο σύστημα μηχανικής εκμάθησης που θα πραγματευτούμε είναι τα Δέντρα Αποφάσεων ή Ταξινόμησης (Desicion/ Classification Trees). Τα δέντρα αποφάσεων είναι από τα πλέον ισχυρά εργαλεία για την ταξινόμηση και πρόβλεψη δεδομένων καθώς προσφέρουν ταχύτητα στον υπολογισμό και σαφήνεια ως προς την γνώση που παρέχει. Πρόκειται για μια μέθοδο λήψης αποφάσεων της οποίας το όνομα βασίζεται στο ότι χρησιμοποιεί ένα δέντρο ως γραφική αναπαράσταση του προγνωστικού μοντέλου και χαρτογραφεί τις παρατηρήσεις σχετικά με ένα αντικείμενο σε συμπεράσματα σχετικά με την τιμή στόχο του αντικειμένου.

Τα δέντρα λειτουργούν με την λογική του 'διαίρει και βασίλευε'. Για να γίνει πιο κατανοητή η λειτουργία τους παραθέτουμε ένα παράδειγμα. Θα δουλέψουμε με το σετ των σπονδυλωτών ζώων ,το οποίο αναλύεται στο από τους Tan, Steinbach και Kumar (2006) και εξετάζει σε ποια κατηγορία (θηλαστικό, μη θηλαστικό) ανήκει ένα σπονδυλωτό ζώο έχοντας ως βάση κάποιες ιδιότητες του. Έστω τώρα ότι έχουμε ένα νέο σύνολο δεδομένων σπονδυλωτών για το οποία όμως δεν ξέρουμε σε ποια κατηγορία ανήκουν. Ένας πιθανός τρόπος προσέγγισης της λύσης του προβλήματος θα ήταν να θέσουμε μια σειρά από ερωτήσεις σχετικά με τα χαρακτηριστικά του κάθε είδους. Για παράδειγμα, θα μπορούσαμε να ρωτήσουμε αρχικά αν

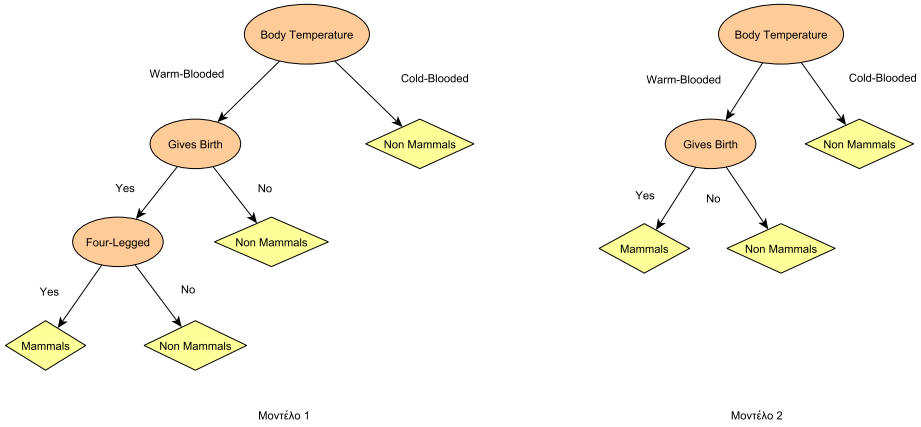
Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Response Variable
eagle	warm-blooded	no	no	no	non-mammal
guppy	warm-blooded	yes	no	no	non-mammal
salamander	cold-blooded	no	yes	no	non-mammal
python	cold-blooded	no	no	yes	non-mammal
cat	warm-blooded	yes	yes	no	mammal
bat	warm-blooded	yes	no	yes	non-mammal
whale	warm-blooded	yes	no	no	mammal
komodo dragon	cold-blooded	no	yes	no	non-mammal
salmon	cold-blooded	no	no	no	non-mammal
porcupine	warm-blooded	yes	yes	yes	mammal

Σχήμα 2.1: Σύνολο εκπαίδευσης για το πρόβλημα ταξινόμησης σπονδυλωτών ζώων

το ζώο είναι θερμόαιμο ή ψυχρόαιμο. Αν είναι ψυχρόαιμο, τότε σίγουρα δεν είναι θηλαστικό ενώ αν είναι θερμόαιμο δεν μπορούμε να πούμε με βεβαιότητα σε ποια κατηγορία ανήκει. Επειδή βλέπουμε ότι δεν παίρνουμε ένα σαφές συμπέρασμα με αυτή την ερώτηση θα προχωρήσουμε σε μια ακόμα: τα θηλυκά αυτού του ζώου γεννάνε τα μικρά τους; Αν ναι, τότε είναι αναμφισβήτητα θηλαστικά ενώ αν όχι μπορούμε να ισχυριστούμε απλά ότι δεν είναι θηλαστικά.

Όπως βλέπουμε, η μέθοδος των στοχευμένων ερωτήσεων για να ανακαλύψουμε την τιμή της μεταβλητής απόκρισης του αντικειμένου είναι μια πιθανή λύση στο πρόβλημα της ταξινόμησης. Αυτή η διαδικασία των ερωτήσεων-απαντήσεων επαναλαμβάνεται μέχρι να καταλήξουμε σε μια απάντηση που να αντιπροσωπεύει μια τιμή της μεταβλητής απόκρισης. Μια πιο οργανωμένη παρουσίαση της διαδικασίας που ακολουθήσαμε για να λάβουμε τις αποφάσεις μας είναι αυτή του δέντρου. Το δέντρο είναι μια ιεραρχική κατασκευή που αποτελείται από κόμβους και από κλαδιά.

Στο επόμενο γράφημα παρουσιάζεται το δέντρο αποφάσεων για το παράδειγμα που παραθέσαμε προηγουμένως:



Σχήμα 2.2: Δύο πιθανά δέντρα αποφάσεων για τα δεδομένα του πίνακα

Το παραπάνω γράφημα μας βοηθάει με την σειρά του να ορίσουμε τις εξής βασικές έννοιες για τα δέντρα αποφάσεων:

- ***Κόμβος-ρίζα:*** είναι ο μοναδικός κόμβος σε ένα δέντρο στον οποίο δεν καταλήγει κανένα κλαδί αλλά από τον οποίο ξεκινούν είτε κανένα είτε περισσότερα κλαδιά.

- ***Εσωτερικοί κόμβοι:*** είναι οι κόμβοι οι οποίοι είναι η κατάληξη ενός κλαδιού και ταυτόχρονα η αρχή ενός ή περισσότερων κλαδιών.

- ***Φύλλα-τερματικοί κόμβοι:*** είναι οι κόμβοι στους οποίους καταλήγει ένα κλαδί αλλά δεν ξεκινάει κάποιο άλλο.

Είναι προφανές ότι οι τερματικοί κόμβοι αντιπροσωπεύουν τις κατηγορίες της μεταβλητής απόκρισης. Οι μη τερματικοί κόμβοι, όπως ο κόμβος-ρίζα και οι εσωτερικοί κόμβοι αντιπροσωπεύουν τα γνωρίσματα με τα οποία γίνεται ο χωρισμός των δεδομένων διαφορετικών χαρακτηριστικών. Με αυτό τον τρόπο, η ταξινόμηση ενός αντικειμένου είναι μια αρκετά άμεση διαδικασία αφού κατασκευαστεί το δέντρο αποφάσεων για το πρόβλημα.

2.2.2 Ο αλγόριθμος κατασκευής των Δέντρων Αποφάσεων

Έχοντας στην διαθεσή μας όλα τα γνωρίσματα (επεξηγηματικές μεταβλητές) ενός συνόλου δεδομένων, είναι προφανές ότι μπορούμε να κατασκευάσουμε πολλά διαφορετικά δέντρα αποφάσεων, τα οποία όμως δεν μας παρέχουν όλα το ίδιο ποσοστό ακρίβειας στη λήψη των αποφάσεων. Από την άλλη πλευρά, η εύρεση ενός αλγορίθμου που να υπολογίζει το βέλτιστο δέντρο είναι μια αρκετά απαιτητική διαδικασία αφού ο όγκος των δεδομένων που δουλεύουμε λειτουργεί ως τροχοπέδη

υπολογιστικά. Παρολ' αυτα, έχουν παρουσιαστεί κατά καιρούς αρκετοί αλγόριθμοι οι οποίοι κατασκευάζουν δέντρα με ένα ικανοποιητικό ποσοστό ακρίβειας και σε ένα εύλογο χρονικό πλαίσιο. Ένας τέτοιος αλγόριθμος είναι αυτός του Hunt, ο οποίος αποτελεί την βάση πολλών σύγχρονων μεθόδων κατασκευής δέντρων αποφάσεων όπως ο CART και ο C4.5 που θα αναλύσουμε σε επόμενη παράγραφο.

Ο αλγόριθμος του Hunt.

Σύμφωνα με τον αλγόριθμο του Hunt, ένα δέντρο αποφάσεων κατασκευάζεται σταδιακά χωρίζοντας σε κάθε βήμα τα δεδομένα σε 'καθαρότερα' σύνολα. Έστω D_t το σύνολο των δεδομένων εκπαίδευσης τα οποία σχετίζονται με τον κόμβο t και $y_c = \{y_1, y_2, ..., y_c\}$ το σύνολο των κατηγοριών της μεταβλητής απόκρισης. Γνωρίζοντας αυτά, προχωράμε στην παρουσίαση του αλγορίθμου του Hunt:

- **1ο βήμα:** Αν όλα τα δεδομένα του D_t ανήκουν στην ίδια κατηγορία y_t , τότε ο tείναι τερματικός κόμβος και έχει την ετικέτα y_t.
- **2ο βήμα**: Αν το σύνολο D_t περιέχει δεδομένα που αντιστοιχούν σε διαφορετικές κατηγορίες της μεταβλητής απόκρισης, τότε εφαρμόζουμε μια συνθήκη δοκιμής ενός γνωρίσματος προκειμένου να γίνει ένας διαχωρισμός δεδομένων σε μικρότερα υποσύνολα. Κάθε υποσύνολο εκφράζει κάθε δυνατό αποτέλεσμα που προκύπτει από την συνθήκη δοκιμής και αντιστοιχεί σε ένα κόμβο-απόγονο. Ο αλγόριθμος επαναλαμβάνεται αναδρομικά για κάθε κόμβο απόγονο.

Μια απλή εφαρμογή του αλγορίθμου είναι το πρόβλημα πρόβλεψης για το αν μπορούμε να παίξουμε γκολφ βασιζόμενοι σε πιθανές καιρικές συνθήκες. Στον παρακάτω πίνακα βλέπουμε τα μετεωρολογικά δεδομένα καθώς επίσης και το δέντρο που προκύπτει για αυτά τα δεδομένα.

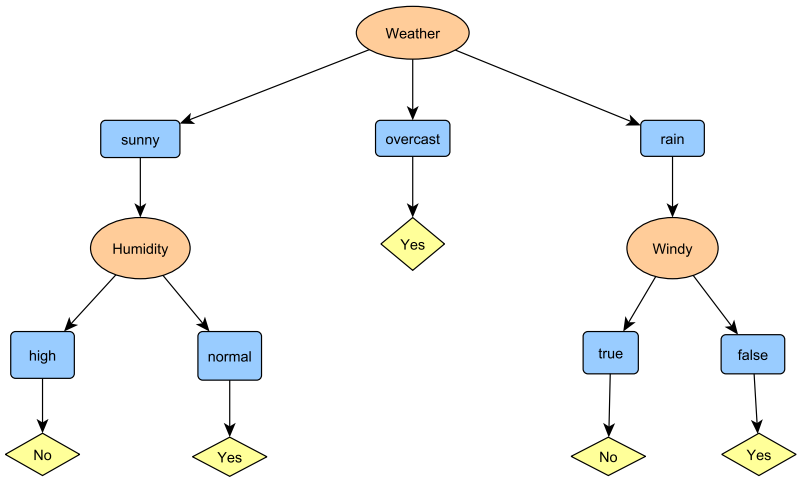
Εξετάζοντας το δέντρο που προέκυψε, βλέπουμε ότι το αρχικό κριτήριο διαχωρισμού των δεδομένων που τοποθετείται στη ρίζα του δέντρου είναι ο καιρός. Από αυτό το διαχωριστικό κριτήριο προκύπτουν τρεις κόμβοι-απόγονοι, οι οποίοι αντιστοιχούν στις κατηγορίες της μεταβλητής του καιρού. Ο δεύτερος κόμβος αφορά όλες τις παρατηρήσεις στις οποίες θα έχουμε συννεφιά και άρα από την Εικόνα 2.4 βλέπουμε ότι θα μπορούμε να παίξουμε. Συνεπώς ο κόμβος αυτός είναι τερματικός. Ο πρώτος και ο τρίτος κόμβος από τον πρώτο διαχωρισμό αφορά τις παρατηρήσεις που αντιστοιχούν σε ηλιοφάνεια και βροχή οι οποίες όμως δεν μας παρέχουν ικανό ποσό πληροφορίας και επομένως δεν μπορούμε να γνωρίζουμε αν μπορούμε να παίξουμε ή όχι. Για την περαιτέρω διερεύνηση αυτών των υποσυνόλων, θα εφαρμόσουμε ως κριτήρια διαχωρισμού την υγρασία και τον αέρα αντίστοιχα, προκειμένου να καταλήξουμε, όπως φαίνεται και στην Εικόνα 2.4, σε τερματικούς κόμβους.

2.2.3 Μέθοδοι διαχωρισμού των δεδομένων ανάλογα με τον τύπο τους

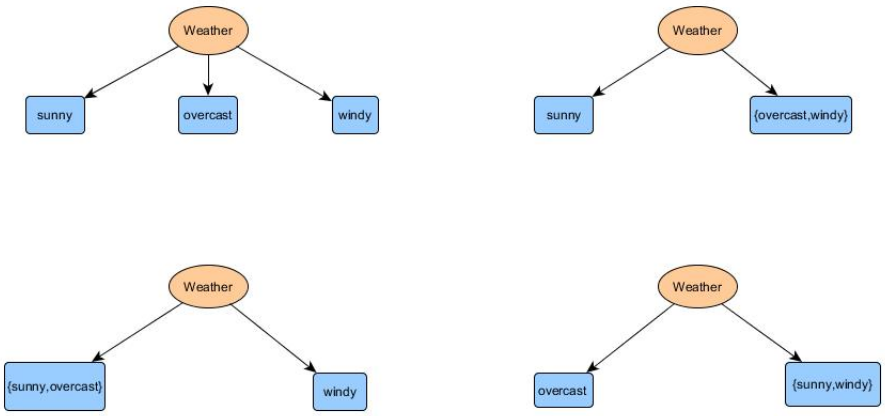
Σε αυτό το σημείο πρέπει να παρατηρήσουμε ότι ο αλγόριθμος κατασκευής του δέντρου πρέπει να διαθέτει μια μέθοδο που να ανιχνεύει τον τύπο των μεταβλητών

Weather	Temperature	Humidity	Windy	Play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rain	mild	high	false	yes
rain	cool	normal	false	yes
rain	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rain	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
rain	mild	high	true	no

Σχήμα 2.3: Σύνολο δεδομένων εκπαίδευσης για το πρόβλημα πρόβλεψης του αν μπορούμε να παίξουμε γκολφ



Σχήμα 2.4: Το δέντρο απόφασης για τα δεδομένα εκπαίδευσης της Εικόνας 2.3



Σχήμα 2.5: Πιθανοί διαχωρισμοί κατηγορικής μεταβλητής

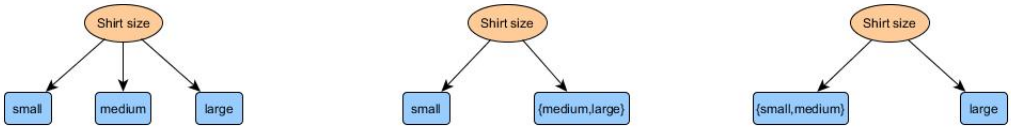
(αριθμητικές, κατηγορικές, ταξικές) και να επιτρέπει τον σωστό διαχωρισμό στις αντίστοιχες κατηγορίες τους. Ας δούμε αναλυτικά τι συμβαίνει για τους διάφορους τύπους των μεταβλητών:

- ***Κατηγορικές μεταβλητές***: θα λέγαμε ότι είναι ο απλούστερος τύπος μεταβλητής, καθώς αν μια τέτοια μεταβλητή επιλεχθεί ως κριτήριο διαχωρισμού, με απλή λογική διαφαίνεται ότι τα υποσύνολα που θα προκύψουν θα αντιστοιχούν σε κάθε μια από τις κατηγορίες της (multiway split). Ωστόσο, αυτός δεν είναι ο μοναδικός διαχωρισμός καθώς αλγόριθμοι σαν τον CART δέχονται μόνο δυαδικές μεταβλητές (binary split). Αυτό πρακτικά σημαίνει ότι για μια μεταβλητή με k κατηγορίες θα έχουμε $2^{k-1} - 1$ πιθανούς διαχωρισμούς.

- ***Αριθμητικές μεταβλητές***: ο διαχωρισμός στις αριθμητικές ή συνεχείς μεταβλητές εκφράζεται σε διαστήματα της μορφής $A < u$ και $A \geq u$ με διάσπαση σε δύο υποσύνολα ή στη μορφή $u_i < A < u_{i+1}$. Η σταθερά u που χρησιμοποιείται για τον διαχωρισμό επιλέγεται από τον αλγόριθμο ως εκείνη που παρέχει τον βέλτιστο διαχωρισμό. Όπως και στις κατηγορικές μεταβλητές, ο διαχωρισμός μπορεί να γίνει και σε παραπάνω από δύο υποσύνολα. Για τις αριθμητικές μεταβλητές αυτό εκφράζεται στην κατασκευή διαδοχικών διαστημάτων, με την προϋπόθεση να διατηρείται αυστηρά η σειρά τους.

- ***Ταξικές μεταβλητές***: οι ταξικές μεταβλητές, ως ένα είδος των κατηγορικών μεταβλητών, συμπεριφέρονται ακριβώς όπως οι κατηγορικές μεταβλητές όσον αφορά τον διαχωρισμό τους, αν επιλεχθούν ως κριτήριο διαχωρισμού από τον αλγόριθμο. Κοινώς, μπορούμε να έχουμε διάσπαση σε δύο και σε παραπάνω υποσύνολα. Η μόνη τους διαφορά εντοπίζεται στο γεγονός ότι στα υποσύνολα που δημιουργούνται κάθε φορά δεν πρέπει να παραβιάζεται η σειρά των τάξεων της μεταβλητής.

Σχήμα 2.6: Πιθανοί διαχωρισμοί συνεχούς μεταβλητής



Σχήμα 2.7: Πιθανοί διαχωρισμοί ταξικής μεταβλητής

2.2.4 Κατασκευή του βέλτιστου δέντρου απόφασης

Το παραπάνω παράδειγμα ήταν αρκετά απλό προκειμένου να κατανοήσουμε την λογική λειτουργίας των δέντρων αποφάσεων. Ωστόσο, η επεξήγηση που έγινε παραπάνω έχει κάποια 'θολά' σημεία. Ένα από αυτά είναι το ότι παραλείψαμε να αναφέρουμε με ποιο τρόπο επιλέγονται τα κριτήρια διαχωρισμού των μεταβλητών μας. Επιπλέον, τα σύνολα δεδομένων που θα αντιμετωπίσουμε στην πραγματικότητα είναι πολύ μεγαλύτερα σε όγκο με άμεση συνέπεια ο αλγόριθμος να δίνει πολύ μεγάλα σε μέγεθος δέντρα και άρα πολύ περίπλοκα μοντέλα. Στην εισαγωγή μας όμως έχουμε εξηγήσει ότι ένα τέτοιο εξιδανικευμένο μοντέλο δεν είναι επιθυμητό αφού δεν έχει μεγάλη προβλεπτική ικανότητα. Οι παρατηρήσεις αυτές αποτελούν τα δύο μέτρα βελτιστοποίησης ενός δέντρου αποφάσεων που θα εξετάσουμε σε αυτή την παράγραφο.

2.2.4.1 Μέθοδοι εύρεσης των βέλτιστων κριτηρίων διαχωρισμού των μεταβλητών

Έστω ότι βρισκόμαστε σε ένα κόμβο του δέντρου ο οποίος έχει ένα σύνολο μεταβλητών και πρέπει να διαλέξουμε την μεταβλητή με την οποία θα διαχωρίσουμε τα δεδομένα μας. Με ποιο τρόπο όμως θα γίνει αυτή η επιλογή; Υπάρχουν πολλές προτάσεις για αυτό το ερώτημα προερχόμενες από διαφορετικές σκοπιές. Εμείς θα επιλέξουμε να απαντήσουμε βασιζόμενοι στον πληθυσμό (σύνολο δεδομένων). Θα θεωρήσουμε τον χώρο X ως τον χώρο των παρατηρήσεών μας και τον χώρο C ως τον χώρο των κλάσεων των μεταβλητών του συνόλου δεδομένων μας. Θεωρώντας το χώρο $X \times C$ που αντιπροσωπεύει τις παρατηρήσεις που θα υπάρχουν σε αυτό το κόμβο μπορούμε να ορίσουμε πλέον μια γνωστή κατανομή πιθανότητας. Αυτό μας δίνει μια οριακή κατανομή πιθανότητας p_k στο χώρο C. Έστω ότι επιλέγουμε την μεταβλητή A ως κριτήριο διαχωρισμού με κατηγορίες $\alpha_1, \alpha_2, .., \alpha_m$ με $i = 1, ..., m$.

Τότε έχουμε μια πιθανότητα κατανομής p_{ik} για τα γνωρίσματα και τις κλάσεις έτσι ώστε κάθε κόμβος-απόγονος που θα αντιστοιχεί στην $A = a_i$ να έχει κατανομή πιθανότητας $p(k|a_i) = {}^{p_{ik}}/_{p_i}$ σε κλάσεις k.

Τώρα είμαστε σε θέση να εξετάσουμε αν οι κόμβοι-απόγονοι είναι 'καθαρότεροι' από τους αρχικούς.Ένα μέτρο για να ελέγξουμε την μη καθαρότητα είναι σύμφωνα με τον Breiman et al. (1984 p.24) το μηδέν αν η πιθανότητα p_j συγκεντρώνεται σε μια κλάση, δηλαδή αν όλες οι παρατηρήσεις βρίσκονται σε μια κλάση, ενώ γίνεται μέγιστη και ίση με 1 όταν η p_j είναι ομοιόμορφη, δηλαδή έχουμε ισόποσα σύνολα παρατηρήσεων. Τα πλέον γνωστά μέτρα μη καθαρότητας που έχουν αναπτυχθεί είναι :

1. **Εντροπία (entropy):**$-\sum p_j \log p_j$
2. **Δείκτης Gini (Gini Index):** $\sum p_i p_j = 1 - \sum p_j^2$
3. **Δείκτης Twoing (Twoing Index):** $\Phi(s,t) = p_L p_R \left[\sum |p(j/t_L) - p(j/t_R)|\right]^2$
4. **Κόστος εσφαλμένης ταξινόμησης:** $1 - \max[p(i/t)]$
5. **Κέρδος πληροφορίας (information gain)**.

Ο δείκτης Gini μπορεί να μεταφραστεί ως ο αναμενόμενος βαθμός σφάλματος εάν η κατηγορία έχει επιλεγεί τυχαία από την κατανομή των κλάσεων του κόμβου. Κατά συνέπεια, ο δείκτης Gini παίρνει την μεγαλύτερη τιμή του, η οποία ισούται με $1 - \frac{1}{k}$ όπου κ οι υποψήφιες κατηγορίες για το κόμβο απόγονο, όταν όλες οι παρατηρήσεις διανέμονται ομαλά διαμέσου των κατηγοριών, και την ελάχιστη τιμή του όταν όλες οι παρατηρήσεις σε ένα κόμβο καταχωρούνται σε μια κατηγορία.

Από την άλλη πλευρά, ο δείκτης Twoing περιορίζεται σε προβλήματα διαχωρισμού των κατηγοριών της μεταβλητής απόκρισης σε δύο μόνο κόμβους και έπειτα στην εύρεση του καλύτερου διαχωρισμού βασιζόμενος στους δύο αυτούς κόμβους. Οι δύο κόμβοι που προκύπτουν ορίζονται ως εξής:

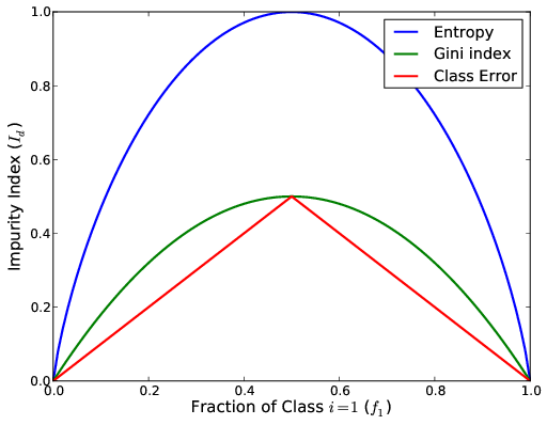
$$C_1 = \{j : p(j/t_L) \geq p(j/t_R)\}$$

και

$$C_2 = C - C_1$$

όπου C είναι το σύνολο των κατηγοριών της μεταβλητής απόκρισης και οι πιθανότητες $p(t/t_L), p(j/t_R)$ είναι ουσιαστικά οι πιθανότητες $p(j/t)$ που αντιστοιχούν στους αριστερούς και δεξιούς κόμβους αντίστοιχα. Συνεπώς, με βάση την συνάρτηση που ορίσαμε παραπάνω, ο καλύτερος διαχωρισμός ς είναι αυτός που μεγιστοποιεί την συνάρτηση αυτή.

Συνοψίζοντας τα παραπάνω κριτήρια ως προς την χρήση τους για τις διάφορες κατηγορίες μεταβλητών βλέπουμε ότι οι δείκτες Gini και Twoing ενδείκνυται για κατηγορικές μεταβλητές ενώ το σφάλμα εσφαλμένης ταξινόμησης για αριθμητικές. Η εντροπία, αντιθέτως δουλεύει εξίσου για όλους τους τύπους μεταβλητών. Όσον αφορά τώρα την ακρίβεια των κριτηρίων, σύμφωνα με τις προτάσεις της διεθνούς βιβλιογραφίας, η εντροπία και ο δείκτης Gini αναδεικνύονται ως τα πιο αξιόπιστα, αφού παρουσιάζουν μεγαλύτερη ευαισθησία στις μεταβολές πιθανότητας των



Σχήμα 2.8: Τα μέτρα μη καθαρότητας κόμβου για την ταξινόμηση δύο τάξεων,ως συνάρτηση του ποσοστού p στην τάξη 2

κόμβων σε σχέση με το σφάλμα εσφαλμένης ταξινόμησης. Για να γίνει αυτός ο ισχυρισμός περισσότερο κατανοητός θα δώσουμε ένα παράδειγμα. Έστω ένα σύνολο 800 παρατηρήσεων και μια μεταβλητή απόκρισης με δύο κατηγορίες. Σε κάθε κατηγορία της απόκρισης αντιστοιχούν 400 παρατηρήσεις. Ένας πιθανός διαχωρισμός των δεδομένων αυτών είναι σε δύο κόμβους των (300,100) και (100,300), ενώ μια άλλη πιθανή διάσπαση είναι πάλι σε δύο κόμβους των (200,400) και (200,0). Και για τις δύο διασπάσεις, το ποσοστό εσφαλμένης ταξινόμησης είναι 0.25, άρα όποια διάσπαση και να επιλέγαμε δεν θα υπήρχε τυπικά πρόβλημα. Ωστόσο στην δεύτερη διάσπαση βλέπουμε ότι σύντομα θα προκύψει καθαρός κόμβος (pure node) οπότε είναι προτιμότερη. Η εντροπία και ο δείκτης Gini για την δεύτερη διάσπαση είναι χαμηλότερα σε σχέση με το ποσοστό μη ταξινόμησης.

Συνεπώς όταν αυξάνεται το δέντρο, είναι προτιμότερο να χρησιμοποιούμε είτε την εντροπία είτε τον δείκτη Gini. Αν θέλουμε να μειώσουμε τώρα την πολυπλοκότητα του δέντρου, οποιοδήποτε από τα τέσσερα μέτρα που παρουσιάσαμε είναι ικανοποιητικό, αλλά έχει επικρατήσει να χρησιμοποιείται το ποσοστό μη ταξινόμησης.

Μια άλλη προσέγγιση για να βρούμε το κατάλληλο κριτήριο διαχωρισμού είναι να συγκρίνουμε το βαθμό μη καθαρότητας ενός κόμβου πριν τον διαχωρισμό με τον βαθμό μη καθαρότητας των κόμβων-απογόνων του. Όσο μεγαλύτερη είναι η διαφορά τους τόσο καλύτερο το κριτήριο διαχωρισμού. Το κέρδος Δ είναι ένα κριτήριο αξιολόγησης ενός διαχωρισμού και ορίζεται ως:

$$\Delta = \mathrm{I}(parent) - \sum_{j=1}^{k} \frac{N(u_j)}{N} I(u_j)$$

όπου $I(\cdot)$ είναι το μέτρο μη καθαρότητας ενός τυχαίου κόμβου, N είναι ο αριθμός των παρατηρήσεων στον αρχικό κόμβο, u_j ο κόμβος- απόγονος και k ο αριθμός των μεταβλητών. Η αύξηση του κέρδους Δ είναι ισοδύναμη με την μείωση του

ισοσταθμισμένων μέσων μέτρων μη καθαρότητας των κόμβων-απογόνων. Τέλος, αν στην παραπάνω εξίσωση χρησιμοποιήσουμε ως μέτρο μη καθαρότητας την εντροπία, τότε παίρνουμε το κέρδος πληροφορίας Δ_{info} (information gain).

2.2.4.2 Υπερπροσαρμογή Μοντέλου (Overfitting)

Το πιο συχνό πρόβλημα που εμφανίζεται όταν εφαρμόζουμε τα δέντρα αποφάσεων ως μέθοδο ταξινόμησης σε δεδομένα μεγάλων διαστάσεων είναι η παραγωγή ενός εκτενούς δέντρου και κατ΄ επέκταση ενός υπερπροσαρμοσμένου στα δεδομένα εκπαίδευσης μοντέλου. Πολύ συχνά αυτού του είδους τα μοντέλα εμφανίζουν πολύ μικρό σφάλμα προσαρμογής, οπότε θα περίμενε κανείς ότι είναι 'ιδανικά'.

Προτού όμως καταλήξουμε στο αν ένα μοντέλο είναι ιδανικό ή μη, θα πρέπει πρώτα να προσδιορίσουμε για ποιό είδος σφάλματος μιλάμε. Σε ένα πρόβλημα ταξινόμησης μεγάλων δεδομένων παρατηρούνται δύο τύποι σφαλμάτων: τα σφάλματα εκπαίδευσης (training errors) και τα γενικευμένα σφάλματα (generalization errors). Αναλυτικότερα,το σφάλμα εκπαίδευσης είναι ο αριθμός των σφαλμάτων εσφαλμένης ταξινόμησης που παρατηρούνται στο σύνολο εκπαίδευσης, ενώ το γενικευμένο σφάλμα είναι τα αναμενόμενο σφάλμα κατά την εφαρμογή του μοντέλου στο σύνολο δεδομένων εξέτασης.

Συνεπώς, σε ένα πρόβλημα μεγάλων δεδομένων που δεν επιζητούμε μόνο την καλή προσαρμογή ενός μοντέλου στο σύνολο εκπαίδεσης αλλά και στο σύνολο εξέτασης, εύκολα αντιλαμβάνεται κανείς ότι επιθυμητό μοντέλο είναι αυτό το οποίο θα έχει εξίσου χαμηλά και τα δύο είδη σφαλμάτων που αναφέραμε παραπάνω.

Στη συνέχεια θα παρουσιάσουμε της βασικότερες αιτίες στις οποίες οφείλεται η υπερπροσαρμογή μοντέλου. Για την μεγαλύτερη κατανόησή τους θα χρησιμοποιήσουμε το σύνολο δεδομένων εκπαίδευσης για το πρόβλημα ταξινόμησης των σπονδυλωτών ζώων σε θηλαστικά και μη θηλαστικά, όπως αυτό παρουσιάστηκε από τους Tan, Steinbach και Kumar (2006).

- **Ύπαρξη 'θορύβου' στα δεδομένα:** Έχοντας υπ΄όψιν το πρόβλημα ταξινόμησης των σπονδυλωτών ζώων που παρουσιάστηκε στην αρχή του κεφαλαίου,παραθέτουμε το αντίστοιχο σύνολο δεδομένων εξέτασης, όπως αυτό παρουσιάζεται στην Εικόνα 2.9:

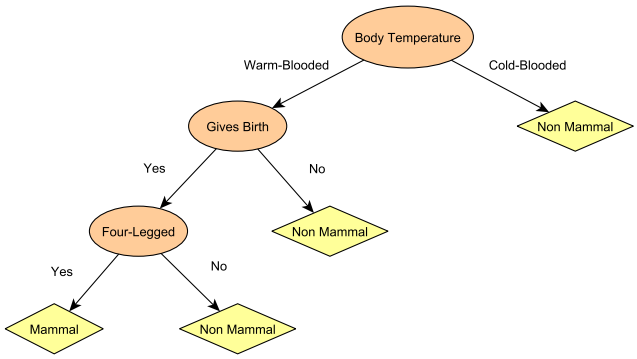
Στην Εικόνα 2.2, βλέπουμε ότι το Μοντέλο 1 δίνει ένα πιο αναλυτικό δέντρο, για το οποίο ενώ το σφάλμα εκπαίδευσης είναι 0, ο ρυθμός σφάλματος για το σύνολο εξέτασης είναι 30%. Με άλλα λόγια έχουμε περιπτώσεις λάθους ταξινόμησης των δεδομένων, όπως για παράδειγμα οι άνθρωποι και τα δελφίνια ταξινομούνται ως μη θηλαστικά διότι οι αντίστοιχες τιμές τους για τα γνωρίσματα Body Temperature, Gives Birth και Four-legged είναι πανομοιότυπες με τις λάθος ταξινομημένες παρατηρήσεις στο σύνολο εκπαίδευσης. Από την άλλη, το Μοντέλο 2 δίνει ρυθμό σφάλματος εκπαίδευσης 20% (σαφώς μεγαλύτερο από το Μοντέλο 1) αλλά ρυθμό σφάλματος για το σύνολο εξέτασης 10% (χαμηλότερο από το Μοντέλο 1). Αυτό σημαίνει ότι εφόσον υπάρχει ένα μοντέλο με χαμηλότερο ρυθμό σφάλματος, το Μοντέλο 1 προφανώς υπερπροσαρμόζεται στα δεδομένα εκπαίδευσης. Πιο συγκεκριμένα το μόνο κριτήριο διαχωρισμού που προκαλεί το σφάλμα στην περίπτωση

Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Response Variable
human	warm-blooded	yes	no	no	mammal
pigeon	warm-blooded	no	no	no	non-mammal
elephant	warm-blooded	yes	yes	no	mammal
leopard shark	cold-blooded	yes	no	no	non-mammal
turtle	cold-blooded	no	yes	no	non-mammal
penguin	cold-blooded	no	no	no	non-mammal
eel	cold-blooded	no	no	no	non-mammal
dolphin	warm-blooded	yes	no	no	mammal
spiny anteater	warm-blooded	no	yes	yes	mammal
gila monster	cold-blooded	no	yes	yes	non-mammal

Σχήμα 2.9: Σύνολο δεδομένων εξέτασης για το πρόβλημα ταξινόμησης σπονδυλωτών ζώων σε θηλαστικά και μη θηλαστικά

Name	Body temperature	Gives Birth	Four-legged	Hibernates	Response Variable
eagle	warm-blooded	no	no	no	non-mammal
guppy	warm-blooded	yes	no	no	non-mammal
platypus	warm-blooded	no	yes	yes	mammal
poorwill	cold-blooded	no	no	yes	non-mammal
salamander	cold-blooded	no	yes	yes	non-mammal

Σχήμα 2.10: Σύνολο δεδομένων εκπαίδευσης για το πρόβλημα ταξινόμησης σπονδυλωτών



Σχήμα 2.11: Το δέντρο ταξινόμησης για τα δεδομένα της Εικόνας 2.9

αυτή είναι το Four-legged, αφού η χρήση του οδηγεί στην εσφαλμένη ταξινόμηση των δεδομένων εκπαίδευσης.

- **Απουσία ικανού αριθμού παρατηρήσεων:** Τα πολύ μικρά σύνολα δεδομένων μπορούν και αυτά με την σειρά τους να οδηγήσουν σε υπερπροσαρμογή μοντέλου διότι οι αλγόριθμοι εκμάθησης έχουν κατασκευαστεί με τέτοιο τρόπο ώστε να εκτελούν εξαντλητικές μεθόδους προκειμένου να μας παρέχουν το βέλτιστο μοντέλο ακόμα και αν δεν έχουν ένα επαρκή αριθμό παρατηρήσεων. Για να γίνει εμφανής ο παραπάνω ισχυρισμός απομονώνουμε ένα κομμάτι του συνόλου εκπαίδευσης που παρουσιάζεται στον πίνακα 2.10:

Με βάση το σύνολο δεδομένων εξέτασης που δώσαμε παραπάνω παρατηρούμε ότι το μοντέλο που προκύπτει από την εκπαίδευση είναι εσφαλμένο γιατί οδηγεί σε λανθασμένη ταξινόμηση των δεδομένων εξέτασης. Πιο συγκεκριμένα, οι άνθρωποι,τα δελφίνια και οι ελέφαντες καταττάσονται στα μη θηλαστικά επειδή είναι

θερμόαιμα και δεν πέφτουν σε χειμερία νάρκη. Ο λόγος που προκύπτει αυτός ο συλλογισμός στο δέντρο απόφασης είναι η ύπαρξη μιας παρατήρησης (αυτής του αετού) η οποία έχει αυτά τα χαρακτηριστικά.

Η αντιμετώπιση της υπερπροσαρμογής του μοντέλου γίνεται με την μείωση του μεγέθους του δέντρου μέσω της διαδικασίας του κλαδέματος (pruning). Ουσιαστικά πρόκειται για την απομάκρυνση κατώτερων κλάδων οι οποίοι δεν συνεισφέρουν σημαντικά στην ακρίβεια του δέντρου ώστε το σφάλμα ταξινόμησης ενός μικρότερου δέντρου να μην διαφέρει σημαντικά από αυτό ενός μεγαλύτερου δέντρου. Το κλάδεμα μπορεί να λάβει χώρα είτε κατά την διάρκεια της κατασκευής του δέντρου είτε μετά την κατασκευή του. Όμως, τις περισσότερες φορές προτιμάται να γίνει μετά το πέρας της κατασκευής του δέντρου διότι πιθανώς να προσφέρει καλύτερα αποτελέσματα από ένα πρόωρο τερματισμό της διαδικασίας όταν το κλάδεμα γίνεται ταυτόχρονα με την κατασκευή του δέντρου.

Όπως εισήχθη για πρώτη φορά από τον Breiman et al.(1984), η διαδικασία του κλαδέματος είναι ουσιαστικά μια διαδικασία μείωσης του κόστους της πολυπλοκότητας του δέντρου χρησιμοποιώντας ένα δείκτη που μετρά το ρίσκο εσφαλμένης ταξινόμησης και την πολυπλοκότητα του δέντρου, αφού θέλουμε να ελαχιστοποιήσουμε και τα δύο. Ο δείκτης αυτός ορίζεται από την κάτωθι σχέση:

$$R_a(T) = R(T) + a \times size$$

όπου ο όρος $R(T)$ είναι το ρίσκο λανθασμένης ταξινόμησης, το $size$ το πλήθος των τερματικών κόμβων του δέντρου T και το a το κόστος της πολυπλοκότητας ανά τερματικό κόμβο για το δέντρο, το οποίο υπολογίζεται από τον αλγόριθμο του κλαδέματος. Στην τιμή $a = 0$ αντιστοιχεί το πλήρως αναπτυσσόμενο δέντρο πριν κλαδευτεί. Όσο αυξάνει το a, τόσο μικραίνει ο αριθμός των τερματικών κόμβων του δέντρου T_a και άρα τόσο μικρότερο το κόστος πολυπλοκότητάς του. Ο αλγόριθμος του κλαδέματος ουσιαστικά παράγει μια ακολουθία δέντρων $T_1, T_{2,...}$ με λιγότερους τερματικούς κόμβους, αφαιρώντας κάθε φορά τον κόμβο $\{t\}$ στον οποίο αντιστοιχεί ο πιο 'αδύναμος' διαχωρισμός. Αν T_t είναι ο υπο-κλάδος του κόμβου t, τότε με βάση τις εξισώσεις:

$$R_a(\{t\}) = R(t) + a \tag{2.1}$$

$$R_a(T_t) = R(T_t) + a \times size \tag{2.2}$$

θα προχωρήσουμε σε αφαίρεση του υπο-κλάδου T_t όταν $R_a(T_t) > R_a(\{t\})$ για μια τιμή του a ,δηλαδή όταν η πολυπλοκότητα του υπο-κλάδου ξεπεράσει την πολυπλοκότητα του κόμβου και συνεπώς αυξάνει την συνολική πολυπλοκότητα του δέντρου. Έχοντας παραθέσει όλα τα παραπάνω ,μπορούμε πλέον να δώσουμε τον αλγόριθμο του κλαδέματος (Αλγόριθμος 1):

Αλγόριθμος 2.1 Αλγόριθμος κλαδέματος

- Όρισε $a_1 = 0$ και ξεκίνα με το δέντρο για το οποίο $T_1 = T(0)$(πλήρως αναπτυσσόμενο δέντρο).
- Αύξησε το a μέχρι το κλάδεμα ενός κλαδιού. Κλάδεψε το κλαδί από το δέντρο και υπολόγισε την εκτίμηση του ρίσκου του δέντρου που λεχει προκύψει.
- Επανέλαβε το προηγούμενο βήμα μέχρι να απομείνει μόνο ο αρχικός κόμβος-ρίζα, αποδίδωντας μια σειρά από δέντρα $T_1, T_2, ..., T_k$.
- Αν επιλεχθεί ο κανόνας του τυπικού σφάλματος, διάλεξε το μικρότερο δέντρο T_{opt} για το οποίο $R(T_{opt}) \leq \min_k R(T_k) + m \times Std.Error(R(T))$.
- Αν δεν επιλεχθεί ο κανόνας του τυπικού σφάλματος, τότε διαλέγεται το δέντρο με την μικρότερη εκτίμηση ρίσκου $R(T)$.

2.2.5 Αξιολόγηση της απόδοσης των Δέντρων Αποφάσεων ως ταξινομητές στο πρόβλημα της ταξινόμησης.

Ο υπολογισμός των γενικευμένων σφαλμάτων που αναφέραμε στην προηγούμενη παράγραφο επιτρέπει στον επαγωγικό αλγόριθμο κατασκευής δέντρων να επιλέξει το καταλληλότερο μοντέλο[1] κατά την ανάλυση των δεδομένων εκπαίδευσης, το οποίο θα έχει ικανοποιητική απόδοση κατά την εφαρμογή του στα δεδομένα εξέτασης. Συχνά, είναι ιδιαίτερα χρήσιμο να υπολογίσουμε την απόδοση του μοντέλου πάνω στα δεδομένα εξέτασης αφού αυτό μας παρέχει μια αμερόληπτη εκτίμηση του γενικευμένου σφάλματος. Επιπλέον, η ακρίβεια του μοντέλου πάνω σε ένα σύνολο δεδομένων εξέτασης μπορεί να χρησιμοποιηθεί με την σειρά της για την σύγκριση της απόδοσης διαφορετικών ταξινομητών πάνω στο ίδιο σύνολο δεδομένων. Κάτι τέτοιο όμως είναι εφικτό μόνο αν είναι γνωστές οι τιμές της μεταβλητής απόκρισης. Στη συνέχεια παραθέτουμε μερικές πολύ γνωστές μεθόδους αξιολόγησης:

Διασταυρωμένη Επικύρωση (Cross Validation)

Έστω ότι έχουμε ένα σύνολο δεδομένων το οποίο χωρίζουμε σε δύο ισόποσα υποσύνολα. Το ένα υποσύνολο είναι το σύνολο εκπαίδευσης και το άλλο το σύνολο εξέτασης. Αφού εκτελέσουμε την ανάλυση των δεδομένων κατά τα γνωστά, εναλλάσουμε τους ρόλους των δύο υποσυνόλων και εκτελούμε πάλι την ανάλυση. Έτσι, κάθε παρατήρηση έχει χρησιμοποιηθεί δύο φόρες: μια στην εκπαίδευση και μια στην εξέταση. Αυτή η προσέγγιση είναι γνωστή και ως διπλή διασταυρωμένη επικύρωση. Το συνολικό σφάλμα υπολογίζεται αθροίζοντας τα επιμέρους σφάλματα από τις δύο αναλύσεις. Η γενίκευση της παραπάνω προσέγγισης είναι η k-οστή διασταυρωμένη επικύρωση, όπου το σύνολο δεδομένων χωρίζεται σε k

[1] δηλαδή εκείνο με την χαμηλότερη πολυπλοκότητα και που δεν είναι ευάλωτο σε υπερπροσαρμογή

ισόποσα σύνολα. Σε κάθε ανάλυση, ένα από τα k υποσύνολα χρησιμοποιείται ως σύνολο εξέτασης και τα υπόλοιπα ως σύνολο εκπαίδευσης. Αυτή η διαδικασία επαναλαμβάνεται k φορές ώστε κάθε σύνολο να έχει χρησιμοποιηθεί μόνο μια φορά ως σύνολο εξέτασης. Ο υπολογισμός των σφαλμάτων γίνεται πάλι με το ίδιο σκεπτικό, αθροίζοντας τα επιμέρους σφάλματα από τις k αναλύσεις.

Μέθοδος Bootstrap

Η μέθοδος Bootstrap είναι μια εξαιρετικά διαδεδομένη μέθοδος στην στατιστική καθώς μας επιτρέπει να μελετήσουμε πολλές ιδιότητες του συνόλου των δεδομένων μας. Από θεωρία, η διασταυρωμένη επικύρωση είναι μια μέθοδος δειγματοληψίας χωρίς αντικατάσταση, το οποίο σημαίνει ότι μια τυχαία παρατήρηση του συνόλου δεδομένων δεν μπορεί να χρησιμοποιηθεί παραπάνω από μια φορά είτε σε σύνολο εκπαίδευσης είτε σε σύνολο εξέτασης. Αντιθέτως η μέθοδος Bootstrap εκτελεί δειγματοληψία με αντικατάσταση, δηλαδή μια τυχαία παρατήρηση η οποία έχει χρησιμοποιηθεί έστω στο σύνολο εκπαίδευσης μπορεί να αφαιρεθεί από αυτό και να τοποθετηθεί στα υπόλοιπα δεδομένα με την ίδια πιθανότητα να επιλεγεί για να επανέλθει στο σύνολο εκπαίδευσης. Το σύνολο εκπαίδευσης που προκύπτει από την προσθαφαίρεση παρατηρήσεων ονομάζεται bootstrap δείγμα. Ό,τι δεν ανήκει στο δείγμα αυτό ανήκει στο σύνολο εκπαίδευσης. Εκτελώντας τον αλγόριθμο εκμάθησης για το σύνολο εκπαίδευσης, το μοντέλο που προκύπτει εφαρμόζεται στο σύνολο εξέτασης προκειμένου να υπολογιστεί η εκτιμήτρια της ακρίβειας ε_i. Η διαδικασία που περιγράφηκε παραπάνω επαναλαμβάνεται b φορές κατά τις οποίες δημιουργούνται b bootstrap δείγματα. Η συνολική ακρίβεια του μοντέλου υπολογίζεται συνδυάζοντας την ακρίβεια που προκύπτει από κάθε bootstrap δείγμα (ε_i) με την ακρίβεια του συνόλου εκπαίδευσης που περιέχει όλες τις παρατηρήσεις των αρχικών δεδομένων (acc_s):

$$acc_{boot} = \frac{1}{b}\sum_{i=1}^{b}(0.632 \times \varepsilon_i + 0.368 \times acc_s)$$

2.2.6 Αλγόριθμοι Δέντρων Αποφάσεων

Κλείνοντας αυτό το κεφάλαιο θα παρουσιάσουμε μερικούς από τους διασημότερους αλγόριθμους δέντρων αποφάσεων με τους οποίους θα ασχοληθούμε σε αυτή την διπλωματική και θα αναλύσουμε τα τεχνικά χαρακτηριστικά τους:

1. Classification And Regression Trees (CART) (βλ. Breiman et al.(1984))
2. Quick Unbiased Efficient Statistical Tree (QUEST) (βλ. Loh and Shih (1997))
3. Chi-Square Automatic Interaction Detection (CHAID) (βλ. Kass (1980))
4. C4.5 (βλ. Quinlan (1993))

2.2.6.1 Αλγόριθμος CART

Ο αλγόριθμος CART παρουσιάστηκε για πρώτη φορά στο ομώνυμο βιβλίο των Breiman & Fisher (1984) και τα αρχικά του σημαίνουν δέντρα ταξινόμησης και παλινδρόμησης. Είναι ένας δυαδικός αλγόριθμος, που σημαίνει ότι σε κάθε κόμβο ο διαχωρισμός γίνεται αυστηρά σε δύο υποσύνολα. Η διαδικασία που ακολουθεί ο CART για την κατασκευή του δέντρου είναι η εξής: χρησιμοποιώντας τον δείκτη Gini ως κριτήριο μη καθαρότητας και θεωρώντας ότι για κάθε διαχωρισμό προκύπτουν δύο κόμβοι-απόγονοι t_L και t_R με p_L και p_R οι αναλογίες των δεδομένων στους κόμβους t_L και t_R αντίστοιχα, η μεταβλητή που επιλέγεται ως βέλτιστο κριτήριο διαχωρισμού είναι αυτή για την οποία η εξίσωση:

$$i\left(t\right) - p_L i\left(t_L\right) - p_R i\left(t_R\right)$$

παίρνει την μεγαλύτερη τιμή. Ο CART δίνει την δυνατότητα αυτόματου 'κλαδέματος' του κόστους της πολυπλοκότητας ενός δέντρου με την μέθοδο της διασταυρωμένης επικύρωσης. Πιο συγκεκριμένα, υπολογίζει το κόστος εσφαλμένης ταξινόμησης για κάθε υπο-δέντρο και κρατάει μόνο αυτά με το μικρότερο εκτιμώμενο κόστος. Εν κατακλείδι, ο CART είναι μεροληπτικός ως προς τις μεταβλητές με διακριτές τιμές αλλά και ως προς τις ελλιπείς τιμές. Όσον αφορά τις μεταβλητές και με δεδομένο ότι ο CART είναι δυαδικός, για μια μεταβλητή m τιμών έχουμε $(m-1)$ πιθανούς διαχωρισμούς αν είναι αριθμητική και $\left(2^{m-1}-1\right)$ διαχωρισμούς αν είναι κατηγορική. Ειδικά για τις κατηγορικές μεταβλητές, αν το m είναι μεγάλο και η μεταβλητή απόκρισης έχει παραπάνω από δύο κατηγορίες δημιουργούνται σοβαρά υπολογιστικά προβλήματα. Η μεροληψία του αλγορίθμου για τις μεταβλητές με ελλιπείς τιμές έχει να κάνει με το κριτήριο μη καθαρότητας, το οποίο βασίζεται σε αναλογίες και όχι στο μέγεθος κάθε δείγματος. Για την ακρίβεια, οι ελλιπείς παρατηρήσεις δεν συμπεριλαμβάνονται στον υπολογισμό του κριτηρίου μη καθαρότητας. Με αυτό τον τρόπο, διευκολύνεται η διαδικασία καθαρισμού ενός κόμβου επιλέγοντας ως κριτήριο διαχωρισμού μια μεταβλητή με μεγάλο ποσοστό ελλιπών παρατηρήσεων.[2]

2.2.6.2 Αλγόριθμος QUEST

Από το όνομά του, ο αλγόριθμος QUEST κατασκευάζει γρήγορα, αμερόληπτα, αποτελεσματικά στατιστικά δέντρα και είναι από τους νεότερους αλγορίθμους που χρησιμοποιούνται στα δέντρα αποφάσεων. Όπως και ο CART, ο QUEST ανήκει και αυτός στην κατηγορία των δυαδικών αλγορίθμων αλλά με κάποιες σημαντικές διαφορές από τον CART. Αποφεύγοντας την μεροληψία επιλογής και τα προβλήματα υπολογισμού των κατηγορικών μεταβλητών που εμφανίζονται στον CART, ο QUEST πρώτα επιλέγει την μεταβλητή διαχωρισμού x_i και μετά επιλέγει το σημείο ή το σύνολο με βάση το οποίο θα διασπαστεί η μεταβλητή. Αυτό είναι εξαιρετικά οικονομικό από πλευράς υπολογισμών διότι ο αλγόριθμος δεν μπαίνει στην διαδικασία να ψάξει όλα τα δυνατά σύνολα ή τα σημεία διάσπασης για κάθε μεταβλητή. Η μεταβλητή διαχωρισμού επιλέγεται, ανάλογα με τον τύπο της μεταβλητής, με

[2] H.Kim,W-Y.Loh (2001)

χρήση τεστ υποθέσεων. Πιο συγκεκριμένα, για τις αριθμητικές μεταβλητές εκτελείται ανάλυση διασποράς με χρήση $F-tests$ και για τις κατηγορικές ανάλυση διασποράς με $X^2-tests$. Η μεταβλητή με την μικρότερη σημαντικότητα επιλέγεται ως μεταβλητή διαχωρισμού για κάθε κόμβο. Τέλος,επειδή ο QUEST είναι δυαδικός αλγόριθμος, αν η μεταβλητή απόκρισης έχει παραπάνω από δύο κατηγορίες τότε αυτές χωρίζονται σε δύο υπερκλάσεις, πριν ο αλγόριθμος προχωρήσει στην επιλογή μεταβλητών διαχωρισμού για τους κόμβους. Ο QUEST μπορεί και αυτός πολλές φορές να καταλήξει σε εκτενή δέντρα αλλά επιτρέπει το αυτόματο κλάδεμα του κόστους πολυπλοκότητας για να μειωθεί το μέγεθός του.

2.2.6.3 Αλγόριθμος C4.5

Ο C4.5 είναι ένας από τους γρηγορότερους αλγόριθμους για τα δέντρα ταξινόμησης. Δεν ανήκει στην κατηγορία των δυαδικών αλγορίθμων, το οποίο σημαίνει ότι επιτρέπει στις κατηγορικές μεταβλητές να διασπαστούν σε όσες κατηγορίες διαθέτουν. Αντίστοιχα, οι αριθμητικές μεταβλητές διασπώνται κατά τα γνωστά σε διαστήματα της μορφής $x_i \leq c$. Ερχόμενοι στο ζήτημα της επιλογής των μεταβλητών που θα χρησιμοποιηθούν ως κριτήρια διαχωρισμού,ο C4.5 επιλέγει την μεταβλητή που έχει την χαμηλότερη εντροπία ή εντελώς ισοδύναμα το μεγαλύτερο κέρδος πληροφορίας. Όπως και για τους προηγούμενους αλγορίθμους, ο C4.5 μπορεί να κατασκευάσει μεγάλα δέντρα αλλά μπορεί να τα συρρικνώσει κλαδεύοντας τα. Ωστόσο, κατά την διαδικασία του κλαδέματος δεν κάνει χρήση της διασταυρωμένης επικύρωσης αλλά σε κάθε κόμβο που κλαδεύει υπολογίζει μια εκτίμηση του σφάλματος. Αυτό έχει ως συνέπεια τα δέντρα που προκύπτουν από αυτό τον αλγόριθμο να έχουν περισσότερους τερματικούς κόμβους συγκριτικά με αυτά άλλων αλγορίθμων. Τέλος, ο C4.5 θεωρείται ένας 'μεροληπτικός' αλγόριθμος,αφού προτιμά τις μεταβλητές που διασπώνται σε περισσότερες κατηγορίες ως κριτήρια διαχωρισμού.

2.2.6.4 Αλγόριθμος CHAID

Ο αλγόριθμος CHAID αναπτύχθηκε από τον Kass το 1980. Τα αρχικά του ονόματος του σημαίνουν X^2- Αυτόματος Ανιχνευτής Αλληλεπίδρασης και ανήκει στις μη δυαδικές, μεροληπτικές μεθόδους. Ο CHAID προσεγγίζει το πρόβλημα κατασκευής δέντρου ταξινόμησης ως εξής: αξιολογεί τις παρατηρήσεις του συνόλου δεδομένων εκπαίδευσης με βάση την σημαντικότητα ενος στατιστικού τεστ. Ανάλογα με τον τύπο των μεταβλητών το στατιστικό τεστ αλλάζει: για αριθμητικές μεταβλητές χρησιμοποιείται F-test και για τις κατηγορικές X^2-test. Χρησιμοποιώντας αυτό το κριτήριο, οι παρατηρήσεις που κρίνονται ως στατιστικά ομογενείς σε σχέση με την μεταβλητή απόκρισης συγχωνεύονται ενώ οι υπόλοιπες μένουν ως έχουν. Η κατασκευή του πρώτου κλάδου του δέντρου βασίζεται πάλι στα αποτελέσματα του τεστ προκειμένου οι κόμβοι-απόγονοι να είναι ομοιογενείς και με την ίδια λογική χτίζεται όλο το δέντρο. Πολλές φορές ωστόσο, ο CHAID δεν μπορεί να βρει τη βέλτιστη μεταβλητή για το κριτήριο διαχωρισμού ακόμα και αν σταματάει τις συγχωνεύσεις όταν οι εναπομείνασες κατηγορίες είναι στατιστικά διαφορετικές.

Εξαντλητικός CHAID

Τα προβλήματα που παρουσιάζει ο CHAID έρχεται να διορθώσει ο εξαντλητικός CHAID, ο οποίος αποτελεί μια τροποποίηση του απλού CHAID και παρουσιάστηκε από τους Biggs, De Ville και Suen (1991). Για την εύρεση της βέλτιστης μεταβλητής διαχωρισμού, ο εξαντλητικός CHAID ακολουθεί διαφορετική τακτική σε σχέση με τον απλό: συνεχίζει τις συγχωνεύσεις κατηγοριών των μεταβλητών μέχρι να καταλήξει σε δύο υπερκλάσεις. Αφού εξετάσει την σειρά των συγχωνεύσεων που πραγματοποιήθηκαν για μια μεταβλητή, θα βρει το σύνολο των κατηγοριών που έχουν την ισχυρότερη σχέση με την μεταβλητή απόκρισης και θα υπολογίσει μια τιμή της μεταβλητής για την σχέση αυτή. Συγκριτικά με τον CHAID, αυτή η στρατηγική καθιστά την διαδικασία περάτωσης του εξαντλητικού CHAID αρκετά χρονοβόρα αλλά υπερέχει σε αξιοπιστία όσον αφορά τα αποτελέσματά της, ενώ πολλές φορές τα αποτελέσματα των δύο αλγορίθμων τυχαίνει να ταυτίζονται.

Κεφάλαιο 3

Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks)

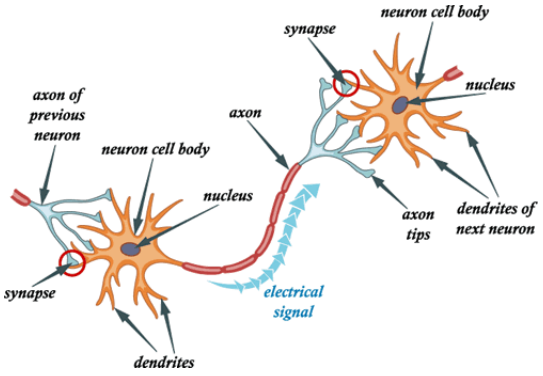
3.1 Το βιολογικό πρότυπο

Ο όρος «Νευρωνικά Δίκτυα» έχει τις ρίζες του στην προσπάθεια που έγινε προκειμένου να βρούμε μαθηματικά μοντέλα τα οποία να προσομοιάζουν τον τρόπο επεξεργασίας της πληροφορίας από τον ανθρώπινο εγκέφαλο (McCullogh and Pitts ,1943; Widrow and Hoff, 1960; Rosenblatt, 1962; Rumelhart et al ,1985). Η εμπλοκή του βιολογικού παράγοντα έκανε τα Νευρωνικά Δίκτυα εξαιρετικά δημοφιλή στην επιστημονική κοινότητα με αποτέλεσμα όμως το ενδιαφέρον να μετατοπιστεί από την εύρεση πρακτικών εφαρμογών για την αναγνώριση προτύπων σε δεδομένα. Ειδικότερα τα Νευρωνικά Δίκτυα περιλαμβάνουν ένα ευρύ φάσμα μεθόδων οι οποίες έχουν μια ασθενή σύνδεση με το βιολογικό μοντέλο αφού συνήθως δεν το ακολουθούν πιστά. Οι ορισμοί για τα νευρωνικά δίκτυα ποικίλλουν όσο και οι τομείς στους οποίους χρησιμοποιούνται. Αν και δεν υπάρχει κάποιος ακριβής ορισμός που να περιγράφει όλη την οικογένεια μεθόδων που ανήκουν στα νευρωνικά δίκτυα, παραθέτουμε μια γενική περιγραφή, η οποία διατυπώθηκε από τον Haykin (1998):

> *Νευρωνικό δίκτυο είναι ένας μαζικός παράλληλος διανεμημένος επεξεργαστής ο οποίος εκ φύσεως αποθηκεύει εμπειρική γνώση και την καθιστά διαθέσιμη για χρήση. Προσομοιάζει τον ανθρώπινο εγκέφαλο σε δύο τομείς:*

- *η γνώση αποκτάται από το δίκτυο μέσω μιας διαδικασίας εκμάθησης,*
- *οι ενδονευρωνικές συνδέσεις χρησιμοποιούνται για την διαφύλαξη της γνώσης.*

Ο ανθρώπινος εγκέφαλος είναι ένας ιδιαίτερα πολύπλοκος, μη γραμμικός και παράλληλος ηλεκτρονικός υπολογιστής. Έχει την ικανότητα να οργανώνει τους νευ-



Σχήμα 3.1: Αναπαράσταση των ανθρώπινων νευρώνων

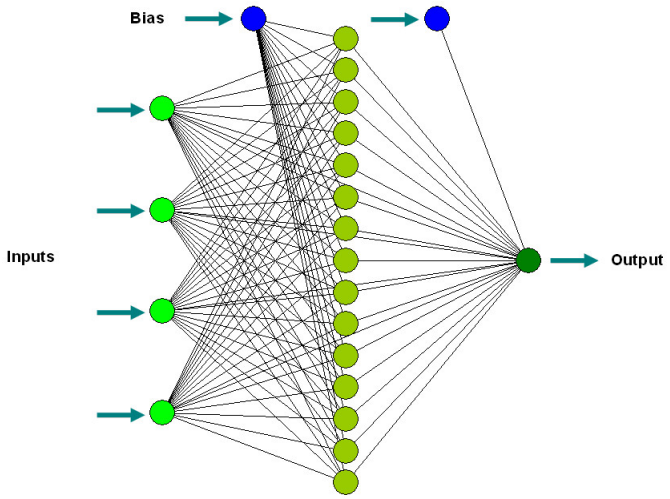
ρώνες με τέτοιο τρόπο προκειμένου να εκτελέσει συγκεκριμένους υπολογισμούς όπως η αντίληψη (perception), η κίνηση, η αναγνώριση προτύπων με ασύγκριτη ταχύτητα. Αποτελείται από 100 δισεκατομμύρια νευρικά κύτταρα, τους νευρώνες, τα οποία συνδέονται μεταξύ τους με την βοήθεια ινών, γνωστές και ως άξονες.

Οι άξονες λειτουργούν ως πύλη εξόδου, μεταφέροντας τα διάφορα νευρικά ερεθίσματα από τον ένα νευρώνα στον άλλο, όταν οι νευρώνες διεγείρονται. Οι νευρώνες-παραλήπτες συνδέονται με τους άξονες μέσω των δενδριτών, οι οποίοι αποτελούν επεκτάσεις του κυτταρικού σώματος του νευρώνα. Το σημείο επαφής ενός δενδρίτη με τον άξονα καλείται σύναψη. Έχει αποδειχθεί ότι η «εκμάθηση» του ανθρώπινου εγκεφάλου πραγματοποιείται με την αλλαγή έντασης στην συναπτική σύνδεση μεταξύ των νευρώνων μετά από επαναλαμβανόμενες διεγέρσεις από το ίδιο ερέθισμα. Πιο αναλυτικά, έστω ότι ένας νευρώνας Α συλλέγει ένα ηλεκτρικό φορτίο που δέχεται από κάθε σύναψη στους δενδρίτες του ζυγίζοντας το εισερχόμενο φορτίο με το αντίστοιχο συναπτικό του βάρος. Όσο πιο ισχυρή είναι η συναπτική σύνδεση τόσο πιο σημαντική είναι η συμβολή του φορτίου αυτού στο συνολικό άθροισμα. Αν δε το συνολικό άθροισμα φορτίων από τις συνάψεις ξεπερνάει κάποιο όριο, τότε ο άξονας του Α ξεκινάει να παράγει ηλεκτρικούς παλμούς (firing) με μεγάλη συχνότητα. Στην περίπτωση που το συνολικό φορτίο δεν ξεπερνάει το όριο αυτό, ο νευρώνας παράγει πολύ «αραιούς» παλμούς και χαρακτηρίζεται ως αδρανής. Τέλος, κάθε παλμός έχει ένα συγκεκριμένο χρονικό πλάτος t_p ενώ ο νευρώνας χρειάζεται ένα ελάχιστο χρόνο ανάπαυσης t_r. Συνεπώς, ο μέγιστος αριθμός παλμών δεν ξεπερνά το όριο:

$$FiringFrequency < \frac{1}{t_p + t_r}.$$

Στη συνέχεια οι παλμοί μέσω του άξονα τροφοδοτούν τους νευρώνες που συνδέονται με τον νευρώνα Α. Κλείνοντας, είναι βασικό να παρατηρήσουμε ότι όλη η διαδικασία που περιγράψαμε παραπάνω γίνεται αποκλειστικά προς μια μόνο κατεύθυνση και δεν υπάρχει αμφίδρομη επικοινωνια των νευρώνων.

Εντελώς ανάλογα με την δομή του ανθρώπινου εγκεφάλου λειτουργούν και τα Τεχνητά Νευρωνικά Δίκτυα, χωρίς ωστόσο να καταφέρνουν να προσομοιάσουν



Σχήμα 3.2: Σχηματική αναπαράσταση ενός Τεχνητού Νευρωνικού Δικτύου με ένα κρυφό στρώμα (hidden layer).

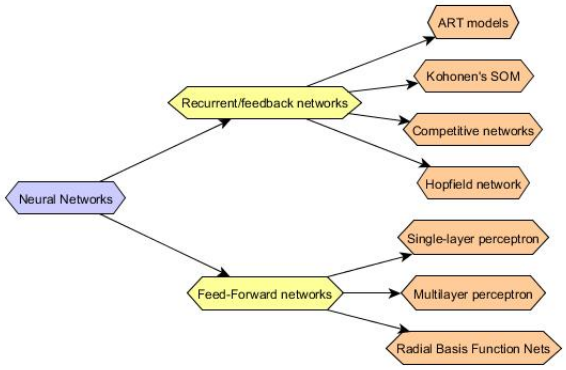
πλήρως την πολυπλοκότητά της ανθρώπινης σκέψης. Τα Τεχνητά Νευρωνικά Δίκτυα ανήκουν στην κατηγορία των μεθόδων εκμάθησης που αναπτύχθηκαν τόσο από την στατιστική όσο και από τον τομέα της τεχνητής νοημοσύνης. Είναι ένα εργαλείο με ποικίλες εφαρμογές στην εξόρυξη δεδομένων λόγω της δυναμικής τους, της ευελιξίας τους και της ευκολίας στην χρήση τους. Στόχος τους είναι η παραγωγή γραμμικών μοντέλων με βάση τις εισροές, δηλαδή τις επεξηγηματικές μεταβλητές, και στην πορεία η μοντελοποίηση της μεταβλητής απόκρισης (μεταβλητή-στόχο) ως ένας μη γραμμικός συνδυασμός των εισροών.

Όπως βλέπουμε και από την Εικόνα 3.2, τα Τεχνητά Νευρωνικά Δίκτυα είναι ουσιαστικά μια μέθοδος ταξινόμησης ή αλλιώς μια μέθοδος δύο βημάτων παλινδρόμησης που αντιπροσωπεύεται από ένα διάγραμμα δικτύου. Το διάγραμμα αυτό χωρίζεται σε τρία βασικά στρώματα. Σε κάθε στρώμα υπάρχει ένα σύνολο κόμβων, οι οποίοι αντιπροσωπεύουν τους νευρώνες:

- *Στρώμα εισόδου (input layer):* στο σημείο αυτό εισάγονται στο δίκτυο όλες οι επεξηγηματικές μεταβλητές του προβλήματος που μελετάμε μέσω των νευρώνων εισόδου. Το πλήθος των νευρώνων εισόδου ταυτίζεται με το πλήθος των μεταβλητών.

- *Κρυφό στρώμα (hidden layer):* αυτά τα στρώματα παρεμβάλλονται μεταξύ του στρώματος εισόδου και εξόδου και η ύπαρξη τους ή μη, καθώς και ο αριθμός τους εξαρτάται από το είδος δικτύου που δουλεύουμε. Η λειτουργία τους είναι η κωδικοποίηση των δεδομένων που εισάγονται στο στρώμα εισόδου και ο καθορισμός των εξόδων του δικτύου.

- *Στρώμα εξόδου (output layer):* στο στρώμα αυτό παρουσιάζονται οι έξοδοι (αποτελέσματα) του δικτύου. Ο αριθμός των νευρώνων εδώ καθορίζεται από



Σχήμα 3.3: Μια ταξινόμηση των Feed-forward και Recurrent/Feedback δικτύων.

το πλήθος των μεταβλητών απόκρισης (αν είναι ποσοτικές) ή το πλήθος των κατηγοριών μιας κατηγορικής μεταβλητής.[1]

Στις επόμενες παραγράφους θα αναφερθούμε εκτενώς στις βασικές κατηγορίες των νευρωνικών δικτύων καθώς και στο μαθηματικό υπόβαθρο που βρίσκεται πίσω από αυτές.

3.2 Ταξινόμηση Τεχνητών Νευρωνικών Δικτύων.

Ο κλάδος των τεχνητών νευρωνικών δικτύων, όπως αναφέρθηκε και παραπάνω, είναι ταχύτατα αναπτυσσόμενος, περιλαμβάνοντας πλέον και μεθόδους οι οποίες έχουν μόνο μια τυπική σχέση με το ισοδύναμο βιολογικό μοντέλο αλλά χωρίς ωστόσο να στερούνται αποτελεσματικότητας για τις χρήσεις που προορίζονται. Για παράδειγμα, κάποια νευρωνικά δίκτυα λειτουργούν ως ≪προσαρμοστικά συστήματα≫ και χρησιμοποιούνται με αρκετά καλή προσέγγιση για την μοντελοποίηση διαρκώς μεταβαλλόμενων πληθυσμών και περιβαλλόντων. Οι διαφορές που εντοπίζονται στους διάφορους τύπους νευρωνικών δικτύων έχουν να κάνουν κυρίως με κάποιες αλλαγές στην αρχιτεκτονική τους αλλά και με μικρότερης κλίμακας διαφορές, όπως λόγου χάρη η χρήση ελαφρώς τροποιημένων αλγορίθμων εκμάθησης. Ένας αρχικός διαχωρισμός που μπορεί να γίνει στο σύνολο των νευρωνικών δικτύων και οφείλεται στην διαφορετική αρχιτεκτονική, είναι σε feed-forward και recurrent/feedback δίκτυα.

Τα feed-forward νευρωνικά δίκτυα είναι τα πρώτα νευρωνικά δίκτυα που αναπτύχθηκαν και είναι αναμφισβήτητα ο πιο απλός τύπος νευρωνικού δικτύου που μπορούμε να συναντήσουμε. Σε αυτά τα δίκτυα η πληροφορία, όπως και στο βιολογικό μοντέλο, κινείται μόνο προς μια κατεύθυνση-προς τα εμπρός (εξ' ου και

[1] εξαιρείται η περίπτωση της παλινδρόμησης, όπου το δίκτυο μπορεί να διαχειριστεί μόνο μια μεταβλητή εξόδου (είτε ποσοτική είτε κατηγορική)

ο όρος feed-forward). Από τους εισερχόμενους κόμβους, τα δεδομένα κινούνται προς τους κρυφούς κόμβους (αν υπάρχουν) και στη συνέχεια προς τους κόμβους εξόδου. Δεν υπάρχουν κύκλοι ή βρόγχοι σε αυτό το είδος δικτύου. Γενικότερα, τα feed-forward νευρωνικά δίκτυα είναι από τα πλέον διαδεδομένα και απλά στην χρήση τους και είναι κατάλληλα για μεγάλο εύρος εφαρμογών. Εξαιρώντας τα Single -layer perceptron και Multilayer perceptron δίκτυα με τά οποία θα ασχοληθούμε εκτενώς σε αυτή την διπλωματική, θα παρουσιάσουμε ένα ακόμα είδος νευρωνικού δικτύου:

- *Radial Basis Function (RBF) network:* Τα RBF δίκτυα παρουσιάστηκαν για πρώτη φορά το 1988 από τους Broomhead και Lowe. Πρόκειται για ένα δίκτυο με στρώμα εισόδου, εξόδου και ένα κρυμμένο στρώμα. Σε κάθε νευρώνα του κρυφού στρώματος εφαρμόζεται μια ακτινική συνάρτηση βάσης (radial basis function). Η έξοδος του δικτύου είναι ένας γραμμικός συνδυασμός των ακτινικών συναρτήσεων βάσης των επεξηγηματικών μεταβλητών και των παραμέτρων του δικτύου. Τέτοιου είδους δίκτυα εφαρμόζονται συχνά για την προσέγγιση συναρτήσεων, την πρόβλεψη χρονοσειρών και την ταξινόμηση δεδομένων.

Σε αντίθεση με τα feed-forward δίκτυα, τα recurrent ή feedback δίκτυα είναι μοντέλα τα οποία επιτρέπουν την αμφίδρομη ροή της πληροφορίας. Ενώ στα feed-forward δίκτυα η διάδοση των δεδομένων γίνεται «γραμμικά» από την είσοδο στην έξοδο, τα recurrent δίκτυα μπορούν να διαδόσουν τα δεδομένα από τα τελευταία στάδια της διαδικασίας σε προήγουμενα. Επιπλέον, τα recurrent δίκτυα λειτουργούν ως επεργαστές γενικευμένων ακολουθιών δεδομένων. Αυτό σημαίνει ότι μπορούν να χρησιμοποιήσουν την εσωτερική τους μνήμη για να επεξεργαστούν τυχαίες ακολουθίες εισερχόμενων δεδομένων. Το χαρακτηριστικό αυτό επιτρέπει την εφαρμογή τους σε θέματα όπως η αναγνώριση άτμητου γραπτού κειμένου (unsegmented connected handwriting recognition), όπου προσφέρουν τα καλύτερα δυνατά αποτελέσματα που υπάρχουν. Κάποια πολύ γνωστά είδη δικτύων που υπάγονται στην κατηγορία των recurrent νευρωνικών δικτύων είναι τα κάτωθι:

- *Hopfield network:* Τα Hopfield δίκτυα είναι απλά νευρωνικά δίκτυα ενός στρώματος ο οποίος χρησιμοποιείται για αναγνώριση προτύπων. Ειδικότερα, ο αλγόριθμος εκπαίδευσης που χρησιμοποιεί «εκπαιδεύει» το δίκτυο να αναγνωρίζει πρότυπα. Συνεπώς, μόλις το δίκτυο αναγνωρίσει ένα πρότυπο στην έξοδο, ενημερώνει αυτόματα και το προηγούμενο στρώμα .

- *Kohonen Self-Organizing Map (SOM) network:* Το SOM δίκτυο είναι ένα νευρωνικό δίκτυο δύο στρωμάτων (εισόδου και εξόδου) που εφαρμόζει την στρατηγική «ο νικητής τα παίρνει όλα» στο στρώμα της εξόδου. Αντί να εμφανίζει το αποτέλεσμα του κάθε νευρώνα εξόδου, ο νευρώνας με το υψηλότερο αποτέλεσμα είναι αυτός που «κερδίζει» και αποτελεί ουσιαστικά το αποτέλεσμα από την ανάλυση. Τα SOMs δίκτυα χρησιμοποιούνται κυρίως σε προβλήματα ταξινόμησης, όπου οι νευρώνες εξόδου αντιπροσωπεύουν τις κατηγορίες στις οποίες θα ταξινομηθούν τα δεδομένα.

- *Ανταγωνιστικά (Competitive) δίκτυα:* Είναι η ευρύτερη κατηγορία δικτύων στην οποία ανήκουν και τα SOM δίκτυα, η οποία χρησιμοποιεί ανταγωνιστική εκμάθηση (competitive learning). Στα δίκτυα αυτά, οι νευρώνες εξόδου ανταγωνίζονται μεταξύ τους ως προς το ποιος από αυτούς θα ενεργοποιηθεί. Συνεπώς, κάθε στιγμή μόνο ένας νευρώνας εξόδου είναι ενεργός (η στρατηγική του «ο νικητής τα παίρνει όλα»). Μέσω της ανταγωνιστικής εκμάθησης, τα δεδομένα συνήθως συσταδοποιούνται. Παρόμοιο μοτίβο ακολουθεί και το δίκτυο, με τα αποτελέσματα της συσταδοποίησης να εκφράζονται στον μοναδικό ενεργό νευρώνα εξόδου.Τέλος, πρέπει να αναφερθεί ότι η ανταγωνιστική εκμάθηση έχει αποδειχθεί ότι εμφανίζεται και στα βιολογικά νευρωνικά δίκτυα.

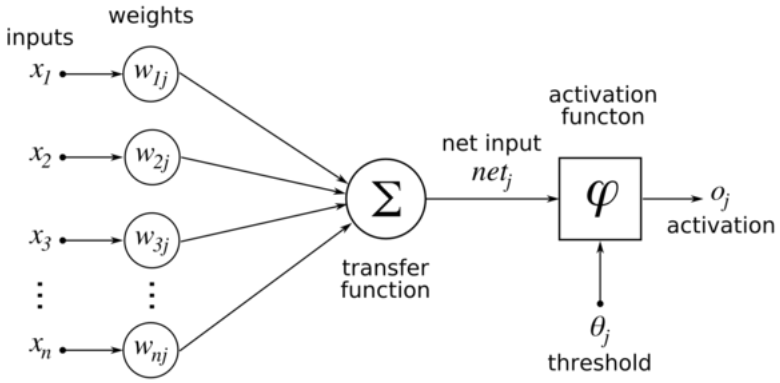
- *Adaptive Resonance Theory (ART) models:*Τα ART μοντέλα αναπτύχθηκαν από τους S.Grossberg και G.Carpenter και χρησιμοποιούνται κυρίως σε προβλήματα πρόβλεψης και αναγνώρισης προτύπων. Διαισθητικά,η κεντρική ιδέα τους είναι η εξής: η αναγνώριση ενός αντικειμένου είναι αποτέλεσμα της αλληλεπίδρασης μεταξύ των «προσδοκιών» που έχει θέσει ένας παρατηρητής στο μοντέλο και των πραγματικών τιμών που ανιχνεύονται για το αντικείμενο. Στην πράξη, οι «προσδοκίες» του παρατηρητή παίρνουν την μορφή ενός προτύπου μνήμης ή αλλιώς πρωτοτύπου (memory template or prototype) το οποίο συγκρίνεται με τις πραγματικές τιμές του αντικειμένου (που έχουν ανιχνευθεί από το μοντέλο). Εαν η διαφορά των δύο τιμών δεν ξεπερνά ένα όριο που έχει τεθεί από τον παρατηρητή, τότε το αντικείμενο ταξινομείται στην «προσδοκόμενη» κατηγορία. Με αυτό το τρόπο μπορούμε να κερδίζουμε νέα γνώση για ένα πρόβλημα χωρίς όμως να διαταρράσουμε την ήδη υπάρχουσα.

3.3 Το μαθηματικό υπόβαθρο των Τεχνητών Νευρωνικών Δικτύων.

Ξεκινώντας από το πιο απλό τεχνητό νευρωνικό δίκτυο που υπάρχει, το Perceptron ή Single Layer Perceptron και καταλήγοντας στην εξέλιξη αυτού, που είναι το Multilayer Perceptron, σε αυτή την ενότητα θα παρουσιαστεί αναλυτικά το μαθηματικό μοντέλο που εδράζεται πίσω από αυτά τα δίκτυα καθώς και οι αντίστοιχοι αλγόριθμοι εκμάθησης τους.

3.3.1 Το Perceptron δίκτυο.

Το Single Layer Perceptron δίκτυο εισήχθη για πρώτη φορά από τον Frank Rosenblatt το 1962 και αποτελεί την πιο απλοποιημένη μορφή ενός βιολογικού νευρωνικού δικτύου καθώς δεν διαθέτει κανένα κρυφό στρώμα (single layer). Λόγω αυτής της αρχιτεκτονικής του, θα μπορούσε να πει κανείς ότι ο όρος δίκτυο για το απλό Perceptron είναι καταχρηστικός αφού δεν υπάρχουν παραπάνω από έναν νευρώνες και η διασύνδεση γίνεται μόνο μεταξύ των νευρώνων εισόδου και εξόδου.



Σχήμα 3.4: Διάγραμμα δικτύου για το Single Layer Perceptron

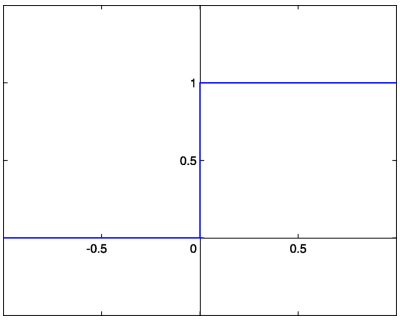
Η μετάφραση της λειτουργίας ενός βιολογικού νευρώνα σε μαθηματικά γίνεται λαμβάνοντας υπ΄ όψιν μας τρεις βασικές συνιστώσες: τις συνάψεις, τους νευρώνες αποστολείς και την συνάρτηση ενεργοποίησης. Σε πρώτη φάση, οι συνάψεις των βιολογικών νευρώνων είναι υπεύθυνες για την διασύνδεση μεταξύ των νευρώνων και μάλιστα εκφράζουν την δύναμη των διασυνδέσεων. Σε ένα τεχνητό νευρώνα, οι συνάψεις μοντελοποιούνται ως συναπτικά βάρη (synaptic weights). Τα βάρη συμβολίζονται ως w_{kj} και είναι πραγματικοί αριθμοί που το πρόσημο τους καθορίζει το είδος της διασύνδεσης των νευρώνων. Για παράδειγμα, ένα θετικό πρόσημο αντιπροσωπεύει μια διεγερτική σύνδεση ενώ ένα αρνητικό μια ανασταλτική σύνδεση. Επιπλέον, πολλά μοντέλα νευρώνων χρησιμοποιούν και ένα εξωτερικό βάρος, το οποίο δεν εξαρτάται από καμία μεταβλητή εισόδου και ονομάζεται μεροληψία (bias). Το πρόσημο της μεροληψίας επηρεάζει την τελική τιμή που δίνει το δίκτυο στη συνάρτηση ενεργοποίησης. Τέλος, η μεροληψία b_k λογίζεται και αυτή ως μια σύναψη με τιμή εισόδου $x_o = \pm 1$ (ανάλογα αν αυξάνει ή μειώνει την τιμή εισόδου στο δίκτυο). Συνοψίζοντας τα όσα έχουμε πει μέχρι στιγμής έχουμε:

- τα συναπτικά βάρη $w_{k1}, w_{k2}, ..., w_{kp}$,
- τις μεταβλητές εισόδου $x_1, x_2, ...x_p$ του νευρώνα k,
- και την μεροληψία b_k.

Το άθροισμα του φορτίου που δέχεται ο νευρώνας στο στρώμα εξόδου συμβολίζεται με u_k και εκφράζεται ως (Ρίζος,1996):

$$u_k = \sum_{j=1}^{p} w_{kj} x_j$$

Η παραπάνω έκφραση δεν αντιπροσωπεύει ωστόσο την τελική τιμή του νευρώνα στο στρώμα εξόδου. Η έξοδος του νευρώνα y_k είναι η εφαρμογή μιας συνάρτησης ενεργοποίησης πάνω στην διαφορά του αθροίσματος u_k και της μεροληψίας b_k.



Σχήμα 3.5: Γράφημα συνάρτησης βήματος

Από την διαφορά $u_k - b_k$ και το γεγονός ότι η μεροληψία αντιπροσωπεύει μια μεταβλητή εισόδου $x_o = \pm 1$, προκύπτει το νέο άθροισμα :

$$v_k = \sum_{j=0}^{p} w_{kj} x_j$$

Συνεπώς, το αποτέλεσμα του δικτύου θα είναι μια συνάρτηση της μορφής:

$$y_k = \varphi\left(u_k - b_k\right) \iff y_k = \varphi\left(v_k\right)$$

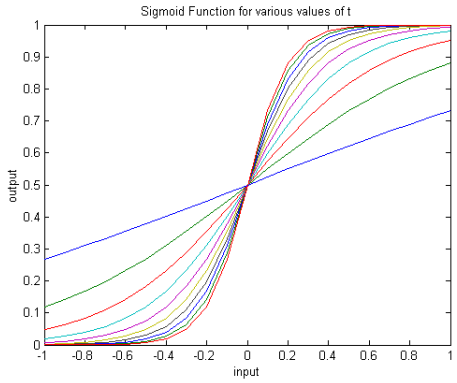
Στα παραπάνω βήματα κάναμε συχνά αναφορά σε μια συνάρτηση ενεργοποίησης. Ο λόγος που γίνεται χρήση της συνάρτησης ενεργοποίησης (activation function) είναι προκειμένου να ελεγχθεί το εύρος των τιμών y_k του νευρώνα εξόδου. Για παράδειγμα, σε ένα νευρωνικό δίκτυο συνήθως θέλουμε οι τιμές y_k του νευρώνα εξόδου να περιορίζονται στο διάστημα $[0, 1]$ ή $[-1, 1]$.Οι κυριότεροι τύποι συναρτήσεων ενεργοποιήσης που μπορούμε να συναντήσουμε είναι οι παρακάτω:

1. *Συνάρτηση βήματος (step function)*: Η συνάρτηση βήματος είναι η συνάρτηση που χρησιμοποιήθηκε στο πρώτο perceptron που παρουσιάστηκε και χρησιμοποιείται κυρίως σε προβλήματα ταξινόμησης όπου η μεταβλητή απόκρισης έχει δύο κατηγορίες. Οι τιμές που παίρνει είναι 0 (ή -1) αν το άθροισμα v_k είναι κάτω από ένα όριο και 1, αν είναι μεγαλύτερο ή ίσο με αυτό.

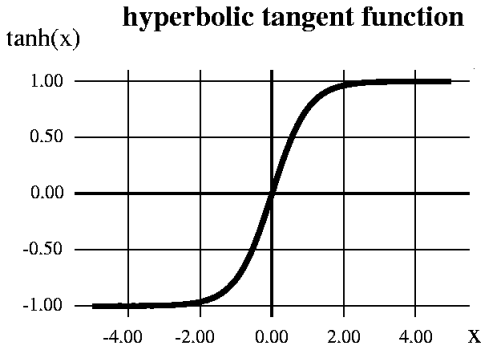
2. *Σιγμοειδής συνάρτηση (sigmoid function)*: Η οικογένεια των σιγμοειδών συναρτήσεων χωρίζεται σε δύο τύπους: την λογαριθμο-σιγμοειδή και την σιγμοειδή υπερβολικής εφαπτομένης. Η λογαριθμο-σιγμοειδής συνάρτηση ή αλλιώς λογιστική συνάρτηση δίνεται από την σχέση:

   $$\sigma\left(t\right) = \frac{1}{1 + e^{\beta t}}$$

   και θεωρείται η πλέον διαδεδομένη στην χρήση της, ειδικά στα κρυφά στρώματα των Multilayer δικτύων. Αυτό οφείλεται στο γεγονός ότι οι συναρτήσεις



Σχήμα 3.6: Γράφημα της λογαριθμο-σιγμοειδούς συνάρτησης για διάφορες τιμές της παραμέτρου κλίμακας t. Βλέπουμε ότι η παράμετρος κλίμακας t ρυθμίζει το ποσοστό ενεργοποίησης



Σχήμα 3.7: Γράφημα σιγμοειδούς συνάρτησης υπερβολικής εφαπτομένης

Σχήμα 3.8: Γραφήματα σιγμοειδών συναρτήσεων

αυτές μπορούν να εκτιμηθούν γιατί οι παράγωγοί τους μπορούν να υπολογιστούν εύκολα. Η εκτίμηση των συναρτήσεων αυτών είναι ιδιαίτερα χρήσιμη για ορισμένους αλγορίθμους στους οποίους τα συναπτικά βάρη ανανεώνονται μέχρι να καταλήξουν στην τελική τιμή τους. Από την άλλη, η σιγμοειδής συνάρτηση υπερβολικής εφαπτομένης έχει πάρει το όνομά της προφανώς από την υπερβολική εφαπτομένη που έχει χρησιμοποιηθεί για την κατασκευή της. Δίνεται από την σχέση:

$$\sigma(t) = \tanh(t) = \frac{e^t - e^{-t}}{e^t + e^{-t}}$$

και χρησιμοποιείται εξίσου συχνά με την λογαριθμο-σιγμοειδή συνάρτηση, αφού εμφανίζει την ίδια υπολογιστική ευκολία.

3. *Συνάρτηση Softmax*: Πρόκειται για ένα σχετικά νέο τύπο συνάρτησης ενεργοποίησης, η οποία δρα συμπιεστικά πάνω σε διάνυσμα με στόχο τον περιορισμό του εύρους των τιμών του στο διάστημα $[0, 1]$. Δίνεται από την σχέση:

$$\sigma_k(T) = \frac{e^{T_k}}{\sum_{k=1}^{K} e^{T_k}}$$

όπου T_k είναι το διάνυσμα που περιλαμβάνει τις τιμές των $k-$νευρώνων εξόδων. Αυτή η συνάρτηση χρησιμοποιείται στο μοντέλο multilogit (βλ. Hastie et al.2001) καθώς και στο κρυφό στρώμα των κανονικοποιημένων RBF δικτύων.

Οι προηγούμενοι τύποι συναρτήσεων ενεργοποίησης χρησιμοποιούνται από όλους τους άλλους τύπους νευρωνικών δικτύων, εκτός του απλού Perceptron. Ειδικότερα, για το Single Layer Perceptron η συνάρτηση ενεργοποίησης που χρησιμοποιείται είναι η συνάρτηση βήματος, με την μεταβλητή απόκρισης να εκφράζεται ως εξής:

$$\hat{y} = sign\left(w_1x_1 + w_2x_2 + \ldots + w_{p-1}x_{p-1} + w_px_p - w_0x_0\right) = sign\left(w \cdot x\right)$$

όπου το $w \cdot x$ είναι το εσωτερικό γινόμενο του διανύσματος των βαρών με το διάνυσμα των εισερχόμενων μεταβλητών (συμπεριλαμβανόμενης και της μεροληψίας).

3.3.1.1 Αλγόριθμος εκμάθησης του Single-Layer Perceptron δικτύου

Από την παρουσίαση του μαθηματικού μοντέλου που βρίσκεται πίσω από τα νευρωνικά δίκτυα αντιλαμβανόμαστε ότι καθοριστικό ρόλο στην διαμόρφωση ενός κατάλληλου μοντέλου για το εκάστοτε πρόβλημα έχουν τα συναπτικά βάρη. Άρα το πρόβλημα ταξινόμησης ισοδυναμεί ουσιαστικά με την εύρεση του σωστού συνδυασμού συναπτικών βαρών. Κατά την εκπαίδευση ενός perceptron δικτύου, τα συναπτικά βάρη w προσαρμόζονται διαρκώς μέχρι οι μεταβλητές εξόδου που παράγωνται από αυτά να ταυτίζονται με τις μεταβλητές απόκρισης του συνόλου εκπαίδευσης. Η διαδικασία αυτή περιγράφεται από την εξής σχέση:

$$w_j^{(k+1)} = w_j^{(k)} + \lambda\left(y_i - \hat{y_i^{(k)}}\right)x_{ij}$$

όπου $w_j^{(k)}$ είναι το συναπτικό βάρος που αντιστοιχεί στην $i-$οστή μεταβλητή εισόδου μετά από την $k-$οστή επανάληψη του αλγορίθμου, λ είναι ο ρυθμός εκμάθησης (learning rate) και το x_{ij} αντιστοιχεί στην τιμή x_i της $j-$οστής επεξηγηματικής μεταβλητής.

Η προηγούμενη σχέση είναι αναδρομική, αφού το νέο συναπτικό βάρος $w_j^{(k+1)}$ εξαρτάται από το αμέσως προηγούμενο βάρος $w_j^{(k)}$ και από έναν όρο ανάλογο του σφάλματος πρόβλεψης $\left(y_i - \hat{y_i^{(k)}}\right)$. Ο ρυθμός πρόβλεψης λ είναι μια παράμετρος του μοντέλου που παίρνει τιμές στο διάστημα $(0, 1)$ και εκφράζει τον αριθμό των τροποποιήσεων που γίνονται σε κάθε επανάληψη του αλγορίθμου. Αν το λ είναι κοντά στο 0 αυτό σημαίνει ότι το καινούριο βάρος θα είναι πολύ κοντά στο αμέσως προηγούμενο (δηλαδή θα εξαρτάται περισσότερο από τον όρο $w_j^{(k)}$). Αντίθετα, αν το λ είναι κοντά στο 1, το νέο συναπτικό βάρος θα εξαρτάται περισσότερο από το ποσό των τροποποιήσεων που λαμβάνουν χώρα στην εκάστοτε επανάληψη.[2]

Εάν η πρόβλεψη για το καινούριο συναπτικό βάρος είναι σωστή (δηλαδή έχουμε μηδενικό σφάλμα πρόβλεψης), τότε το συναπτικό βάρος παραμένει σταθερό. Στην αντίθετη περίπτωση, δουλεύουμε ως εξής:

[2] Σε μερικούς αλγόριθμους το λ είναι μεταβαλλόμενο: στις πρώτες επαναλήψεις παίρνει σχετικά μεγάλες τιμές και μετά σταδιακά φθίνει

Αλγόριθμος 3.1 Αλγόριθμος εκμάθησης του δικτύου

1. Έστω το σύνολο εκπαίδευσης $D = \{(x_i, y_i) \,|i = 1, 2, ...N\}$.
2. Αρχικοποίησε το διάνυσμα συναπτικών βαρών με τυχαίες τιμές, έστω $w^{(0)}$.
3. Επανάλαβε
   (α′) Για κάθε ζεύγος $(x_i, y_i) \in D$,
   (β′) υπολόγισε την προβλεπόμενη τιμή $\hat{y_i^{(k)}}$
   (γ′) Για κάθε συναπτικό βάρος w_j,
   (δ′) υπολόγισε το $w_j^{(k+1)} = w_j^{(k)} + \lambda \left(y_i - \widehat{y_i^{(k)}} \right) x_{ij}$.
   (ε′) τερμάτισε
4. τερμάτισε
5. μέχρι να ικανοποιηθεί η συνθήκη διακοπής της διαδικασίας.

- Αν $y = 1$ και $\hat{y} = -1$, το σφάλμα πρόβλεψης είναι $(y - \hat{y}) = 2$. Για να αντισταθμιστεί αυτό το σφάλμα, θα πρέπει να αυξηθεί η τιμή της εξόδου. Αυτό επιτυγχάνεται αυξάνοντας τα συναπτικά βάρη των θετικών μεταβλητών εισόδου και μειώνοντας αντίστοιχα τα βάρη των αρνητικών μεταβλητών εισόδου.

- Εντελώς ανάλογα, αν $y = -1$ και $\hat{y} = 1$ είναι $(y - \hat{y}) = -2$. Συνεπώς, το σφάλμα αυτό θα αντισταθμιστεί αν μειωθεί η τιμή της εξόδου, μειώνοντας τα συναπτικά βάρη των θετικών μεταβλητών εισόδου και αυξάνοντας αυτά των αρνητικών μεταβλητών εισόδου.

Η επιλογή μεταβλητής απόκρισης με δύο κατηγορίες για την παραπάνω επεξήγηση έγινε καθαρά λόγω ευκολίας στην κατανόηση για τον αναγνώστη. Με ακριβώς το ίδιο σκεπτικό δουλεύουμε και για άλλους τύπους μεταβλητών. Από την σχέση (1), είναι προφανές ότι οι επεξηγηματικές μεταβλητές που συνεισφέρουν περισσότερο μέσω των βαρών τους στον όρο του σφάλματος, απαιτούν και τις περισσότερες τροποποιήσεις στα συναπτικά βάρη τους. Αυτό ωστόσο δεν μας δίνει την ελευθερία να κάνουμε και δραστικές αλλαγές, διότι ο όρος του σφάλματος υπολογίζεται κάθε φορά για ένα σύνολο εκπαίδευσης. Αν επιμέναμε σε μεγάλες αυξομειώσεις των βαρών, τότε οι τροποποιήσεις στις αρχικές επαναλήψεις της εκπαίδευσης θα αναιρούνταν.

3.3.2 Perceptron πολλών στρωμάτων (Multilayer Perceptron)

Ένα τυπικό Τεχνητό Νευρωνικό Δίκτυο δεν είναι τόσο απλό όσο το Perceptron δίκτυο που παρουσιάσαμε. Αντίθετα, τα Perceptron δίκτυα πολλών στρωμάτων (Multilayer Perceptron) τείνουν να είναι περισσότερο ο κανόνας παρά η εξαίρεση. Όσον αφορά την αρχιτεκτονική τους, η βασική διαφορά τους με το απλό Perceptron είναι ότι διαθέτουν ένα ή και περισσότερα ενδιάμεσα στρώματα μεταξύ των στρωμάτων εισόδου και εξόδου. Ένα τυπικό παράδειγμα Multilayer Perceptron δικτύου είναι αυτό της Εικόνας 3.2.

Τα ενδιάμεσα αυτά στρώματα ονομάζονται κρυφά στρώματα (hidden layers) και οι κόμβοι που βρίσκονται μέσα σε αυτά κρυφοί κόμβοι (hidden nodes). Μια ακόμα διαφοροποίηση των Perceptron δικτύων πολλών στρωμάτων είναι ότι δεν χρησιμοποιούν εξ' ορισμού κάποια συγκεκριμένη συνάρτηση ενεργοποίησης αλλά δίνεται η δυνατότητα στον χρήστη να επιλέξει εκείνος αυτή που επιθυμεί. Αυτές οι επιπρόσθετες διαφοροποιήσεις που αναφέρθηκαν καθιστούν τα Perceptrons πολλών στρωμάτων ικανά να αντιμετωπίσουν πολυπλοκότερα προβλήματα αφού μπορούν να μοντελοποιήσουν σύνθετες σχέσεις μεταξύ των μεταβλητών εισόδου και εξόδου.

3.3.2.1 Εκμάθηση ενός Τεχνητού Νευρωνικού Δικτύου

Για να βρούμε τα συναπτικά βάρη ενός τεχνητού νευρωνικού δικτύου πρέπει να έχουμε στα χέρια μας έναν αποτελεσματικό αλγόριθμο, ο οποίος θα συγκλίνει στη σωστή λύση όταν δίνεται ένα επαρκές σύνολο δεδομένων. Μια μεθοδολογία που συναντάται πολύ συχνά για την εύρεση των βαρών του δικτύου είναι η μέθοδος καθόδου κλίσεων (gradient descent) την οποία θα εξηγήσουμε αναλυτικά στις παρακάτω γραμμές. Ο στόχος του αλγόριθμου εκμάθησης του τεχνητού νευρωνικού δικτύου είναι να προσδιορίσει ένα σύνολο βαρών w τα οποία θα ελαχιστοποιούν το άθροισμα ελαχίστων τετραγώνων:

$$E(w) = \frac{1}{2}\sum_{i=1}^{N}(y_i - \hat{y_i})^2$$

Όπως έχει ήδη αναφερθεί, η έξοδος των περισσότερων νευρωνικών δικτύων είναι μια μη γραμμική συνάρτηση λόγω των διαφόρων συναρτήσεων ενεργοποίησης που συναντώνται. Συνεπώς, η εύρεση των τιμών των βαρών w που να δίνουν μια ολικά βέλτιστη λύση δεν είναι και τόσο εύκολο ζήτημα. Αυτό το θέμα βελτιστοποίησης που ανακύπτει έχει αντιμετωπιστεί σε μεγάλο βαθμό από αλγόριθμους οι οποίοι χρησιμοιποιούν την «άπληστη» μέθοδο της καθόδου κλίσεων. Η αντίστοιχη σχέση που περιγράφει τον υπολογισμό νέων βαρών από τον αλγόριθμο είναι:

$$w_j \leftarrow w_j - \lambda\frac{\partial E(w)}{\partial w_j}$$

όπου λ είναι ο ρυθμός εκμάθησης. Η πληροφορία που μας δίνει αυτή η σχέση είναι ότι το βάρος πρέπει να αυξάνεται μέχρις ότου επιτευχθεί η απαραίτητη μείωση

στον όρο του συνολικού σφάλματος. Πολλές φορές ωστόσο, εξαιτίας του ότι η συνάρτηση σφάλματος είναι μη γραμμική, η μέθοδος καθόδου κλίσεων μπορεί να «παγιδευτεί» σε ένα τοπικό ελάχιστο.

3.3.2.2 Αλγόριθμος Back-propagation

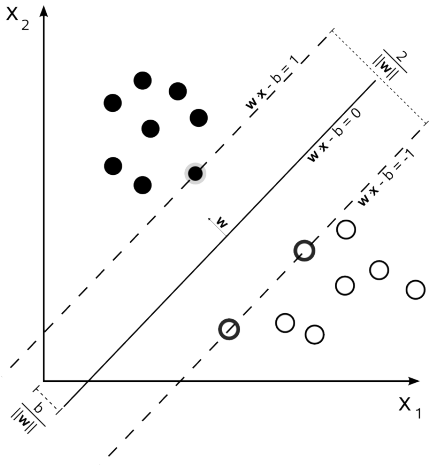
Με τη μέθοδο καθόδου κλίσεων μπορούμε να προσδιορίσουμε τα τελικά βάρη των επεξηγηματικών μεταβλητών στο στρώμα εξόδου. Τι γίνεται όμως αν θέλουμε να γνωρίζουμε τις τιμές των βαρών στα κρυφά στρώματα του δικτύου; Για τα κρυφά στρώματα, ο υπολογισμός αυτός είναι ιδιαίτερα δύσκολος διότι ο αλγόριθμος εκμάθησης δεν μας παρέχει κάποιο τρόπο προσδιορισμού του όρου $\frac{\partial E(w)}{\partial w_j}$ σε αυτά. Αυτό το πρόβλημα έρχεται να λύσει ο αλγόριθμος Back-propagation, ο οποίος δημοσιεύθηκε σε μια εργασία των Rumerhalt, Hinton και Williams το 1986 . Από το όνομα του, καταλαβαίνουμε ότι ο αλγόριθμος έχει δύο στάδια σε κάθε επανάληψή του: ένα μπρος και ένα πίσω. Κατά την ευθεία διαδικασία, το βάρος κάθε επανάληψης υπολογίζεται με βάση το αμέσως προηγούμενο και χρησιμοποιείται για τον υπολογισμό της εξόδου κάθε νευρώνα του δικτύου. Σε αυτό το σημείο πρέπει να παρατηρηθεί ότι ο υπολογισμός των συναπτικών βαρών των νευρώνων εξόδου γίνεται με αύξουσα σειρά (δηλαδή δεν μπορούν να υπολογιστούν τα βάρη του στρώματος $k+1$ πριν από αυτά που αντιστοιχούν στο στρώμα k). Κατά την «προς τα πίσω» διαδικασία, εφαρμόζονται όσα περιγράψαμε παραπάνω αλλά με κατεύθυνση προς τα προηγούμενα στρώματα. Με άλλα λόγια, τα βάρη που αντιστοιχούν στο στρώμα $k+1$ υπολογίζονται πριν τα βάρη του στρώματος k. Το πραγματικό πλεονέκτημα που μας προσφέρει ο αλγόριθμος αυτός δεν περιορίζεται στο απλά να μας παρέχει μια τιμή για τα κρυφά στρώματα. Αυτό που μας επιτρέπει είναι η καλύτερη εποπτεία του δικτύου σε κάθε επίπεδο αφού γνωρίζοντας τα σφάλματα για το στρώμα εξόδου του δικτύου μπορούμε να ελέγξουμε τις διαδικασίες που λαμβάνουν χώρα στα κρυφά στρώματα και άρα να ανιχνεύσουμε και να διορθώσουμε το οποιοδήποτε σφάλμα έχει παρουσιαστεί κατά την εκμάθηση του δικτύου.

3.4 Γεωμετρική ερμηνεία των Τεχνητών Νευρωνικών Δικτύων

Μέχρι στιγμής, έχουμε αντιμετωπίσει τα νευρωνικά δίκτυα ως μια κλάση παραμετρικών μη γραμμικών συναρτήσεων που έχουν ως πεδίο ορισμού τις μεταβλητές εισόδου x_j και πεδίο τιμών τις μεταβλητές εξόδου y_k. Προτού προχωρήσουμε στην γεωμετρική αναπαράσταση των τεχνητών νευρωνικών δικτύων, θα κάνουμε τις εξής δύο παραδοχές: πρώτον θα δουλέψουμε με το απλό Perceptron και δεύτερον η μεταβλητή απόκρισης θα είναι κατηγορική με κατηγορίες τις $y = 1$ και $y = -1$.

Στην προηγούμενη παράγραφο, είδαμε ότι στο Single Layer Perceptron η απόκριση y_k είναι μια γραμμική συνάρτηση της μορφής:

$$y_k = w^T x + w_o$$



Σχήμα 3.9: Ένα γραμμικό διαχωριστικό επίπεδο που αντιστοιχεί στην τιμή $y(x) = 0$ σε ένα δισδιάστατο χώρο μεταβλητών εισόδου. Το διάνυσμα των συνατπικών βαρών w αναπαρίσταται ως διάνυσμα και στον χώρο των μεταβλητών εισόδου και ορίζει τον προσανατολισμό του επιπέδου. Το βάρος που αντιστοιχεί στην μεροληψία w_o ορίζει την θέση του επιπέδου μετρώντας την κάθετη απόστασή του από την αρχή των αξόνων

Η παραπάνω έκφραση έχει την απλή γεωμετρική απεικόνιση (Duda and Hart, 1973) που ακολουθεί. Αρχικά παρατηρούμε ότι η τιμή $y(x) = 0$ αντιστοιχεί σε ένα υπερεπίπεδο $(k-1)-$διαστάσεων στον $k-$διαστάσεων χώρο των μεταβλητών εισόδου . Το υπερεπίπεδο αυτό λειτουργεί ως διαχωριστικό για τις μεταβλητές διότι τις κατατάσσει στην κατηγορία της μεταβλητής απόκρισης που αντιστοιχούν. Αν ο χώρος των x είναι δισδιάστατος, δηλαδή $k = 2$, τότε το διαχωριστικό υπερεπίπεδο θα είναι μια ευθεία γραμμή, όπως φαίνεται και στο παρακάτω σχήμα:

Αν θεωρήσουμε δύο σημεία x^A και x^B πάνω στο επίπεδο τότε $y\left(x^A\right) = y\left(x^B\right) = 0$ και από την προηγούμενη εξίσωση $w^T\left(x^B - x^A\right) = 0$. Συνεπώς βλέπουμε ότι το διάνυσμα των συναπτικών βαρών w είναι κανονικό προς κάθε διάνυσμα του χώρου και κατ΄ επέκταση καθορίζουν τον προσανατολισμό του διαχωριστικού επιπέδου. Έστω τώρα ότι x είναι ένα σημείο του επιπέδου. Από αναλυτική γεωμετρία τότε, η κανονική απόσταση του επιπέδου από την αρχή των αξόνων δίνεται από την σχέση:

$$a = \frac{w^T x}{\|w\|} = -\frac{w_o}{\|w\|}$$

η οποία προκύπτει από την σχέση στην αρχή της παραγράφου αν θέσουμε $y(x) = 0$. Άρα βλέπουμε ότι η μεροληψία w_o ορίζει την θέση του επιπέδου στον χώρο των x.

Παραμένοντας στην περίπτωση του Single Layer Perceptron αλλά με κατηγορική μεταβλητή περισσότερων κατηγοριών θα δούμε ότι δεν παρουσιάζονται σοβαρές

αλλαγές όσον αφορά την γεωμετρική απεικόνιση. Η συνάρτηση που δίνει την έξοδο $y_k(x)$ για κάθε κατηγορία k της y είναι:

$$y_k(x) = w_k^T x + w_{k0}$$

Η ταξινόμηση των σημείων x του επιπέδου των μεταβλητών εισόδου γίνεται με την ακόλουθη λογική: ένα σημείο x ανήκει στην κατηγορία k της εξόδου y αν $y_k(x) > y_j(x)$ για οποιαδήποτε άλλη κατηγορία $j \neq k$. Το επίπεδο που διαχωρίζει τις κατηγορίες k και j δίνεται από την εξίσωση:

$$y_k(x) = y_j(x) \Longleftrightarrow (w_k - w_j)^T x + (w_{k0} - w_{j0}) = 0$$

Εντελώς ανάλογα με την περίπτωση των δύο κατηγοριών, το διάνυσμα των βαρών $(w_k - w_j)^T$ καθορίζει πάλι τον προσανατολισμό του επιπέδου και η απόσταση του επιπέδου από την αρχή των αξόνων είναι:

$$a = -\frac{(w_{k0} - w_{j0})}{\|w_k - w_j\|}$$

όπου $w_{k0} - w_{j0}$ είναι η αντίστοιχη μεροληψία.

3.5 Πολυπλοκότητα ενός Τεχνητού Νευρωνικού Δικτύου

Τα Τεχνητά Νευρωνικά Δίκτυα, όπως έχουν παρουσιαστεί μέχρι στιγμής, φαίνονται σαν μια απλή μέθοδος ταξινόμησης που δεν παρουσιάζει ιδιαίτερα θέματα κατά την εκτέλεσή της. Κάτι τέτοιο όμως δεν ισχύει. Αντιθέτως μάλιστα, η εκπαίδευση των νευρωνικών δικτύων μπορεί να χαρακτηριστεί και ως «τέχνη», αφού απαιτεί αρκετή εμπειρία από έναν αναλυτή προκειμένου να εντοπιστούν κάποιες καλά κρυμμένες παράμετροι που προκαλούν τα τυχόν προβλήματα. Προτού προχωρήσουμε στην παράθεση των θεμάτων που θα εξετάσουμε σε αυτή την παράγραφο πρέπει να κάνουμε κάποιες παρατηρήσεις που θα μας βοηθήσουν στην καλύτερη κατανόηση στην πορεία. Το μοντέλο που παράγεται από ένα τεχνητό νευρωνικό δίκτυο είναι υπερπαραμετρικό και κατά συνέπεια το πρόβλημα βελτιστοποίησης που καλείται να επιλύσει το δίκτυο είναι ασταθές και μη κυρτό, εκτός και αν υπάρχουν κάποιες κατευθυντήριες γραμμές που θα δούμε παρακάτω.

Ποιες είναι αυτές οι κατευθυντήριες γραμμές όμως; Η απάντηση σε αυτή την ερώτηση εξαρτάται από την συνάρτηση που προσεγγίζει την μεταβλητή εξόδου y_k του νευρωνικού δικτύου. Υπάρχουν τρεις τρόποι για να ελέγξουμε την πολυπλοκότητα των συναρτήσεων που παράγονται από ένα νευρωνικό δίκτυο:

- η αποκοπή συνδέσμων μεταξύ των διαφόρων στρωμάτων του δικτύου (κλάδεμα δικτύου),
- η αλλαγή του αριθμού των νευρώνων στα κρυφά στρώματα του δικτύου,

- η αλλαγή της παραμέτρου κανονικοποίησης (regularization)[3] του δικτύου.

Τα νευρωνικά δίκτυα λειτουργούν με την λογική του ≪μαύρου κουτιού≫ όσον αφορά την πρόβλεψη. Συνεπώς, κατά την διαδικασία της πρόβλεψης είναι προτιμότερο να γίνονται ομαλές αλλαγές στο σύστημα (όπως συρρίκνωση της τιμής των βαρών ή συστηματοποίηση) από το να αφαιρούνται ολόκληρα τμήματα του μοντέλου.

3.5.1 Κλάδεμα Τεχνητού Νευρωνικού Δικτύου

Με την έννοια του κλαδέματος σε ένα νευρωνικό δίκτυο υποδεικνύεται η κατάργηση κάποιων συνδέσμων μεταξύ νευρώνων που ανήκουν σε διαφορετικά στρώματα του δικτύου. Μαθηματικά αυτό μεταφράζεται στο να θέσουμε κάποια μεμονωμένα συναπτικά βάρη ίσα με το μηδέν με συνέπεια την πρόσθεση ή την διαγραφή ολόκληρων δομικών μονάδων (νευρώνων) του δικτύου. Η επιλογή των συνδέσμων που θα καταργηθούν γίνεται με κάποιες πολύ γνωστές μεθόδους στατιστικής ανάλυσης όπως η stepwise selection και ο δείκτης AIC.

Η επιστημονική κοινότητα που έχει ως αντικείμενο έρευνας τα νευρωνικά δίκτυα έχει αναπτύξει για το παραπάνω σκεπτικό κάποιες μεθόδους όπως οι Optimal Brain Surgeon (Hassibi and Stork · Hassibi et al.,1994 · Buntine and Weigend,1994) και Optimal Brain Damage (Le Cun et al.,1990b)[4], οι οποίες κλαδεύουν το δίκτυο θέτοντας κάποια βάρη ίσα με το μηδέν. Ωστόσο δεν μπορεί κανείς να προβεί σε μια μαζική εξίσωση βαρών με το μηδέν. Το γεγονός που αποτρέπει αυτή την ενέργεια είναι οι κοντινές συγγραμμικότητες των βαρών. Αυτό πρακτικά σημαίνει ότι αν θέσουμε ένα βάρος ίσο με το μηδέν αυτό μπορεί να οδηγήσει σε σημαντική μείωση των τυπικών σφαλμάτων των υπολοίπων βαρών. Επιπλέον, η χρήση μηδενικών βαρών οδηγεί σε μηδενικές παραγώγους και τέλεια συμμετρία με αποτέλεσμα ο αλγόριθμος να μην κινείται.

Μια άλλη προσέγγιση είναι η χρήση μικρών τιμών για τα βάρη κατά την εκκίνηση της εκπαίδευσης που ενδεχομένως θα μηδενιστούν και θα οδηγήσουν στην απομάκρυνση των αντίστοιχων συνδέσμων και νευρώνων. Η επιλογή τιμών κοντά στο μηδέν για τα βάρη κατά την εκκίνηση δημιουργεί γραμμικά μοντέλα στην αρχή το οποία στην πορεία εξελίσσονται σε μη γραμμικά με την αύξηση των βαρών.

3.5.2 Επιλογή του αριθμού των νευρώνων στα κρυφά στρώματα του δικτύου

Από την ανάλυση που έγινε στην προηγούμενη παράγραφο είδαμε ότι η ύπαρξη αρκετών κρυφών στρωμάτων και αντίστοιχα μεγάλου αριθμού νευρώνων μέσα σε αυτά λειτουργεί θετικά για την έκβαση της επεξεργασίας ενός συνόλου δεδομένων. Πιο συγκεριμένα, με πολύ λίγες κρυφές μονάδες το μοντέλο δεν είναι αρκετά ευέλικτο στο να εντοπίσει μη γραμμικότητες των δεδομένων. Αντιθέτως, με μεγάλο αριθμό κρυφών μονάδων, τα επιπλέον βάρη θα μειωθούν σταδιακά μέχρι μηδενισμού αν χρησιμοποιείται κατάλληλη κανονικοποίηση.

[3] οι τεχνικές κανονικοποίησης ενός τεχνητού νευρωνικού δικτύου αφορούν τον ορισμό περιορισμών ομαλότητας στις συναρτήσεις εξόδου

[4] αποτελούν προσεγγιστικές εκδοχές του κριτηρίου του Wald

Μια συνηθισμένη μέθοδος για να εντοπιστεί ο βέλτιστος αριθμός κρυφών στρωμάτων ή νευρώνων σε αυτά θα μπορούσε να είναι η διασταυρωμένη επικύρωση. Όμως εδώ θα πρέπει να τονιστεί ότι η εκπαίδευση των νευρωνικών δικτύων δεν είναι μια καλά ορισμένη διαδικασία (Ripley,1996) αφού μπορούν να εμφανιστούν πολλαπλά τοπικά ελάχιστα κατά την βελτιστοποίηση της συνάρτησης του σφάλματος που μπορούν να οδηγήσουν σε διαφορετική κάθε φορά απόδοση του μοντέλου. Άρα η διασταυρωμένη επικύρωση δεν θα μπορούσε να μας δώσει πολύ καλά αποτελέσματα αφού ο οποιοσδήποτε χωρισμός των δεδομένων σε σύνολο εκπαίδευσης και εξέτασης θα ήταν μεροληπτικός και θα κατέληγε κάθε φορά και σε διαφορετική λύση.

Μια άλλη προσέγγιση για το πρόβλημα της εύρεσης του αριθμού των νευρώνων στα κρυφά στρώματα είναι η αυξητική κατασκευή δικτύου (incremental network construction). Οι μέθοδοι που ανήκουν σε αυτή την κατηγορία έχουν ως σκεπτικό την σταδιακή αύξηση των δικτύων προσθέτοντας κάθε φορά ένα νευρώνα στο ίδιο ή σε περισσότερα κρυφά στρώματα. Η προηγούμενη διαδικασία συνοψίζεται στην κατασκευή ενός δικτύου -πυραμίδα (pyramid network), η οποία περιγράφεται αναλυτικά από τον Gallant (1990 , 1993, Κεφ.10). Ιδιαίτερα διαδεδομένη μέθοδος που βασίζεται στην κατασκευή ενός δικτύου-πυραμίδα είναι η Cascade correlation (Falhman and Lebiere, 1990). Ο αλγόριθμος που περιγράφει την μέθοδο είναι ο κάτωθι:

1. Η μέθοδος ξεκινά με το πιο απλό δίκτυο που αποτελείται αποκλειστικά από τα στρώματα εισόδου και εξόδου. Και τα δύο στρώματα είναι πλήρως συνδεδεμένα.

2. Για κάθε νευρώνα εξόδου, το δίκτυο εκπαιδεύεται κατά τα γνωστά μέχρι την μη περαιτέρω μείωση του σφάλματος δικτύου.

3. Εισαγωγή ενός υποψήφιου νευρώνα στο κρυφό στρώμα. Κάθε υποψήφιος νευρώνας συνδέεται με όλες τις μεταβλητές εισόδου (και με όλους τους υπόλοιπους νευρώνες του κρυφού στρώματος). Σε αυτό το σημείο δεν έχουν οριστεί ακόμα συνατπικά βάρη μεταξύ των νευρώνων εξόδου και της δεξαμενής των υποψήφιων νευρώνων.

4. Προσπάθησε να μεγιστοποιήσεις την συσχέτιση μεταξύ της συνάρτησης ενεργοποίησης του υποψήφιου νευρώνα και του σφάλματος των υπολοίπων του δικτύου εκπαιδεύοντας όλους τους συνδέσμους που καταλήγουν στο υποψήφιο νευρώνα. Η εκπαίδευση αυτή σταματά όταν οι τιμές για την συσχέτιση δεν μπορούν να βελτιωθούν άλλο.

5. Επέλεξε τον υποψήφιο νευρώνα με τη μεγαλύτερη τιμή συσχέτισης για εισαγωγή στο δίκτυο και «πάγωσε» τις τιμές όλων των εισερχόμενων σε αυτόν συναπτικών βαρών.

6. Για να μετατραπεί ένας υποψήφιος νευρώνας σε νευρώνα του κρυφού στρώματος, πρέπει να παραχθούν συνδέσεις μεταξύ του επιλεγμένου νευρώνα και των νευρώνων εξόδου. Εφόσον τα βάρη προς τον νέο «κρυφό» νευρώνα έχουν «παγώσει», βρες έναν νέο μόνιμο ανιχνευτή μεταβλητών. Η διαδικασία επαναλαμβάνεται ξανά επιστρέφοντας στο βήμα 2.

7. Ο αλγόριθμος επαναλαμβάνεται μέχρι το συνολικό σφάλμα του δικτύου πέσει κάτω από μια τιμή που θα έχει προσδιορίσει ο ερευνητής.

Ο αλγόριθμος αυτός είναι τουλάχιστον 10-φορές ταχύτερος από ολους τους τυπικούς αλγόριθμους Back-propagation, ενώ επιτρέπει στο δίκτυο να καθορίσει το ίδιο το μέγεθός του. Ένας άλλος εξίσου αποτελεσματικός αλγόριθμος είναι ο SMART (Friedman,1984) ο οποίος μπορεί να είναι περισσότερο χρονοβόρος από τον Cascade Correlation αλλά πολλές φορές μπορεί να παράγει μοντέλα μικρότερα σε μέγεθος αλλά με καλύτερες παραμέτρους κανονικοποίησης.

Κεφάλαιο 4

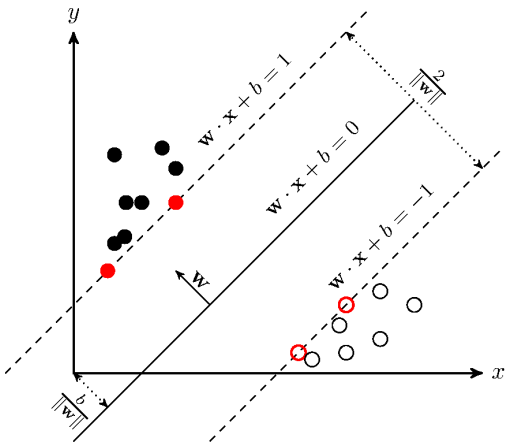
Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines)

4.1 Μια εισαγωγή στις Μηχανές Διανυσμάτων Υποστήριξης

Οι μηχανές διανυσμάτων υποστήριξης (Support Vector Machines) ή αλλιώς SVMs δεν μοιάζουν σε μια πρώτη ανάγνωση με τις τυπικές μεθόδους στατιστικής ανάλυσης που έχουμε γνωρίσει μέχρι στιγμής καθώς όπως θα δούμε σε επόμενη παράγραφο η θεωρία τους περιλαμβάνει περισσότερο γνώσεις μαθηματικής ανάλυσης παρά στατιστικής. Οι SVMs αναπτύχθηκαν κυρίως στην πληροφορική ως μέθοδος ταξινόμησης (χρησιμοποιώντας βέβαια το ανάλογο τεχνικό υπόβαθρο όπως η γλώσσα, ο τρόπος σκέψης και χρήσης κτλ) βασιζόμενα στην θεωρία του Vapnik (1995). Στα χρόνια που ακολούθησαν ωστόσο το εύρος των εφαρμογών τους έχει αμβλυνθεί με αποτέλεσμα οι SVMs να μοιάζουν με μια τυπική μέθοδο παλινδρόμησης (Cortes & Vapnik,2000 · Cristianini & Shawe-Taylor, 2000).

Εστιάζοντας περισσότερο τώρα στο αντικείμενο μελέτης αυτής της διπλωματικής, οι Μηχανές Διανυσμάτων Υποστήριξης είναι μια μέθοδος εκμάθησης με επίβλεψη η οποία χρησιμοποιείται τόσο για ταξινόμηση δεδομένων όσο και για παλινδρόμηση. Έχοντας ως δεδομένο ένα σύνολο εκπαίδευσης, όπου κάθε παρατήρηση[1] ανήκει σε μια κατηγορία της μεταβλητής απόκρισης, ο αλγόριθμος εκπαίδευσης για τις SVM παράγει ένα μη πιθανοθεωρητικό μοντέλο το οποίο ταξινομεί τις παρατηρήσεις του συνόλου εξέτασης στις αντίστοιχες κατηγορίες της απόκρισης. Αυτή η ταξινόμηση γίνεται πιο κατανοητή διότι η μέθοδος δίνει την δυνατότητα γραφικής

[1] Πρέπει να τονιστεί ότι δεν μας ενδιαφέρει σε αυτή την μέθοδο με ποιο τρόπο έχουν αποκτηθεί οι παρατηρήσεις ούτε το κατά πόσο επηρεάζουν οι επεξηγηματικές μεταβλητές την μεταβλητή απόκρισης. Το ενδιαφέρον επικεντρώνεται αποκλειστικά στην εύρεση ακριβών διαχωρισμών των δεδομένων



Σχήμα 4.1: Γραφική αναπαράσταση της μεθόδου SVM για γραμμικώς διαχωρίσιμα δεδομένα με μεταβλητή απόκρισης δύο κατηγοριών. Τα σημεία του γραφήματος που είναι κόκκινα και για τις δύο κατηγορίες είναι τα διανύσματα υποστήριξης (support vectors)

απεικόνισης του μοντέλου.

Πιο συγκεκριμένα, οι παρατηρήσεις αντιπροσωπεύουν σημεία στο χώρο που είναι τοποθετημένα με τέτοιο τρόπο ώστε να είναι εμφανής ο διαχωρισμός τους σε κατηγορίες από ένα κενό χώρο. Με άλλα λόγια, η εύρεση του κατάλληλου μοντέλου ταυτίζεται με την εύρεση του βέλτιστου διαχωριστικού επιπέδου (seperating hyperplane) ή αλλιώς ορίου απόφασης (decision boundary) χρησιμοποιώντας την πληροφορία που παρέχεται από τις επεξηγηματικές μεταβλητές ούτως ώστε ο διαχωρισμός των παρατηρήσεων σε κατηγορίες να είναι όσο το δυνατόν πιο ομογενής. Η διαδικασία ταξινόμησης ολοκληρώνεται όταν βρεθεί το διαχωριστικό επίπεδο που απέχει την μεγαλύτερη δυνατή απόσταση[2] από όλες τις πιθανές κατηγορίες. Το επίπεδο αυτό είναι γνωστό και ως επίπεδο μέγιστου περιθωρίου (maximal margin hyperplane).

Οι SVMs αν και δεν είναι μια μέθοδος που αναπτύχθηκε αρχικά με σκοπό να εξυπηρετήσει την επίλυση στατιστικών προβλημάτων έχει αποδειχτεί ότι μπορεί να προσφέρει ιδιαίτερα ακριβή αποτελέσματα σε πρακτικές εφαρμογές όπως η αναγνώριση ή κατηγοριοποίηση κειμένου. Επιπλέον, ενδείκνυται σαν μέθοδος για την ανάλυση δεδομένων υψηλών διαστάσεων καθώς μας παρέχει ένα πολύ σημαντικό πλεονέκτημα: ο τρόπος λειτουργίας της μας επιτρέπει να αποφύγουμε την «κατάρα των διαστάσεων»[3].Αυτό οφείλεται στο γεγονός ότι το βέλτιστο διαχωριστικό επίπεδο δεν εξάγεται με χρήση όλου του όγκου των παρατηρήσεων των

[2] όσο μεγαλύτερη είναι η απόσταση των κατηγοριών, τόσο μικρότερο το σφάλμα ταξινόμησης

[3] κατάρα των διαστάσεων (curse of dimensionality): σε αυτόν τον όρο συνοψίζονται όλα τα προβλήματα υπολογιστικής φύσης που μπορούν να προκληθούν λόγω του μεγάλου όγκου των δεδομένων

επεξηγηματικών μεταβλητών αλλά από ένα υποσύνολο αυτών, οι οποίες είναι γνωστές και ως διανύσματα υποστήριξης (support vectors). Με την βοήθεια της παραπάνω εικόνας, βλέπουμε ότι τα διανύσματα υποστήριξης είναι ουσιαστικά οι παρατηρήσεις που απέχουν την μικρότερη απόσταση από το διαχωριστικό επίπεδο και αντιπροσωπεύουν τις παρατηρήσεις που είναι πιο δύσκολο να ταξινομηθούν.

Μέχρι στιγμής, θα μπορούσε να έχει δημιουργηθεί στον αναγνώστη βάσει της Εικόνας 4.1 η εντύπωση ότι ο διαχωρισμός των δεδομένων σε κατηγορίες είναι αποκλειστικά γραμμικός και ότι γίνεται πάντα στις δύο διαστάσεις. Και οι δύο αυτές υποθέσεις όμως δεν ευσταθούν. Όπως θα παρουσιαστεί εκτενώς στην επόμενη παράγραφο, κατά την εφαρμογή του αλγορίθμου της μεθόδου ο διαχωρισμός που προκύπτει τις περισσότερες φορές δεν είναι γραμμικός. Αυτό όπως γνωρίζουμε και από τις κλασσικές μεθόδους στατιστικής δεν είναι κάτι επιθυμητό και πιο συγκεκριμένα στα SVMs δημιουργεί πρόβλημα ως προς την ασφάλεια πρόβλεψης του μοντέλου καθώς δεν μπορεί να οριστεί ένα σαφές μέγιστο περιθώριο από το διαχωριστικό επίπεδο. Σε αυτό το πρόβλημα έρχονται να δώσουν λύση οι συναρτήσεις πυρήνα (kernel functions) οι οποίες μετασχηματίζοντας καταλλήλως τα δεδομένα επιτυγχάνουν τον γραμμικό διαχωρισμό τους όχι στον χώρο των παρατηρήσεων αλλά στο χώρο των επεξηγηματικών μεταβλητών, ο οποίος είναι υψηλότερης διάστασης από τον χώρο των παρατηρήσεων.

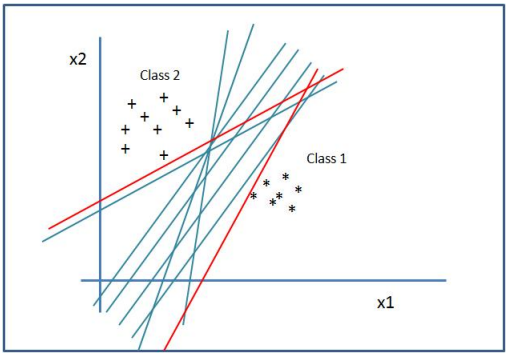
Στις επόμενες παραγράφους θα υπάρξει μια αναλυτική παρουσίαση της θεωρίας των μηχανών διανυσμάτων υποστήριξης για τις διάφορες κατηγορίες προβλημάτων που αναφέραμε παραπάνω καθώς και κάποια μέτρα αξιολόγησης της μεθόδου.

4.2 Ταξινόμηση στις Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Classifier)

Το πρωταρχικό βήμα για την ταξινόμηση των παρατηρήσεων ενός συνόλου δεδομένων κατά την εφαρμογή των SVMs είναι η απεικόνιση τους στον αντίστοιχο διανυσματικό χώρο παρατηρήσεων. Ωστόσο υπάρχουν πολλές περιπτώσεις όσον αφορά το διαχωριστικό επίπεδο που προκύπτει για την εκάστοτε χαρτογράφηση. Στόχος της μεθόδου δεν είναι αποκλειστικά η παραγωγή ενός γραμμικού διαχωριστικού επιπέδου αλλά του βέλτιστου που μπορεί να υπάρξει για τις δεδομένες τιμές μας. Στο υπόλοιπο της παραγράφου, θα παρουσιαστούν όλες οι δυνατές περιπτώσεις που μπορεί να συναντήσει ένας ερευνητής κατά την χρήση της μεθόδου καθώς και οι τρόποι υπολογισμού αντίστοιχα του βέλτιστου διαχωρισμού.Επιπλέον, πρέπει να τονιστεί ότι για τις περιπτώσεις που θα εξετάσουμε θα χρησιμοποιηθεί αποκλειστικά δυαδική μεταβλητή απόκρισης. Το πως συμπεριφέρεται η μέθοδος για μεταβλητές απόκρισης πολλαπλών κατηγορίων θα εξεταστεί στην τρίτη παράγραφο του κεφαλαίου.

4.2.1 Γραμμικά SVMs

Μια γραμμική SVM είναι ένας ταξινομητής που στόχος του είναι η εύρεση του επιπέδου που η απόσταση/ περιθώριο του από τις ακραίες παρατηρήσεις των διαφορετικών κατηγοριών της μεταβλητής απόκρισης του προβλήματος είναι η μεγαλύτερη



Σχήμα 4.2: Μερικά από τα δυνατά διαχωριστικά επίπεδα που μπορούν να προκύψουν για ένα σύνολο δεδομένων

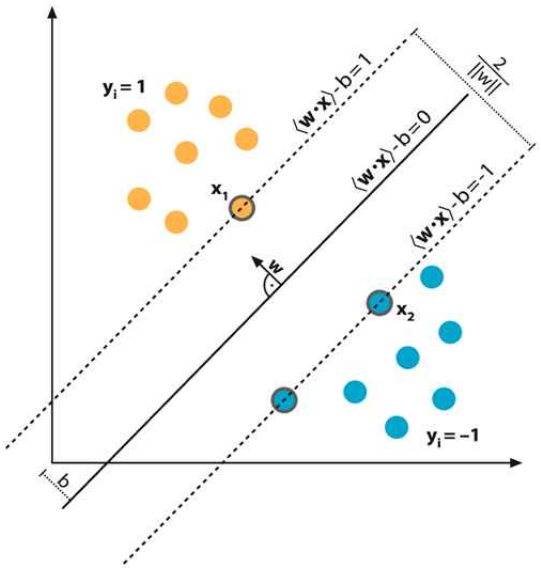
δυνατή. Για αυτό το λόγο πολλές φορές ο ταξινομητής αυτός αναφέρεται και ως ταξινομητής μεγίστου περιθωρίου (maximal margin classifier).

Όπως βλέπουμε και από την Εικόνα 4.2 το να βρούμε ένα διαχωριστικό επίπεδο είναι πολύ εύκολο. Υπάρχουν άπειρες ευθείες που μπορούμε να χαράξουμε προκειμένου να πετύχουμε έναν ομογενή διαχωρισμό των δεδομένων. Αυτό που ψάχνουμε όμως είναι η βέλτιστη ευθεία. Σε αυτή την μέθοδο το βέλτιστο διαχωριστικό επίπεδο ορίζεται ως αυτό το οποίο απέχει την μεγαλύτερη απόσταση από την ευθεία που ορίζεται από τις ακραίες παρατηρήσεις κάθε κατηγορίας της μεταβλητής απόκρισης. Αυτή η απόσταση μεταξύ του διαχωριστικού επιπέδου και της κάθε ευθείας ή ισοδύναμα η απόσταση μεταξύ των δύο ευθειών ονομάζεται περιθώριο (margin) και παίζει κυρίαρχο ρόλο στην θεωρία των SVMs. Ειδικότερα, όσο μεγαλύτερο είναι το περιθώριο τόσο καλύτερος θα είναι ο διαχωρισμός των δεδομένων και άρα το μοντέλο θα έχει μικρότερο γενικευμένο σφάλμα. Αντίθετα, μικρά περιθώρια δεν είναι επιθυμητά διότι το μοντέλο που προκύπτει από αυτά συχνά υπερπροσαρμόζει τα δεδομένα και μειώνει την πιθανότητα σωστής πρόβλεψης στο σύνολο εξέτασης.

Σε μια πιο μαθηματική διατύπωση, η συσχέτιση του περιθωρίου με το γενικευμένο σφάλμα του μοντέλου δίνεται από την διαρθρωτική ελαχιστοποίηση του κινδύνου (structural risk minimization-SRM). Το μέτρο αυτό ουσιαστικά παρέχει ένα άνω φράγμα στο γενικευμένο σφάλμα του ταξινομητή SVM, το οποίο δίνεται από την σχέση:

$$R \leq R_e + \varphi\left(\frac{h}{N}, \frac{\log(h)}{N}\right)$$

όπου R είναι ο ταξινομητής, R_e το σφάλμα εκπαίδευσης, N ο αριθμός των παρατηρήσεων και φ μια αύξουσα μονότονη συνάρτηση της ικανότητας h του μοντέλου. Η ικανότητα του μοντέλου είναι αντιστρόφως ανάλογη του περιθωρίου. Δηλαδή, μοντέλα με μικρά περιθώρια έχουν μεγαλύτερη ικανότητα καθώς είναι πιο ευέλικτα και μπορούν να προσαρμοστούν σε περισσότερα του ενός σύνολα δεδομένων εκπαίδευσης. Από την άλλη όμως, όσο αυξάνεται η ικανότητα ενός μοντέλου ταυ-



Σχήμα 4.3: Διαχωριστικό επίπεδο και περιθώριο μιας SVM

τόχρονα αυξάνεται και το άνω φράγμα του γενικευμένου σφάλματος, κάτι το οποίο είναι απευκτέο. Συνεπώς, το πραγματικό πλεονέκτημα των γραμμικών SVM ταξινομητών βρίσκεται στο γεγονός ότι μπορούν να μεγιστοποιήσουν τα περιθώρια από το διαχωριστικό επίπεδο και άρα να ελαχιστοποιήσουν στο ελάχιστο δυνατό το γενικευμένο σφάλμα του μοντέλου.

Προτού προχωρήσουμε στον τρόπο κατασκευής του διαχωριστικού επιπέδου καθώς και στις μεθόδους που χρησιμοποιούνται για την βελτιστοποίηση αυτού πρέπει να επισημάνουμε ένα ακόμα διαχωρισμό στη μελέτη μας: το ότι μπορούμε πάντα να φέρουμε μια ευθεία που να διαχωρίζει τα δεδομένα μας σε κατηγορίες δεν σημαίνει ότι υπάρχει πάντα και βέλτιστη ευθεία. Τα δεδομένα για τα οποία ισχύει κάτι τέτοιο ονομάζονται μη διαχωρίσιμα και η μελέτη τους διαφοροποιείται ελαφρώς από αυτή των διαχωρίσιμων.

4.2.1.1 Διαχωρίσιμα δεδομένα

Έστω ότι έχουμε ένα δυαδικό πρόβλημα ταξινόμησης όπου το σύνολο εκπαίδευσης αποτελείται από N παρατηρήσεις. Κάθε παρατήρηση έχει μια τιμή για κάθε μια από τις p επεξηγηματικές μεταβλητές και μια τιμή για την μεταβλητή απόκρισης. Συνεπώς τα δεδομένα μας είναι ζεύγη της μορφής $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$ όπου $x_i \in R^p$ και $y_i \in \{-1, 1\}$. Το διαχωριστικό επίπεδο θα είναι διάστασης $p-1$ και θα δίνεται από την σχέση:

$$f(x) = w^{\mathrm{T}} \cdot x + b = 0 \tag{4.1}$$

όπου οι w, b είναι παράμετροι του μοντέλου.

Εφόσον τα δεδομένα μας χωρίζονται σε δύο κατηγορίες, από την Εικόνα 4.3 βλέπουμε ότι κάθε παρατηρήση που αντιστοιχεί στην κατηγορία $+1$ είναι πάνω από το διαχωριστικό επίπεδο και δίνεται από την σχέση $w \cdot x + b = k$ όπου $k > 0$. Αντίστοιχα, για κάθε μια παρατήρηση που είναι κάτω από το διαχωριστικό επίπεδο και ανήκει στην κατηγορία -1, από την ίδια σχέση της αντιστοιχεί μια τιμή $k^{'} < 0$. Συνεπώς, η πρόβλεψη για το που ανήκει μια παρατήρηση x του συνόλου εξέτασης γίνεται ως εξής:

$$y_i = +1, wx + b > 0$$

και

$$y_i = -1, wx + b < 0 \tag{4.2}$$

Ο προσδιορισμός των δύο παράλληλών ευθειών ως προς το διαχωριστικό επίπεδο γίνεται με τον προσδιορισμό σημείων. Αν θεωρήσουμε ένα σημείο x_1 το οποίο ανήκει στην κατηγορία $+1$ και του αντιστοιχεί μια θετική τιμή k και ένα ακόμα στοιχείο x_2 της κατηγορίας -1 με μια αρνητική τιμή $k^{'}$,με κατάλληλη προσαρμογή των παραμέτρων w, b προκύπτουν οι κάτωθι εξισώσεις των δύο ευθειών:

$$w \cdot x_1 + b = 1 \tag{4.3}$$

$$w \cdot x_2 + b = -1 \tag{4.4}$$

Για τον υπολογισμό του περιθωρίου d, αρκεί να αφαιρέσουμε την εξίσωση (4.4) από την εξίσωση (4.3):

$$w \cdot (x_1 - x_2) = 2 \Leftrightarrow \|w\| \times d = 2 \Leftrightarrow d = \frac{2}{\|w\|}. \tag{4.5}$$

Εκμάθηση μιας γραμμικής SVM

Η εκμάθηση μιας γραμμικής SVM συνίσταται στον υπολογισμό των παραμέτρων w και b που εξασφαλίζουν το μεγαλύτερο περιθώριο από το διαχωριστικό επίπεδο. Η ταξινόμηση των δεδομένων γίνεται με τον εξής συνοπτικό κανόνα:

$$G(x) = sign[w \cdot x + b] \tag{4.6}$$

Από βασικές γνώσεις γεωμετρίας, η εξίσωση $f(x)$ της (4.1) μας δίνει την απόσταση ενός σημείου x από το διαχωριστικό επίπεδο $f(x) = w \cdot x + b = 0$. Εφόσον τα δεδομένα μας είναι διαχωρίσιμα, μπορούμε να βρούμε μια συνάρτηση $f(x) = w \cdot x + b$ για την οποία αν ισχύει $y_i f(x_i) > 0 \ \forall i$, τότε οι παρατηρήσεις έχουν ταξινομηθεί σωστά. Αν προστεθούν δε και τα συμπεράσματα των εξισώσεων (4.3) και (4.4), τότε το πρόβλημα είναι ουσιαστικά να ευρεθούν τα w, b που ικανοποιούν τον περιορισμό:

$$y_i f(x_i) \geq 1, i = 1, 2, \ldots, N. \tag{4.7}$$

Επιπλέον, από την σχέση (4.5) η μεγιστοποίηση του περιθωρίου είναι ισοδύναμη με την ελαχιστοποίηση του $\|w\|$ ή ισοδύναμα του $\frac{\|w\|^2}{2}$[4]. Το πλήρες πρόβλημα

[4]η επιλογή αυτού του μετασχηματιμού επιτρέπει την χρήση βελτιστοποίησης Τετραγωνικού Προγραμματισμού (quadratic programming)

βελτιστοποίησης που πρέπει να επιλυθεί συνεπώς είναι:

$$\min_{w} \frac{\|w\|^2}{2} \text{υπό τον περιορισμό } y_i f(x_i) \geq 1, i = 1, 2, \ldots, N. \tag{4.8}$$

Από την παραπάνω εξίσωση βλέπουμε ότι η αντικειμενική συνάρτηση είναι τετραγωνική ενώ συνολικά το πρόβλημα είναι κυρτό και μπορεί να επιλυθεί με την βοήθεια της μεθόδου πολλαπλασιαστών του Lagrange. Για να μπορέσουμε να χρησιμοποιήσουμε την μέθοδο, θα πρέπει η σχέση (4.8) να συμπυκνωθεί στην μορφή της Λανγκρατζιανής συνάρτησης:

$$L_p = \frac{1}{2}\|w\|^2 - \sum_{\iota=1}^{\mathrm{N}} \lambda_i \left(y_i f(x_i) - 1\right) \tag{4.9}$$

όπου $\lambda_i \geq 0$ είναι οι πολλαπλασιαστές Lagrange. Ο λόγος που η Λανγκρατζιανή συνάρτηση έχει αυτή την μορφή δεν είναι τυχαίος. Εκτός από την αντικειμενική συνάρτηση, ο λόγος που επιλέχθηκε να αφαιρείται ο περιορισμός από την αντικειμενική συνάρτηση βασίζεται στο γεγονός ότι για την ελαχιστοποίηση της συνάρτησης L_p παίρνουμε:

$$\frac{\partial L_p}{\partial w} = 0 \Leftrightarrow w = \sum_{i=1}^{N} \lambda_i y_i x_i, \tag{4.10}$$

$$\frac{\partial L_p}{\partial b} = 0 \Leftrightarrow \sum_{i=1}^{N} \lambda_i y_i = 0. \tag{4.11}$$

Ωστόσο, και πάλι δεν μπορούμε να πάρουμε λύση για το πρόβλημα διότι οι πολλαπλασιαστές Lagrange είναι άγνωστοι. Οι συντελεστές λ_i θα μπορούσαν να υπολογιστούν εύκολα αν στην εξίσωση (4.7) κρατούσαμε μόνο την ισότητα. Έτσι θα είχαμε N εξισώσεις οι οποίες μαζί με τις σχέσεις (4.10) και (4.11) θα μας έδιναν όλες τις αποδεκτές λύσεις για τα w, b και λ_i. Η μετατροπή των ανισοτήτων της σχέσης (4.7) σε ισότητες οδηγεί με την σειρά της στην επιβολή των παρακάτω περιορισμών για τους πολλαπλασιαστές λ_i, οι οποίοι είναι γνωστοί και ως συνθήκες των Karush-Kuhn-Tucker (KKT):

$$\lambda_i \geq 0 \tag{4.12}$$

$$\lambda_i \left(y_i f(x_i) - 1\right) = 0. \tag{4.13}$$

Η πληροφορία που παρέχει ο παραπάνω περιορισμός είναι ότι για μια παρατήρηση x_i ο αντίστοιχος πολλαπλασιαστής της $\lambda_i > 0$ αν και μόνο αν $y_i f(x_i) = 1$. Αν ισχύει κάτι τέτοιο, τότε αυτή η παρατήρηση θα κείται πάνω σε κάποια από τις δύο παράλληλες ως προς το διαχωριστικό επίπεδο ευθείες και θα είναι ένα διάνυσμα υποστήριξης. Οι παρατηρήσεις για τις οποίες ισχύει $\lambda_i = 0$ θα είναι εκατέρωθεν των δύο αυτών ευθειών. Σε αυτό το σημείο συνεπώς με την βοήθεια των σχέσεων (4.10) και (4.13) βλέπουμε ότι εξάγεται ένα πολύ σημαντικό συμπέρασμα: οι παράμετροι w, b που χαρακτηρίζουν το διαχωριστικό επίπεδο εξαρτώνται αποκλειστικά και μόνο από τα διανύσματα υποστήριξης.

Αντικαθιστώντας τώρα το αποτελέσματα των σχέσεων (4.10) και (4.11) στην (4.9), προκύπτει η εξίσωση:

$$L_D = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j x_i x_j \Leftrightarrow L_D = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i H_{ij} \lambda_j \Leftrightarrow$$

$$\Leftrightarrow L_D = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \lambda^{\mathrm{T}} H_{ij} \lambda \tag{4.14}$$

όπου $H_{ij} = y_i y_j x_i x_j$.Η εξίσωση (4.14) είναι η διπλή μορφή της (4.9) και η διαφορά της είναι ότι περιλαμβάνει μόνο τους πολλαπλασιαστές και τα δεδομένα από το σύνολο εκπαίδευσης[5] ενώ οι λύσεις που δίνει είναι απολύτως ισοδύναμες. Επιπλέον, επειδή ο τετραγωνικός όρος της εξίσωσης έχει πλέον αρνητικό πρόσημο, δεν έχουμε να λύσουμε ένα πρόβλημα ελαχιστοποίησης αλλά μεγιστοποίησης. Η σχέση (4.14) και αυτή με την σειρά αποτελεί ένα κυρτό τετραγωνικό πρόβλημα βελτιστοποίησης το οποίο μπορεί να επιλυθεί με Τετραγωνικό προγραμματισμό επιστρέφοντας ως αποτέλεσμα τους πολλαπλασιαστές λ_i. Γνωρίζοντας τα λ_i , από την σχέση (4.10) παίρνουμε το w και ακολούθως αντικαθιστούμε την σχέση (4.11) στην (4.10). Τέλος, με χρήση της σχέσης (4.7) και βρίσκοντας τον μέσο όρο των παρατηρήσεων x_i υπολογίζεται το b . Έχοντας πλέον στα χέρια μας τις τιμές w, b μπορούμε να ορίσουμε το βέλτιστο διαχωριστικό επίπεδο και κατ΄ επέκταση την μηχανή SVM.

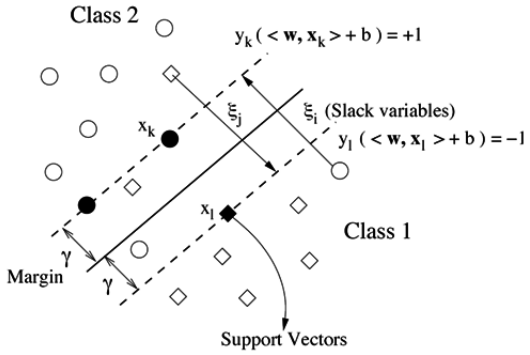
4.2.1.2 Μη διαχωρίσιμα δεδομένα

Τα μη διαχωρίσιμα δεδομένα δεν αποτελούν τόσο μια διαφορετική κατηγορία προβλήματος αλλά περισσότερο μια επέκταση της μελέτης πάνω στις γραμμικές μηχανές SVM. Ξεκινώντας κατ ΄ αρχάς, με τον όρο μη διαχωρίσιμα εννοούμε τα δεδομένα τα οποία δεν είναι πλήρως διαχωρισμένα, δηλαδή ανάμεσα τους υπάρχουν και παρατηρήσεις δεν έχουν ταξινομηθεί σωστά και ανήκουν στον ενδιάμεσο χώρο που ορίζουν οι δύο ευθείες των διανυσμάτων υποστήριξης .

Προκειμένου να ταξινομηθούν σωστά αυτές οι παρατηρήσεις θα χρειαστεί να χαλαρώσουμε λίγο τους περιορισμούς που θέσαμε στην σχέση (4.8) για να επιτρέπουν τα ελαφρώς μη ταξινομημένα στοιχεία. Αυτό θα γίνει με την εισαγωγή μιας θετικής χαλαρής μεταβλητής (slack variable) $\xi = \{\xi_1, \xi_2, \ldots, \xi_{\mathrm{N}}\}, \xi_i \geq 0$ που εκτιμά την απόσταση από το σημείο της εσφαλμένα ταξινομημένης παρατήρησης μέχρι την ευθεία των διανυσμάτων υποστήριξης που αντιστοιχεί στην κατηγορία που θα έπρεπε να ανήκουν. Όσο μεγαλύτερη είναι η απόσταση του σημείου από την ευθεία, τόσο μεγαλύτερη είναι και η τιμή της χαλαρής μεταβλητής. Προφανώς, η μεταβλητή παίρνει την τιμή 0 για τις σωστά ταξινομημένες παρατηρήσεις εκατέρωθεν των δύο ευθειών των διανυσμάτων υποστήριξης. Με την εισαγωγή της νέας αυτής μεταβλητής, οι εξισώσεις (4.2), (4.3) και (4.4) μετασχηματίζονται ως εξής:

$$w \cdot x_i + b \geq +1 - \xi_i, \ \gamma\iota\alpha \ y_i = +1 \tag{4.15}$$

[5]αυτό το τέχνασμα θα αποδειχθεί πολύ σημαντικό κατά την χρήση συναρτήσεων-πυρήνα που θα δούμε παρακάτω



Σχήμα 4.4: Διαχωριστικό επίπεδο για μια γραμμική μηχανή SVM μη διαχωριζόμενων δεδομένων

$$w \cdot x_i + b \leq -1 + \xi_i, \ \gamma\iota\alpha \ y_i = -1 \tag{4.16}$$

όπου $\xi_\iota \geq 0 \ \forall i$. Αντίστοιχα, ο συνδυασμός αυτών των εξισώσεων-περιορισμών οδηγεί στην συνοπτική μορφή:

$$w \cdot x_i + b - 1 + \xi_i \geq 0, \ \xi_i \geq 0 \ \forall i \tag{4.17}$$

Εφόσον έχουμε το σύνολο το περιορισμών μας θα μπορούσε κανείς να πει ότι το επόμενο βήμα στην ανάλυση θα ήταν να χρησιμοποιήσουμε και πάλι την σχέση (4.9) προκειμένου να υπολογίσουμε κατά την παραπάνω διαδικασία το διαχωριστικό επίπεδο. Αυτό όμως θα ήταν λάθος, διότι σε αντίθεση με τα πλήρως διαχωριζόμενα δεδομένα, στην περίπτωση που εξετάζεται τώρα δεν υπάρχουν κανείς περιορισμός για τον αριθμό των εσφαλμένων ταξινομήσεων που μπορεί να προκύψουν από την κατασκευή του διαχωριστικού επιπέδου. Αυτό πρακτικά σημαίνει ότι το διαχωριστικό επίπεδο που θα προέκυπτε θα είχε ναι μεν μεγάλο περιθώριο αλλά θα ταξινομούσε λάθος τις παρατηρήσεις. Σε αυτού του είδους το πρόβλημα έρχεται να δώσει λύση η μέθοδος «μαλακού» περιθωρίου (soft margin), η οποία ουσιαστικά θέτει τους όρους για την μεγιστοποίηση του περιθωρίου με ταυτόχρονη ελαχιστοποίηση των λάθος κατηγοριοποιήσεων.

Σύμφωνα με την μέθοδο, για να αποτραπεί ο μεγάλος αριθμός εσφαλμένων παρατηρήσεων, η σχέση (4.9) θα τροποποιηθεί καταλλήλως με την εισαγωγή μιας παραμέτρου ποινής C, η οποία θα αυξάνει όταν η απόσταση των εσφαλμένα ταξινομημένων σημείων από την εκάστοτε ευθεία των διανυσμάτων υποστήριξης παίρνει πολύ μεγάλες τιμές. Η τροποποιημένη εξίσωση θα είναι:

$$\min \left(\frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} \xi_i \right) \tag{4.18}$$

υπό τον περιορισμό της εξίσωσης (4.17). Η τροποποιημένη Λανγκρατζιανή που ακολουθεί θα ελαχιστοποιείται σε σχέση με τις w, b και ξ_i και θα μεγιστοποιείται

ως προς τους πολλαπλασιαστές λ_i:

$$L_P = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\lambda_i\left(y_i f\left(x_i\right) - 1 + \xi_i\right) - \sum_{i=1}^{N}\mu_i\xi_i \tag{4.19}$$

όπου οι δύο πρώτοι όροι αποτελούν την αντικειμενική συνάρτηση που πρέπει να ελαχιστοποιηθεί, ο τρίτος όρος τους περιορισμούς από την ανισότητα (4.17) για τις χαλαρές μεταβλητές και ο τελευταίος όρος εκφράζει την θετικότητα των χαλαρών μεταβλητών. Αντίστοιχα πάλι, οι ανισοτικές σχέσεις (4.17) των περιορισμών του προβλήματος μπορούν να μετασχηματιστούν κατά τον ίδιο τρόπο στις συνθήκες Karush-Kuhn-Tucker (KKT):

$$\xi_i \geq 0,\ \lambda_i \geq 0,\ \mu_i \geq 0 \tag{4.20}$$

$$\lambda_i\left\{y_i f\left(x_i\right) - 1 + \xi_\iota\right\} = 0 \tag{4.21}$$

$$\mu_i\xi_i = 0 \tag{4.22}$$

Διαφορίζοντας την L_P ως προς τις w, b και ξ_i και τις θέτουμε ίσες με 0:

$$\frac{\partial L_P}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{N}\lambda_\iota y_i x_i \tag{4.23}$$

$$\frac{\partial L_P}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N}\lambda_i y_i = 0 \tag{4.24}$$

$$\frac{\partial L_P}{\partial \xi_i} = 0 \Rightarrow C = \mu_i + \lambda_\iota \tag{4.25}$$

Στην εξίσωση (4.25) βρίσκεται κρυμμένος ένας ακόμα περιορισμός για τους πολλαπλασιαστές λ_i. Από την θεωρία της μεθόδου πολλαπλασιαστών Lagrange ξέρουμε ότι $\lambda_i \geq 0$. Ωστόσο, από την σχέση (4.25) βλέπουμε ότι το λ_i δεν θα μπορούσε ποτέ να είναι μεγαλύτερο του C. Συνεπώς, ο περιορισμός των λ_i για το πρόβλημα των μη διαχωριζόμενων δεδομένων σε μια γραμμική SVM είναι $0 \leq \lambda_i \leq C$. Αντικαθιστώντας τις τρεις παραπάνω εξισώσεις στην Λανγκρατζιανή της σχέσης (4.19) παίρνουμε μετά από πράξεις:

$$L_D = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i,j}\lambda_i\lambda_j x_i x_j y_i y_j = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\lambda^{\mathrm{T}} H_{ij}\lambda \tag{4.26}$$

με $H_{ij} = y_i y_j x_i x_j$. Συνεπώς καταλήγουμε πάλι σε ένα πρόβλημα μεγιστοποίησης:

$$\max\left\{\sum_{i=1}^{N}\lambda_i - \frac{1}{2}\lambda^{\mathrm{T}} H_{ij}\lambda\right\} \text{υπό τους περιορισμούς } 0 \leq \lambda_i \leq C,\ \sum_{i=1}^{N}\lambda_i y_i = 0$$

το οποίο επιλύεται πάλι αριθμητικά με τετραγωνικό προγραμματισμό για τον υπολογισμό των πολλαπλασιαστών λ_i. Στη συνέχεια,για τον υπολογισμό και των υπολοίπων παραμέτρων του διαχωριστικού επιπέδου, οι τιμές των λ_i αντικαθίστανται στις εξισώσεις (4.23)-(4.25) και στις συνθήκες (KKT).

4.2.2 Μη-γραμμικές SVMs

Στην προηγούμενη παράγραφο αναλύθηκαν όλες οι πιθανές περιπτώσεις για την ταξινόμηση δεδομένων με γραμμικό διαχωριστικό επίπεδο. Όμως δεν είναι όλα τα δεδομένα γραμμικώς διαχωρίσιμα. Για την περίπτωση των δεδομένων αυτών έχει αναπτυχθεί μια ξεχωριστή μεθοδολογία SVM που έχει ως κεντρική ιδέα την μεταφορά των δεδομένων από τον αρχικό χώρο συντεταγμένων x σε ένα χώρο $\Phi(x)$ στον οποίο θα είναι εφικτή η κατασκευή γραμμικού διαχωριστικού επιπέδου. Την διαδικασία αυτή έρχονται να πραγματοποιήσουν οι συναρτήσεις πυρήνα (kernel functions) που θα αναπτυχθούν παρακάτω.

4.2.2.1 Εκμάθηση μη-γραμμικών SVMs

Έστω ένα σύνολο δεδομένων εκπαίδευσης με μια δυαδική μεταβλητή απόκρισης για το οποίο το διαχωριστικό επίπεδο που προκύπτει με εφαρμογή του αλγορίθμου SVM δεν είναι γραμμικό. Εφόσον, το πρόβλημα αυτό είναι απόρροια της «κακής» τοποθέτησης των δεδομένων στον χώρο των παρατηρήσεων, μια πιθανή λύση για την αντιμετώπισή του είναι η εύρεση μιας μη γραμμικής συνάρτησης $\Phi(x)$ που να μεταφέρει τα δεδομένα από τον χώρο των παρατηρήσεων σε ένα νέο χώρο, όπου το διαχωριστικό επίπεδο θα ήταν γραμμικό. Για να γίνει περισσότερο κατανοητή η χρήση αυτής της συνάρτησης παραθέτουμε το ακόλουθο παράδειγμα:

Παράδειγμα: Έστω η συνάρτηση στόχου που αφορά το νόμο του Νεύτωνα για την βαρύτητα που εκφράζει την βαρυτική δύναμη μεταξύ δύο σωμάτων m_1, m_2 σε απόσταση r από το έδαφος

$$y = f(m_1, m_2, r) = G\frac{m_1 m_2}{r^2}$$

Οι παρατηρούμενες ποσότητες (επεξηγηματικές μεταβλητές) είναι οι μάζες m_1, m_2 και η απόσταση r. Με βάση την παραπάνω εξίσωση αλλά και όσα είπαμε στην προηγούμενη παράγραφο για τις γραμμικές SVMs είναι προφανές ότι το πρόβλημα αυτό δεν εμπίπτει στην κατηγορία των γραμμικών SVMs. Ωστόσο, μια μικρή αλλαγή στις συντεταγμένες

$$(m_1, m_2, r) \mapsto (x, y, z) = (\ln m_1, \ln m_2, \ln r)$$

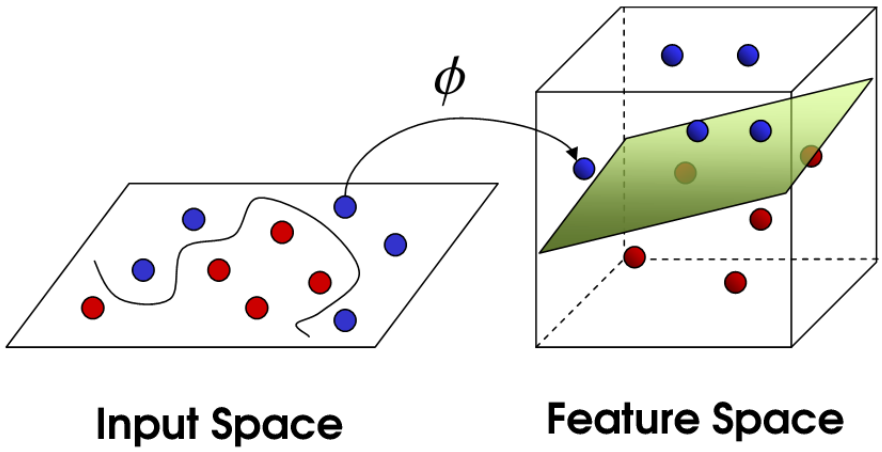
θα μπορούσε να δώσει την παρακάτω απεικόνιση

$$g(x, y, z) = \ln f(m_1, m_2, r) = \ln G + \ln m_1 + \ln m_2 - 2\ln r = G^{'} + x + y - 2z$$

που είναι μια γραμμική συνάρτηση, στην οποία μπορεί κατ΄ επέκταση να εφαρμοστεί η θεωρία των γραμμικών SVMs.

Συνεπώς, η εκμάθηση της SVM δεν βασίζεται πλέον πάνω στις παρατηρήσεις x_i αλλά στις μετασχηματισμένες $\Phi(x_i)$. Αυτή η τροποποίηση των δεδομένων μεταφράζεται μαθηματικά στο ακόλουθο πρόβλημα βελτιστοποίησης:

$$\min \frac{\|w\|^2}{2}$$



Σχήμα 4.5: Απεικόνιση των δεδομένων από τον χώρο των παρατηρήσεων στον χώρο των επεξηγηματικών μεταβλητών με την συνάρτηση Φ και κατασκευή του γραμμικού διαχωριστικού επιπέδου

$$\text{υπό τον περιορισμό } y_i\,(w \cdot \Phi(x_i) + b) \geq 1,\ \ i = 1, 2, \ldots, N. \tag{4.27}$$

Συγκρίνοντας αυτό το πρόβλημα με το αντίστοιχο των γραμμικών SVMs, παρατηρούμε ότι η μόνη τους διαφορά εντοπίζεται στην αντικατάσταση των αρχικών δεδομένων από τα τροποποιημένα. Είναι φανερό ότι ο τρόπος επίλυσης του προβλήματος παραμένει ίδιος κατά τα γνωστά. Η Λανγκρατζιανή για το πρόβλημα των μη γραμμικών SVMs είναι:

$$L_D = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j \Phi(x_i) \Phi(x_j) \tag{4.28}$$

Μηδενίζοντας τις μερικές παραγώγους της (4.28) ως προς w, b παίρνουμε:

$$w = \sum_{i=1}^{N} \lambda_i y_i \Phi(x_i) \tag{4.29}$$

$$\lambda_i \left\{ y_i \left(\sum_j \lambda_j y_j \Phi(x_j) \Phi(x_i) + b \right) - 1 \right\} = 0 \tag{4.30}$$

όπου οι πολλαπλασιαστές λ_i προσδιορίζονται με χρήση τετραγωνικού προγραμματισμού.

4.2.2.2 Συναρτήσεις Πυρήνα (Kernel Functions)

Στην προηγούμενη ενότητα αποφύγαμε εσκεμμένα να δώσουμε διευκρινίσεις σε ένα πολύ σημαντικό ερώτημα: ποια ή ποιες είναι οι συναρτήσεις που θα μας επιτρέπουν κάθε φορά τον γραμμικό διαχωρισμό των δεδομένων. Προφανώς, μια οποιαδήποτε συνάρτηση που δίνει ένα γραμμικό διαχωρισμό δεν είναι το ζητούμενο σε αυτή την περίπτωση καθώς μπορεί να αποδειχθεί μια πολύ ακριβή υπολογιστικά διαδικασία σε χώρους υψηλότερων διαστάσεων. Την απάντηση σε αυτό το ερώτημα έρχεται να δώσει η μέθοδος συναρτήσεων πυρήνα, θέτοντας τους απαραίτητους περιορισμούς.

Η μέθοδος των συναρτήσεων πυρήνα είναι από τις πλέον δημοφιλείς και αποτελεσματικές στον τομέα της μηχανικής εκμάθησης. Το σκεπτικό τους βασίζεται στο τέχνασμα του πυρήνα (kernel trick) το οποίο μπορεί να εφαρμοστεί σε κάθε γραμμικό αλγόριθμο που βασίζεται σε δεδομένα από την άποψη των εσωτερικών γινομένων μεταξύ των παρατηρήσεων.

Για να εξηγήσουμε καλύτερα τον ρόλο των εσωτερικών γινομένων στην μέθοδο θα θεωρήσουμε ότι έχουμε ένα πίνακα δεδομένων $N \times M$, όπου N είναι ο αριθμός των παρατηρήσεων και M ο αριθμός των μεταβλητών (επεξηγηματικών και απόκρισης). Για τον πίνακα αυτό θα εκτελέσουμε ένα τυπικό διάγραμμα διασποράς, στο οποίο οι παρατηρήσεις τοποθετούνται στο διανυσματικό χώρο που ορίζεται από τις μεταβλητές. Στη συνέχεια, θα κατασκευάσουμε τον συμμετρικό $M \times M$ πίνακα εξωτερικού γινομένου (cross- product matrix) των δεδομένων, όπου κάθε μη-διαγώνιο στοιχείο εκφράζει το μέτρο της συσχέτισης μεταξύ δύο μεταβλητών[6]. Ο λόγος που μπήκαμε στη διαδικασία να κατασκευάσουμα αυτόν το πίνακα δεν είναι άλλος από τον ορισμό του εσωτερικού γινομένου μεταξύ των παρατηρήσεων, μιας και το άθροισμα[7] των εξωτερικών γινομένων δύο παρατηρήσεων, έστω i και j, ισούται με το εσωτερικό τους γινόμενο.

Στην στατιστική έχουμε δει αρκετές φορές ότι το εσωτερικό γινόμενο δύο παρατηρήσεων x_i, x_j θεωρείται ως ένα μέτρο συσχέτισης τους. Το τέχνασμα του πυρήνα επιτρέπει ουσιαστικά τον απολύτως ανάλογο ορισμό αυτού του μέτρου συσχέτισης μέσω του εσωτερικού γινομένου και για τις μετασχηματισμένες παρατηρήσεις $\Phi(x_i), \Phi(x_j)$ θέτοντας:

$$K(x_i, x_j) = \Phi(x_i)\Phi(x_j) \tag{4.31}$$

Η συνάρτηση $K(x_i, x_j)$ αποτελεί το ισοδύναμο μέτρο συσχέτισης παρατηρήσεων στον μετασχηματισμένο χώρο και ονομάζεται συνάρτηση πυρήνα. Μερικές ευρέως γνωστές συναρτήσεις πυρήνα που μπορεί να συναντήσει κανείς σε σχετική με τις SVMs βιβλιογραφία είναι οι:

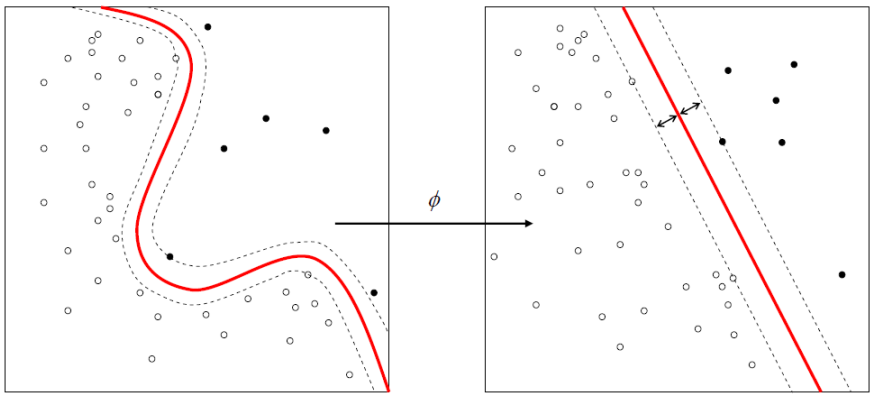
$$\text{Πολυωνυμική } d\text{−ου βαθμού: } K(x_i, x_j) = (1 + x_i x_j)^d$$

$$\text{Ακτινικής Βάσης: } K(x_i, x_j) = \exp\left(-\gamma (x_i - x_j)^2\right)$$

$$\text{Νευρωνικών Δικτύων: } K(x_i, x_j) = \tanh\left(\kappa_1 (x_i x_j) + \kappa_2\right)$$

[6] με την χρήση κατάλληλης κλιμακοποίησης ο πίνακας μετασχηματίζεται σε έναν τυπικό πίνακα συσχέτισης

[7] το άθροισμα αυτό αποτελεί ουσιαστικά την απόσταση μεταξύ των παρατηρήσεων i και j



Σχήμα 4.6: Απεικόνιση των δεδομένων με χρήση συνάρτησης πυρήνα

Το βασικότερο πλεονέκτημα που μας προσφέρει ο ορισμός αυτής της συνάρτησης είναι ότι πλέον δεν χρειάζεται να γνωρίζουμε την ακριβή τιμή της Φ. Αυτό γίνεται γιατί κάθε συνάρτηση πυρήνα που χρησιμοποιείται σε πρόβλημα εύρεσης γραμμικού επιπέδου για μη γραμμικές SVMs πρέπει να ικανοποιεί το ακόλουθο θεώρημα:

Θεώρημα Mercer: *Μια συνάρτηση πυρήνα K ορίζεται ως το εσωτερικό γινόμενο*

$$K(x_i, x_j) = \Phi(x_i)\,\Phi(x_j)$$

αν και μόνο αν για κάθε συνάρτηση $g(x)$, για την οποία το αόριστο ολοκλήρωμα $\int g(x)^2\,dx$ είναι πεπερασμένο, ισχύει ότι:

$$\int K(x_i, x_j)\,g(x)\,g(y)\,dxdy \geq 0.$$

Το θεώρημα του Mercer μας εξασφαλίζει ότι η συνάρτηση πυρήνα θα εκφράζεται πάντα σαν το εσωτερικό γινόμενο δύο μετασχηματισμένων παρατηρήσεων σε ένα χώρο υψηλών διαστάσεων. Το συμπέρασμα αυτό είναι κομβικής σημασίας για το υπολογιστικό κομμάτι της μεθόδου καθώς οι συναρτήσεις πυρήνα είναι σαφώς οικονομικότερες υπολογιστικά από την συνάρτηση Φ. Επιπλέον, από τις σχέσεις (4.31-4.34) βλέπουμε ότι η συνάρτηση πυρήνα υπολογίζεται από τις αρχικές παρατηρήσεις και όχι από τις μετασχηματισμένες. Επομένως, υπολογιστικά θέματα που μπορούν να προκύψουν[8] από τον μετασχηματισμό των δεδομένων εξαλείφονται. Κλείνοντας, ο χώρος που παράγεται με την χρήση πυρήνων στις SVMs ονομάζεται χώρος Hilbert αναπαραγόμενου πυρήνα (Reproducing Kernel

[8] και σχετίζονται με την κατάρα των διαστάσεων

Hilbert Space- RKHS). Για περισσότερες πληροφορίες πάνω στις συναρτήσεις πυρήνα στην ταξινόμηση παραπέμπουμε τον αναγνώστη στους Herbrich (2002) και N.Cristianini,J.Shawe-Taylor (2000).

4.3 Ταξινόμηση στις SVMs για προβλήματα πολλαπλών κατηγοριών

Οι μηχανές διανυσμάτων υποστήριξης είναι από τον ορισμό τους ταξινομητές για δυαδικά προβλήματα. Αυτό δεν σημαίνει ωστόσο ότι δεν μπορούν να χρησιμοποιηθούν σαν τεχνική ταξινόμησης για προβλήματα με μεταβλητές απόκρισης πολλαπλών κατηγοριών. Σε αυτή την παράγραφο θα παρουσιάσουμε κάποιες από τις τεχνικές που έχουν αναπτυχθεί και συνδυάζουν ταξινομητές δυαδικών προβλημάτων με τελικό στόχο την κατασκευή ενός ταξινομητή SVM για προβλήματα όπου η μεταβλητή απόκρισης έχει $K > 2$ κατηγορίες.

Η πρώτη προσέγγιση που προτάθηκε για την αντιμετώπιση του προβλήματος ήταν από τον Vapnik (1998) και αφορούσε στην κατασκευή K διαφορετικών δυαδικών προβλημάτων για ένα πρόβλημα με K κατηγορίες στην μεταβλητή απόκρισης. Αναλυτικότερα, έστω ότι επιλέγουμε την κατηγορία k από το σύνολο των K κατηγοριών της απόκρισης. Αν θέσουμε την τιμή $+1$ στην κατηγορία k που επιλέξαμε και στις υπόλοιπες $K-1$ κατηγορίες την τιμή -1, προκύπτει άμεσα ένα δυαδικό πρόβλημα, το οποίο μπορεί να λυθεί κατά τα γνωστά. Η ίδια διαδικασία επαναλαμβάνεται για κάθε μια από τις κατηγορίες της απόκρισης και μας δίνει στο σύνολο $K(K-1)/2$ δυαδικά προβλήματα. Η μέθοδος αυτή είναι γνωστή και «ένας-εναντίον-των-υπολοίπων» (one-against-the-rest). Για κάθε παρατήρηση του συνόλου εξέτασης, η ταξινόμηση γίνεται με βάση τη σχέση:

$$y(x) = \max_k y_k(x).$$

Η παραπάνω τεχνική, αν και φαίνεται απλή στην εφαρμογή της, εμφανίζει τα δύο ακόλουθα σημαντικά προβλήματα. Πρώτον, η ομαδοποίηση κατηγοριών που κάνουμε σε κάθε ένα από τα K δυαδικά προβλήματα μπορεί να οδηγήσει σε ασυνεπή αποτελέσματα διότι επιτρέπει στις παρατηρήσεις να κατατάσσονται ταυτόχρονα σε διάφορες κατηγορίες. Το δεύτερο πρόβλημα εντοπίζεται στο ότι τα σύνολα εκπαίδευσης που χρησιμοποιούνται σε κάθε δυαδικό πρόβλημα δεν είναι «ισορροπημένα». Για να αντιληφθεί ο αναγνώστης την έννοια του «ισορροπημένου» συνόλου παραθέτουμε το κάτωθι παράδειγμα:

> **Παράδειγμα**: Έστω ότι έχουμε ένα σύνολο εκπαίδευσης όπου η μεταβλητή απόκρισης έχει δέκα κατηγορίες. Όλες οι κατηγορίες έχουν ισάριθμες παρατηρήσεις. Άρα, για την επίλυση του εκάστοτε δυαδικού προβλήματος θα έχουμε 10% θετικές παρατηρήσεις και 90% αρνητικές! Με άλλα λόγια, το πρόβλημα έχει χάσει την συμμετρία του.

Για να διορθωθεί το θέμα που προκύπτει με την ισορροπία των συνόλων εκπαίδευσης, οι Lee et al.(2001) πρότειναν μια ελαφριά παραλλαγή της «ένας-εναντίον-των-υπολοίπων» τεχνικής. Σύμφωνα με αυτή την πρόταση, κατά την ομαδοποίηση των

κατηγοριών της μεταβλητής απόκρισης, η επιλεγμένη την κάθε φορά κατηγορία θα παίρνει την τιμή $+1$ ενώ οι υπόλοιπες κατηγορίες την τιμή $-\frac{1}{K-1}$.

Η δεύτερη προσέγγιση για την ταξινόμηση σε προβλήματα πολλαπλών κατηγοριών ακολουθεί την ίδια λογική ως προς την κατασκευή των δυαδικών προβλημάτων αλλά διαφοροποιείται ως προς την εκτέλεση. Σε αντίθεση με την προηγούμενη μέθοδο όπου κατασκευάστηκαν ξεχωριστά $K\left(K-1\right)/2$ ταξινομητές, εδώ η κατασκευή όλων των δυνατών ταξινομητών γίνεται ταυτόχρονα. Η ταξινόμηση των παρατηρήσεων του συνόλου εξέτασης γίνεται με ένα σύστημα «ψήφων». Η παρατήρηση τοποθετείται στην κατηγορία με τον μεγαλύτερο αριθμό «ψήφων». Η παραπάνω διαδικασία είναι γνωστή και ως «ένας-εναντίον-ενός» (one-against-one). Το βασικό μειονέκτημα αυτής της μεθόδου γίνεται αισθητό για πολύ μεγάλο αριθμό κατηγοριών καθώς όσο αυξάνουν οι κατηγορίες ανάλογα αυξάνει και ο χρόνος εκπαίδευσης. Αντίστοιχα, χρονοβόρα διαδικασία είναι και η ταξινόμηση των δεδομένων του συνόλου εξέτασης λόγω των αρκετών υπολογισμών.

Εν κατακλείδι, η ταξινόμηση για προβλήματα πολλαπλών κατηγοριών στις SVMs είναι ακόμα ένας ενεργός ερευνητικός κλάδος. Ωστόσο, παρά τα μειονεκτήματα που παρουσιάζουν οι παραπάνω τεχνικές, η μέθοδος του «ενός-εναντίον-των-υπολοίπων» επικρατεί έναντι αυτής του «ενός-εναντίον-ενός» και θεωρείται η πιο διαδεδομένη στη χρήση της.

4.4 Βελτίωση της απόδοσης ενός μοντέλου SVM

Είδαμε ότι κατά την δημιουργία ενός μοντέλου SVM, κεντρικό ρόλο διαγραμματίζουν οι παράμετροι w, b, οι οποίες με την χρήση κατάλληλων μαθηματικών μεθόδων, βελτιστοποιούνται με στόχο την εύρεση του ακριβέστερου γραμμικού επιπέδου. Συνήθως όμως, εξαιτίας του μεγάλου αριθμού δεδομένων που καλούνται να αξιοποιήσουν οι μέθοδοι βελτιστοποίησης, η γενικότερη διαδικασία παραγωγής του μοντέλου καταλήγει να είναι αρκετά χρονοβόρα. Ενδεικτικά αναφέρουμε ότι μια τέτοια αρκετά διαδεδομένη μέθοδος είναι το πλέγμα αναζήτησης (GS).

Είναι φανερό λοιπόν ότι και σε αυτή την μεθοδο ταξινόμησης είναι η παραγωγή ενός ακριβούς και ταυτόχρονα σύντομου σε εκτέλεση μοντέλου. Στην προσπάθειες για την μείωση του υπολογιστικού χρόνου των SVM έχουν συμβάλλει κατά καιρούς αρκετές επιστημονικές προτάσεις χωρίς αυτό όμως να σημαίνει ότι οι έρευνες έχουν σταματήσει. Οι Ou et al.(2003) πρότειναν ένα μηχανισμό μείωσης των δεδομένων προκειμένου να εξοικονομήσουμε χρόνο για την περάτωση της μεθόδου. Από τα πειραματικά αποτελέσματα δε, έχει αποδειχθεί ότι η επιτάχυνση της διαδικασίας έχει το ελάχιστο δυνατό κόστος στην ακρίβεια του μοντέλου. Τον επόμενο χρόνο, οι Zhu et al.(2004), έχοντας ως κεντρικό αντικείμενο το πλέγμα αναζήτησης (GS), πρότειναν έναν ενιαίο σχεδιασμό (uniform design-UD), ο οποίος σε συνδυασμό με την χρήση της μεθόδου παλινδρόμησης των μηχανών διανυσμάτων υποστήριξης (Support Vector Regression-SVR), μειώνει σημαντικά το κόστος υπολογισμού και άρα επιταχύνει την διαδικασία. Στο ίδιο μήκος κύματος, μια άλλη προσέγγιση έγινε από τους Lebrun et al.(2006), οι οποίοι πρότειναν μια νέα μέθοδο μάθησης για

την κατασκευή δίτιμης συνάρτησης αποφάσεων (Binary Decision Function-BDF) στις SVMs. Μέσω αυτής της μεθόδου μειώνεται η πολυπλοκότητα του μοντέλου και καθίσταται αποτελεσματικότερη η γενίκευση. Τέλος, οι Hwang et al.(2007) πρότειναν ένα σύνολο ενιαίων σχεδιασμών για την εύρωστη (robust) και αυτόματη επιλογή του μοντέλου στις SVMs. Σύμφωνα με αυτή την μέθοδο, επιλέγεται ένα σύνολο υποψήφιων συνδυασμών των παραμέτρων και εκτελείται μια k-fold cross validation για να αξιολογηθεί η απόδοση του κάθε συνδυασμού. Ο συνδυασμός με την καλύτερη απόδοση αποτελεί και το ζητούμενο μοντέλο.

Κεφάλαιο 5

Αξιολόγηση μοντέλου

5.1 Εισαγωγή

Ως απόδοση ενός μοντέλου ορίζουμε την ικανότητα σωστής πρόβλεψης του πάνω σε ένα σύνολο τυχαίων δεδομένων εξέτασης. Η αξιολόγηση της απόδοσης έχει καθοριστικό ρόλο στην συνολική διαδικασία της επεξεργασίας μιας βάσης δεδομένων καθώς αποτελεί ένα μέτρο της ποιότητας του τελικώς επιλέγομενου μοντέλου είτε καθορίζει την επόμενη μέθοδο εκμάθησης που πρέπει να ακολουθήσει ο ερευνητής. Σε αυτό το κεφάλαιο θα δώσουμε αρχικά κάποια θεωρητικά στοιχεία που αφορούν την απόδοση ενός μοντέλου και στη συνέχεια θα παρουσιάσουμε κάποιες βασικές μεθόδους για την αξιολόγηση της απόδοσης του εκάστοτε μοντέλου.

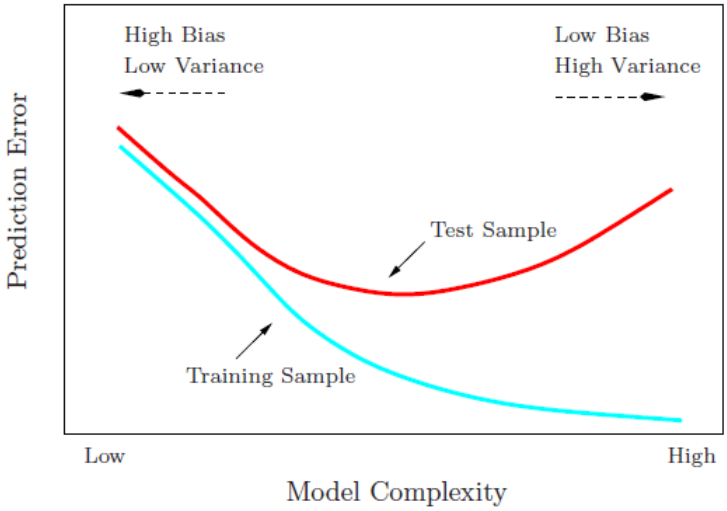
5.2 Μεροληψία, Διασπορά και Πολυπλοκότητα μοντέλου

Πριν προχωρήσουμε στην παράθεση των εννοιών του τίτλου της παραγράφου, θα ορίσουμε πρώτα το πρόβλημα στο οποίο θα δουλέψουμε. Έστω ένα σύνολο δεδομένων εξέτασης με N αριθμό παρητηρήσεων και μια μεταβλητή απόκρισης y(είτε ποσοτική είτε κατηορική). Αν μεταφράσουμε τα παραπάνω σε επίπεδο διανυσμάτων, θα έχουμε το διάνυσμα των δεδομένων X, το οποίο θα έχει ως συνιστώσες τις N παρατηρήσεις, και το N - διαστάσεων διάνυσμα της απόκρισης Y, ενώ μέσω αυτών μπορούμε να ορίσουμε την πρόβλεψη του μοντέλου $\hat{f}(X)$, η οποία έχει υπολογιστεί από ένα τυχαίο σύνολο εκπαίδευσης. Από την κλασσική στατιστική, οι δύο πιο χαρακτηριστικές μορφές συνάρτησης σφάλματος είναι το τετραγωνικό και το απόλυτο σφάλμα:

$$L\left(Y, \hat{f}(X)\right) = \left(Y - \hat{f}(X)\right)^2$$

και

$$L\left(Y, \hat{f}(X)\right) = |Y - \hat{f}(X)|$$



Σχήμα 5.1: Συμπεριφορά του αναμενόμενου σφάλματος εκπαίδευσης και του αναμενόμενου σφάλματος δοκιμών καθώς μεταβάλλεται η πολυπλοκότητα του μοντέλου. Η κόκκινη γραμμή αντιπροσωπεύει το αναμενόμενο σφάλμα δοκιμών, ενώ η μπλε το αναμενόμενο σφάλμα εκπαίδευσης

Σε όλα τα παραπάνω κεφάλαια γινόταν συχνή αναφορά στον όρο γενικευμένο σφάλμα χωρίς όμως να εξηγήσουμε περί τίνος πρόκειται. Έχοντας κάνει την παραπάνω εισαγωγή, είμαστε πλέον σε θέση να ορίσουμε ως γενικευμένο σφάλμα ή σφάλμα δοκιμών (test error) το σφάλμα πρόβλεψης για ένα τυχαίο σύνολο δεδομένων εξέτασης, το οποίο θα δίνεται από την κάτωθι σχέση:

$$Err_T = E\left(L\left(Y, \hat{f}\left(X\right)\right)|T\right) \tag{5.1}$$

Με βάση το παραπάνω σφάλμα μπορεί να οριστεί και ένας ακόμα τύπος σφάλματος, γνωστό και ως αναμενόμενο σφάλμα πρόβλεψης ή αναμενόμενο σφάλμα δοκιμών (expected prediction error / expected test error):

$$Err = E\left[L\left(Y, \hat{f}\left(X\right)\right)\right] = E\left[Err_T\right] \tag{5.2}$$

Σε αντιστοιχία με τα σφάλματα επί του συνόλου εξέτασης, ορίζεται το σφάλμα εκπαίδευσης ως η μέση απώλεια επί του συνόλου εκπαίδευσης και δίνεται από την σχέση:

$$\overline{err} = \frac{1}{N}\sum_{i=1}^{N} L\left(y_i, \hat{f}\left(x_i\right)\right) \tag{5.3}$$

Κατά την εφαρμογή του μοντέλου πρόβλεψης στο σύνολο εκπαίδευσης θα θέλαμε να γνωρίζουμε εξ΄ αρχής το αναμενόμενο σφάλμα δοκιμών. Όσο αυξάνει η πολυπλοκότητα του μοντέλου τόσο αυξάνεται και η εξάρτησή του από τα δεδομένα

εκπαίδευσης, καθώς περιλαμβάνει όλο και μεγαλύτερο αριθμό μεταβλητών. Φυσικό επακόλουθο αυτής της αύξησης των μεταβλητών είναι η ανάλογη αύξηση της διασποράς και αντίθετα η μείωση της μεροληψίας του μοντέλου. Από την Εικόνα 5.1 βλέπουμε ότι το σφάλμα εκπαίδευσης δεν μπορεί να αποτελέσει αξιόπιστο εκτιμητή του σφάλματος δοκιμών διότι όσο αυξάνεται η πολυπλοκότητα του μοντέλου, αυτό τείνει στο μηδέν. Εάν δε η πολυπλοκότητα πάρει πολύ μεγάλες τιμές, δεν αποκλείεται και ο μηδενισμός του σφάλματος εκπαίδευσης. Μηδενισμός του σφάλματος εκπαίδευσης ωστόσο ισοδυναμεί με υπερπροσαρμογή του μοντέλου στα δεδομένα εκπαίδευσης, κάτι που όπως έχει αποδειχθεί οδηγεί σε εσφαλμένα αποτελέσματα.

Η εκτίμηση του αναμενόμενου σφάλματος δοκιμών αποτελεί στόχο των μεθόδων που θα αναπτυχθούν παρακάτω σε αυτό το κεφάλαιο. Το παραπάνω πρόβλημα εκτίμησης στην πραγματικότητα διαιρείται σε δύο ξεχωριστά προβλήματα, που στόχο έχουν την βελτιστοποίηση της τιμής του αναμενόμενου σφάλματος:

- *Επιλογή μοντέλου (model selection)*: εκτίμηση της απόδοσης όλων των μοντέλων που μπορούν να παραχθούν και επιλογή του καλύτερου,
- *Εκτίμηση μοντέλου (model assessment)*: μετά την επιλογή του καλύτερου μοντέλου, εκτιμάται το σφάλμα δοκιμών πάνω στο σύνολο εξέτασης.

Εφόσον δουλεύουμε με δεδομένα υψηλών διαστάσεων, η καλύτερη προσέγγιση για την επίλυση των προβλημάτων αυτών και κατά συνέπεια την εύρεση της βέλτιστης τιμής για το σφάλμα δοκιμών είναι η τυχαία διαίρεση του συνόλου δεδομένων σε τρια μέρη. Ειδικότερα, τα μέρη αυτά είναι:

- *Σύνολο εκπαίδευσης (training set)*: χρησιμοποιείται για την παραγωγή των μοντέλων,
- *Σύνολο επικύρωσης (validation set)*: χρησιμοποιείται για την εκτίμηση των σφαλμάτων πρόβλεψης των παραπάνω μοντέλων και βάσει αυτών επιλέγεται το καλύτερο μοντέλο,
- *Σύνολο εξέτασης (test set)*: χρησιμοποιείται για την εκτίμηση του γενικευμένου σφάλματος του παραπάνω επιλεγμένου μοντέλου.

Όπως βλέπουμε το σύνολο επικύρωσης έχει μεγάλη σημασία στην πορεία για την εύρεση του σφάλματος δοκιμών αφού σε αυτό το στάδιο θα επιλεχθεί θεωρητικά το καλύτερο μοντέλο. Η εκτίμηση της απόδοσης των μοντέλων γίνεται είτε με τυπικές μεθόδους στατιστικής όπως οι δείκτες AIC και BIC είτε με την χρήση τεχνικών όπως η διασταυρωμένη επικύρωση και η bootstrap. Ωστόσο, δεν πρέπει να ξεχνάμε ότι κάθε μέθοδος εκμάθησης μπορεί να διαθέτει αντίστοιχα κάποια μέτρα για την εκτίμηση του σφάλματος δοκιμών τα οποία μπορούν, σε συνδυασμό με τις παραπάνω τεχνικές, να μας παρέχουν μια αξιόπιστη εκτίμηση του σφάλματος δοκιμής του τελικού μοντέλου. Τέλος, ο χωρισμός του συνόλου δεδομένων δεν είναι μια εύκολη υπόθεση καθώς δεν υπάρχει ούτε κάποιος γενικός κανόνας ούτε επαρκή στοιχεία τις περισσότερες φορές που να οδηγούν σε ένα διαχωρισμό. Μια τυπική διάσπαση που συναντάται είναι 50% για την εκπαίδευση και 25% για κάθε ένα από τα δύο επόμενα σύνολα.

		Πραγματική Τιμή	
Αποτέλεσμα Μοντέλου		Αληθές (T)	Ψευδές (F)
	Θετικά (p)	Αληθώς Θετικά (TP)	Ψευδώς Θετικά (FP)
	Αρνητικά (n)	Αληθώς Αρνητικά (TN)	Ψευδώς Αρνητικά (FN)

Πίνακας 5.1: Πίνακας συνάφειας

5.3 Απόδοση Ταξινομητή

Θα ξεκινήσουμε πάλι την ανάλυση μας ορίζοντας ένα τυχαίο πρόβλημα ταξινόμησης δεδομένων. Για περισσότερη ευκολία, θα επιλέξουμε ένα δυαδικό πρόβλημα ταξινόμησης, όπου η τιμή $+1$ αντιστοιχεί στις θετικές παρατηρήσεις και η τιμή -1 στις αρνητικές[1]. Από τον ορισμό του, ένα πρόβλημα ταξινόμησης είναι ένας κανόνας αντιστοίχισης παρατηρήσεων (του συνόλου εξέτασης) σε προβλεπόμενες κλάσεις. Με άλλα λόγια, κάθε παρατήρηση αντιστοιχίζεται σε ένα στοιχείο του συνόλου $\{p, n\}$ που εκπροσωπεί τις θετικές (positive) και αρνητικές (negative) κατηγορίες της μεταβλητής απόκρισης.

Έχοντας στην διάθεσή μας μια οποιαδήποτε παρατήρηση του συνόλου εξέτασης και τον ταξινομητή που έχουμε κατασκευάσει, έχουμε τέσσερα πιθανά διαφορετικά αποτελέσματα όσον αφορά την ταξινόμηση της παρατήρησης:

- *TP (True Positive)*: όταν ξέρουμε ότι μια παρατήρηση ανήκει στις θετικές παρατηρήσεις και ταξινομείται ως θετική, προσμετράται ως αληθώς θετική,
- *FP (False Positive)*:όταν ξέρουμε ότι μια παρατήρηση ανήκει στις αρνητικές παρατηρήσεις και ταξινομείται ως θετική, προσμετράται ως ψευδώς θετική,
- *TN (True Negative)*:όταν ξέρουμε ότι μια παρατήρηση ανήκει στις αρνητικές παρατηρήσεις και ταξινομείται στις αρνητικές, προσμετράται ως αληθώς αρνητική,
- *FN (False Negative)*:όταν ξέρουμε ότι μια παρατήρηση ανήκει στις θετικές παρατηρήσεις και ταξινομείται στις αρνητικές, προσμετράται ως ψευδώς αρνητική.

Δεδομένου του συνόλου παρατηρήσεων των δεδομένων εξέτασης και του ταξινομητή, οι παραπάνω ποσότητες ορίζουν ένα 2×2 πίνακα συνάφειας (contingency table). Οι αριθμοί κατά μήκος των διαγωνίων ανιπροσωπεύουν τις σωστές αποφάσεις ενώ οι αριθμοί εκτός της διαγωνίου τα λάθη ή αλλιώς την σύγχυση μεταξύ των κατηγοριών της μεταβλητής απόκρισης.

Ο πίνακας αυτός αποτελεί την βάση για τον ορισμό των παρακάτω ποσοτήτων:

[1]εντελώς ανάλογα δουλεύουμε και για τις ποσοτικές μεταβλητές. Ο χωρισμός τους σε κατηγορίες για την ευκολότερη διαχείριση του προβλήματος γίνεται με την επιλογή μιας τιμής που έχει το ρόλο του ορίου μεταξύ των κατηγοριών

1. Το Αληθώς Θετικό Ποσοστό (True Positive Rate - TPR) ή αλλιώς ποσοστό επιτυχίας (hit rate) ενός ταξινομητή ορίζεται η ποσότητα:

$$TP\ rate \cong \frac{TP}{P}$$

2. Το Ψευδώς Θετικό Ποσοστό (False Positive Rate - FPR) ενός ταξινομητή ορίζεται ως:

$$FP\ rate \cong \frac{FP}{N}$$

3. Η Ακρίβεια (accuracy) είναι η αναλογία των πραγματικών αποτελεσμάτων σε σχέση με τον πληθυσμό και ορίζεται ως:

$$accuracy = \frac{TP + TN}{P + N}$$

Το να έχουμε ακρίβεια 100% πρακτικά σημαίνει ότι οι τιμές που προκύπτουν μέσω του μοντέλου πρόβλεψης είναι ακριβώς ίδιες με τις τιμές απόκρισης που δίνονται εξ' αρχής από το σύνολο.

(α') Η Θετική Προγνωστική Αξία[2] (precision) είναι μια διαφορετική έκφραση της ακρίβειας και ορίζεται ως το ποσοστό των αληθώς θετικών παρατηρήσεων επί του συνόλου των θετικών παρατηρήσεων (θετικών και αρνητικών):

$$precision = \frac{TP}{TP + FP}$$

Η Θετική Προγνωστική Αξία ουσιαστικά αντιπροσωπεύει το σφάλμα τύπου I που είχαμε στους αντίστοιχους ελέγχους υποθέσεων.

4. Η Αρνητική Προγνωστική Αξία (Negative Predictive Value - NPV) ορίζεται ως:

$$NPV = \frac{TN}{FN + TN}$$

και εκφράζει αντιστοίχως το σφάλμα τύπου II των ελέγχων υποθέσεων.

Ευαισθησία και Ειδικότητα

Η ευαισθησία και η ειδικότητα είναι δύο στατιστικά μέτρα της απόδοσης για δυαδικά προβλήματα και σχετίζονται έντονα με τα σφάλματα τύπου I και II. Ειδικότερα έχουμε:

- το ποσοστό των αληθώς θετικών αποτελεσμάτων (δηλαδή το ποσοστο των θετικών ενδείξεων του πληθυσμού) (True Positive Rate - TPR) ή αλλιώς ευαισθησία (sensitivity) του προβλήματος, είναι η πιθανότητα το μοντέλο να

[2] η τροποποιημένη ελληνική μετάφραση του αγγλικού όρου γίνεται καθαρά για την διαφοροποίηση του μεγέθους αυτού από αυτό της ακρίβειας (accuracy)

μας δώσει θετικά αποτελέσματα δεδομένου ότι κάποιος έχει το χαρακτηριστικό που εξετάζουμε και δίνεται από την σχέση:

$$TPR = \frac{TP}{TP + FN}$$

Η ευαισθησία σχετίζεται με την ικανότητα του μοντέλου να δώσει θετικά αποτελέσματα. Ένα πρόβλημα με υψηλό δείκτη ευαισθησίας έχει χαμηλό ποσοστό σφάλματος τύπου ΙΙ.

- το ποσοστό των αληθώς αρνητικών αποτελεσμάτων (δηλαδή το ποσοστό των αρνητικών ενδείξεων του προβλήματος) (True Negative Rate - TNR) ή ειδικότητα (specificity) του προβλήματος είναι η πιθανότητα το μοντέλο να μας δώσει αρνητικά αποτελέσματα δεδομένου ότι κάποιος δεν έχει το χαρακτηριστικό που εξετάζουμε και δίνεται από την σχέση:

$$TNR = \frac{TN}{FP + TN}$$

Η ειδικότητα σχετίζεται με την ικανότητα του μοντέλου να δώσει αρνητικά αποτελέσματα, ενώ ένα πρόβλημα με υψηλό δείκτη ευαισθησίας έχει χαμηλό ποσοστό σφάλματος τύπου Ι.

Εφόσον τα παραπάνω ποσοστά εκφράζουν πιθανότητες, μπορούν να οριστούν αντίστοιχα και τα συμπληρωματικά τους ποσοστά:

- ποσοστό ψευδώς αρνητικών αποτελεσμάτων (False Negative Rate - FNR): $FNR = 1 - TPR$.
- ποσοστό ψευδώς θετικών αποτελεσμάτων (False Positive Rate - FPR): $FPR = 1 - TNR$.

Όλα τα παραπάνω ποσοστά μαζί με τα συμπληρωματικά τους ονομάζονται πιθανοφάνειες (likelihood) ή αλλιώς λειτουργικά χαρακτηριστικά (operating characteristics) του μοντέλου πρόβλεψης. Τέλος, είναι φανερό από τους ίδιους τους ορισμούς της ευαισθησίας και της ειδικότητας ότι ένας ιδανικός ταξινομητής θα πρέπει να έχει 100% ευαισθησία και 100% ειδικότητα.

Επιπολασμός

Με τον όρο επιπολασμό (prevalence) ορίζουμε το σύνολο των θετικών παρατηρήσεων (αληθών και ψευδών) προς το σύνολο όλων των παρατηρήσεω:

$$prevalence = \frac{TP + FP}{P + N}.$$

Ο επιπολασμός εκφράζει την πιθανότητα προ πειράματος (Pre-test Probability) και κατά συνέπεια τα ποσοστά PPV και NPV λειτουργούν συμπληρωματικά ως προς αυτόν. Επιπλέον, αν είναι γνωστός ο επιπολασμός του μοντέλου, μπορούμε

		Πραγματική Τιμή		
		Αληθές (T)	Ψευδές (F)	
Αποτέλεσμα Μοντέλου	Θετικό (p)	TP	FP	⟶ Θετική Προγνωστική Αξία (PPV)
	Αρνητικό (n)	TN	FN	⟶ Αρνητική Προγνωστική Αξία (NPV)
		↓ Ευαισθησία (sensitivity)	↓ Ειδικότητα (specificity)	Ακρίβεια (accuracy)

Πίνακας 5.2: Συγκεντρωτικός πίνακας συνάφειας με τα αντίστοιχα μέτρα

να προσδιορίσουμε μέσω της ακρίβειας και της ειδικότητας την ακρίβεια (accuracy) του μοντέλου από την σχέση:

$$accuracy = (sensitivity)\,(prevalence) + (specificity)\,(1 - prevalence)$$

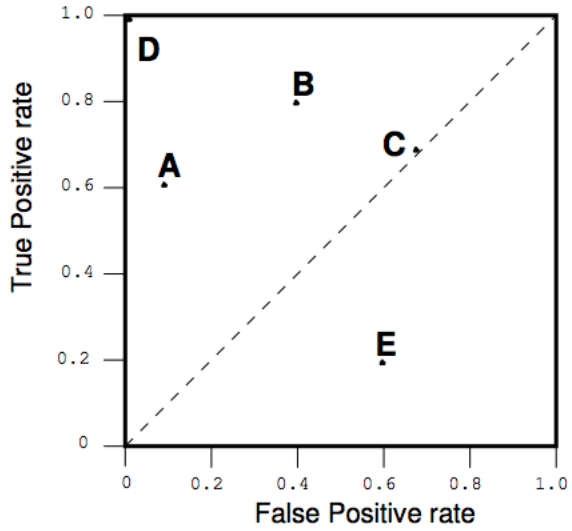
Εν κατακλείδι, όλα τα παραπάνω μέτρα που είδαμε παρουσιάζονται πιο συνοπτικά στον παρακάτω πίνακα:

5.4 Καμπύλες ROC

5.4.1 Εισαγωγή

Οι καμπύλες ROC (Receiver Operating Characteristics) ή καμπύλες λειτουργικών χαρακτηριστικών είναι μια ιδιαίτερα χρήσιμη μέθοδος για την εξασφάλιση της επιθυμητής προγνωστικής ακρίβειας σε ένα σχεδιασμό και την σύγκριση μοντέλων,αλγορίθμων και τεχνολογιών που παράγουν προβλέψεις. Αν και αρχικά χρησιμοποιήθηκαν σε πειράματα ανάλυσης δεδομένων για ιατρικούς σκοπούς (λόγου χάρη ανάλυση συμπεριφοράς διαγνωστικών συστημάτων), δεν άργησαν να χρησιμοποιηθούν σε προβλήματα μηχανικής εκμάθησης. Μάλιστα, οι πρώτες αναφορές για την υιοθέτηση των καμπυλών ROC από την κοινότητα της μηχανικής εκμάθησης έγιναν από τον Spackman (1989), ο οποίος επεσήμανε τα πλεονεκτήματα των ROC καμπυλών στην αξιολόγηση και την σύγκριση αλγορίθμων.

Η σχέση του ποσοστού των αληθώς θετικών (TP rate) και ψευδώς θετικών (FP rate) αποτελεσμάτων της διαγνωστικής διαδικασίας, καθώς μεταβάλλεται προοδευτικά προς μια κατεύθυνση το διαχωριστικό όριο αυτής, παριστάνεται γραφικά από μια καμπύλη ROC. Το εμβαδόν που ορίζεται κάτω από την καμπύλη αποτελεί ένα μέτρο διαχωρισμού του θορύβου - σήματος και συμβάλλει σημαντικά στην συμπερασματολογία για την απόδοση ενός μοντέλου - ταξινομητή.



Σχήμα 5.2: Ένα τυπικό γράφημα ROC όπου απεικονίζονται πέντε ταξινομητές

Όπως διαφαίνεται, οι καμπύλες ROC είναι εύκολες εννοιολογικά χωρίς αυτό να σημαίνει ωστόσο ότι δεν μπορούν να προκύψουν παρερμηνείες και παγίδες στην πράξη. Στη συνέχεια της παραγράφου θα παρουσιάσουμε αναλυτικά όλο το υπόβαθρο των καμπυλών ROC και θα παρουσιάσουμε συνοπτικά κάποια από τα θέματά τους καθώς και τις λύσεις τους.

5.4.2 Ερμηνεία ROC γραφημάτων

5.4.2.1 Χώρος ROC

Αν θεωρήσουμε πάλι προς χάριν ευκολίας ένα πρόβλημα πρόβλεψης δύο κατηγοριών (δυαδική ταξινόμηση), όπου οι παρατηρήσεις ταξινομούνται είτε ως θετικές (positive) είτε ως αρνητικές (negative). Όπως είδαμε και στην παραπάνω παράγραφο, για ένα δυαδικό ταξινομητή υπάρχουν τέσσερις πιθανές εκβάσεις οι οποίες παρουσιάστηκαν στους αντίστοιχους πίνακες συνάφειας.

Το γράφημα ROC είναι ένα δισδιάστατο γράφημα όπου το Αληθώς Θετικό Ποσοστό (TP rate) απεικονίζεται στον $Y-$ άξονα και το Ψευδώς Θετικό Ποσοστό (FP rate) στον $X-$ άξονα. Κάθε ταξινομητής παράγει ένα ζεύγος (FP rate, TP rate) τιμών, το οποίο αντιστοιχεί σε ένα σημείο στον χώρο μιας ROC καμπύλης.

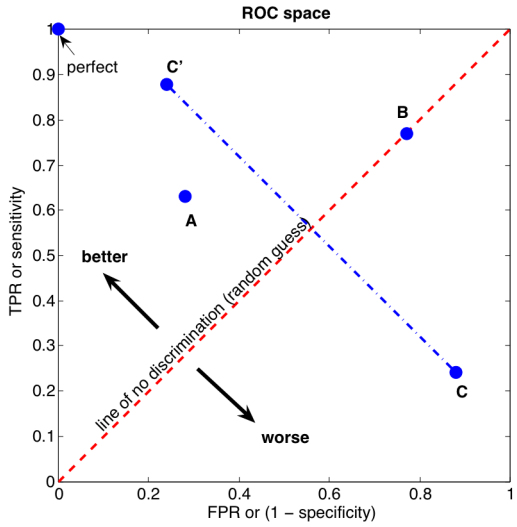
Από την Εικόνα 5.2 βλέπουμε ότι μέσω ενός ROC γραφήματος απεικονίζεται η σχετική μεταβολή μεταξύ του κέρδους (αληθώς θετικών παρατηρήσεων) και του κόστους (ψευδώς θετικών παρατηρήσεων). Ειδικότερα, μια καμπύλη ROC ορίζεται ως το μοναδιαίο τετράγωνο $[0,1] \times [0,1]$ που ξεκινά από το σημείο $[0,0]$ (όταν το σημείο απόφασης είναι μεγαλύτερο από όλες τις άλλες μετρήσεις θορύβου - σήματος) και καταλήγει στο σημείο $[1,1]$ (όταν το σημείο είναι μικρότερο από όλες τις άλλες μετρήσεις).

Το σημείο (0,0) εκφράζει την περίπτωση της μη θετικού ταξινομητή, όπου ναι μεν δεν υπάρχουν ψευδώς θετικές παρατηρήσεις αλλά δεν υπάρχουν και αντίστοιχα αληθώς θετικές. Αντιθέτως, η καλύτερη δυνατή πρόβλεψη δίνεται από το σημείο που βρίσκεται στην επάνω αριστερή γωνία ή αλλιώς στην συντεταγμένη (0,1) του χώρου ROC και αντιπροσωπεύει την 100% ευαισθησία (μηδέν ψευδώς αρνητικά) και την 100% ειδικότητα (μηδέν ψευδώς θετικά). Το σημείο (0,1) ονομάζεται αλλιώς και τέλεια ταξινόμηση (perfect classification). Μια εντελώς τυχαία διαδικασία θα δώσει ένα σημείο πάνω στην διαγώνιο $y = x$ ή αλλιώς γραμμή μη - διάκρισης, η οποία διαιρεί στα δύο τον χώρο ROC. Ένα σημείο στο χώρο ROC θα κινείται μπρος και πίσω στη διαγώνιο με βάση την συχνότητα με την οποία εικάζει την θετική τάξη. Ένα πολύ χαρακτηριστικό παράδειγμα τυχαίου ταξινομητή που έχουμε δει στις πιθανότητες είναι το κέρμα. Η πιθανότητα, ανεξαρτήτως του μεγέθους του δείγματος, να πάρουμε κορώνα (έστω θετική κατηγορία) είναι 50% και είναι ακριβώς ίση με την πιθανότητα να πάρουμε γράμματα (αρνητική κατηγορία). Ένας τέτοιος ταξινομητής αντιστοιχεί στο σημείο (0.5, 0.5). Εντελώς ανάλογα, αν ένα μοντέλο ταξινομεί τις παρατηρήσεις ως θετικές με 90% πιθανότητα, αυτό σημαίνει ότι το 90% των παρατηρήσεων που είναι όντως θετικές θα ταξινομηθεί σωστά αλλά ταυτόχρονα το ποσοστό των Ψευδώς Θετικών παρατηρήσεων θα αυξηθεί κατά 90%! Άρα, ο ταξινομητής αυτό θα αντιστοιχεί στο σημείο (0.9, 0.9).

Για να βρίσκεται ένα σημείο εκτός διαγωνίου (είτε στην πάνω τριγωνική περιοχή είτε στην κάτω), θα πρέπει κατά την διάρκεια της ταξινόμησης να ληφθούν υπ΄ όψιν κάποιες πληροφορίες που αφορούν τα δεδομένα. Θεωρούμε άτυπα ότι ένα σημείο ROC (που εκφράζει ένα ταξινομητή) είναι καλύτερο αν βρίσκεται στην πάνω δεξιά τριγωνική περιοχή του γραφήματος και όσο πιο κοντά γίνεται στο σημείο (0,1), δηλαδή θα πρέπει να έχει χαμηλή τιμή FP rate, υψηλή τιμή TP rate είτε και τις δύο τιμές πολύ χαμηλά.

Από την Εικόνα 5.3 και με βάση τα συμπεράσματα που έχουμε από την προηγούμενη ανάλυση, καταλαβαίνουμε ότι ο ταξινομητής A μας παρέχει μεγαλύτερη προβλεπτική δύναμη σε σύγκριση με τους B, C. Ο B βρίσκεται στην γραμμή μη - διάκρισης (άρα η ακρίβεια που θα δίνει σαν μοντέλο θα είνα 50%) ενώ ο C είναι θεωρητικά ο χειρότερος από τους τρεις ταξινομητές αφού έχει αρνητική προγνωστική δύναμη και βρίσκεται στην κάτω τριγωνική περιοχή. Ωστόσο, αν εκμεταλλευτούμε την συμμετρικότητα του χώρου, αντικατοπτρίζοντας το σημείο C, παίρνουμε το σημείο C', το οποίο δίνει ένα ταξινομητή σαφώς καλύτερο από τον A. Οι παρατηρήσεις του νέου ταξινομητή C' είναι ουσιαστικά οι προβλέψεις που παράγονται από τον πίνακα συνάφειας του ταξινομητή C αλλά αντεστραμμένες. Αυτό πρακτικά σημαίνει ότι όταν η μέθοδος C προβλέπει θετικά (p) ή αρνητικά (n), η νέα μέθοδος C' θα προβλέπει αρνητικά ή θετικά αντίστοιχα.

Εναλλακτικά λοιπόν, μπορούμε να πούμε ότι όσο πιο κοντά είναι ένα αποτέλεσμα του πίνακα συνάφειας στην επάνω αριστερή γωνία του χώρου ROC, τόσο καλύτερη είναι η πρόβλεψη. Γενικότερα όμως, η απόσταση από την ευθεία μη - διάκρισης προς οποιαδήποτε κατεύθυνση είναι ο πιο έγκυρος δείκτης για την προβλεπτική ικανότητα μιας μεθόδου. Κάθε ταξινομητής που εμφανίζεται στην κάτω τριγωνική περιοχή του γραφήματος θεωρείται χειρότερο μοντέλο και από ένα που βρίσκεται πάνω στην $y = x$. Ως εκ τούτου επιλέγεται αυτή η περιοχή να είναι άδεια στα ROC γραφήματα. Αυτό όμως δεν σήμαινει ότι οι ταξινομητές αυτοί



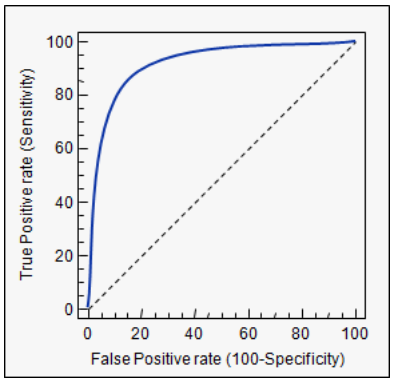
Σχήμα 5.3: Ο χώρος ROC με την αναπαράσταση τεσσάρων ταξινομητών.

μένουν τελείως αναξιοποίητοι διότι αντιτρέφοντας τις προβλέψεις τους, δίνουν ένα υπολογίσιμο προβλεπτικό μοντέλο.

5.4.2.2 Η περιοχή κάτω από την ROC καμπύλη (Area Under Curve - AUC)

Η καμπύλη ROC είναι μια δισδιάστατη απεικόνιση της απόδοσης ενός ταξινομητή. Για να συγκρίνουμε ταυτόχρονα πολλούς ταξινομητές, μπορούμε να περιορίσουμε την απόδοση ROC σε μια ενιαία βαθμωτή τιμή που να αντιπροσωπεύει την αναμενόμενη απόδοση. Μια τέτοια μέθοδος είναι ο υπολογισμός του εμβαδού της περιοχής κάτω από την καμπύλη ROC (Area Under Curve - AUC) (Hanley & McNeil, 1982 · Bradley, 1997). Εφόσον η AUC είναι μέρος του μοναδιαίου τετραγώνου στο οποίο ορίζεται μια καμπύλη ROC, η τιμή της θα είναι πάντα μεταξύ του 0 και του 1. Επιπλέον, η γραμμή μη - διάκρισης διαιρεί τον χώρο ROC σε δύο τριγωνικές περιοχές με εμβαδό 0.5 έκαστη. Συνεπώς, γνωρίζοντας ότι μια καμπύλη ROC ορίζεται πάντα πάνω από την διαγώνιο, κανένας ρεαλιστικός ταξινομητής δεν θα πρέπει να έχει AUC μικρότερη από 0.5.

Η AUC έχει μια ακόμα σημαντική στατιστική ιδιότητα: η τιμή της AUC για ένα ταξινομητή είναι ισοδύναμη με την πιθανότητα ο ταξινομητής να ταξινομήσει μια τυχαία θετική παρατήρηση υψηλότερα από μια τυχαία αρνητική παρατήρηση. Αυτή η ιδιότητα εκφράζεται και από το Mann - Whitney U test (Hanley & McNeil, 1982 · Mason & Graham, 2002) καθώς επίσης και με την δοκιμή Wilcoxon των βαθμίδων (Hanley & McNeil, 1982). Η AUC είναι στενά συνδεδεμένη και με τον δείκτη Gini (Breiman, Friedman, Olshen & Stone, 1984), ο οποίος είναι διπλάσιος από τον χώρο ανάμεσα στην διαγώνιο και την καμπύλη ROC. Ειδικότερα, οι Hand



Σχήμα 5.4: Τυπικό παράδειγμα καμπύλης ROC

& Till (2001) επισημαίνουν ότι:

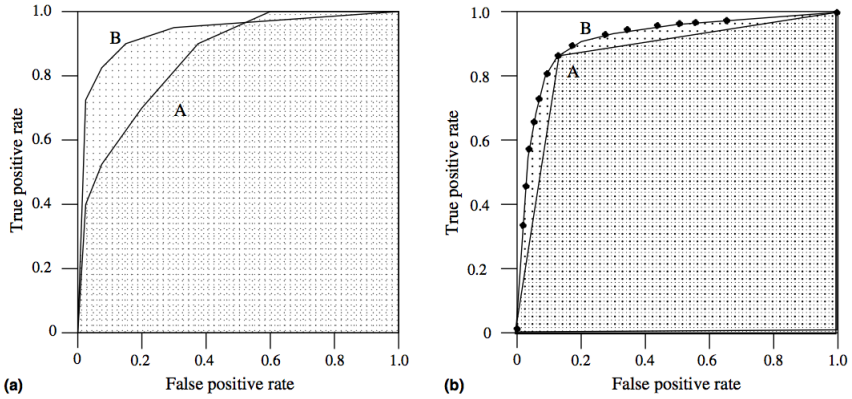
$$Gini + 1 = 2 \times AUC.$$

Ο υπολογισμός της AUC μπορεί να γίνει εύκολα με μια μικρή τροποποίηση στον αλγόριθμο κατασκευής μιας ROC καμπύλης. Ειδικότερα, ο αλγόριθμος αντί να συλλέγει σημεία στο χώρο ROC, προσθέτει διαδοχικά περιοχές τραπεζοειδών στην περιοχή. Τέλος, το άθροισμα αυτό διαιρείται από το συνολικό εφικτό εμβαδόν της περιοχής προκειμένου να κανονικοποιηθεί η τιμή της AUC ως προς το μοναδιαίο τετράγωνο.

Πριν κλείσουμε αυτή την παράγραφο, θα δούμε και ένα πιθανό θέμα που μπορεί να προκύψει κατά τον υπολογισμό της AUC για ένα ταξινομητή. Στο πρώτο σχήμα της Εικόνας 5.5, βλέπουμε τις AUC δύο ταξινομητών A και B. Ο ταξινομητής B έχει μεγαλύτερο εμβαδό συνεπώς θα έχει καλύτερη μέση απόδοση. Στο δεύτερο σχήμα, έχουμε την AUC για ένα δυαδικό ταξινομητή A και έναν αθροιστικό ταξινομητή B. Ο ταξινομητής A αντιπροσωπεύει την απόδοση του ταξινομητή B όταν αυτός δουλεύει με ένα συγκεκριμένο όριο. Αν και η απόδοση και των δύο είναι ίδια για αυτό το συγκεκριμένο όριο, η απόδοση του B γίνεται χειρότερη του A από το σημείο αυτό και παραπέρα. Άρα βλέπουμε ότι γενικά δύναται ένας ταξινομητής υψηλής απόδοσης (δηλαδή υψηλής AUC) να έχει χειρότερες επιδόσεις σε μια περιοχή του χώρου ROC από έναν άλλο ταξινομητή χαμηλότερης απόδοσης.

5.4.3 Γραφήματα ROC σε προβλήματα με πολλαπλές κατηγορίες

Η χρήση των καμπυλών ROC καθιερώθηκε κυρίως μέσω της ιατρικής, όπου εξετάζονται ως επί το πλείστον προβλήματα με δύο καταστάσεις. Συνεπώς, η γενίκευση της χρήσης τους σε προβλήματα με δύο κατηγορίες εκτός ιατρικής ήταν το πιο εύκολο κομμάτι και για αυτό το λόγο επιλέξαμε να μιλήσουμε αρχικά για αυτό. Ωστόσο, η κατάσταση περιπλέκεται αρκετά όταν έρχεται η σειρά των προβλημάτων πολλαπλών κατηγοριών. Για παράδειγμα, εάν η μεταβλητή απόκρισης



Σχήμα 5.5: Δύο γραφήματα ROC. Το πρώτο γράφημα δείχνει την περιοχή κάτω από δύο καμπύλες ROC. Το δεύτερο γράφημα δείχνει την περιοχή κάτω από τις καμπύλες του διακριτού ταξινομητή A και του πιθανού ταξινομητή B

έχει n κατηγορίες, τότε ο πίνακας συσχέτισης είναι διάστασης $n \times n$, όπου τα διαγώνια στοιχεία του εκφράζουν τις n σωστές ταξινομήσεις και τα μη διαγώνια τις $n^2 - n$ τα πιθανά σφάλματα. Με βάση την προηγούμενη θεωρία, καταλαβαίνουμε ότι η αναγωγή του πίνακα αυτού στο χώρο θα οδηγούσε στην κατασκευή ενός πολύτοπου, το οποίο σε καμία περίπτωση δεν θα μπορούσαμε να διαχειριστούμε.

Η πιο κοινή μέθοδος που υπάρχει για την αντιμετώπιση του προβλήματος για τις n κατηγορίες είναι η κατασκευή n ROC γραφημάτων. Πιο συγκεκριμένα, εάν η μεταβλητή απόκρισης έχει nτο πλήθος κατηγορίες τότε το i γράφημα ROC απεικονίζει την απόδοση του ταξινομητή θέτοντας την κατηγορία n_i ως την θετική κατηγορία και όλες τις υπόλοιπες ως αρνητικές. Αυτή η διαδικασία γίνεται διαδοχικά για όλες τις κατηγορίες της μεταβλητής απόκρισης. Παρά το γεγονός ότι υπάρχουν αντιρρήσεις για την ευαισθησία της μεθόδου, έχει αποδειχθεί ότι δουλεύει καλά στην πράξη και προσφέρει επαρκή ελαστικότητα στην εκτίμηση.

5.4.3.1 Η περιοχή κάτω από την ROC καμπύλη (AUC) για προβλήματα πολλαπλών κατηγοριών.

Όπως είδαμε για τα προβλήματα δύο διαστάσεων, το εμβαδό της περιοχής κάτω από την καμπύλη είναι μια και μοναδική τιμή. Τι γίνεται όμως για τα προβλήματα με περισσότερες από δύο διαστάσεις, όπου πλέον δεν υπάρχει μια αλλά nδιαφορετικές τιμές για την AUC; Για το πρόβλημα αυτό υπάρχουν δύο προσεγγίσεις.

Η πρώτη προσέγγιση και προτάθηκε από τους Hand & Hill (2001), οι οποίοι επιθυμούσαν ένα μέτρο για την AUC πολλαπών κατηγοριών που να μην επηρεάζεται από την αλλαγή της κατανομής των κατηγοριών της μεταβλητής απόκρισης και από τα κόστη σφαλμάτων. Η προσέγγιση αυτή είναι πιθανοθεωρητική καθώς βασίζεται στο γεγονός ότι μέσω της AUC εκφράζεται η πιθανότητα μια τυχαία επιλεγμένη θετική παρατήρηση να ταξινομηθεί υψηλότερα από μια άλλη τυχαία ε-

πιλεγμένη αρνητική παρατήρηση. Το μέτρο διαχωρισμού των κατηγοριών με βάση την παραπάνω λογική είναι:

$$AUC_{total} = \frac{2}{|n|\left(|n|-1\right)} \sum_{\{n_i,n_j\}\in n} AUC\left(n_i, n_j\right).$$

Η άθροιση γίνεται πάνω σε όλα τα πιθανά ζεύγη κατηγοριών με τυχαία σειρά. Εφόσον όλα τα πιθανά ζεύγη είναι $|n|\left(|n|-1\right)/2$ το πλήθος, η πολυπλοκότητα της μεθόδου είναι $O\left(|n|^2 n \log n\right)$. Τέλος, αν και αυτό το μέτρο ικανοποιεί τους στόχους των δημιουργών του, δεν μπορούμε να απεικονίσουμε εύκολα την επιφάνεια της οποίας το εμβαδό υπολογίζουμε.

Μια δεύτερη πρόταση για τον υπολογισμό της AUC για πολλαπλές κατηγορίες είναι αυτή των Provost & Domingos (2001). Σύμφωνα με αυτή την πρόταση, υπολογίζονται πάλι οι nδιαφορετικές AUC για κάθε κατηγορία, οι οποίες εν συνεχεία πολλαπλασιάζονται με την κατηγορία αναφοράς που «επικρατεί» σε κάθε διάγραμμα. Η συνολική AUC για το πρόβλημα ταξινόμησης πολλαπλών κατηγοριών είναι:

$$AUC_{total} = \sum_{n_i\in n} AUC(n_i)p\left(n_i\right).$$

Εξ ΄ ορισμού, η μέθοδος αυτή απαιτεί $|n|$ το πλήθος υπολογισμού, άρα η πολυπλοκότητά της είναι $O\left(|n|n\log n\right)$. Το πλεονέκτημα αυτής της μεθόδου είναι ότι παράγεται άμεσα από τις επιμέρους AUC που μπορούν να παρουσιαστούν πολύ εύκολα γραφικά. Από την άλλη, οι AUC αυτές είναι ευαίσθητες ως προς τα κόστη εσφαλμένης ταξινόμησης καθώς και τις αλλαγές στην κατανομή των κατηγοριών της μεταβλητής απόκρισης και άρα κατ ΄ επέκταση και η AUC_{total}.

5.5 Διασταυρωμένη Επικύρωση (Cross - Validation)

Έχοντας μιλήσει πιο επισταμένα πλέον για σφάλματα, δεν θα μπορούσαμε να μην αναφερθούμε ξανά σε μια ιδιαίτερα διαδεδομένη μέθοδο αξιολόγησης μοντέλου όπως η διασταυρωμένη επικύρωση. Η διασταυρωμένη επικύρωση, για την οποία έχουμε μιλήσει και σε προηγούμενα κεφάλαια, είναι μια μέθοδος εκτίμησης του σφάλματος πρόβλεψης (prediction error). Πιο συγκεκριμένα, με την μέθοδο αυτή μπορεί να υπολογιστεί άμεσα το αναμενόμενο γενικευμένο σφάλμα $Err = E\left[L\left(Y, \hat{f}\left(X\right)\right)\right]$, όταν το μοντέλο $\hat{f}\left(X\right)$ εφαρμόζεται σε ένα τυχαίο δείγμα δοκιμής από την κατανομή των X και Y.

Στην ιδανική περίπτωση όπου ο αριθμός των δεδομένων είναι επαρκής, θα μπορούσαμε να χρησιμοποιήσουμε το σύνολο επικύρωσης για την εκτίμηση της απόδοσης του μοντέλου, αλλά ένα τέτοιο ενδεχόμενο είναι εξαιρετικά σπάνιο. Την λύση σε αυτό το πρόβλημα έρχεται να δώσει η $k-$φορές διασταυρωμένη επικύρωση ($k-$fold cross - validation). Αυτό που κάνει επί της ουσίας η μέθοδος αυτή είναι να χωρίσει τυχαία το σύνολο των δεδομένων σε k τμήματα, εκ των οποίων τα $k-1$ σύνολα θα χρησιμοποιηθούν για την εκπαίδευση του μοντέλου και το

$k-$ οστό σύνολο για τον υπολογισμό του σφάλματος πρόβλεψης. Η διαδικασία αυτή επαναλαμβάνεται k φορές ούτως ώστε κάθε σύνολο να έχει χρησιμοποιηθεί ως σύνολο επικύρωσης.

Επομένως, το ερώτημα που φυσιολογικά γεννάται είναι το εξής: υπάρχει κάποια τιμή για το k για την οποία να παίρνουμε βέλτιστο αποτέλεσμα; Προφανώς, η απάντηση είναι όχι, διότι κάθε σύνολο δεδομένων που μελετάμε έχει ξεχωριστές ιδιότητες και άρα με βάση αυτές πρέπει να προσδιορίσει ο εκάστοτε αναλυτής την βέλτιστη τιμή για το k.

Κεφάλαιο 6

Εφαρμογή σε πραγματικά δεδομένα

Στο τελευταίο κεφάλαιο αυτής της εργασίας, θα συγκρίνουμε όλες τις μεθόδους ταξινόμησης από το πεδίο της μηχανικής εκμάθησης που παρουσιάστηκαν στα προηγούμενα κεφάλαια και στη συνέχεια θα παρουσιάσουμε τα αποτελέσματα της διαδικασίας μοντελοποίησης. Λόγω του πολύ μεγάλου όγκου του συνόλου εξέτασης (15.000 μεταβλητές), δεν κατέστη εφικτή η επεξεργασία του με τα τυπκά προγράμματα στατιστικής ανάλυσης και κατ΄ επέκταση η χρήση του για επαλήθευση του παραγόμενου μοντέλου από το σύνολο εκπαίδευσης. Για την εκτίμηση της απόδοσης των μοντέλων ταξινόμησης που κατασκευάστηκαν από τις διάφορες μεθόδους χρησιμοποιήθηκαν τεχνικές αξιολόγησης που συμπεριλαμβάνονται στους αλγορίθμους κατασκευής των ταξινομητών, καθώς επίσης και καμπύλες ROC. Η γενικευμένη απόδοση ενός ταξινομητή συνήθως εκτιμάται με holdout επικύρωση.

6.1 Εισαγωγή στα πακέτα SPSS 22 και R

SPSS 22

Το στατιστικό πακέτο SPSS 22 προσφέρει μια πολύ μεγάλη ποικιλία μεθόδων μοντελοποίησης που λαμβάνονται από την μηχανική εκμάθηση, την τεχνητή νοημοσύνη αλλά και την κλασσική στατιστική. Οι μέθοδοι που διατίθενται στην παλέτα του μας επιτρέπουν να αντλήσουμε νέες πληροφορίες από τα δεδομένα μας και να αναπτύξουμε μοντέλα πρόβλεψης. Κάθε μέθοδος έχει ορισμένες δυναμικές και μπορεί να χρησιμοποιηθεί σε συγκεκριμένα είδη προβλημάτων. Ως μια εφαρμογή εξόρυξης δεδομένων, το SPSS 22 προσφέρει μια στρατηγική προσέγγιση στην εύρεση χρήσιμων σχέσεων σε σύνολα δεδομένων μεγάλων διαστάσεων.

R

Η R είναι ταυτόχρονα μια προγραμματιστική γλώσσα ελεύθερου λογισμικού και ένα προγραμματιστικό περιβάλλον που ενδείκνυται για προβλήματα στατιστικής και κατασκευή γραφημάτων. Τα τελευταία χρόνια, έρευνες στην κοινότητα του data mining έχουν δείξει ότι η δημοτικότητα της R έχει εκτοξευτεί όσον αφορά την χρήση της σε προβλήματα δεδομένων μεγάλων διαστάσεων. Όπως και το SPSS 22, έτσι και η R διαθέτει το ίδιο μεγάλο εύρος μεθόδων για την εξόρυξη δεδομένων με το πλεονέκτημα όμως ότι η R είναι πολύ πιο ευέλικτη καθώς επιτρέπει στους χρήστες να τροποποιήσουν τους υπάρχοντες κώδικες ή ακόμα και να προσθέσουν δικούς τους προκειμένου να προσαρμόσουν τις μεθόδους καλύτερα στα εκάστοτε δεδομένα τους.

6.2 Περιγραφή του προβλήματος

Η διαχείριση πελατειακών σχέσεων (Customer Relationship Management - CRM) είναι ένα από τα σημαντικότερα συστατικά των στρατηγικών που εφαρμόζονται στο σύγχρονο μάρκετινγκ. Ο πιο πρακτικός τρόπος να παράγουμε γνώση από μια βάση δεδομένων για CRM είναι να κατασκευάσουμε ένα μοντέλο αποτελεσμάτων (scores). Τα αποτελέσματα υπολογίζονται με την χρήση των επεξηγηματικών μεταβλητών μέσω των παρατηρήσεων και στη συνέχεια εφαρμόζονται στο αντίστοιχο σύστημα πληροφορίας (information system - IS) προκειμένου να διαμορφωθεί μια πελατειακή σχέση.

Μια τέτοια βιομηχανική πλατφόρμα πελατειακής ανάλυσης ανέπτυξε και η γαλλική εταιρεία τηλεπικοινωνιών Orange, την οποία διέθεσε προς αξιοποίηση στα πλαίσια του Παγκόσμιου Διαγωνισμού για Εξόρυξη Δεδομένων KDD Cup την χρονιά 2009. Στόχος του προβλήματος που τέθηκε στο διαγωνισμό είναι η πρόβλεψη της ροπής των πελατών για αλλαγή παροχέα κινητής τηλεφωνίας (churn), για αγορά νέων υπηρεσιών ή προϊόντων (appetency) ή να αγοράσουν αναβαθμίσεις των προϊόντων που ήδη έχουν, αυξάνοντας έτσι τον τζίρο της εταιρείας (up-selling). Οι έννοιες που εκφράζονται από τις μεταβλητές απόκρισης του προβλήματος είναι ιδιαίτερα διαδεδομένες στον χώρο των επιχειρήσεων. Για αυτό το λόγο, κρίνουμε λοιπόν πως θα ήταν χρήσιμο για τον αναγνώστη να τις παρουσιάσουμε πιο αναλυτικά:

- Appetency: η πιθανότητα ένας πελάτης να προβεί σε αγορά νέων προϊόντων ή υπηρεσιών,
- Churn: πολύ συχνά συναντάται και ως ρυθμός τριβής (attrition rate) και υπό μια ευρύτερη έννοια, εκφράζει τον αριθμό των ατόμων ή αντικειμένων που προσθαφαιρούνται σε ένα σύνολο σε ένα συγκεκριμένο χρονικό διάστημα. Ειδικότερα, θεωρείται κομβικής σημασίας για επιχειρήσεις που διαθέτουν πελατειακές βάσεις δεδομένων καθώς τους επιτρέπει να γνωρίζουν την πρόθεση των πελατών να παραμείνουν ή να φύγουν από αυτές.
- Up - selling: είναι μια από τις πιο γνωστές τεχνικές πωλήσεων, στην οποία ο πωλητής προσπαθεί να πείσει τον πελάτη να αγοράσει ακριβότερα προϊόντα

ή άλλες πρόσθετες υπηρεσίες, τις οποίες ο πελάτες αγνοεί, με στόχο να επιτευχθεί μια πιο επικερδής πώληση.

Συνεπώς για το πρόβλημα μας έχουμε τρεις κατηγορικές μεταβλητές απόκρισης, οι οποίες κωδικοποιούνται σε (+1) (εκφράζει θετικό αποτέλεσμα) - (-1) (αρνητικό αποτέλεσμα). Τα δεδομένα εκπαίδευσης είναι ένα σύνολο δεδομένων υψηλών διαστάσεων 100.000 παρατηρήσεων που αποτελείται από 230 επεξηγηματικές μεταβλητές (40 κατηγορικές και 190 συνεχείς). Για τις μεταβλητές μας δεν έχουμε καμία πληροφορία για το τι εκφράζουν καθώς το περιεχόμενό τους έχει κρυπτογραφηθεί για λόγους προστασίας των δεδομένων των πελατών.

Επειδή το σύνολο εξέτασης που διατίθεται για το πρόβλημα δεν μπορούμε να το διαχειριστούμε με τα πακέτα που διαθέτουμε, επιλέξαμε αντ΄ αυτού την διάσπαση του συνόλου εκπαίδευσης σε δύο υποσύνολα: το 70% του αρχικού συνόλου χρησιμοποιήθηκε ως σύνολο εκπαίδευσης και το 30% ως σύνολο εξέτασης. Αυτή η τεχνική εφαρμόστηκε στη μέθοδο των νευρωνικών δικτύων όπου δεν διατίθενταν μέτρα αξιολόγησης του παραγόμενου μοντέλου.

Πριν προχωρήσουμε στην ανάλυση των δεδομένων μας με την χρήση των ταξινομητών που παραθέσαμε προηγουμένως, χρειάστηκε να περάσουμε το σύνολο εκπαίδευσης από μια διαδικασία καθαρισμού. Ο λόγος που ακολουθήσαμε αυτή την πρακτική είναι λόγω του πολύ μεγάλου αριθμού των ελλιπών τιμών καθώς επίσης και της μείωσης του πλήθους των μεταβλητών. Οι δύο αυτοί παράγοντες προκαλούν σοβαρά προβλήματα στην των αλγοριθμικών μεθόδων καθώς λόγω του μεγάλου θορύβου που περικλείουν τα δεδομένα διαστρεβλώνεται το μοντέλο και επιπλέον αυξάνεται κατά πολύ ο χρόνος περάτωσης της διαδικασίας. Τα βήματα που ακολουθήσαμε είναι τα εξής:

1. Αρχικά εισάγουμε τα δεδομένα στο πρόγραμμα SPSS 22 μέσω ενός αρχείου xls (excel).

2. Επιλέγουμε την διαδρομή: *Analyze* $\longrightarrow$ *Multiple Imputation* $\longrightarrow$ *Impute Missing Data Values*. Ο λόγος που κάναμε αυτή την επιλογή είναι για να γίνει ένας πρώτος «καθαρισμός» των δεδομένων από τις ελλιπείς τιμές. Στην καρτέλα που ανοίγει κάνουμε τις εξής επιλογές:

   (α΄) Στο πεδίο *Variables In Model* εισάγουμε όλες τι επεξηγηματικές μεταβλητές.

   (β΄) Στο πεδίο *Imputations* επιλέγουμε τον αριθμό των επαναλήψεων που θέλουμε να έχουμε. Για το σύνολο δεδομένων μας επιλέξαμε πέντε επαναλήψεις.

   (γ΄) Στη συνέχεια επιλέγουμε σε τι μορφή θέλουμε να αποθηκευτεί το νέο σύνολο δεδομένων μας. Οι υπόλοιπες καρτέλες μένουν στις προεπιλογές του SPSS καθώς δεν συνεισφέρουν ιδιαίτερα στην ανάλυσή μας.

3. Επιλέγουμε την διαδρομή: $Transform \longrightarrow Prepare\ Data\ For\ Modelling \longrightarrow$ *Interactive*. Στο μενού που ανοίγει κάνουμε τις παρακάτω επιλογές:

(α′) Στην καρτέλα *Objective* επιλέγουμε *Optimize For Accuracy* προκειμένου η διαλογή των μεταβλητών να γίνει με έμφαση στην ακρίβεια του μοντέλου που θα προκύψει.

(β′) Στην καρτέλα *Fields* εισάγουμε τις μεταβλητές που θέλουμε να «καθαριστούν» (δηλαδή τις επεξηγηματικές μεταβλητές) και επιλέγουμε *Use Predefined Roles* για να καθορίσουμε το είδος των μεταβλητών (επεξηγηματικές ή απόκρισης)

(γ′) Στην καρτέλα *Settings*, θα κάνουμε τις εξής επιλογές στο υπό - μενού που παρατίθεται:

i. στην κατηγορία *Exclude Fields* θα επιλέξουμε να εξαιρεθούν οι μεταβλητές «χαμηλής ποιότητας», και συγκεκριμένα οι μεταβλητές που ποσοστό ελλιπων τιμών μεγαλύτερο του 50%.

ii. στην κατηγορία *Select And Construct* θα επιλέξουμε να γίνει επιλογή χαρακτηριστικών του συνόλου για να επιλεχθούν οι πιο σημαντικές μεταβλητές.

Οι υπόλοιπες επιλογές που περιέχονται στην παραπάνω διαδικασία παρέμειναν στις προεπιλογές του ίδιου του προγράμματος. Η διαδικασία αυτή εκτελέστηκε δύο φορές για να δούμε αν θα προκύψει διαφορετικό σύνολο δεδομένων από το αν επιλέγαμε ή όχι την επιλογή χαρακτηριστικών του μοντέλου. Εκ του αποτελέσματος, είδαμε ότι και για τις δύο επιλογές, το σύνολο που προκύπτει είναι ακριβώς ίδιο και αποτελείται πλέον από 51 επεξηγηματικές μεταβλητές.

Έχοντας πραγματοποιήσει τον καθαρισμό των δεδομένων πλέον έχουμε το τελικό σύνολο στο οποίο θα εφαρμόσουμε τις μεθόδους ταξινόμησης που παρουσιάσαμε στα προηγούμενα κεφάλαια. Πιο συγκεκριμένα, θα χρησιμοποιήσουμε το πακέτο SPSS 22 για την εκτέλεση των Δέντρων Αποφάσεων και των Τεχνητών Νευρωνικών Δικτύων, ενώ για τις Μηχανές Διανυσματικής Υποστήριξης το πακέτο της R.

6.3 Μέτρα αξιολόγησης

Στα προηγούμενα κεφάλαια αναφερθήκαμε εκτενέστερα στα μέτρα αξιολόγησης που περιλαμβάνονται εντός και εκτός των αλγορίθμων των μεθόδων. Ωστόσο, ειδικά σε κλειστά πακέτα λογισμικού ,δεν έχουμε πάντα στα χέρια μας όλο το εύρος των πιθανών αξιολογητών που μπορούν να εφαρμοστούν σε ένα αλγόριθμο εκμάθησης. Για αυτό το λόγο, για την αξιολόγηση των μοντέλων που θα προκύψουν για το πρόβλημα με το οποίο θα ασχοληθούμε, θα χρησιμοποιηθούν τα παρακάτω μέτρα αξιολόγησης:

Προφανώς κάθε μέθοδος μας παρέχει και το αντίστοιχο μέτρο ακρίβειας (accuracy) για τα αποτελέσματά της.

Μέθοδος ταξινόμησης	Μέτρα αξιολόγησης
Δέντρα Αποφάσεων	$k-$διασταυρωμένη επικύρωση
Τεχνητά Νευρωνικά Δίκτυα	ROC καμπύλες
Μηχανές Διανυσμάτων Υποστήριξης	$k-$διασταυρωμένη επικύρωση

Πίνακας 6.1: Πίνακας μεθόδων ταξινόμησης - μέτρων αξιολόγησης

6.4 Δέντρα Αποφάσεων

6.4.1 CHAID

Ο CHAID δημιουργεί δέντρα απόφασης προσδιορίζοντας την βέλτιστη διάσπαση βάσει των $X-$τετράγωνο στατιστικών. Σε αντίθεση με τις μεθόδους εκμάθησης CRT και QUEST, ο CHAID μπορεί να δημιουργήσει και μη δυαδικά δέντρα αποφάσεων, το οποίο πρακτικά σημαίνει ότι σε κάποιες διασπάσεις έχουμε παραπάνω από δύο κλάδους. Επιπλέον, μπορεί να δεχθεί ως ορίσματα στα πεδία στόχου και πρόβλεψης εξίσου κατηγορικές και συνεχείς μεταβλητές.

Τα βήματα που ακολουθήσαμε για την εκτίμηση του μοντέλου με τον αλγόριθμο CHAID και για τις τρεις μεταβλητές απόκρισης περιγράφονται παρακάτω:

$$Analyze \longrightarrow Classify \longrightarrow Tree$$

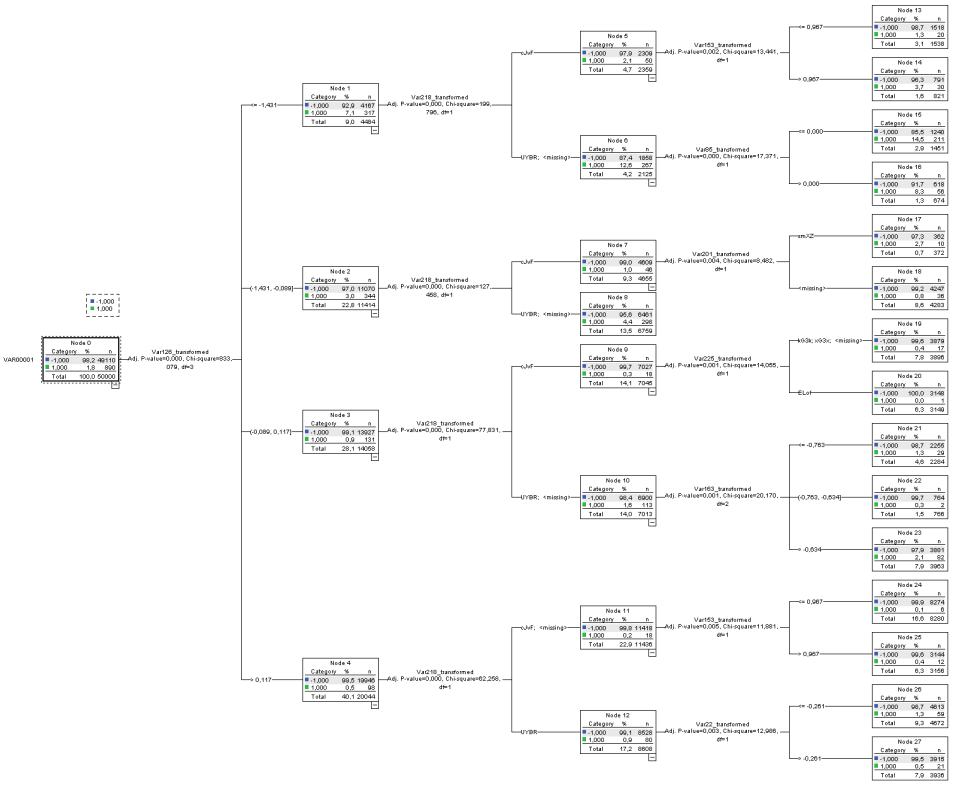
Παράμετροι Μοντέλου:

- *Growing Method*: CHAID
- *Output*: σε αυτή την επιλογή ορίζουμε την μορφή που θέλουμε να έχουν τα αποτελέσματα της ανάλυσης. Ειδικότερα:
  - *Tree*: σε αυτή την καρτέλα ορίζουμε τον προσανατολισμό του δέντρου, το περιεχόμενο των κόμβων του,
  - *Statistics*: επιλέγουμε ποια στατιστικά θέλουμε να μας εμφανιστούν για την ανάλυση του μοντέλου
- *Validation*: εδώ ορίζουμε την μέθοδο επικύρωσης του μοντέλου. Για το πρόβλημα μας επιλέξαμε την $10-fold$ διασταυρωμένη επικύρωση.
- *Criteria*: στην καρτέλα αυτή επιλέγουμε τα επίπεδα σημαντικότητας για τον διαχωρισμό και την συγχώνευση των μεταβλητών κατά την κατασκευή του δέντρου. Αναλυτικότερα:
  - *CHAID*: στην καρτέλα αυτή επιλέγουμε ως στατιστικό ελέγχου για τον διαχωρισμό των μεταβλητών σε κάθε κόμβο τον λόγο πιθανοφάνειας ενώ σαν επίπεδο σημαντικότητας για τον διαχωρισμό και την συγχώνευση στους κόμβους την τιμή 0.01.
- *Options*: εδώ μπορούμε να διευθετήσουμε κάποια επιπλέον κατασκευαστικά ζητήματα του δέντρου όπως η διαχείριση των ελλιπών τιμών και τα κόστη εσφαλμένης ταξινόμησης.

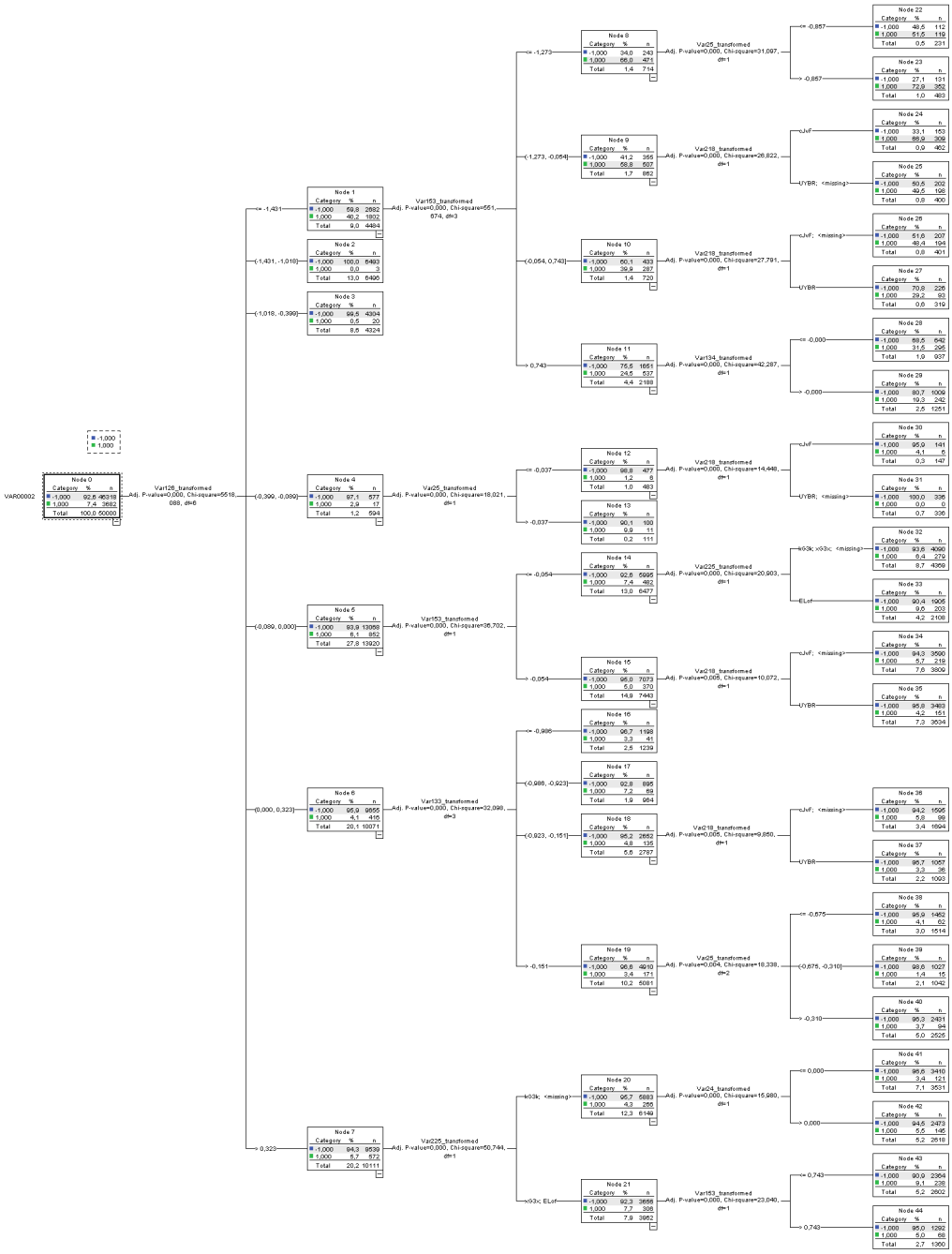
– *Missing Values*: επιλέγουμε ο αλγόριθμος να διαχειριστεί τις ελλιπείς τιμές ως ελλιπείς.

– *Misclassification costs*: επιλέγουμε να είναι το ίδιο για όλες τις κατηγορίες μιας μεταβλητής.

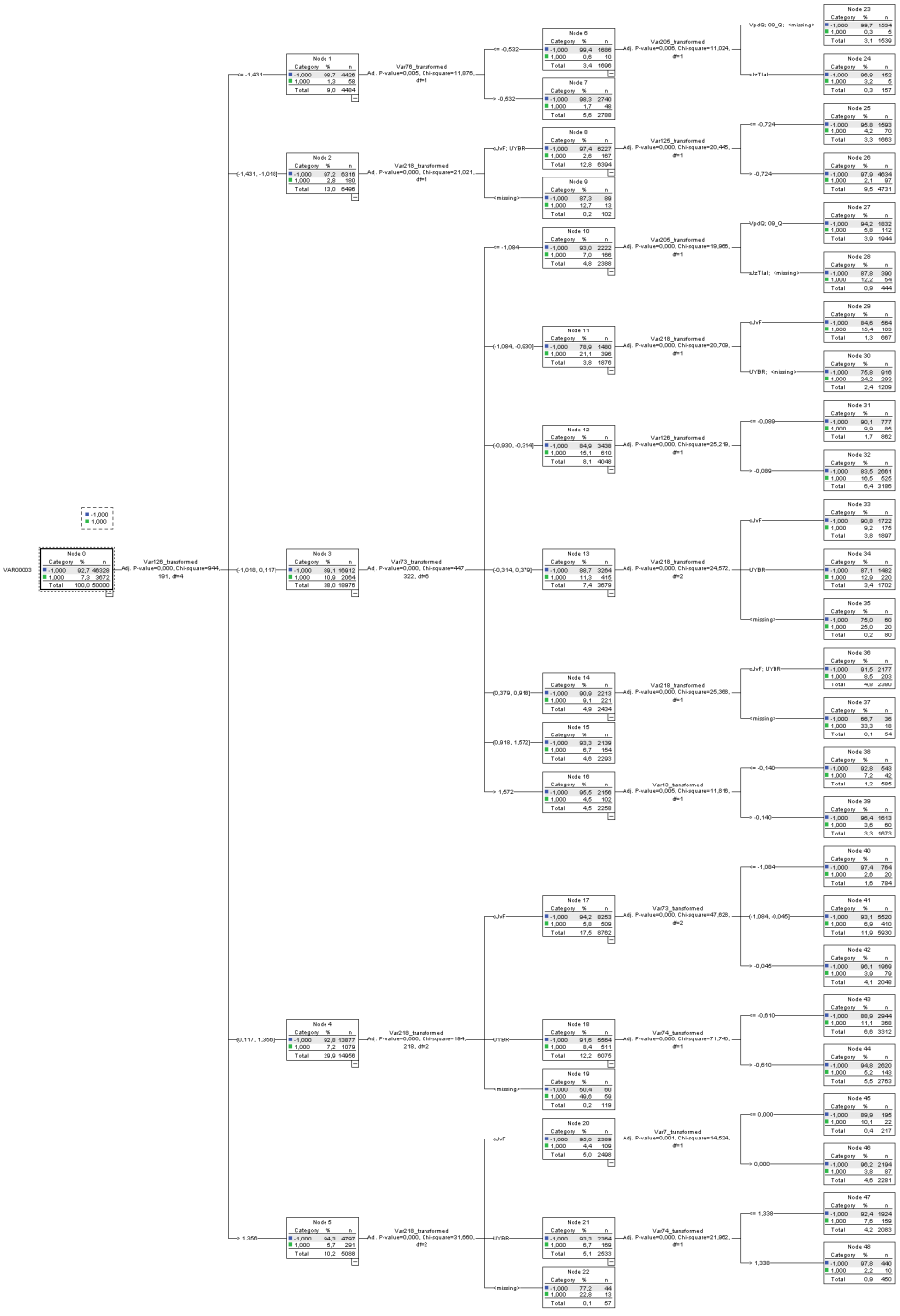
Τα αποτελέσματα που προέκυψαν από την ανάλυση του μοντέλου είναι τα παρακάτω:



Σχήμα 6.1: Το δέντρο που προέκυψε εφαρμόζοντας τον CHAID για την μεταβλητή απόκρισης appetency



Σχήμα 6.2: Δέντρο που προέκυψε εφαρμόζοντας τον CHAID στο πρόβλημα ταξινόμησης για την μεταβλητή churn



Σχήμα 6.3: Δέντρο που προέκυψε εφαρμόζοντας τον CHAID στο πρόβλημα ταξινόμησης για την μεταβλητή απόκρισης Up - selling

Στο τέλος της παραγράφου θα παρουσιαστούν τα μέτρα αξιολόγησης για την μέθοδο για να συγκριθούν με αυτά των υπολοίπων μεθόδων.

6.4.2 Exhaustive CHAID

Η εξαντλητική CHAID αποτελεί μια τροποποίηση της μεθόδου CHAID που εκτελεί μια πιο εμπεριστατωμένη ανάλυση εξετάζοντας όλες τις πιθανές διασπάσεις. Αυτό έχει σαν επακόλουθο την εκτέλεση περισσότερων υπολογισμών και άρα να είναι πιο αργή σε σύγκριση με την απλή CHAID. Και αυτή η μέθοδος μας δίνει μη δυαδικά δέντρα ταξινόμησης ενώ μπορεί να δεχθεί σε όλα της τα ορίσματα και κατηγορικές και συνεχείς μεταβλητές.

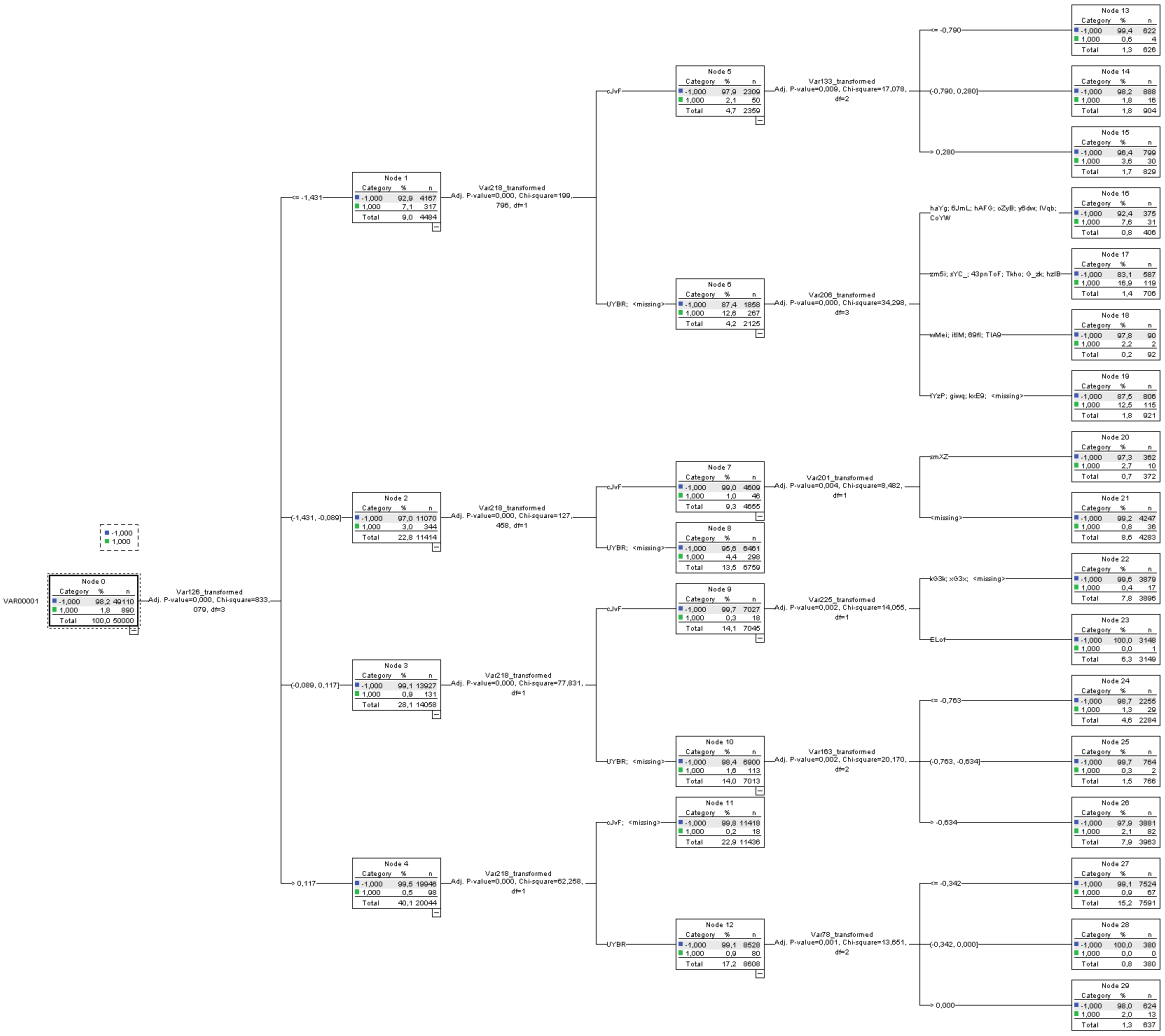
Για την εκτέλεση της μεθόδου, ακολουθήσαμε την παρακάτω διαδρομή:

$$Analyze \longrightarrow Classify \longrightarrow Tree$$

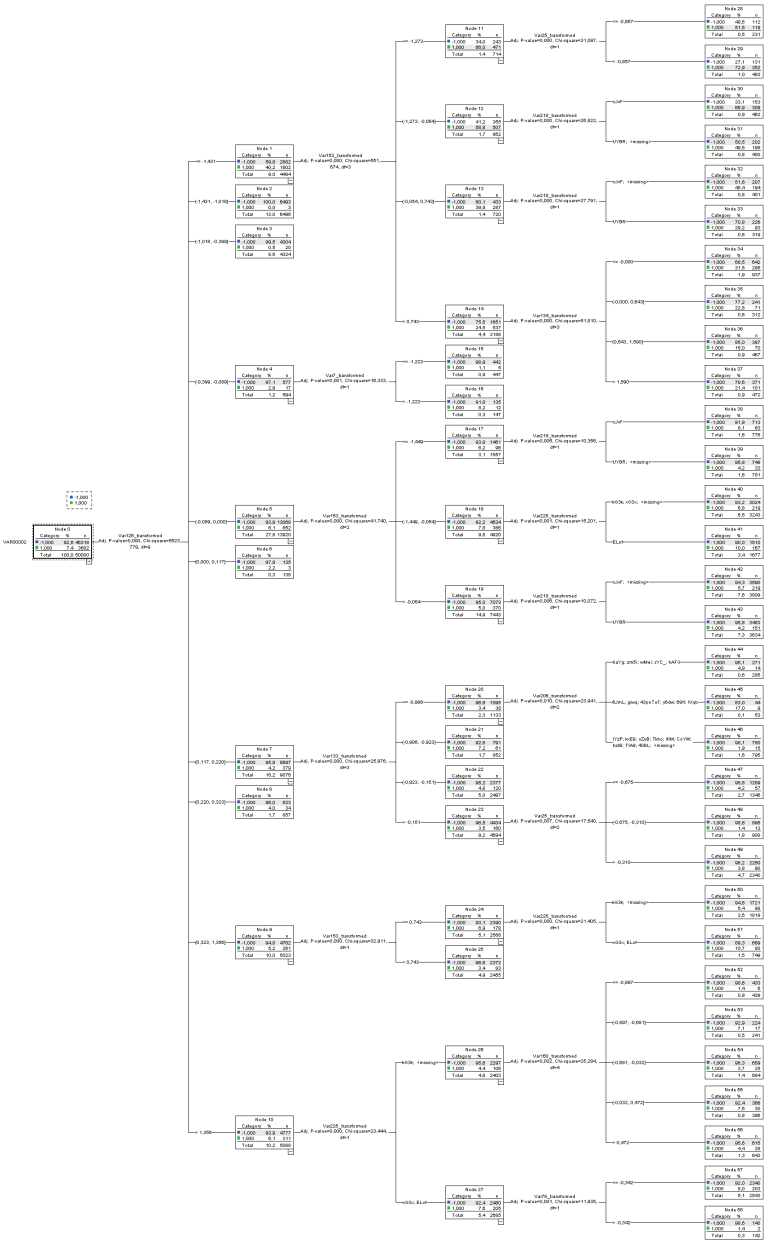
Παράμετροι Μοντέλου:

- *Growing Method*: Exhaustive CHAID
- *Output*: σε αυτή την επιλογή ορίζουμε την μορφή που θέλουμε να έχουν τα αποτελέσματα της ανάλυσης. Ειδικότερα:
  - *Tree*: σε αυτή την καρτέλα ορίζουμε τον προσανατολισμό του δέντρου, το περιεχόμενο των κόμβων του,
  - *Statistics*: επιλέγουμε ποια στατιστικά θέλουμε να μας εμφανιστούν για την ανάλυση του μοντέλου
- *Validation*: εδώ ορίζουμε την μέθοδο επικύρωσης του μοντέλου. Για το πρόβλημα μας επιλέξαμε την $10 - fold$ διασταυρωμένη επικύρωση.
- *Criteria*: στην καρτέλα αυτή επιλέγουμε τα επίπεδα σημαντικότητας για τον διαχωρισμό και την συγχώνευση των μεταβλητών κατά την κατασκευή του δέντρου. Αναλυτικότερα:
  - *Exhaustive CHAID*: στην καρτέλα αυτή επιλέγουμε ως στατιστικό ελέγχου για τον διαχωρισμό των μεταβλητών σε κάθε κόμβο τον λόγο πιθανοφάνειας ενώ σαν επίπεδο σημαντικότητας για των διαχωρισμό και την συγχώνευση στους κόμβους την τιμή 0.01.
- *Options*: εδώ μπορούμε να διευθετήσουμε κάποια επιπλέον κατασκευαστικά ζητήματα του δέντρου όπως η διαχείριση των ελλιπών τιμών και τα κόστη εσφαλμένης ταξινόμησης.
  - *Missing Values*: επιλέγουμε ο αλγόριθμος να διαχειριστεί τις ελλιπείς τιμές ως ελλιπείς.
  - *Misclassification costs*: επιλέγουμε να είναι το ίδιο για όλες τις κατηγορίες μιας μεταβλητής.

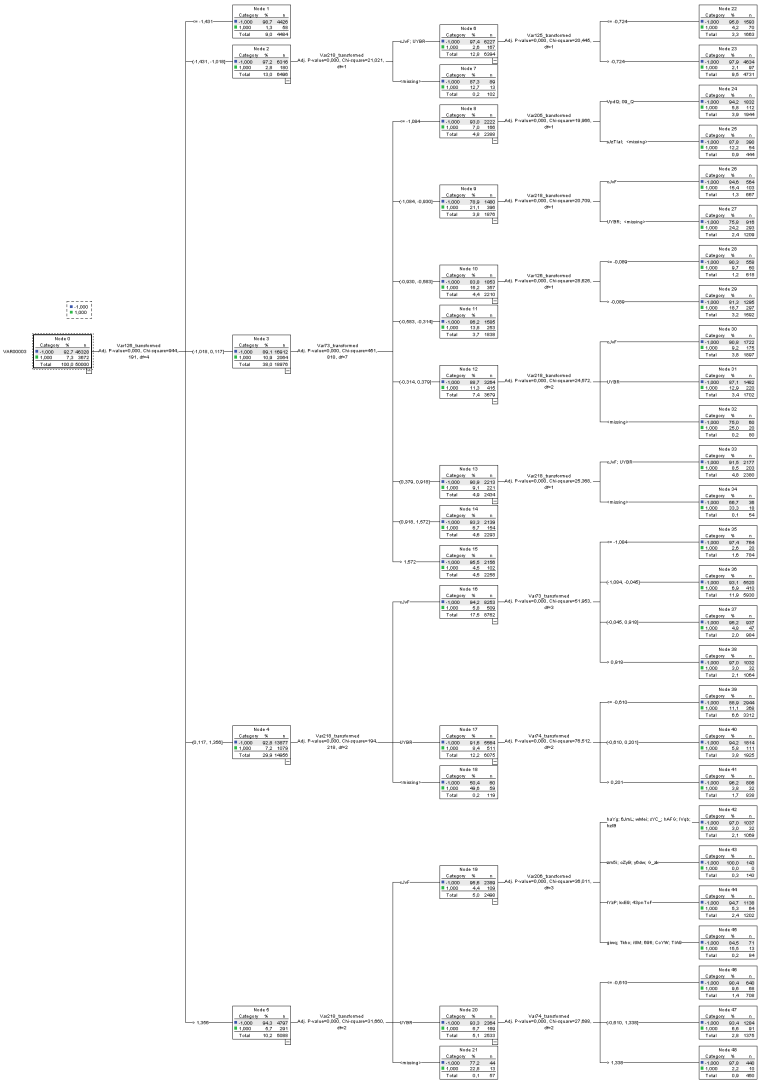
Τα αποτελέσματα που προέκυψαν από την ανάλυση του μοντέλου είναι τα παρακάτω:



Σχήμα 6.4: Το δέντρο που προέκυψε εφαρμόζοντας τον Exhaustive CHAID για την μεταβλητή απόκρισης appetency



Σχήμα 6.5: Δέντρο που προέκυψε εφαρμόζοντας τον Exhaustive CHAID στο πρόβλημα ταξινόμησης για την μεταβλητή churn



Σχήμα 6.6: Δέντρο που προέκυψε εφαρμόζοντας τον Exhaustive CHAID στο πρόβλημα ταξινόμησης για την μεταβλητή απόκρισης up - selling

6.4.3 CRT

Ο CRT μας επιτρέπει την κατασκευή ενός δέντρου απόφασης που μπορεί να χρησιμοποιηθεί για την ταξινόμηση μελλοντικών παρατηρήσεων. Η μέθοδος αυτή διαχωρίζει αναδρομικά τις παρατηρήσεις σε τμήματα, ελαχιστοποιώντας σε κάθε βήμα την μη καθαρότητα κάθε κόμβου. Υπενθυμίζουμε ότι ένας κόμβος θεωρείται

καθαρός όταν το 100% των παρατηρήσεων που καταλήγουν σε αυτόν ανήκουν αποκλειστικά σε μια κατηγορία του πεδίου στόχου. Σε αυτό τον αλγόριθμο μπορούμε να εισάγουμε όλους τους πιθανούς τύπους μεταβλητών στα πεδία της απόκρισης και των επεξηγηματικών μεταβλητών με την διαφορά όμως ότι οι διασπάσεις θα είναι αποκλειστικά δυαδικές.

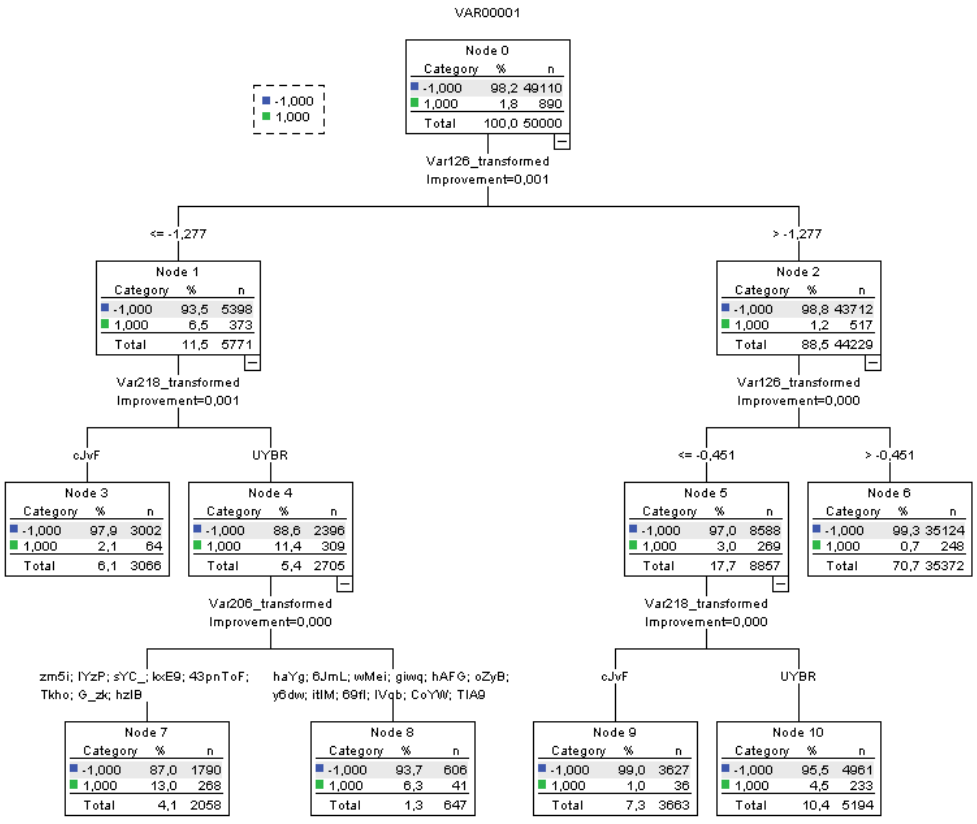
Η διαδικασία που ακολουθήσαμε για την κατασκευή του δέντρου είναι η κάτωθι:

$$Analyze \longrightarrow Classify \longrightarrow Tree$$

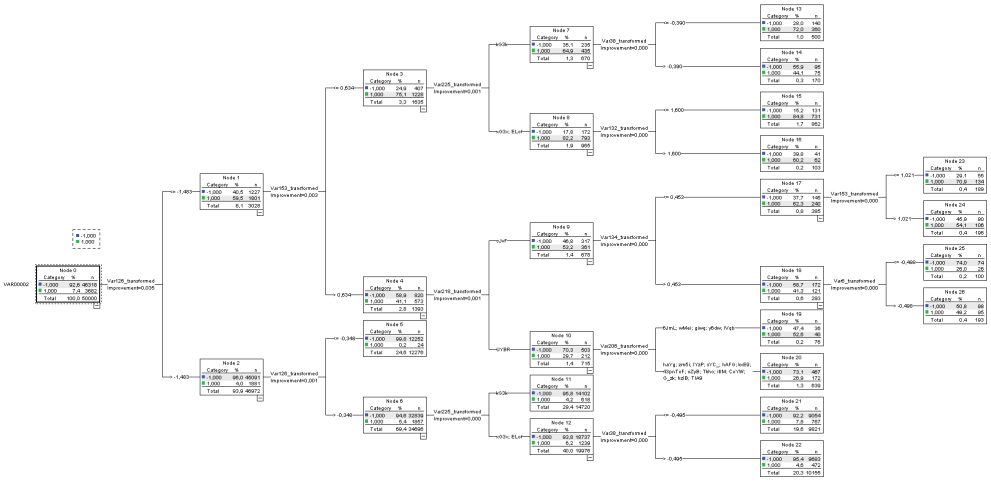
Παράμετροι Μοντέλου:

- *Growing Method*: CRT
- Output: σε αυτή την επιλογή ορίζουμε την μορφή που θέλουμε να έχουν τα αποτελέσματα της ανάλυσης. Ειδικότερα:
  - *Tree*: σε αυτή την καρτέλα ορίζουμε τον προσανατολισμό του δέντρου, το περιεχόμενο των κόμβων του,
  - *Statistics*: επιλέγουμε ποια στατιστικά θέλουμε να μας εμφανιστούν για την ανάλυση του μοντέλου. Σε σχέση με τα προηγούμενα μοντέλα, εδώ μπορούμε να υπολογίσουμε και την σημαντικότητα κάθε μεταβλητής στο μοντέλο καθώς και τις μεταβλητές υποκατάστασης από τον διαχωρισμό σε κάθε κόμβο.
  - *Plots*: κατασκευάζουμε το γράφημα σημαντικότητας όλων των επεξηγηματικών μεταβλητών.
- *Validation*: εδώ ορίζουμε την μέθοδο επικύρωσης του μοντέλου. Για το πρόβλημα μας επιλέξαμε την $10 - fold$ διασταυρωμένη επικύρωση.
- *Criteria*: στην καρτέλα αυτή επιλέγουμε τα επίπεδα σημαντικότητας για τον διαχωρισμό και την συγχώνευση των μεταβλητών κατά την κατασκευή του δέντρου. Αναλυτικότερα:
  - *CRT*: στην καρτέλα αυτή επιλέγουμε ως στατιστικό ελέγχου για τον διαχωρισμό των μεταβλητών σε κάθε κόμβο τον λόγο πιθανοφάνειας ενώ σαν επίπεδο σημαντικότητας για των διαχωρισμό και την συγχώνευση στους κόμβους την τιμή 0.01.
- *Options*: εδώ μπορούμε να διευθετήσουμε κάποια επιπλέον κατασκευαστικά ζητήματα του δέντρου όπως η διαχείριση των ελλιπών τιμών και τα κόστη εσφαλμένης ταξινόμησης.
  - *Missing Values*: επιλέγουμε ο αλγόριθμος να διαχειριστεί τις ελλιπείς τιμές ως ελλιπείς.
  - *Misclassification costs*: επιλέγουμε να είναι το ίδιο για όλες τις κατηγορίες μιας μεταβλητής.

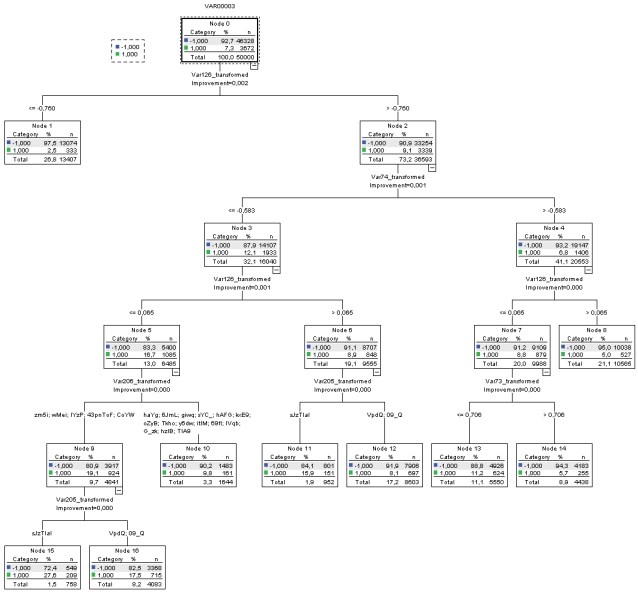
Για τα προβλήματα ταξινόμησης που αντιμετωπίσαμε προέκυψαν τα παρακάτω δέντρα απόφασεων:



Σχήμα 6.7: Δέντρο απόφασης της μεθόδου CRT στο πρόβλημα της ταξινόμησης της μεταβλητής appetency



Σχήμα 6.8: Δέντρο απόφασης της μεθόδου CRT στο πρόβλημα ταξινόμησης της μεταβλητής churn



Σχήμα 6.9: Δέντρο ταξινόμησης της μεθόδου CRT για την μεταβλητή up - selling

6.4.4 QUEST

Η μέθοδος QUEST είναι κατά βάση ένας βελτιωτικός αλγόριθμος όσον αφορά τις προηγούμενες μεθόδους. Όπως ο CRT, μας παρέχει δέντρα αποφάσεων δυαδικής ταξινόμησης αλλά με την διαφορά ότι είναι αρκετά πιο γρήγορος. Σε σύγκριση δε με τους άλλους αλγορίθμους κατασκευής δέντρων αποφάσεων, ο QUEST μειώνει σημαντικά την μεροληπτική τάση των μεθόδων δέντρων ταξινόμησης όπου ευνοούνται μεταβλητές που προσφέρουν περισσότερες διασπάσεις. Όσον αφορά τα ορίσματά του, ο QUEST δέχεται όλους τους τύπους μεταβλητών στο πεδίο των επεξηγηματικών μεταβλητών αλλά αποκλειστικά κατηγορικές στο πεδίο της μεταβλητής απόκρισης.

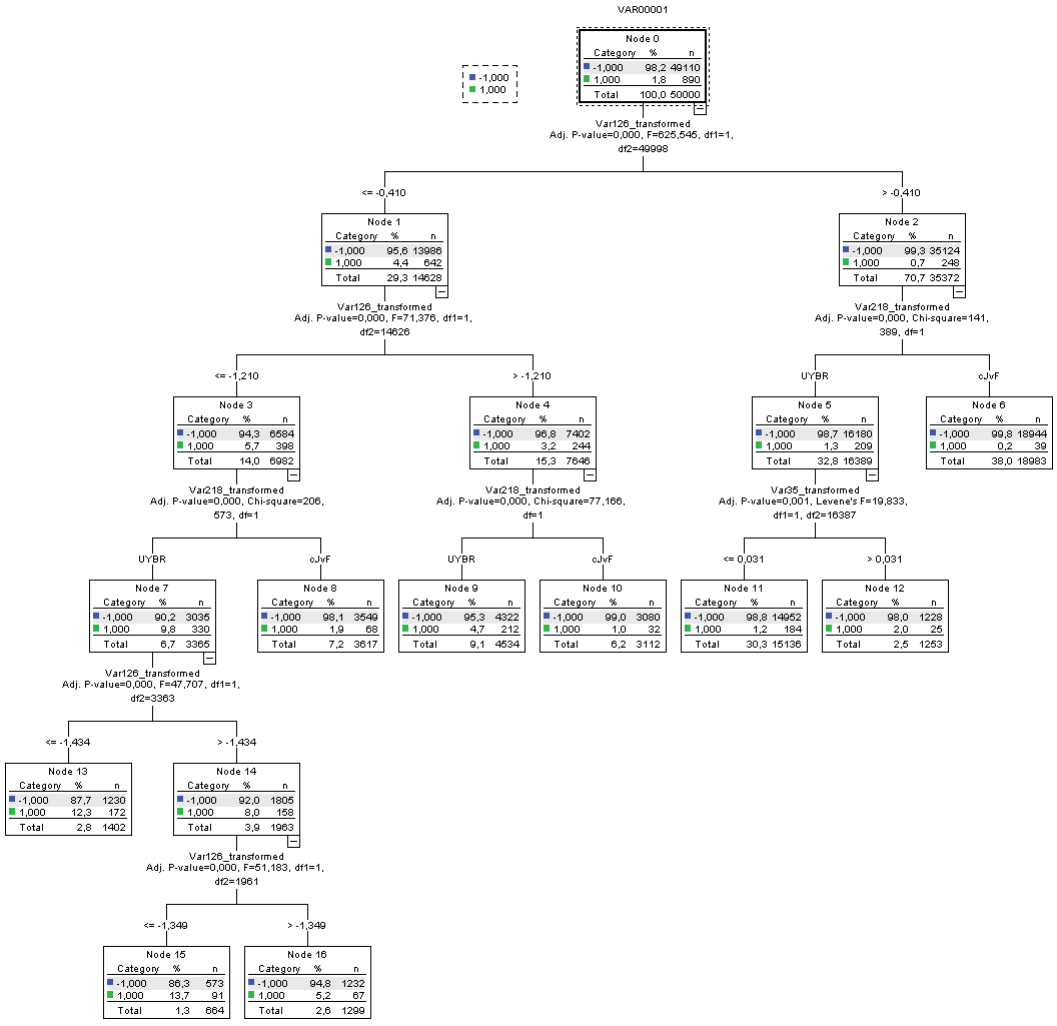
Για την εκτίμηση του μοντέλου με τον QUEST ακολουθήσαμε την παρακάτω διαδικασία:

$$Analyze \longrightarrow Classify \longrightarrow Tree$$

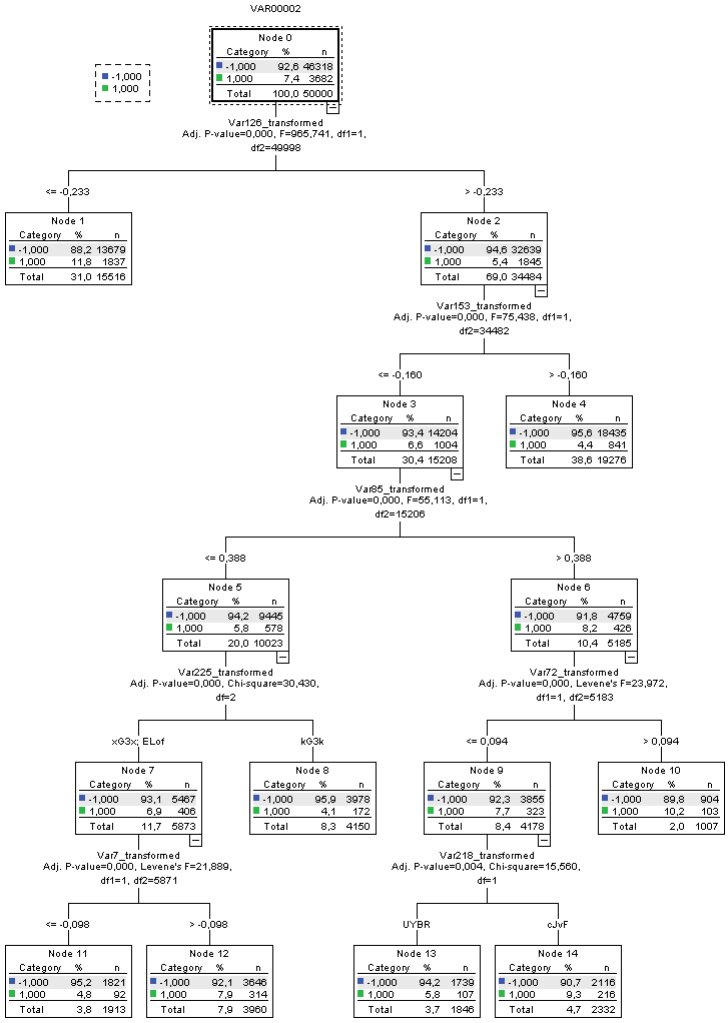
Παράμετροι Μοντέλου:

- *Growing Method*: QUEST
- *Output*: σε αυτή την επιλογή ορίζουμε την μορφή που θέλουμε να έχουν τα αποτελέσματα της ανάλυσης. Ειδικότερα:
  - *Tree*: σε αυτή την καρτέλα ορίζουμε τον προσανατολισμό του δέντρου, το περιεχόμενο των κόμβων του,
  - *Statistics*: επιλέγουμε ποια στατιστικά θέλουμε να μας εμφανιστούν για την ανάλυση του μοντέλου. Επιπρόσθετα, μας δίνεται η δυνατότητα να υπολογίσουμε τις μεταβλητές υποκατάστασης του μοντέλου.
- *Validation*: εδώ ορίζουμε την μέθοδο επικύρωσης του μοντέλου. Για το πρόβλημα μας επιλέξαμε την $10 - fold$ διασταυρωμένη επικύρωση.
- *Criteria*: στην καρτέλα αυτή επιλέγουμε τα επίπεδα σημαντικότητας για τον διαχωρισμό και την συγχώνευση των μεταβλητών κατά την κατασκευή του δέντρου. Αναλυτικότερα:
  - *QUEST*: στην καρτέλα αυτή επιλέγουμε σαν επίπεδο σημαντικότητας για των διαχωρισμό και την συγχώνευση στους κόμβους την τιμή 0.01.
- *Options*: εδώ μπορούμε να διευθετήσουμε κάποια επιπλέον κατασκευαστικά ζητήματα του δέντρου όπως η διαχείριση των ελλιπών τιμών και τα κόστη εσφαλμένης ταξινόμησης.
  - *Missing Values*: επιλέγουμε ο αλγόριθμος να διαχειριστεί τις ελλιπείς τιμές ως ελλιπείς.
  - *Misclassification costs*: επιλέγουμε να είναι το ίδιο για όλες τις κατηγορίες μιας μεταβλητής.

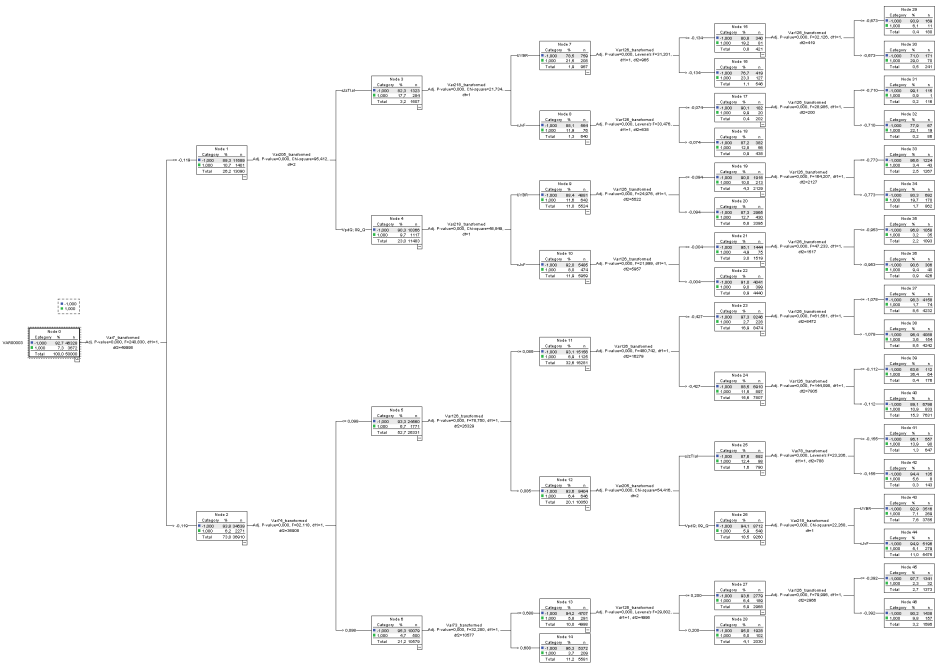
Τα δέντρα που προέκυψαν με χρήση του αλγορίθμου QUEST για τα τρία προβλήματα ακολουθούν παρακάτω:



Σχήμα 6.10: Δέντρο απόφασης για την μεταβλητή appetency κατά την εφαρμογή του QUEST



Σχήμα 6.11: Δέντρο απόφασης για την μεταβλητή churn μετά την εφαρμογή του QUEST



Σχήμα 6.12: Δέντρο απόφασης για την μεταβλητή up - selling κατά την εφαρμογή της μεθόδου QUEST

6.4.5 Σύγκριση απόδοσης των μεθόδων

Με βάση τα αποτελέσματα που πήραμε από τις παραπάνω αναλύσεις για τα τρία προβλήματα ταξινόμησης , μπορούμε να κατασκευάσουμε τον ακόλουθο συγκεντρωτικό πίνακα για τα δέντρα αποφάσεων που πήραμε για τις τρεις διαφορετικές μεταβλητές απόκρισης και να συγκρίνουμε τις αποδόσεις τους:

	Ακρίβεια			Cross - Validation		
Ταξινομητής	Appetency	Churn	Up - selling	Appetency	Churn	Up - selling
CHAID	98.2%	93.4%	92.7%	0.018	0.067	0.074
Exhaustive CHAID	98.2%	93.4%	92.7%	0.018	0.067	0.074
CRT	98.2%	94.5%	92.7%	0.018	0.057	0.073
QUEST	98.2%	92.6%	92.7%	0.018	0.074	0.074

Πίνακας 6.2: Πίνακας σύγκρισης δέντρων αποφάσεων για τα τρία προβλήματα ταξινόμησης

Κοιτώντας τα αποτελέσματα του παραπάνω πίνακα, είναι ορατό ότι ο αλγόριθμος CRT είναι ο καταλληλότερος και για τα τρία προβλήματα ταξινόμησης αφού συνδυάζει και τα πιο ικανοποιητικά ποσοστά ακρίβειας με τις χαμηλότερες τιμές

στο σφάλμα πρόβλεψης. Ξεχωριστά τώρα για κάθε πρόβλημα, βλέπουμε αρχικά ότι για το πρόβλημα ταξινόμησης της μεταβλητής appetency όλες οι μέθοδοι κρίνονται εξίσου ικανοποιητικές αφού για κάθε μέθοδο οι τιμές της ακρίβειας και του σφάλματος πρόβλεψης αντίστοιχα ταυτίζονται. Κάτι περίπου ανάλογο συμβαίνει και για το πρόβλημα ταξινόμησης της μεταβλητής up - selling, όπου όλοι αλγόριθμοι έχουν ναι μεν την ίδια ακρίβεια αλλά η διαφορά (αν και ελάχιστη) εντοπίζεται στο σφάλμα πρόβλεψης όπου ο CRT επικρατεί. Τέλος, όσον αφορά το πρόβλημα ταξινόμησης για την μεταβλητή churn, παρατηρούνται οι μεγαλύτερες διακυμάνσεις σε επίπεδο ακρίβειας και σφάλματος πρόβλεψης, με το καλύτερο συνδυασμό και των δύο να εκτιμάται από τον αλγόριθμο CRT.

6.5 Τεχνητά Νευρωνικά Δίκτυα

Η λογική πάνω στην οποία αναπτύχθηκαν τα Τεχνητά Νευρωνικά Δίκτυα είναι ουσιαστικά ένα απλοποιημένο μοντέλο του ανθρώπινου εγκεφάλου. Για την ακρίβεια, ο τρόπος λειτουργίας τους είναι μέσω της προσομοίωσης απλών διασυνδεδεμένων μονάδων επεξεργασίας που αντιπροσωπεύουν τους ανθρώπινους νευρώνες. Ένα αρκετά παράδοξο χαρακτηριστικό τους είναι ότι ενώ αποτελούν μια πολύ ισχυρή μέθοδο εκτίμησης στον τομέα της εξόρυξης δεδομένων, δεν απαιτούν μεγάλη μαθητική ή στατιστική γνώση για να εφαρμοστούν.

Για τα προβλήματα ταξινόμησης που θα αντιμετωπίσουμε σε αυτή την εργασία θα χρησιμοποιήσουμε δύο πολύ γνωστούς τύπους δικτύων: τα Multi Layer Perceptrons και τα Radial Basis Function Perceptrons. Τα βήματα για την εύρεση του μοντέλου για κάθε τύπο δικτύου παρουσιάζονται στις επόμενες δύο παραγράφους.

6.5.1 Multi Layer Perceptron (MLP)

Επιλέγουμε την διαδρομή:

$$Analyze \longrightarrow Neural\ Networks \longrightarrow Multilayer\ Perceptron$$

Στο πεδίο επιλογών που ανοίγει κάνουμε τις παρακάτω τροποποιήσεις:

- *Variables*: Στην καρτέλα αυτή ορίζουμε το είδος των μεταβλητών μας. Για τις επεξηγηματικές μας μεταβλητές δίνονται δύο ξεχωριστά πεδία: στο πεδίο *Factors* εισάγονται οι κατηγορικές μεταβλητές ενώ στο πεδίο *Covariates* οι συνεχείς μεταβλητές.

- *Partitions*: Σε αντίθεση με τα δέντρα, εδώ η SPSS μας δίνει την δυνατότητα να διαχωρίσουμε το υπάρχον σύνολο εκπαίδευσης σε δύο υποσύνολα. Επιλέγοντας *Randomly Assign Cases Based On Several Number Of Cases* μπορούμε να ορίσουμε το ποσοστό των δεδομένων του αρχικού συνόλου θα ανήκει στο σύνολο εκπαίδευσης και στο σύνολο εξέτασης. Και για τα τρία προβλήματα που θα δούμε θέσαμε ως 70% του αρχικού συνόλου το σύνολο εκπαίδευσης και 30% σύνολο εξέτασης.

- *Architecture*: Εδώ μπορούμε να επιλέξουμε όλες τις ρυθμίσεις που αφορούν την αρχιτεκτονική ενός νευρωνικού δικτύου, όπως τον αριθμό των κρυφών στρωμάτων, τις συναρτήσεις ενεργοποίησης για τα κρυφά στρώματα και τα στρώματα εξόδου. Για το κάθε ένα πρόβλημα που έχουμε, μέσω της επιλογής *Automatic Architecture Selection*, θα κατασκευάσουμε τρία νευρωνικά δίκτυα με 3,5 και 9 μονάδες στο κρυφό στρώμα.

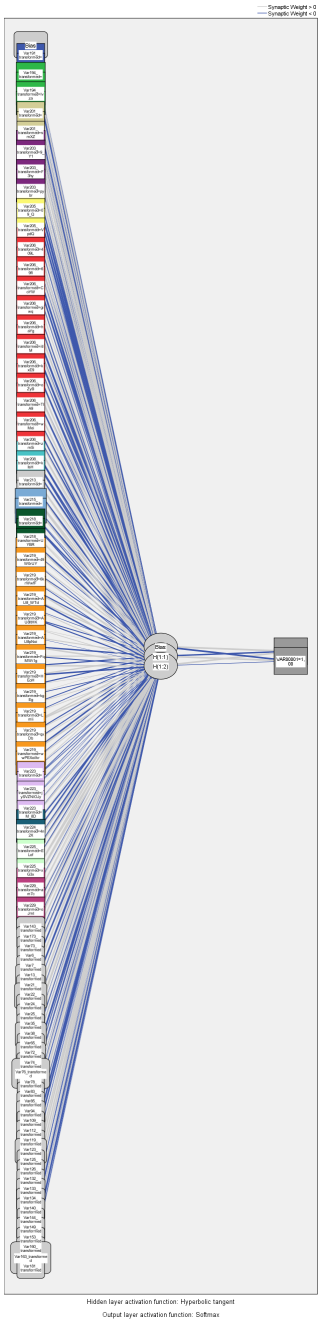
- *Training*: Το επίπεδο αυτό ορίζει το είδος της μεθόδου εκπαίδευσης καθώς και την μέθοδο βελτιστοποίησης της συνάρτησης εξόδου. Με βάση το μέγεθος του δείγματος, για μεγαλύτερη ακρίβεια επιλέξαμε ως μέθοδο εκπαίδευσης το *Batch Training* και ως μέθοδο βελτιστοποίησης την *Scaled Conjugate Gradient*.

- *Output*: Μέσω αυτής της καρτέλας καθορίζουμε την μορφή των αποτελεσμάτων του μοντέλου ταξινόμησης καθώς και των μέτρων αξιολόγησης αυτού. Στις προεπιλογές του SPSS θα προσθέσουμε την εμφάνιση του πίνακα συναπτικών βαρών, το γράφημα ROC και το διάγραμμα σημαντικότητας των επεξηγηματικών μεταβλητών.

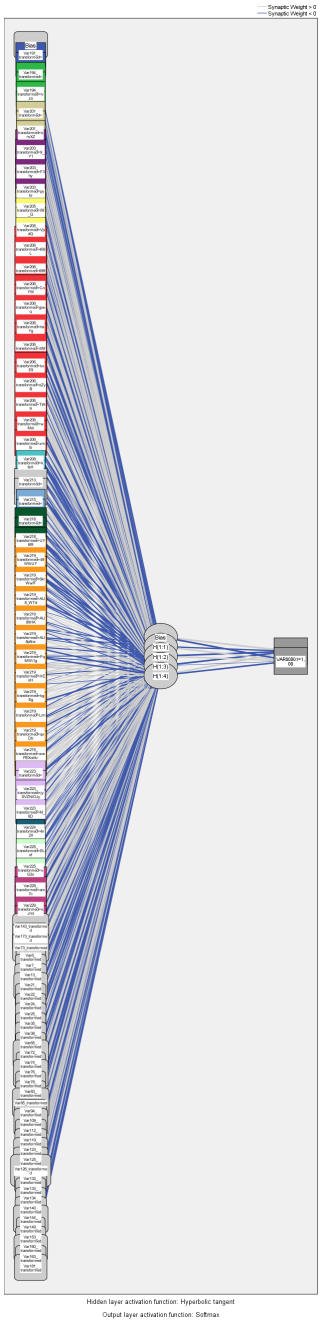
- *Options*: Στο τελευταίο αυτό πεδίο καθορίζονται κάποιες επιπλεόν λεπτομέρειες του δικτύου προς κατασκευή όπως οι κανόνες διακοπής της διαδικασίας και οι ελλιπείς τιμές. Οι ρυθμίσεις που επιλέξαμε για τα προβλήματά μας αφορούν την εισαγωγή των ελλιπών τιμών στο μοντέλο (εφόσον το SPSS τις παραλείπει αυτόματα κατά την διαδικασία) ενώ οι κανόνες διακοπής μένουν ως έχουν.

Στη συνέχεια ακολουθούν τα αποτελέσματα για το κάθε ένα από τα τρία προβλήματα ταξινόμησης:

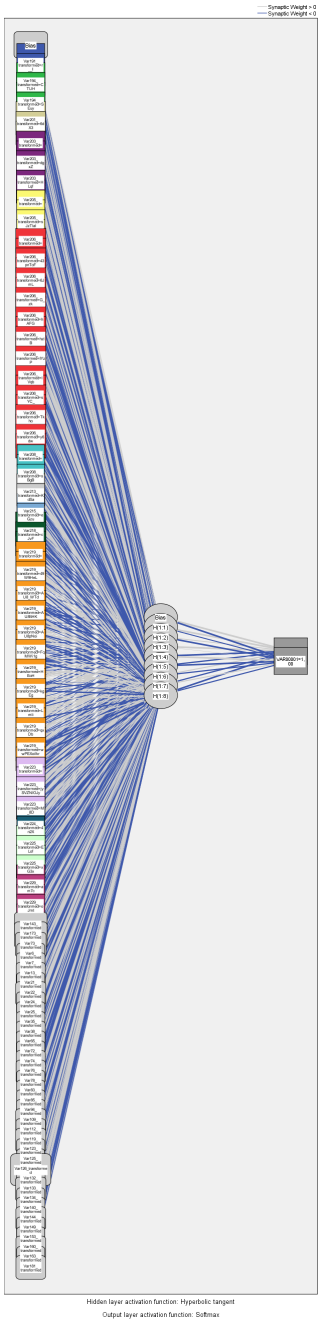
Appetency:



Σχήμα 6.13: Γραφική απεικόνιση του νευρωνικού δικτύου της μεταβλητής appetency για τρεις νευρώνες στο κρυφό στρώμα

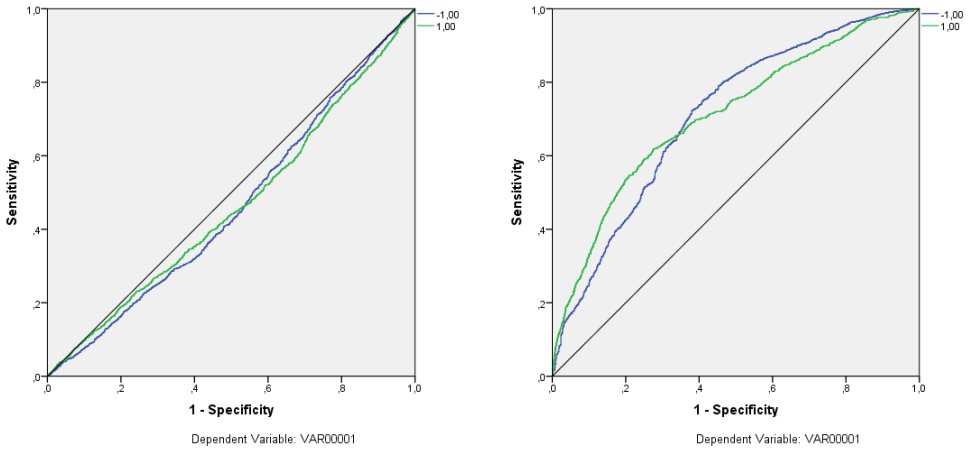


Σχήμα 6.14: Γραφική απεικόνιση του νευρωνικού δικτύου της μεταβλητής appetency για πέντε νευρώνες στο κρυφό στρώμα

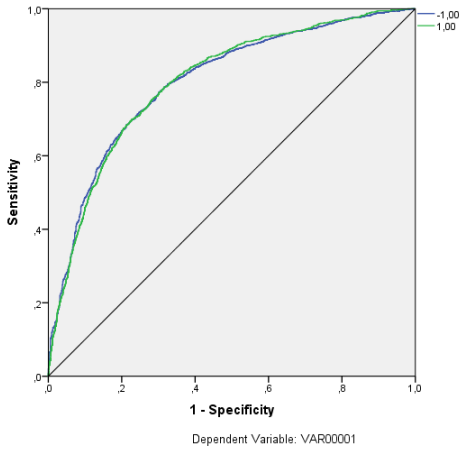


Σχήμα 6.15: Γραφική απεικόνιση του νευρωνικού δικτύου της μεταβλητής appetency για εννιά νευρώνες στο κρυφό στρώμα

Τα αντίστοιχα γραφήματα ROC είναι τα παρακάτω:

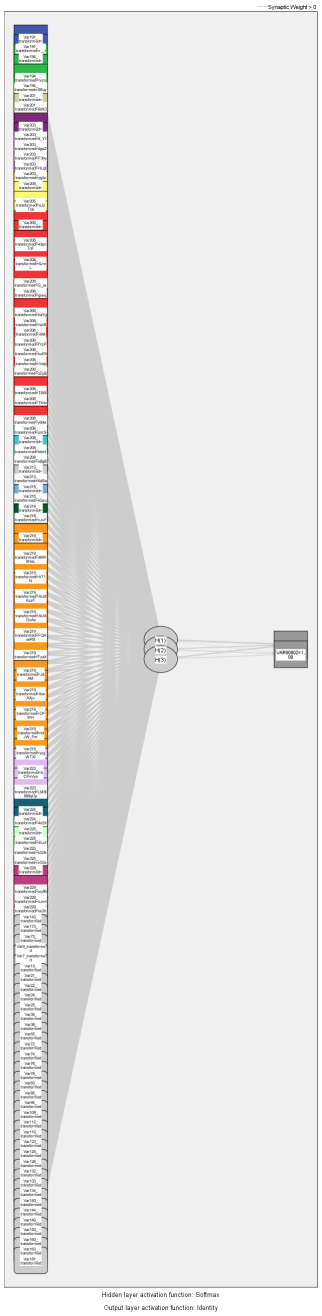


Σχήμα 6.16: Καμπύλες ROC για την μεταβλητή appetency για τρεις και πέντε κρυφές μονάδες αντίστοιχα

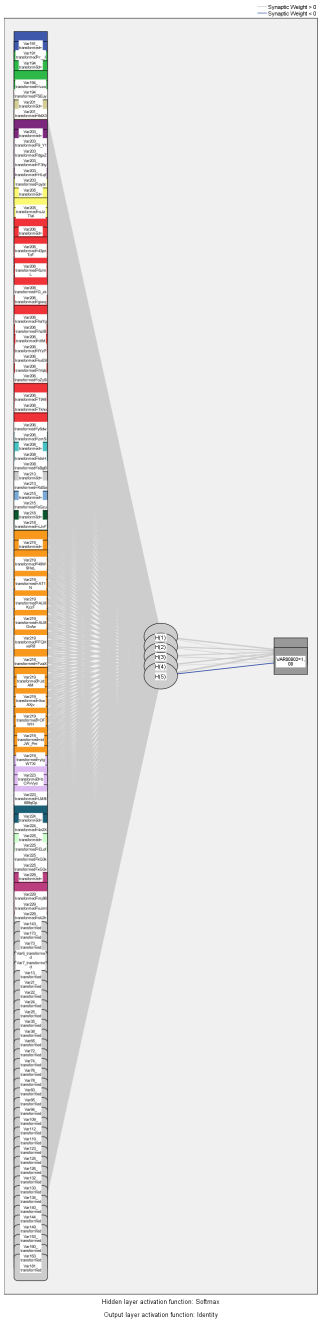


Σχήμα 6.17: Καμπύλη ROC για την μεταβλητή appetency για εννιά κρυφές μονάδες

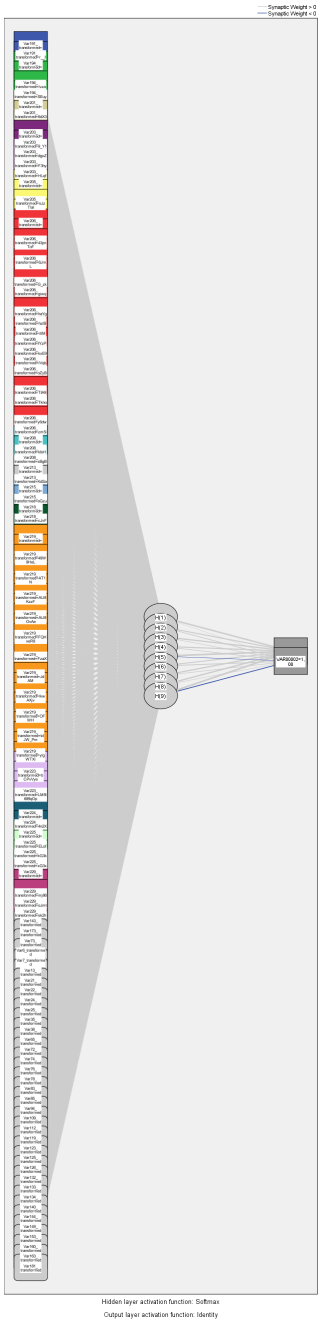
Churn:



Σχήμα 6.18: Νευρωνικό δίκτυο για την μεταβλητή churn με τρεις κρυφές μονάδες

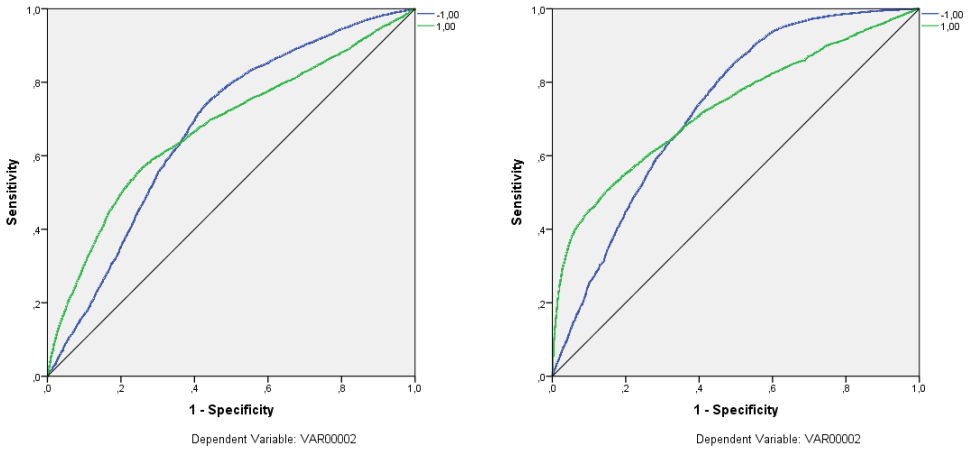


Σχήμα 6.19: Νευρωνικό δίκτυο για την μεταβλητή churn με πέντε κρυφές μονάδες

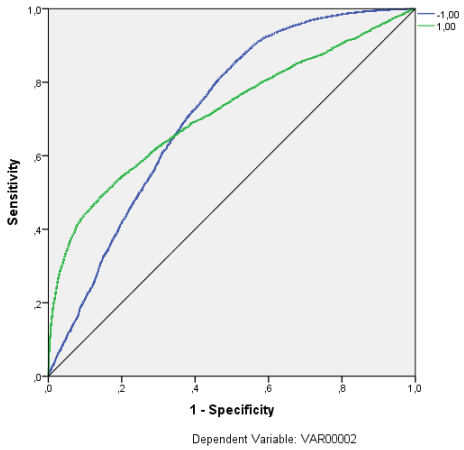


Σχήμα 6.20: Νευρωνικό δίκτυο για την μεταβλητή churn με εννιά κρυφές μονάδες

Τα αντίστοιχα γραφήματα ROC είναι τα παρακάτω:

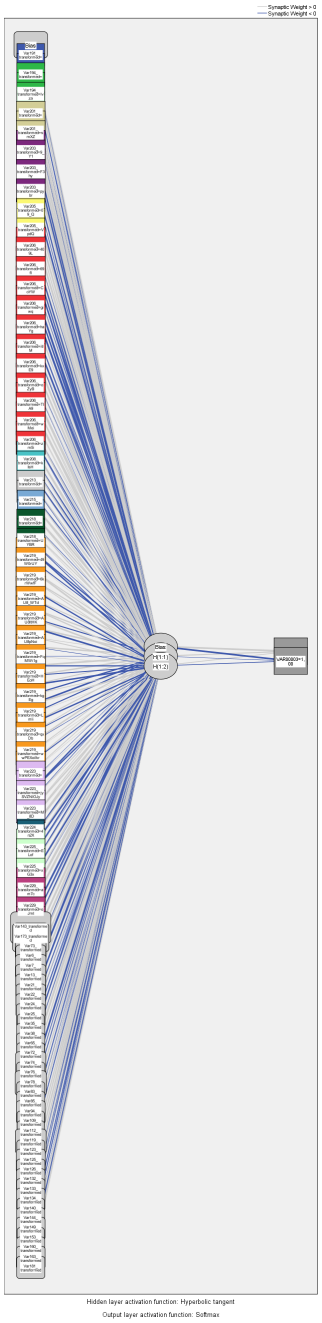


Σχήμα 6.21: Καμπύλες ROC για την μεταβλητή churn για τρεις και πέντε κρυφές μονάδες αντίστοιχα

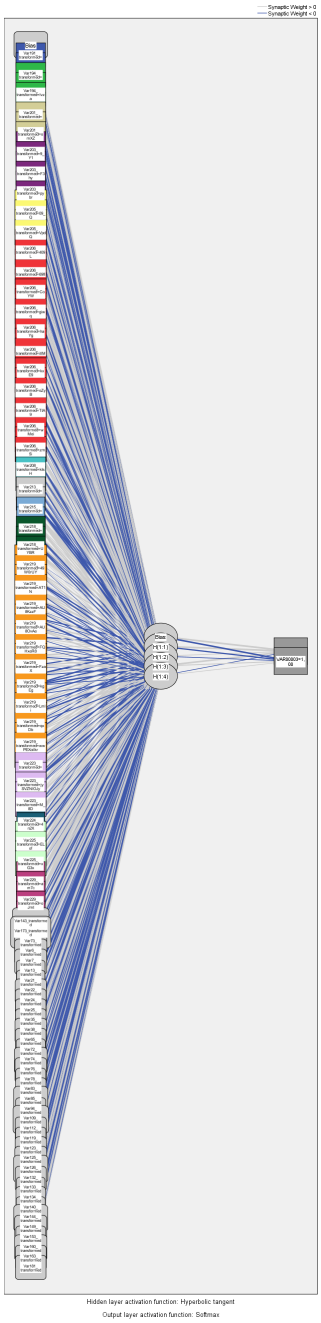


Σχήμα 6.22: Καμπύλη ROC για την μεταβλητή churn για εννιά κρυφές μονάδες

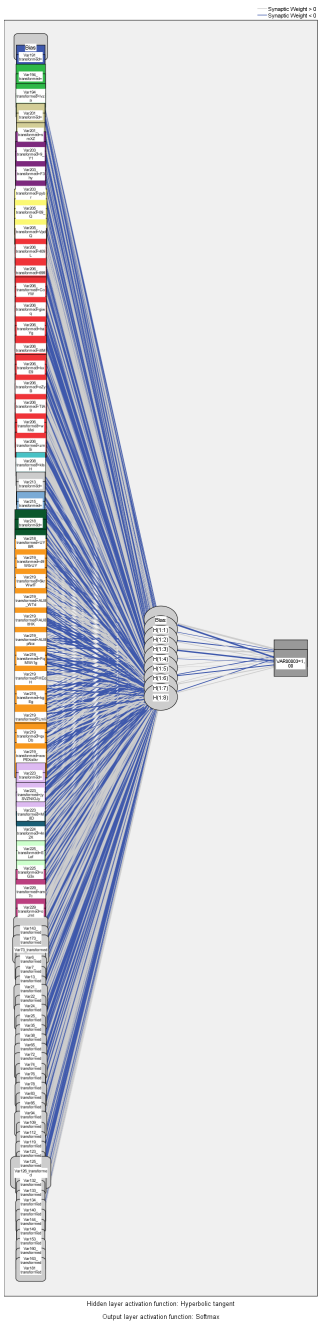
Up - selling:



Σχήμα 6.23: Νευρωνικό δίκτυο για την μεταβλητή up - selling με τρεις νευρώνες στο κρυφό στρώμα

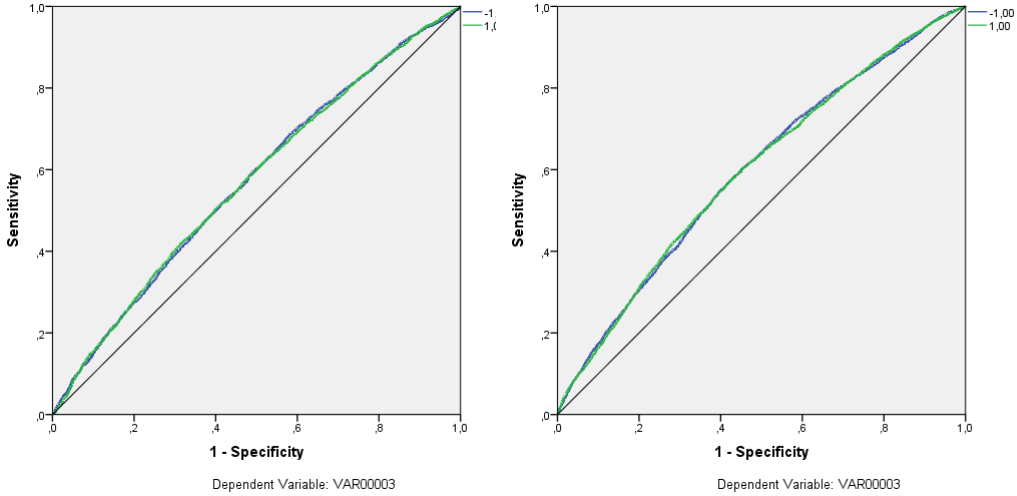


Σχήμα 6.24: Νευρωνικό δίκτυο για την μεταβλητή up - selling με πέντε νευρώνες στο κρυφό στρώμα

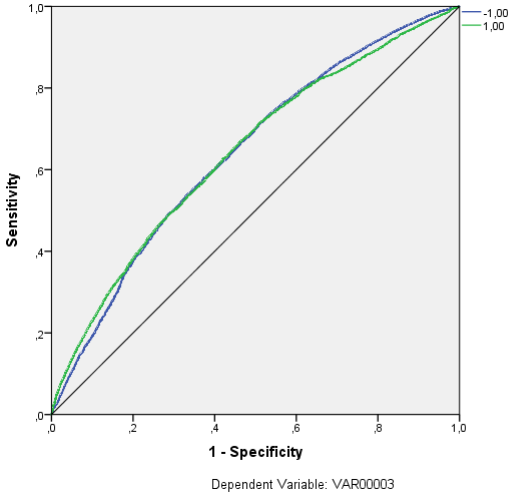


Σχήμα 6.25: Νευρωνικό δίκτυο για την μεταβλητή up - selling με εννιά νευρώνες στο κρυφό στρώμα

Τα αντίστοιχα γραφήματα ROC είναι τα παρακάτω:



Σχήμα 6.26: Γραφήματα ROC για την μεταβλητή up - selling με τρεις και πέντε νευρώνες στο κρυφό στρώμα αντίστοχα



Σχήμα 6.27: Γράφημα ROC για την μεταβλητή up - selling με εννιά νευρώνες στο κρυφό στρώμα

6.5.2 Radial Basis Function (RBF) Perceptron

Επιλέγουμε την διαδρομή:

$$Analyze \longrightarrow Neural\ Networks \longrightarrow Radial\ Basis\ Function$$

Στο πεδίο επιλογών που ανοίγει κάνουμε τις παρακάτω τροποποιήσεις:

- *Variables*: Στην καρτέλα αυτή ορίζουμε το είδος των μεταβλητών μας. Για τις επεξηγηματικές μας μεταβλητές δίνονται δύο ξεχωριστά πεδία: στο πεδίο $Factors$ εισάγονται οι κατηγορικές μεταβλητές ενώ στο πεδίο $Covariates$ οι συνεχείς μεταβλητές.

- *Partitions*: Σε αντίθεση με τα δέντρα, εδω η SPSS μας δίνει την δυνατότητα να διαχωρίσουμε το υπάρχον σύνολο εκπαίδευσης σε δύο υποσύνολα. Επιλέγοντας $Randomly\ Assign\ Cases\ Based\ On\ Several\ Number\ Of\ Cases$ μπορούμε να ορίσουμε το ποσοστό των δεδομένων του αρχικού συνόλου θα ανήκει στο σύνολο εκπαίδευσης και στο σύνολο εξέτασης. Και για τα τρία προβλήματα που θα δούμε θέσαμε ως 70% του αρχικού συνόλου το σύνολο εκπαίδευσης και 30% σύνολο εξέτασης.

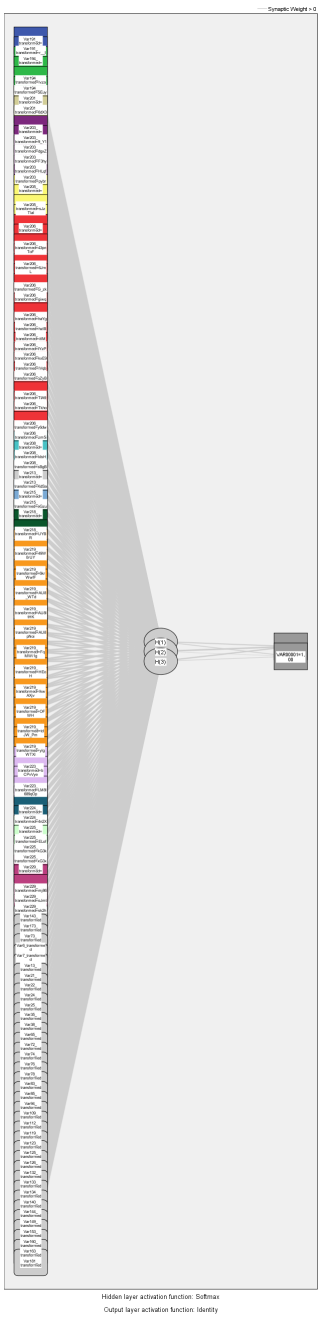
- *Architecture*: Εδώ μπορούμε να επιλέξουμε όλες τις ρυθμίσεις που αφορούν την αρχιτεκτονική ενός νευρωνικού δικτύου, όπως τον αριθμό των κρυφών στρωμάτων, τις συναρτήσεις ενεργοποίησης για τα κρυφά στρώματα και τα στρώματα εξόδου. Για το κάθε ένα πρόβλημα που έχουμε, μέσω της επιλογής $Use\ Specified\ Number\ Of\ Units$, θα κατασκευάσουμε τρία νευρωνικά δίκτυα με 3,5 και 9 μονάδες στο κρυφό στρώμα. Επιπλέον, οριζούμε ως συνάρτηση ενεργοποίησης του κρυφού στρώματος την *Normalized Radial Basis Function*.

- *Output*: Μέσω αυτής της καρτέλας καθορίζουμε την μορφή των αποτελεσμάτων του μοντέλου ταξινόμησης καθώς και των μέτρων αξιολόγησης αυτού. Στις προεπιλογές του SPSS θα προσθέσουμε την εμφάνιση του πίνακα συναπτικών βαρών, το γράφημα ROC και το διάγραμμα σημαντικότητας των επεξηγηματικών μεταβλητών.

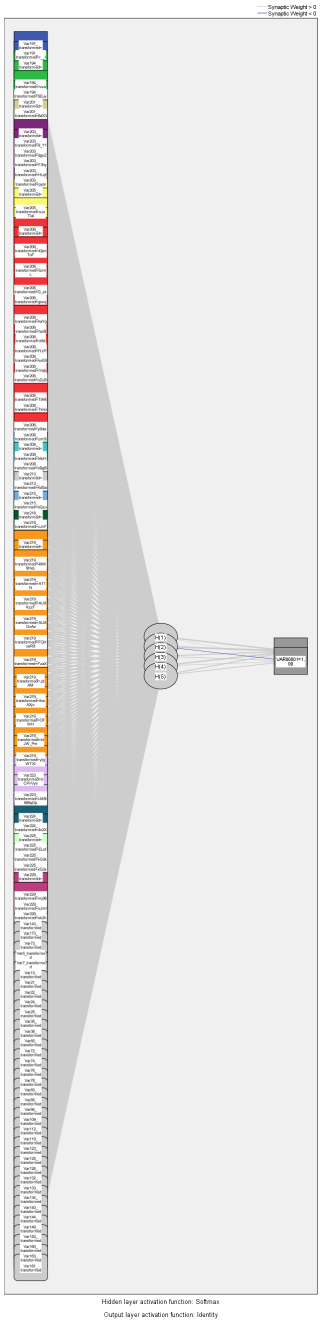
- *Options*: Στο τελευταίο αυτό πεδίο καθορίζονται κάποιες επιπλεόν λεπτομέρειες του δικτύου προς κατασκευή όπως οι κανόνες διακοπής της διαδικασίας και οι ελλιπείς τιμές. Οι ρυθμίσεις που επιλέξαμε για τα προβλήματά μας αφορούν την εισαγωγή των ελλιπών τιμών στο μοντέλο.

Στη συνέχεια ακολουθούν τα αποτελέσματα για το κάθε ένα από τα τρία προβλήματα ταξινόμησης:

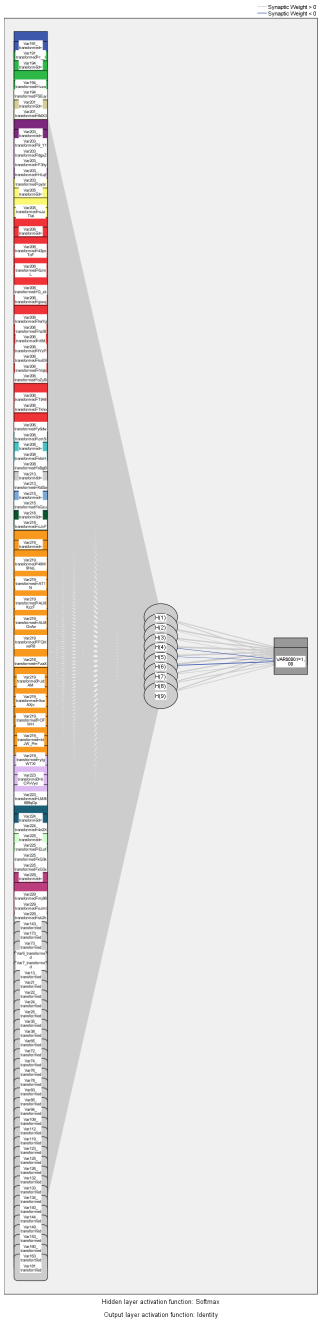
Appetency:



Σχήμα 6.28: Νευρωνικό δίκτυο για την μεταβλητή appetency με τρεις κρυφές μονάδες

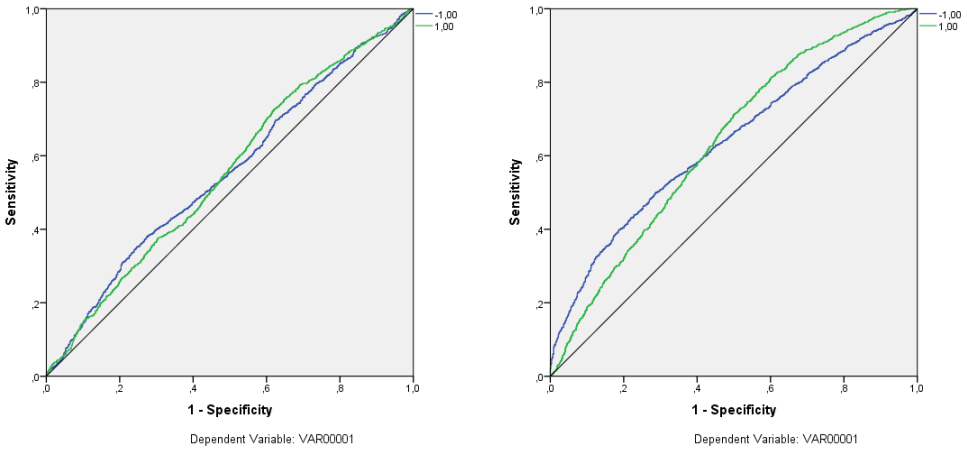


Σχήμα 6.29: Νευρωνικό δίκτυο για την μεταβλητή appetency με πέντε κρυφές μονάδες

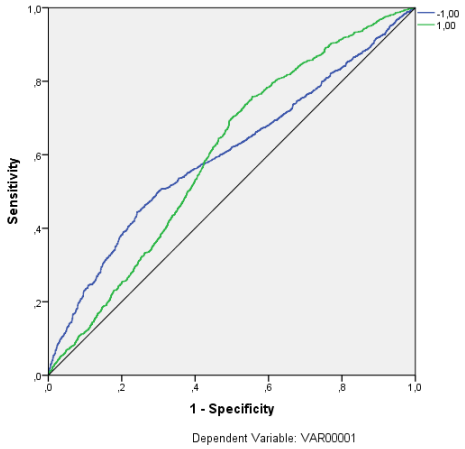


Σχήμα 6.30: Νευρωνικό δίκτυο για την μεταβλητή appetency με εννιά κρυφές μονάδες

Τα αντίστοιχα ROC γραφήματα είναι τα εξής:

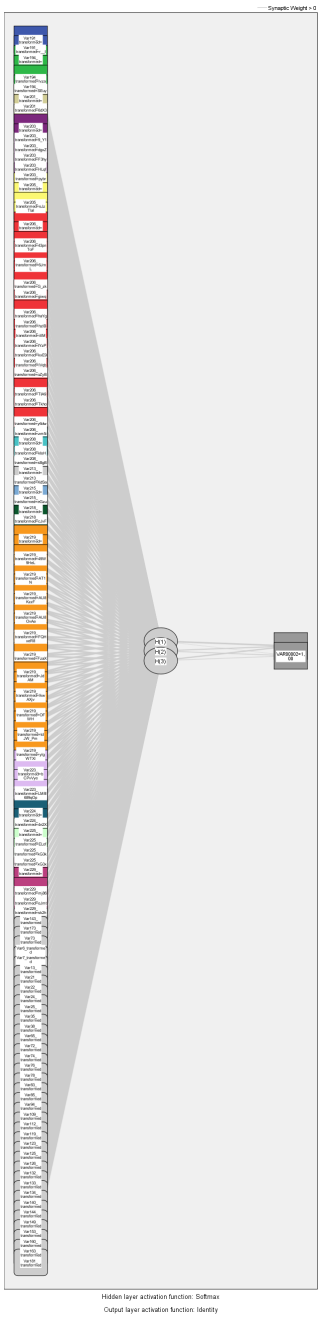


Σχήμα 6.31: Καμπύλες ROC για την μεταβλητή appetency για τρεις και πέντε κρυφές μονάδες αντίστοιχα

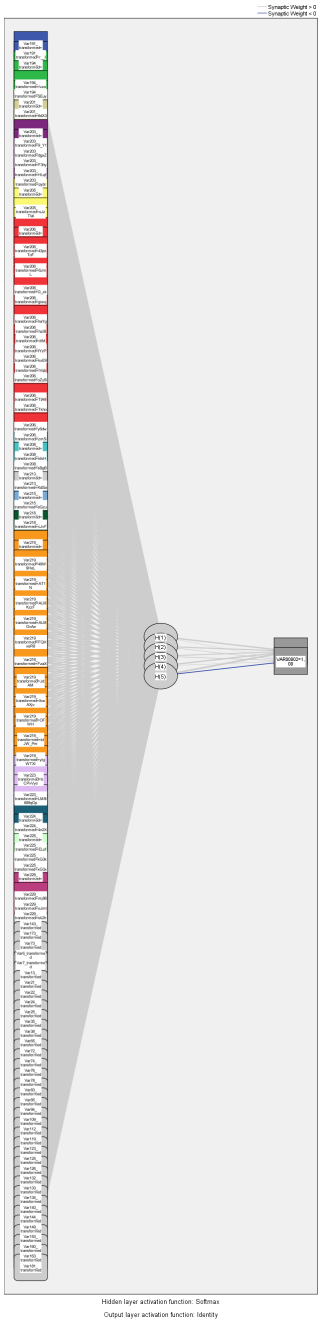


Σχήμα 6.32: Καμπύλη ROC για την μεταβλητή appetency για εννιά κρυφές μονάδες

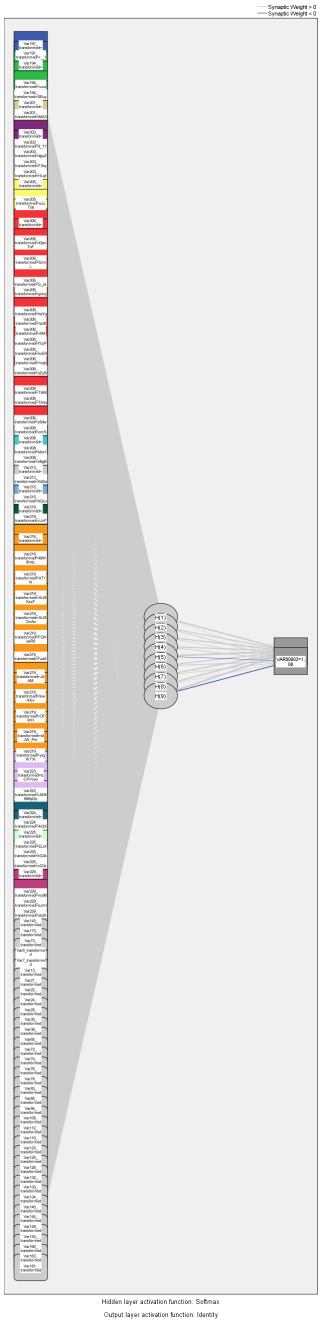
Churn:



Σχήμα 6.33: Νευρωνικό δίκτυο για την μεταβλητή churn με τρεις κρυφές μονάδες

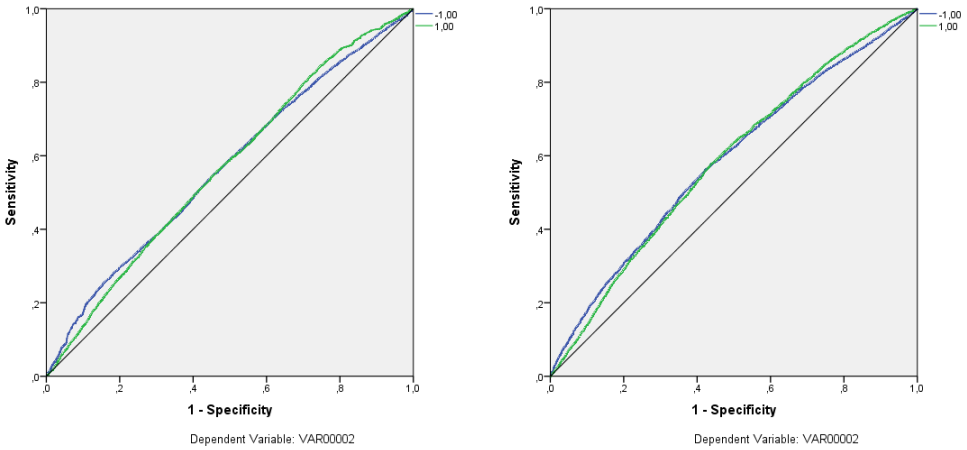


Σχήμα 6.34: Νευρωνικό δίκτυο για την μεταβλητή churn με πέντε κρυφές μονάδες

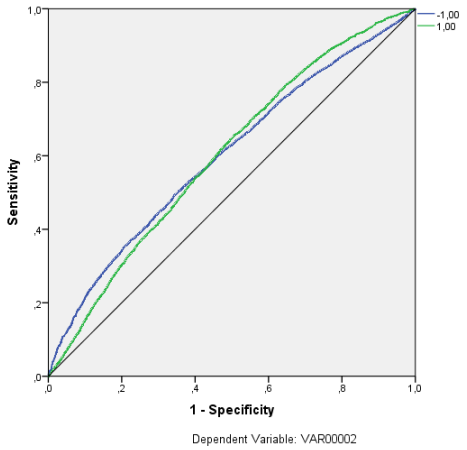


Σχήμα 6.35: Νευρωνικό δίκτυο για την μεταβλητή churn με εννιά κρυφές μονάδες

Τα αντίστοιχα ROC γραφήματα είναι τα εξής:

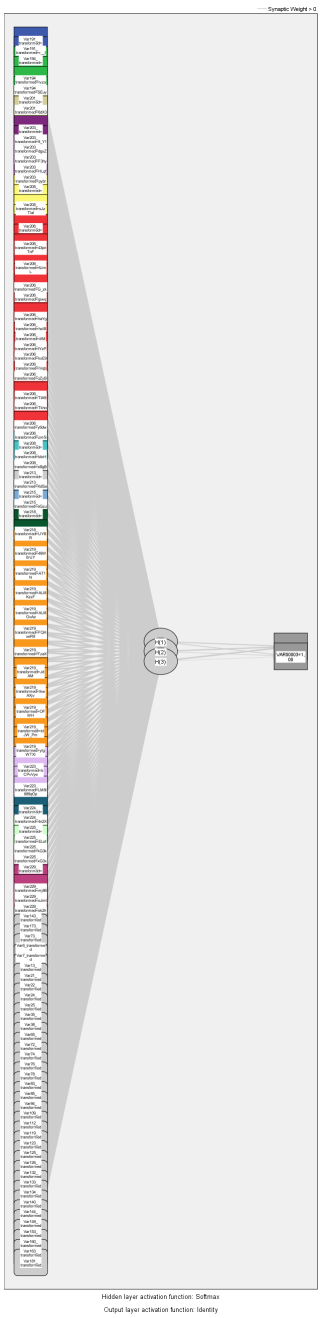


Σχήμα 6.36: Καμπύλες ROC για την μεταβλητή churn για τρεις και πέντε κρυφές μονάδες αντίστοιχα

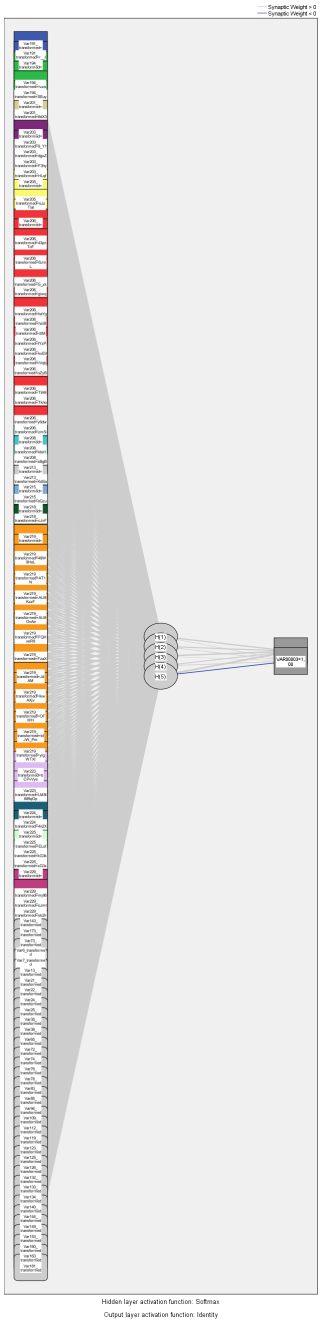


Σχήμα 6.37: Καμπύλη ROC για την μεταβλητή churn για εννιά κρυφές μονάδες

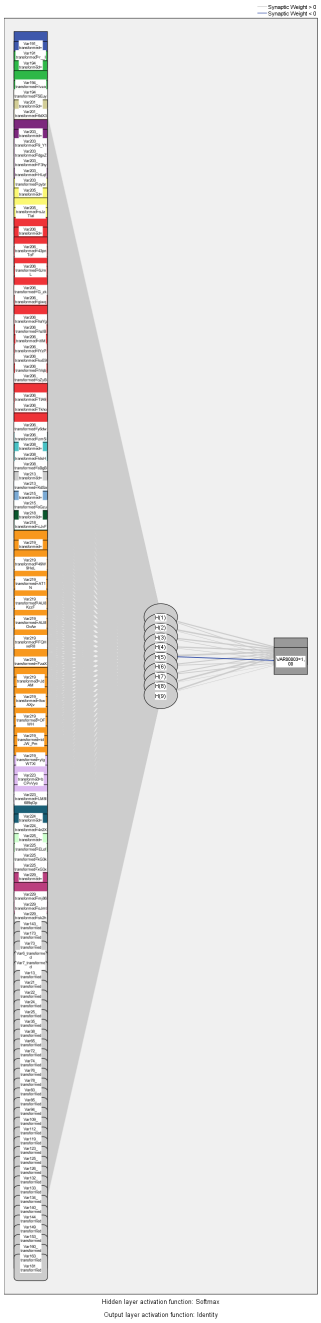
Up - selling:



Σχήμα 6.38: Νευρωνικό δίκτυο για την μεταβλητή up - selling με τρεις κρυφές μονάδες

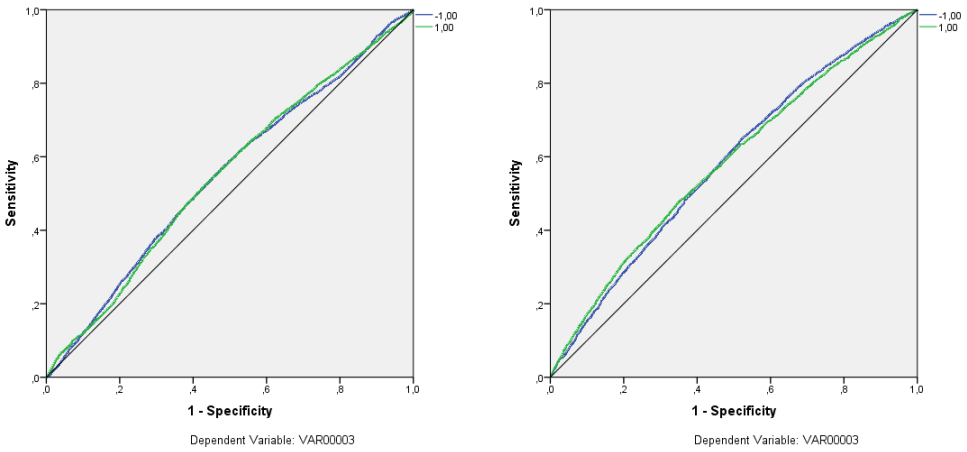


Σχήμα 6.39: Νευρωνικό δίκτυο για την μεταβλητή up - selling με πέντε κρυφές μονάδες

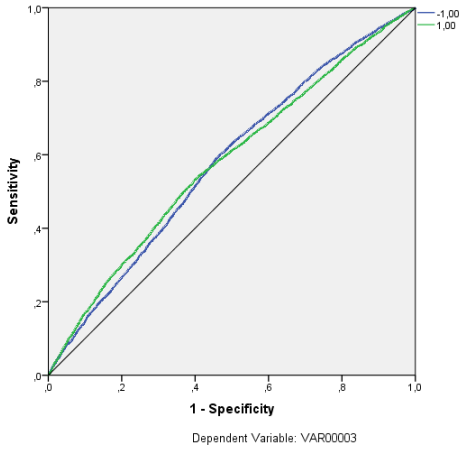


Σχήμα 6.40: Νευρωνικό δίκτυο για την μεταβλητή up - selling με εννιά κρυφές μονάδες

Τα αντίστοιχα ROC γραφήματα είναι τα εξής:



Σχήμα 6.41: Καμπύλες ROC για την μεταβλητή up - selling για τρεις και πέντε κρυφές μονάδες αντίστοιχα



Σχήμα 6.42: Καμπύλη ROC για την μεταβλητή up - selling για εννιά κρυφές μονάδες

6.5.3 Σύγκριση Τεχνητών Νευρωνικών Δικτύων

Η απόδοση ενός Νευρωνικού Δικτύου συνδέεται άμεσα με τον αριθμό των κρυφών στρωμάτων, καθώς είναι πολύ πιθανό αν αυξήσουμε τα κρυφά στρώματα να αυξηθεί και η απόδοση του μοντέλου. Συνεπώς, η σύγκριση των μοντέλων που προέκυψαν από τους τύπους δικτύων που χρησιμοποιήσαμε θα γίνει με βάση τον

αριθμό των κρυφών στρωμάτων που θέσαμε και παρουσιάζεται στον παρακάτω πίνακα. Όπως αναφέραμε και παραπάνω, θα εξετάσουμε το κάθε πρόβλημα για τρεις διαφορετικούς αριθμούς μονάδων στο κρυφό στρώμα προκειμένου να δούμε ποιος είναι ο βέλτιστος για τα δεδομένα μας.

Appetency:

Μέθοδος	Κρυφές Μονάδες	Ακρίβεια	AUC
		Εκπαίδευση / Εξέταση	+1 / -1
MLP	3	98.2 / 98.3	0.463 / 0.463
MLP	5	98.2 / 98.3	0.709 / 0.709
MLP	9	98.3 / 98.1	0.801 / 0.801
RBFN	3	98.2 / 98.2	0.554 / 0.554
RBFN	5	98.2 / 98.3	0.636 / 0.636
RBFN	9	98.2 / 98.3	0.599 / 0.599

Πίνακας 6.3: Πίνακας σύγκρισης αριθμού κρυφών μονάδων,εκτιμώμενης ακρίβειας και εμβαδού κάτω από την καμπύλη (AUC)

Από τα παραπάνω αποτελέσματα για το πρόβλημα ταξινόμησης της μεταβλητής appetency, βλέπουμε ότι ο βέλτιστος αριθμός νευρώνων για το κρυφό στρώμα επιλέγεται να είναι 9 για τον αλγόριθμο MLP και 5 για τον αλγόριθμο RBFN. Από την θεωρία που παρουσιάσαμε στα προηγούμενα κεφάλαια ωστόσο προκύπτει ότι ένας αλγόριθμος, συγκρινόμενους με άλλους, θεωρείται βέλτιστος όταν οι τιμές της ακρίβειας και του εμβαδού κάτω από την καμπύλη ROC (AUC) είναι εξίσου ικανοποιητικά. Αν και η ακρίβεια για τις επιλεγμένες περιπτώσεις των μεθόδων δεν διαφέρει σημαντικά, βασιζόμενοι στο προηγούμενο συμπέρασμα,ο αλγόριθμος RBFN θα πρέπει αναγκαστικά να απορριφθεί, διότι η τιμή της AUC σε σύγκριση με αυτή της μεθόδου MLP είναι σαφώς χειρότερη. Άρα, η βέλτιστη περίπτωση για το πρόβλημα μας είναι ο αλγόριθμος MLP με κρυφό στρώμα 9 νευρώνων.

Churn:

Μέθοδος	Κρυφές Μονάδες	Ακρίβεια	AUC
		Εκπαίδευση / Εξέταση	+1 / -1
MLP	3	92.6 / 92.8	0.676 / 0.676
MLP	5	92.9 / 92.8	0.730 / 0.730
MLP	9	93 / 92.4	0.718 / 0.718
RBFN	3	92.6 / 92.6	0.568 / 0.568
RBFN	5	92.6 / 92.7	0.587 / 0.587
RBFN	9	92.7 / 92.4	0.601 / 0.601

Πίνακας 6.4: Πίνακας σύγκρισης αριθμού κρυφών μονάδων,εκτιμώμενης ακρίβειας και εμβαδού κάτω από την καμπύλη (AUC)

Δουλεύοντας εντελώς ανάλογα, από τον πίνακα βλέπουμε ότι οι αλγόριθμοι MLP και RBFN με 5 και 9 νευρώνες αντίστοιχα στο κρυφό στρώμα δίνουν τα καλύτερα αποτελέσματα από όλες τις εξετασθείσες περιπτώσεις. Ωστόσο, αν συγκρίνουμε τις δύο καλύτερες επιλογές μεταξύ τους θα δούμε ότι η μέθοδος MLP μας προσφέρει σαφώς καλύτερα αποτελέσματα αφού παρουσιάζει υψηλότερες τιμές και στην ακρίβειας αλλά και στην AUC. Συνεπώς, η μέθοδος MLP με κρυφό στρώμα 5 νευρώνων επιλέγεται ως η βέλτιστη για το πρόβλημα ταξινόμησης της μεταβλητής churn.

Up - selling:

Μέθοδος	Κρυφές Μονάδες	Ακρίβεια	AUC
		Εκπαίδευση / Εξέταση	+1 / -1
MLP	3	92.8 / 92.4	0.571 / 0.571
MLP	5	92.6 / 92.8	0.597 / 0.597
MLP	9	92.7 / 92.6	0.642 / 0.642
RBFN	3	92.6 / 92.8	0.550 / 0.550
RBFN	5	92.7 / 92.5	0.583 / 0.583
RBFN	9	92.6 / 92.8	0.579 / 0.579

Πίνακας 6.5: Πίνακας σύγκρισης αριθμού κρυφών μονάδων, εκτιμώμενης ακρίβειας και εμβαδού κάτω από την καμπύλη (AUC)

Για το τελευταίο πρόβλημα ταξινόμησης που θα δούμε, τα αποτελέσματα που προέκυψαν υποδεικνύουν ως τους καλύτερους αλγορίθμους τους MLP με κρυφό στρώμα 9 νευρώνων και RBFN με κρυφό στρώμα 5 νευρώνων. Από την σύγκριση των δύο επιλεγμένων μεθόδων, βλέπουμε πάλι ότι ο αλγόριθμος MLP υπερτερεί του RBFN σε επίπεδο ακρίβειας αλλά και στην τιμή της AUC. Έτσι, για το πρόβλημα ταξινόμησης της μεταβλητής up - selling ως βέλτιστη μέθοδος αναδεικνύεται η MLP με κρυφό στρώμα 9 νευρώνων.

6.6 Μηχανές Διανυσμάτων Υποστήριξης

Οι Μηχανές Διανυσμάτων Υποστήριξης είναι μια από τις σημαντικότερες μεθόδους στην ανάλυση δεδομένων υψηλών διαστάσεων καθώς επιτρέπει την δυαδική ταξινόμηση των επεξηγηματικών μεταβλητών αποφεύγοντας την υπερπροσαρμογή του μοντέλου. Για την ανάλυση που κάναμε για τα τρία προβλήματα ταξινόμησης χρησιμοποιήσαμε πολυωνυμικό,ακτινικό και σιγμοειδή πυρήνα, ενώ για κάθε έναν από αυτούς τους πυρήνες εξετάσαμε πέντε διαφορετικές τιμές της παραμέτρου κανονικοποίησης C (grid search). Υπενθυμίζουμε ότι η παράμετρος κανονικοποίησης C εκφράζει την συσχέτιση μεταξύ του μεγιστοποίησης του περιθωρίου και της ελαχιστοποίησης του σφάλματος εκπαίδευσης.

6.6.1 Συγκεντρωτικοί πίνακες απόδοσης

Στις παρακάτω συγκριτικές μελέτες που προέκυψαν για το κάθε πρόβλημα ταξινόμησης βλέπουμε ότι το βέλτιστο μοντέλο για όλες επιτυγχάνεται, για σταθερή παράμετρο γ, χρησιμοποιώντας C=5. Τα ποσοστά ακρίβειας που παρουσιάζονται στους παρακάτω πίνακες είναι αποτελέσματα της εφαρμογής της μεθόδου της $k-$διασταυρωμένης επικύρωσης για $k = 10$.

	Προγνωστική Ακρίβεια (%)				
Πυρήνες	c=1	c=2	c=3	c=4	c=5
πολυωνυμικός	97.68	97.69	97.68	97.68	97.7
σιγμοειδής	97.24	96.91	96.65	96.47	96.32
ακτινικός	97.66	97.66	97.68	97.67	97.67

Πίνακας 6.6: Σύγκριση της απόδοσης για τις SVMs διαφορετικού πυρήνα με βάση τα αποτελέσματα του grid search για την μεταβλητή appetency

Στο αρχικό πρόβλημα ταξινόμησης που αφορά την μεταβλητή appetency, βλέπουμε ότι τα καλύτερα μοντέλα επιτυγχάνονται για τις τιμές C=5, C=1 και C=3 με πυρήνες πολυωνυμικό, σιγμοειδή και ακτινικό αντίστοιχα. Συγκρίνοντας τις ακρίβειες των τριών επικρατέστερων μοντέλων , αμέσως θα εξαιρούσαμε το μοντέλο που προκύπτει για C=1 με σιγμοειδή πυρήνα καθώς η τιμή της ακρίβειας που παρέχει είναι σαφώς μικρότερη από αυτή των άλλων δύο μοντέλων. Τα δύο εναπομείναντα μοντέλα όπως βλέπουμε από τον πίνακα, έχουν πολύ μικρή διαφορά στην ακρίβειά τους, με αυτό που επικρατεί να είναι για C=5 για πολυωνυμικό πυρήνα.

	Προγνωστική Ακρίβεια (%)				
Πυρήνες	c=1	c=2	c=3	c=4	c=5
πολυωνυμικός	94.47	94.51	94.49	94.5	94.52
σιγμοειδής	92.79	91.73	91.23	90.82	90.56
ακτινικός	94.43	94.44	94.47	94.48	94.46

Πίνακας 6.7: Σύγκριση της απόδοσης για τις SVMs διαφορετικού πυρήνα με βάση τα αποτελέσματα του grid search για την μεταβλητή churn

Για την μεταβλητή churn, τα τρία καλύτερα μοντέλα που προκύπτουν για κάθε πυρήνα είναι για C=5, C=1 και C=4 για πολυωνυμικό, σιγμοειδή και ακτινικό πυρήνα αντίστοιχα. Όπως και στο προηγούμενο πρόβλημα ταξινόμησης, η ακρίβεια του μοντέλου που προκύπτει για την τιμή C=1 του σιγμοειδή πυρήνα είναι πάλι μικρότερη σε σύγκριση με τις άλλες δύο τιμές οπότε και απορρίπτεται. Όσο για τα μοντέλα πολυωνυμικού και ακτινικού πυρήνα που μένουν, αυτή την φορά υπάρχει μια πιο σημαντική διαφορά στις τιμές της ακρίβειας τους. Το μοντέλο που τελικά επιλέγουμε είναι αυτό του πολυωνυμικού πυρήνα καθώς εμφανίζει την υψηλότερη ακρίβεια.

	Προγνωστική Ακρίβεια (%)				
Πυρήνες	c=1	c=2	c=3	c=4	c=5
πολυωνυμικός	92.52	92.53	92.54	92.58	92.68
σιγμοειδής	89.52	88.32	87.80	87.73	87.44
ακτινικός	92.47	92.48	92.48	92.52	92.58

Πίνακας 6.8: Σύγκριση της απόδοσης για τις SVMs διαφορετικού πυρήνα με βάση τα αποτελέσματα του grid search για την μεταβλητή up - selling

Τέλος, στο πρόβλημα ταξινόμησης για την μεταβλητή up - selling, τα τρία βέλτιστα μοντέλα αντιστοιχούν στις τιμές C=5, C=1 και C=5 για πολυωνυμικό, σιγμοειδή και ακτινικό πυρήνα. Το μοντέλο που προκύπτει με εφαρμογή του σιγμοειδή πυρήνα κρίνεται πάλι ακατάλληλο διότι εμφανίζει για ακόμα μια φορά την χαμηλότερη ακρίβεια σε σχέση με τα άλλα δύο μοντέλα. Ως καλύτερο μοντέλο αναδεικνύεται αυτό που αντιστοιχεί στον πολυωνυμικό πυρήνα καθώς εξασφαλίζει την μεγαλύτερη ακρίβεια για το μοντέλο μας.

6.7 Συνολική σύγκριση των ταξινομητών

Στον παρακάτω πίνακα κατατάσσονται τα καλύτερα μοντέλα που προέκυψαν με την χρήση των παραπάνω ταξινομητών για τα τρία προβλήματα ταξινόμησης που μελετήσαμε προκειμένου να εντοπιστεί η καλύτερη προσέγγιση για την ανάλυσή τους.

	Ακρίβεια (%)		
Ταξινομητής	appetency	churn	up - selling
CHAID	98.2	93.4	92.7
Exhaustive CHAID	98.2	93.4	92.7
CRT	98.2	94.5	92.7
QUEST	98.2	92.6	92.7
MLP	98.1	92.8	92.6
Polynomial (SVM)	97.7	94.52	92.68

Πίνακας 6.9: Αναλυτική σύγκριση των ταξινομητών για όλα τα προβλήματα ταξινόμησης που είδαμε μέσω της συνολικής ακρίβειας

Μέσω του συγκεντρωτικού πίνακα είμαστε πλέον σε θέση να επιλέξουμε το καταλληλλότερο μοντέλο (με βάση του ταξινομητές που χρησιμοποιήσαμε) για το κάθε πρόβλημα ταξινόμησης. Έτσι λοιπόν βλέπουμε ότι για το πρόβλημα της μεταβλητής appetency όλοι οι αλγόριθμοι των δέντρων δίνουν ακριβώς την ίδια ακρίβεια και κατά συνέπεια είναι όλοι τους εξίσου κατάλληλοι για την περιγραφή του προβλήματος. Στο πρόβλημα ταξινόμησης που αφορά την μεταβλητή churn, η SVM πολυωνυμικού πυρήνα και με παράμετρο κανονικοποίησης C=5 αναδεικνύεται ως το αποδοτικότερο μοντέλο με ακρίβεια 94.52%. Τέλος, στο πρόβλημα ταξινόμησης

της μεταβλητής up - selling, αν και όλοι οι αλγόριθμοι των δέντρων δίνουν την ίδια ακρίβεια, ως βέλτιστος επιλέγεται ο CRT αφού μας παρέχει το μικρότερο σφάλμα πρόβλεψης (βλ. Πίνακα 5.2).

Εν κατακλείδι, πρέπει να αναφέρουμε ότι τα παραπάνω βέλτιστα μοντέλα που υπολογίστηκαν για τα προβλήματα ταξινόμησης δεν ταυτίζονται με τα τελικά αποτελέσματα του διαγωνισμού KDD Cup 2009, καθώς στα πλαίσια του διαγωνισμού χρησιμοποιήθηκαν όλες οι διαθέσιμες μέθοδοι για data mining, εκ των οποίων κάποιες προσέφεραν καλύτερα αποτελέσματα από αυτά που παρουσιάσαμε στην μελέτη μας. Για περισσότερες πληροφορίες επί των τελικών αποτελεσμάτων του διαγωνισμού για τα σύνολα εκπαίδευσης και εξέτασης, παραπέμπουμε τον αναγνώστη στις εργασίες των Lemaire et al. (2009) και Nicolescu - Mizil et al. (2009).

Βιβλιογραφία

[1] R. Berk., Statistical Learning from a Regression Perspective, Springer. 2008.

[2] De Ville, B., Suen, E. and Biggs, D., A method for choosing multiway partitions for classication and decision trees, Journal Of Applied Statistics, 1991.

[3] C.M. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, Oxford, 1996.

[4] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[5] Friedman J.H. Olshen R.A. Breiman, L. and C.J. Stone, Classication and regression trees, Chapman & Hall, 1984.

[6] Buntine W.L & Weigend, A.S., Computing Second Derivatives in Feed - Forward Networks: A review, Neural Networks, IEEE Transactions on , 1994.

[7] Carpenter,S. & Grossberg, G.A., The Handbook of Brain Theory and Neural Networks, The MIT Press, 1998.

[8] Vapnik V. N. & Cortes, C., Support vector networks, Machine Leaming, 20, 273-297, 1995.

[9] Cristianini, N. & Shawe-Taylor,J., An Introduction to SVMs and Other Kernel-Based Learning Methods, Cambridge University Press, 2000.

[10] Doyle,P., The use of automatic intreraction detector and similar research procedures, Operarional Research Quarterly, 1973.

[11] Duda, R.O. & Hart,P.E., Pattern Classication and Scene Analysis, Wiley, 1973.

[12] Falhman S.E & Lebiere, C., The Cascade Correlation Learning Architecture in Advances in Neural Information Processing Systems, Morgan-Kaufmann, 1990.

[13] Fawcett,T. , ROC graphs: Notes and practical considerations for data mining researchers, Intelligent Enterprise Technologies Laboratory,HP Laboratories Palo Alto, 2003.

[14] Gallant S.I., Neural Network Expert Systems, MIT Press, 1993.

[15] Gallant S.I., Perceptron-based learning algorithms, Neural Networks, IEEE Transactions on, 1990.

[16] Gaudel, R. & Cornuejols, A., Combining feature ranking methods for high dimensional data analysis, 5th Workshop on Statistical Methods for Post-Genomic Data, 2007.

[17] Jones M., Giroso, F. and Poggio,T. , Regularization theory and neural networks architectures, MIT Press, 1995.

[18] Cataltepe Z., Gulgezen, G. and Yu, L., Stable and accurate feature selection. Master's thesis, Istanbul Technical University, Computer Engineering Department, 2009.

[19] Hand, D.J. , Data mining: Statistics and more?, The American Statistician, Vol. 52, No. 2, 1998.

[20] Mannila H., Hand, D. and Smyth,P. , Principles of Data Mining, The MIT Press, 2001.

[21] Hassibi B., & Stork, D. , Second order derivatives for network pruning: Optimal Brain Surgeon, Advances In Neural Information Processing Systems, Vol. 5, 1993.

[22] Hassibi B., Stork, D., Wolf, G. and Watanabe, T., Optimal Brain Surgeon: Extensions, streamlining and performance comparisons, Advances In Neural Information Processing Systems, Vol. 6, 1994.

[23] Tibshirani R., Hastie, T. and Friedman, J., The Elements Of Statistical Learning: Data Mining, Inference And Prediction, Springer, 2001.

[24] Herbrich, R., Learning Kernel Classifiers: Theory and Algorithms, 2002.

[25] Mao J. and Jain, A . K., Articial neural networks: A tutorial. IEEE, 1996.

[26] Johnstone, I.M. &Titterington D.M., Statistical challenges of high - dimensional data, PMC, 2012.

[27] Karatzoglou, A., Meyer D. and Hornik,K. , Support vector machines in r, Journal of Statistical Software, 15(9), 1-28, 2006.

[28] Kass,G.V. , An exploratory technique for investigating large quantities of categorical data, Royal Statistical Society, 1980.

[29] Kim, H. & Loh, W.Y. , Classication trees with unbiased multiway splits, Journal Of the American Statistical Association, 2001.

[30] Lebrun, G., Lezoray O., Charrier C. & Cardot H., A New Model Selection Method for SVM, Springer-Verlag, 2006.

[31] LeCun Y., Denker, J.S., Solla, S., Howard, R.E. and Jackel, L.D., Optimal Brain Damage, Advances In Neural Information Processing Systems, (NIPS 1989).

[32] Lim, T.J.S & Loh,W.Y. , A comparison of prediction accuracy,complexity, and training time of thirty-three old and new classication algorithms.,Machine Learning, 40, 203-229, 2000.

[33] Loh,W.Y., Encyclopedia of Statistics in Quality and Reliability, chapter Classication and Regression Tree Methods, page 315-323, Ruggeri, Kenett, Faltin, Wiley, 2008.

[34] Loh, W.Y. & Shih,Y.S., Split selection methods for classication trees, Statistica Sinica, 1997.

[35] Maroco, J.,Silva D., Rodrigues A., Guerreiro M., Santana I. and De Mendonca,A., Data mining methods in the prediction of dementia: A real-data comparison of the accuracy, sensitivity and specicity of linear discriminant analysis, logistic regression, neural networks, support vector machines, classification trees and random forests, BMC Research Notes, 2011.

[36] McCullogh,W.H. and Pitts,W.S., A logical calculus of the ideas immanent in nervous activity, Bulletin of Mathematical Biophysics, Vol.5, 1943.

[37] Miller, H., Clarke S., Lane S., Lonie A., Lazaridis D., Petrovski S. and Jones,O., Predicting customer behaviour: The university of melbourne's KDD Cup report. the 2009 knowledge discovery in data competition (KDD cup 2009), Challenges in Machine Learning, Volume 3, 2009.

[38] Miller,H.R., Statistical Methods for the Analysis of High-Dimensional Data. PhD thesis, Department of Mathematics and Statistics,The University of Melbourne, 2010.

[39] Novakovic, J., Strbac P. and Bulatovic,D., Toward optimal feature selection using ranking methods and classication algorithms, Yugoslav Journal of Operations Research, 2011.

[40] Ou, Y.Y., Chen, C.Y., Hwang, S.C. & Oyang, Y.J., Expediting model for support vector machines based on data reduction, IEEE International Conference on Systems, Man and Cybernetics, 2003.

[41] Quinlan,J.R., Induction of decision trees, Kluwer Academic Publishers, 1986.

[42] Ripley, B.D., Pattern Recognition and Neural Networks, Cambridge University Press, 1996.

[43] Ripley, B.D., Statistical Data Mining, Springer, 2002.

[44] Rosenblatt,F., The perceptron - a perceiving and recognizing automation, Technical report, Cornell Aeronautical Laboratory, 1957.

[45] Rumelhart, D.E., Hinton G.E. and Williams,R.J., Learning internal representations by error propagation. In Rumelhart, D.E. and McClelland, J.L., editors, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations, The MIT Press, 1986.

[46] Singh, K. & Xie,M., Bootstrap: A statistical method, International Encyclopedia of Education, 2010.

[47] Smola, A.J. and Scholkopf,B., Learning with Kernels, SVMs,Regularization,Optimization and Beyond (Adaptive Computation and Machine Learning), The MIT Press, 2001.

[48] Sun, Y., Todorovic S. and Goodison,S. , Local learning based feature selection for high dimensional data analysis, Pattern Analysis and Machine Intelligence, IEEE Transactions on (Volume:32 , Issue: 9), 2009 .

[49] Tan, P.N., Steinbach M. and Kumar, V., Introduction To Data MIning, Pearson, 2006.

[50] Tian,T.S., Dimensionality Reduction For Classication With High- Dimensional Data. PhD thesis, Faculty Of The Graduate School University Of Southern California, 2009.

[51] Vapnik,V.N. , The Nature of Statistical Learning Theory, Springer-Verlag, 1995.

[52] Vapnik,V.N., Statistical Learning Theory, Wiley-Interscience, 1998.

[53] Widrow, B. & Ho, M., Adaptive switching circuits, IRE WESCON Convention record, Vol. 4, 1960.

[54] Witten, I.H. & Frank,E. , Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, 2005.

[55] Yu, L. & Liu, H., Feature selection for high-dimensional data: A fast correlation - based lter solution, Department of Computer Science & Engineering, Arizona State University, 2003.

[56] Zhu, Y., Li, C. and Zhang,Y., A practical parameters selection method for svm, ISNN 2004, 2004.