ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΟΜΕΑΣ ΜΑΘΗΜΑΤΙΚΩΝ

ΣΤΑΤΙΣΤΙΚΗ ΤΑΞΙΝΟΜΗΣΗ ΚΑΙ ΕΦΑΡΜΟΓΕΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ευθυμιάδης Νικόλαος

Επιβλέπων: Δημήτριος Φουσκάκης

Περιεχόμενα

1. Εισαγωγή

1.1. Γενικά

Με τη **στατιστική ταξινόμηση** ή **κατηγοριοποίηση** (statistical classification) ασχολούνται οι κλάδοι της **Μηχανικής Μάθησης** (Machine Learning), της **Εξόρυξης Δεδομένων** (Data Mining) και της **Αναγνώρισης Προτύπων** (Pattern Recognition). Ο πρώτος όρος –Μηχανική Μάθηση- έχει επικρατήσει στους κύκλους των επιστημόνων υπολογιστών και έχει άμεση σχέση με την τεχνητή νοημοσύνη. Τμήμα της μηχανικής μάθησης είναι και η στατιστική ταξινόμηση η οποία συγκαταλέγεται στην **επιβλεπόμενη μάθηση** (Supervised Learning) ή **μάθηση με δάσκαλο**, στον κλάδο δηλαδή που ασχολείται με τη δημιουργία μοντέλων εισόδου-εξόδου που **μαθαίνουν** (learn) μέσω παραδειγμάτων που έχουν και είσοδο και έξοδο (Sammut & Webb, 2011). Τα παραδείγματα της εξόδου θεωρούνται ο δάσκαλος, μιας και υποδεικνύουν τη "σωστή απάντηση" για δεδομένη είσοδο. Ο δεύτερος όρος –Εξόρυξη Δεδομένων- προέρχεται από τη στατιστική και τμήμα του κλάδου αυτού είναι η στατιστική ταξινόμηση, που επικρατεί και με τον όρο **Διαχωριστική Ανάλυση** (Discriminant Analysis). Εδώ για τα μοντέλα, αντί για τη λέξη "μαθαίνουν" χρησιμοποιείται η λέξη "**εκτιμούν**" (estimate). Ο όρος "**προσαρμόζονται**" (fitted) παρατηρείται και στους δύο κλάδους. Οι δύο αυτοί κλάδοι έχουν κοινή θεωρία αλλά υπονοούν διαφορετική διαχείριση των μεθόδων τους. Στην μηχανική μάθηση, υπονοείται ότι χρησιμοποιούνται οι μέθοδοι για τη δημιουργία αλγορίθμων προς χρήση, όπως για παράδειγμα τα φίλτρα ανεπιθύμητης ηλεκτρονικής αλληλογραφίας. Στην εξόρυξη δεδομένων υπονοείται η χρήση των μεθόδων για την παραγωγή γνώσης μέσω των δεδομένων, δηλαδή πολλές φορές ο στόχος της εξόρυξης δεδομένων μπορεί να είναι μία αναφορά. Αν και κάποιες διαδικασίες είναι πιο παραδοσιακές σε κάποιον από τους δύο κλάδους, κατά τα άλλα οι δύο αυτοί κλάδοι είναι πανομοιότυποι. Ο τρίτος όρος –Αναγνώριση Προτύπων- έχει επικρατήσει στους μηχανικούς και ταυτίζεται με τη στατιστική ταξινόμηση. Υπονοεί κυρίως την παραγωγή γνώσης μέσω πειραμάτων με τη χρήση πειραματικών διατάξεων ή την κατηγοριοποίηση στην όραση υπολογιστών. Στην αναγνώριση προτύπων τον στοχαστικό μη προβλέψιμο παράγοντα ενός προβλήματος συνηθίζουν να τον αποκαλούν **υπόβαθρο** (background) ενώ στη μηχανική μάθηση και στην εξόρυξη δεδομένων επικρατεί ο όρος "**θόρυβος**" (noise). Η περιοχή που μελετά τις στατιστικές ιδιότητες των παραπάνω κλάδων ονομάζεται **Στατιστική Μάθηση** (Statistical Learning). Με βάση τα παραπάνω θα μπορούσε κανείς να δει τους κλάδους αυτούς ως έναν ενιαίο διεπιστημονικό κλάδο και οι διαφορές στις κουλτούρες των επιμέρους ερευνητών γίνονται αμέσως εμφανείς:

"Αντιλαμβανόμαστε τις ανεξήγητες διακυμάνσεις σαν να έχουν δημιουργηθεί από κάποια "πραγματική" ή "υποθάλπτουσα" κατανομή. Σε μια **παραμετρική** ανάλυση υποθέτουμε ότι οι κατανομές αυτές είναι άγνωστα μέλη μιας γνωστής οικογένειας κατανομών"

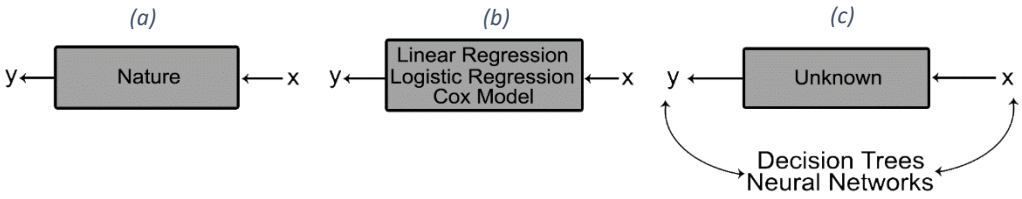
Chris Wild (2006) "The Concept of Distribution" Statistics Education Research Journal σ. 22

"[…] Έχει το μοντέλο σταθερό αριθμό παραμέτρων ή ο αριθμός των παραμέτρων αυξάνεται μαζί με τον αριθμό των δεδομένων; Το πρώτο καλείται **παραμετρικό μοντέλο** και το δεύτερο **μη παραμετρικό μοντέλο**"

Kevin P. Murphy (2012) "Machine Learning: A Probabilistic Perspective" The MIT Press σ. 16

Οι παραπάνω ιδέες για το τι είναι παραμετρικό και τι όχι δεν διαφέρουν μόνο σε επιφανειακό/περιγραφικό επίπεδο αλλά στη φιλοσοφία των κλάδων. Διαφέρουν στο κατά πόσο δηλαδή ο κανόνας που θα φτιαχτεί για να προβλεφθεί ένα φαινόμενο περιγράφει ή πρέπει να περιγράφει τον μηχανισμό δημιουργίας του φαινομένου αυτού ή αν αρκεί η απλή πρόβλεψή του φαινομένου από έναν μη εποπτικό μηχανισμό.

Ο Leo Breiman (Breiman, 2001) διαχωρίζει την επιστημονική κοινότητα σε δύο κουλτούρες με κριτήριο το πώς προσεγγίζουν το γενικότερο πρόβλημα της μοντελοποίησης. Ας σκεφτούμε ότι τα δεδομένα παράγονται μέσα σε ένα μαύρο κουτί στο οποίο μπαίνουν κάποιες τιμές $\boldsymbol{x}$ κάποιων μεταβλητών $\boldsymbol{X}$ από τη μία μεριά και βγαίνουν κάποιες τιμές $\boldsymbol{y}$ μιας μεταβλητής $\boldsymbol{Y}$ από την άλλη. Μέσα στο μαύρο κουτί η φύση δρα με τρόπο που αντιστοιχίζει τα εισαγόμενα $\boldsymbol{x}$ με κάποιο $\boldsymbol{y}$. Στόχος μας σε κάθε περίπτωση είναι είτε να προβλέψουμε τα $\boldsymbol{y}$ δεδομένων των $\boldsymbol{x}$, είτε να εξάγουμε κάποια πληροφορία για τη φύση της αντιστοίχισης των $\boldsymbol{X}, \boldsymbol{Y}$. Η πρώτη κουλτούρα σύμφωνα με τον Breiman θεωρεί ένα στοχαστικό μοντέλο μέσα στο μαύρο κουτί, δηλαδή εδώ το μοντέλο και ο μηχανισμός της φύσης ταυτίζονται. Οι τιμές των παραμέτρων του μοντέλου εκτιμώνται μέσω των δεδομένων και έπειτα το μοντέλο χρησιμοποιείται είτε για πρόβλεψη είτε για μελέτη του φαινομένου. Την κουλτούρα αυτή την αποκαλεί **κουλτούρα στατιστικής μοντελοποίησης** (Data Modeling Culture). Η δεύτερη κουλτούρα από την άλλη θεωρεί το περιεχόμενο του μαύρου κουτιού περίπλοκο και άγνωστο. Η προσέγγισή τους είναι να βρουν μια συνάρτηση ή έναν αλγόριθμο που δρα πάνω στα $\boldsymbol{x}$ για να προβλεφθούν τα $\boldsymbol{y}$. Η συνάρτηση αυτή δεν θεωρείται ότι ταυτίζεται με τον φυσικό μηχανισμό του φαινομένου. Την κουλτούρα αυτή την αποκαλεί **κουλτούρα αλγοριθμικής μοντελοποίησης** (Algorithmic Modeling Culture). Οι προσεγγίσεις αυτές φαίνονται στο Διάγραμμα 1.1.



Διάγραμμα 1.1: Οι δύο κουλτούρες στη μοντελοποίηση σύμφωνα με τον Breiman. Στο (a) φαίνεται η διαδικασία παραγωγής των y από τα x μέσω του μαύρου κουτιού που ορίζει η φύση. Στο (b) φαίνεται η στατιστική κουλτούρα της μοντελοποίησης η οποία θεωρεί ότι μέσα στο μαύρο κουτί υπάρχει ένα στοχαστικό μοντέλο και καλείται να το εκτιμήσει. Στο (c) φαίνεται η αλγοριθμική κουλτούρα που θεωρεί το μαύρο κουτί άγνωστο και καλείται να βρει αλγοριθμικούς τρόπους για να προβλέψει τα y μέσω των x χωρίς να θεωρεί ότι η φύση χρησιμοποιεί τις ίδιες διαδικασίες.

Ο Breiman εκτιμά ότι στην επιστημονική κοινότητα των στατιστικών μόνο το 2% ανήκει στην δεύτερη κατηγορία, ενώ άλλοι κλάδοι τη δέχονται σε πολύ μεγαλύτερο ποσοστό. Μεθόδους της πρώτης προσέγγισης θα αναλύσουμε στο Κεφάλαιο 2 ενώ μεθόδους της δεύτερης προσέγγισης στο Κεφάλαιο 3.

Διχασμοί υπάρχουν επιπλέων και μέσα στην κουλτούρα της στατιστικής μοντελοποίησης. Συγκεκριμένα το παράδειγμα της στατιστικής ταξινόμησης προσφέρει μια καλή ευκαιρία στο να αναπτυχθούν οι διαφορές της κλασικής και της Μπεϋζιανής στατιστικής. Τα βασικότερα ερωτήματα που δημιουργούνται είναι τα παρακάτω:

- Σε ένα στατιστικό μοντέλο πρόβλεψης, οι παράμετροι είναι κάποιες άγνωστες σταθερές ή μπορούν να θεωρηθούν τυχαίες μεταβλητές;
- Είναι εν γένει αποδεκτό να χρησιμοποιηθεί κάποια εκ των προτέρων γνώση για το προς μελέτη φαινόμενο ή πρέπει να βασιστούμε μόνο στο δείγμα μας; Τι γίνεται όταν το φαινόμενο αυτό έχει μελετηθεί πολλές φορές στο παρελθόν και υπάρχει εμπειρία;
- Μας ενδιαφέρει περισσότερο μια αμερόληπτη προσέγγιση η μια που να συγκλίνει γρηγορότερα; Αν προτιμήσουμε την αμεροληψία, τι θα κάνουμε στην περίπτωση που έχουμε λίγα δεδομένα;

Ασυμπτωτικά η κλασική και η Μπεϋζιανή προσέγγιση συμφωνούν αλλά στην πρακτική της ανάλυσης δεδομένων κάτι τέτοιο δεν είναι αρκετό μιας και πολλές φορές τα δεδομένα δεν είναι όσα θα θέλαμε. Επίσης δεν είναι σαφή τα όρια μεγέθους δείγματος ώστε να θεωρηθεί ότι από κάποιο μέγεθος N και πάνω οι ασυμπτωτικές ιδιότητες ισχύουν και έτσι έχουμε ταύτιση μεθόδων. Η Μπεϋζιανή προσέγγιση θα παρουσιαστεί στην Ενότητα 2.4.

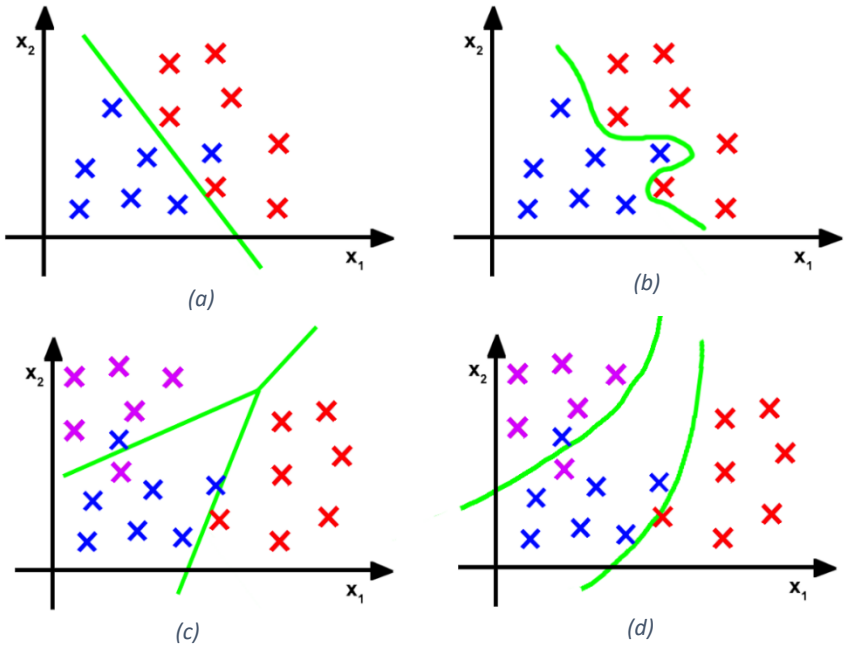
1.2. Το Πρόβλημα της Στατιστικής Ταξινόμησης

Έστω ότι μας δίνεται ένα **σύνολο από δεδομένα** (Dataset) ή **σύνολο εκπαίδευσης** (Training set) $\boldsymbol{D}$ με N παρατηρήσεις από τις μεταβλητές $X_1, X_2, \dots, X_d, Y$ με την προϋπόθεση ότι η Y είναι κατηγορική μεταβλητή, με τις τιμές $1, 2, \dots, C$. Μας δίνεται στη συνέχεια και μια ακόμα παρατήρηση $\boldsymbol{x}^* = (x_1^*, x_2^*, \dots, x_d^*)$. Σκοπός μας είναι να προβλέψουμε την τιμή της μεταβλητής Y, δηλαδή να κατηγοριοποιήσουμε τη νέα παρατήρηση σε μία από τις κατηγορίες των τιμών της Y. Ένα τέτοιο πρόβλημα καλείται πρόβλημα στατιστικής ταξινόμησης και η λύση του βασίζεται στη δημιουργία κάποιου μοντέλου πρόβλεψης μέσω του συνόλου δεδομένων $\boldsymbol{D}$. Το πρόβλημα της ταξινόμησης είναι ουσιαστικά η διακριτή περίπτωση του προβλήματος της παλινδρόμησης. Ένα τέτοιο μοντέλο μπορεί να έχει μαθηματική μορφή αλλά υπάρχουν περιπτώσεις που η πρόβλεψη γίνεται μέσω μιας αλγοριθμικής διαδικασίας όπως στην περίπτωση της διαδικασίας των k-πλησιέστερων γειτόνων (k-nearest neighbor) που θα δούμε παρακάτω. Τις μεταβλητές X_i θα τις καλούμε **ανεξάρτητες** ή **επεξηγηματικές** μεταβλητές και τη μεταβλητή Y **εξαρτημένη** ή μεταβλητή **απόκρισης**. Επειδή στη βιβλιογραφία πολλές φορές οι όροι "ταξινομητής" και "μοντέλο" συγχέονται, στην παρούσα εργασία θα θεωρούμε τον παρακάτω διαχωρισμό: **Μοντέλο** θα καλούμε ένα σύνολο από κανόνες με βάση τους οποίους μια παρατήρηση $\boldsymbol{x}^*$ ταξινομείται σε μία από της κατηγορίες της μεταβλητής Y. Τη διαδικασία με την οποία δημιουργείται το μοντέλο με βάση το σύνολο των δεδομένων $\boldsymbol{D}$ θα την καλούμε **ταξινομητή** (Classifier).

Στην περίπτωση που ο ταξινομητής δημιουργεί μοντέλα που εκφράζονται μαθηματικά, η μορφή του μοντέλου θα περιέχει τις επεξηγηματικές μεταβλητές και κάποιες άγνωστες παραμέτρους. Για τον υπολογισμό των παραμέτρων αυτών θα χρειαστεί βελτιστοποίηση (μεγιστοποίηση ή ελαχιστοποίηση) κάποιας συνάρτησης. Η συνάρτηση αυτή καλείται **αντικειμενική συνάρτηση** (objective function). Στην περίπτωση που ελαχιστοποιούμε, η συνάρτηση καλείται και **συνάρτηση κόστους** (cost function) ή **συνάρτηση απώλειας** (loss function). Στην κουλτούρα της μηχανικής μάθησης συνήθως χρησιμοποιούμε συναρτήσεις κόστους και για τον λόγο αυτό μια αντικειμενική συνάρτηση f τη μετασχηματίζουμε αν χρειαστεί σε συνάρτηση κόστους. Δηλαδή αντί να μεγιστοποιήσουμε την f ελαχιστοποιούμε

την $-f$. Για παράδειγμα σε ένα μοντέλο για να εκτιμήσουμε τις παραμέτρους του μπορούμε είτε να μεγιστοποιήσουμε τη συνάρτηση λογαριθμο-πιθανοφάνειας, είτε να ελαχιστοποιήσουμε την αρνητική συνάρτηση λογαριθμο-πιθανοφάνειας η οποία ταυτίζεται με τη **διεντροπία** (cross-entropy) της εμπειρικής κατανομής του συνόλου εκπαίδευσης και του μοντέλου.

Οι ανεξάρτητες μεταβλητές $X_1, X_2, \dots, X_d$ ορίζουν έναν υπερχώρο d διαστάσεων για τις παρατηρήσεις του συνόλου δεδομένων $\boldsymbol{D}$. Στην περίπτωση δύο κλάσεων, κάθε ταξινομητής δημιουργεί μια υπερεπιφάνεια που χωρίζει τον υπερχώρο έτσι ώστε κάθε παρατήρηση προς ταξινόμηση να ταξινομείται ανάλογα με το που βρίσκεται σε σχέση με την υπερεπιφάνεια. Η υπερεπιφάνεια αυτή καλείται **σύνορο απόφασης** (Decision Boundary). Ένας ταξινομητής που δημιουργεί μόνο υπερεπίπεδα για σύνορα απόφασης καλείται **γραμμικός ταξινομητής** (Linear Classifier). Οι έννοιες αυτές γενικεύονται στην περίπτωση παραπάνω κλάσεων. Αν οι κλάσεις είναι c, ο διαχωρισμός μπορεί να γίνει με c σύνορα απόφασης με το καθένα σύνορο να χωρίζει την κάθε κλάση από τις υπόλοιπες. Αυτή η διαδικασία καλείται **one versus all** ή **one versus rest** classification και δημιουργεί c μοντέλα. Το καθένα ταξινομεί την προς μελέτη τιμή σε μια κλάση με την αντίστοιχη πιθανότητα και επιλέγεται ως πρόβλεψη η κλάση με τη μέγιστη πιθανότητα. Παραδείγματα συνόρων απόφασης γραμμικών και μη γραμμικών ταξινομητών σε προβλήματα δύο ή παραπάνω κλάσεων φαίνονται στο Διάγραμμα 1.2.



Διάγραμμα 1.2: Στα (a) και (b) βλέπουμε ένα πρόβλημα δυο διαστάσεων με δύο κλάσεις: Το (a) αναπαριστά ένα σύνορο απόφασης κάποιου γραμμικού ταξινομητή και το (b) ένα σύνορο απόφασης κάποιου μη γραμμικού ταξινομητή. Στα (c) και (d) βλέπουμε ένα πρόβλημα δύο διαστάσεων με τρείς κλάσεις: Στο (c) έχουμε έναν one versus all γραμμικό ταξινομητή και στο (d) έναν one versus all μη γραμμικό ταξινομητή.

Ενδεικτικά αναφέρονται οι παρακάτω εφαρμογές της στατιστικής ταξινόμησης:

- Αν η μεταβλητή Y εκφράζει την κατάσταση ενός δείγματος ανθρώπινου ιστού και λαμβάνει τις τιμές 1 για καρκινικό ή 0 για φυσιολογικό, για κάποιο σύνολο

μεταβλητών X_j (πχ εντάσεις ανά συχνότητα μέσω φασματοσκοπίας Ράμαν) η λύση του προβλήματος ταξινόμησης οδηγεί στη δημιουργία μοντέλου διάγνωσης του καρκίνου.

- Αν η μεταβλητή Y εκφράζει τον τύπο ενός e-mail με την κωδικοποίηση 0 αν το e-mail είναι επιθυμητό και 1 αν είναι ανεπιθύμητο, με τη χρήση μεταβλητών X_j (που το καθένα μπορεί να είναι το πλήθος εμφάνισης μιας λέξης) η λύση του προβλήματος ταξινόμησης οδηγεί στη δημιουργία φίλτρου ανεπιθύμητης αλληλογραφίας.
- Αν η μεταβλητή Y εκφράζει το χειρότερο συναλλαγματικό στιγμιότυπο ενός πελάτη μιας τράπεζας ως προς την αποπλήρωση ενός δανείου- σε δεδομένο χρόνο- με την κωδικοποίηση 0 αν ο πελάτης δεν έχει καθυστερήσει δόση πάνω από τρείς συνεχόμενους μήνες και 1 αν ο πελάτης έχει καθυστερήσει δόση πάνω από τρεις συνεχόμενους μήνες, τότε με τη χρήση κάποιου συνόλου X_j συμπεριφορικών μεταβλητών (πχ σύνολο φορών που άργησε ένα μήνα) η λύση του προβλήματος ταξινόμησης οδηγεί στη δημιουργία μοντέλου πρόβλεψης ρίσκου.

Από το τελευταίο παράδειγμα παρατηρούμε ότι στην περίπτωση που οι μεταβλητές $X_1, X_2, \dots, X_d$ προηγούνται χρονικά από την μεταβλητή Y, δημιουργούμε μοντέλα που προβλέπουν το μέλλον.

1.3. Το δίπολο μεροληψίας-διακύμανσης

1.3.1. Το δίπολο μεροληψίας-διακύμανσης στην παλινδρόμηση

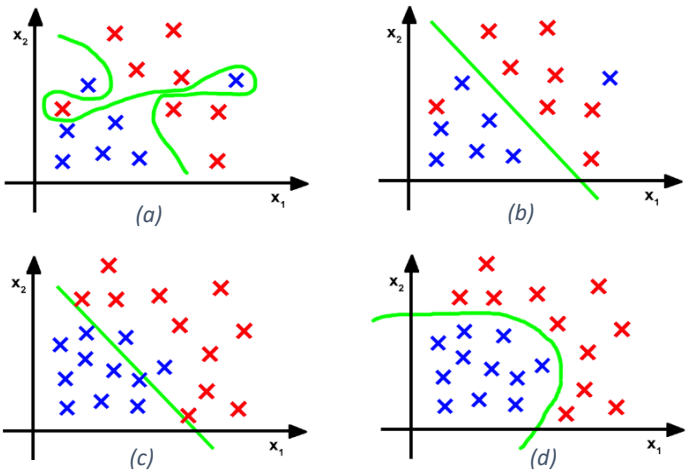
Το δίπολο μεροληψίας-διακύμανσης είναι ένα φαινόμενο που εμφανίζεται σε κάθε στατιστική μοντελοποίηση, γίνεται όμως ευκολότερα εμφανές στο πρόβλημα της παλινδρόμησης, έτσι για λόγους κατανόησης θα το δείξουμε αρχικά σε ένα τέτοιο πρόβλημα. Έστω ότι μελετάμε τη σχέση μεταξύ των μεταβλητών X, Y και έστω ότι υπάρχει μια συνάρτηση τέτοια ώστε $Y = f(X) + \varepsilon$, με ε τον θόρυβο. Θεωρούμε ότι το ε έχει μέση τιμή 0 και διακύμανση σ^2. Μας δίνεται ένα δείγμα από τις μεταβλητές X, Y και εμείς πρέπει να φτιάξουμε μια εκτιμήτρια $\hat{y} = \hat{f}$ της f. Θέλουμε το μέσο τετραγωνικό σφάλμα (MSE) να είναι το ελάχιστο τόσο στις παρατηρούμενες τιμές όσο και στις μελλοντικές. Προφανώς, λόγω του ε γνωρίζουμε ότι το μοντέλο μας δεν θα προσαρμόζεται τέλεια. Ας πάρουμε ένα νέο σημείο $\boldsymbol{x}^*$ με πραγματική μεταβλητή απόκρισης Y^*. Τότε η ποσότητα $E[(Y^* - \hat{f}(\boldsymbol{x}^*))^2]$ καλείται το **αναμενόμενο μέσο τετραγωνικό σφάλμα** στο σημείο $\boldsymbol{x}^*$ και διαισθητικά δείχνει το μέσο σφάλμα στο σημείο $\boldsymbol{x}^*$ μετά από τη δημιουργία πολλών ίδιων μοντέλων, δημιουργημένων από διαφορετικά δεδομένα $\boldsymbol{x}^*, \boldsymbol{y}^*$. Αναλύοντας την ποσότητα αυτή προκύπτει το παρακάτω αποτέλεσμα που καλείται **δίπολο μεροληψίας-διακύμανσης**.

$$
\begin{aligned}
E\left[(Y^* - \hat{f}(\boldsymbol{x}^*))^2\right] &= E\left[Y^{*2} + \hat{f}(\boldsymbol{x}^*)^2 - 2Y^*\hat{f}(\boldsymbol{x}^*)\right] = E\left[Y^{*2}\right] + E\left[\hat{f}(\boldsymbol{x}^*)^2\right] - E\left[2Y^*\hat{f}(\boldsymbol{x}^*)\right] \\
&= Var[Y^*] + E[Y^*]^2 + Var\left[\hat{f}(\boldsymbol{x}^*)\right] + E\left[\hat{f}(\boldsymbol{x}^*)\right]^2 - E\left[2\hat{f}(\boldsymbol{x}^*)(f(\boldsymbol{x}^*) + \varepsilon)\right] \\
&= Var[Y^*] + E[f(\boldsymbol{x}^*) + \varepsilon]^2 + Var\left[\hat{f}(\boldsymbol{x}^*)\right] + E\left[\hat{f}(\boldsymbol{x}^*)\right]^2 - 2f(\boldsymbol{x}^*)E\left[\hat{f}(\boldsymbol{x}^*)\right] \\
&= Var[Y^*] + f(\boldsymbol{x}^*)^2 + Var\left[\hat{f}(\boldsymbol{x}^*)\right] + E\left[\hat{f}(\boldsymbol{x}^*)\right]^2 - 2f(\boldsymbol{x}^*)E\left[\hat{f}(\boldsymbol{x}^*)\right] \\
&= Var[Y^*] + Var\left[\hat{f}(\boldsymbol{x}^*)\right] + \left(f(\boldsymbol{x}^*)^2 - 2f(\boldsymbol{x}^*)E\left[\hat{f}(\boldsymbol{x}^*)\right] + E\left[\hat{f}(\boldsymbol{x}^*)\right]^2\right) \\
&= Var[Y^*] + Var\left[\hat{f}(\boldsymbol{x}^*)\right] + \left(f(\boldsymbol{x}^*) - E\left[\hat{f}(\boldsymbol{x}^*)\right]\right)^2 \\
&= Var[Y^*] + Var\left[\hat{f}(\boldsymbol{x}^*)\right] + E\left[f(\boldsymbol{x}^*) - \hat{f}(\boldsymbol{x}^*)\right]^2 \\
&= \sigma^2 + Var\left[\hat{f}(\boldsymbol{x}^*)\right] + Bias\left[\hat{f}(\boldsymbol{x}^*)\right]^2,
\end{aligned}
$$

γιατί

$$
\begin{aligned}
Var[Y^*] = E[(Y^* - E[Y^*])^2] &= E\left[\left(Y^* - f(\boldsymbol{x}^*)\right)^2\right] = E\left[\left(f(\boldsymbol{x}^*) + \varepsilon - f(\boldsymbol{x}^*)\right)^2\right] = E[\varepsilon^2] \\
&= Var[\varepsilon] + E[\varepsilon]^2 = \sigma^2.
\end{aligned}
$$

Βλέπουμε δηλαδή ότι για να ελαχιστοποιήσουμε το σφάλμα πρέπει να διαλέξουμε μία μέθοδο που να πετυχαίνει ταυτόχρονα και χαμηλή διακύμανση και χαμηλή μεροληψία. Προφανώς για τον παράγοντα σ^2 δεν μπορούμε να κάνουμε κάτι μιας και υπάρχει λόγω της φύσης του προβλήματος. Το **σφάλμα λόγω διακύμανσης** σε ένα μοντέλο δείχνει το κατά πόσο το μοντέλο μεταβάλλεται εάν το προσαρμόσουμε με διαφορετικά δεδομένα, το κατά πόσο δηλαδή το μοντέλο εξηγεί τα δεδομένα και όχι το φαινόμενο. Ένα μοντέλο που εξηγεί καλά τα δεδομένα από τα οποία προσαρμόστηκε αλλά δεν εξηγεί καλά καινούργια δεδομένα πάσχει από **υπερπροσαρμογή** (over-fitting). Το **Σφάλμα λόγω μεροληψίας** σε ένα μοντέλο οφείλεται σε λάθος υποθέσεις για τη φύση του προβλήματος. Συνήθως παρουσιάζεται όταν σε ένα πολύπλοκο πρόβλημα γίνεται μια υπεραπλουστευτική παραδοχή. Αν για παράδειγμα δύο μεταβλητές έχουν τετραγωνική σχέση και προσαρμόσουμε ένα απλό γραμμικό μοντέλο, το μοντέλο μας θα έχει υψηλό σφάλμα λόγω μεροληψίας και λέμε ότι πάσχει από **υποπροσαρμογή** (under-fitting). Στο Διάγραμμα 1.3 φαίνονται παραδείγματα υποπροσαρμοσμένων και υπερπροσαρμοσμένων μοντέλων.



Διάγραμμα 1.3: Τα διαγράμματα αποτελούν παράδειγμα του προβλήματος της ταξινόμησης σε δύο κατηγορίες και με δύο μεταβλητές. Στο (a) βλέπουμε μια εμφανή περίπτωση υπερπσαρμοσμένου μοντέλου στα δεδομένα. Στο (b) παρατηρούμε ότι ένας γραμμικός διαχωρισμός κατηγοριοποιεί πολύ καλά τις παρατηρήσεις και φαίνεται λογικότερος. Στο (c) βλέπουμε ένα γραμμικό υποπροσαρμοσμένο μοντέλο. Στο (d) παρατηρούμε με πόσο απλό τρόπο θα μπορούσαμε να διαχωρίσουμε τα δεδομένα προσαρμόζοντας ένα μη γραμμικό μοντέλο.

Τη σχέση της μεροληψίας-διακύμανσης την λέμε δίπολο γιατί είναι εύκολο να φτιάξουμε ένα μοντέλο με χαμηλή διακύμανση και υψηλή μεροληψία και το αντίστροφο. Σε πολλές περιπτώσεις η προσπάθειά μας να ρίξουμε τον έναν παράγοντα σφάλματος ανεβάζει αυτόματα τον άλλο.

Για να αντιμετωπίσουμε το παραπάνω πρόβλημα έχουμε στη διάθεσή μας δύο προσεγγίσεις. Η πρώτη είναι μέσα από γραφήματα και ελέγχους να αντιληφθούμε την φύση του φαινομένου που μελετάμε. Τη φύση της συνάρτησης f στο πρόβλημα της παλινδρόμησης ή τη φύση του συνόρου απόφασης στο πρόβλημα της ταξινόμησης. Στη συνέχεια θεωρούμε ως λύση τη μορφή που αντιληφθήκαμε. Αυτό μας εξασφαλίζει, για παράδειγμα, ότι αν θεωρήσουμε σε ένα πρόβλημα παλινδρόμησης την f να είναι ένα δεύτερης τάξης πολυώνυμο, η διαδικασία βελτιστοποίησης μέσω της συνάρτησης κόστους δε θα μπορεί να δημιουργήσει μια μορφή τρίτης τάξης πολυωνύμου για την f. Αυτή η προσέγγιση βέβαια δεν είναι πάντα εφικτή. Ειδικά όταν οι διαδικασίες αρχίζουν να γίνονται πολύπλοκες ή όταν οι διαστάσεις του προβλήματος είναι πολλές δεν είμαστε σε θέση πάντα να θεωρήσουμε με σιγουριά τη μορφή του μοντέλου. Χρειαζόμαστε μια διαδικασία λοιπόν που να χειρίζεται την μορφή του μοντέλου με αυτόματο τρόπο.

Έστω ένα μοντέλο $\hat{f}(\boldsymbol{x_i})$ που προσπαθεί να προβλέψει την πραγματική τιμή y_i της μεταβλητές απόκρισης Y. Έστω και μια συνάρτηση κόστους προς ελαχιστοποίηση $\min_f \sum_{i=1}^{N} C\left(\hat{f}(\boldsymbol{x_i}), y_i\right)$, δίνουμε επίσης στο μοντέλο αρκετές ελευθερίες ως προς τη μορφή του, με κίνδυνο να δημιουργηθεί πρόβλημα υπερπροσαρμογής. Αν ορίσουμε μια συνάρτηση $R\left(\hat{f}\right)$ που να ποινικοποιεί τις "πολύπλοκες" f και να επιβραβεύει τις "απλές" f, μπορούμε τότε αντί για την ελαχιστοποίηση της συνάρτησης κόστους να ελαχιστοποιήσουμε την ποσότητα:

$$\min_f \sum_{i=1}^{n} C\left(\hat{f}(\boldsymbol{x_i}), y_i\right) + \lambda R\left(\hat{f}\right),$$

για κάποια παράμετρο λ. Η παραπάνω διαδικασία καλείται **ομαλοποίηση** (Regularization) και είναι μια διαδικασία που αναζητά ταυτόχρονα δυο επιθυμητά χαρακτηριστικά του μοντέλου, το μικρό σφάλμα και την απλότητά του. Η ομαλοποίηση μπορεί να θεωρηθεί και ως μια προσπάθεια να εφαρμοστεί η αρχή του **ξυραφιού του Occam** (Occam's razor) στο μοντέλο (Sammut & Webb, 2011, p. 736).

1.3.2. Η ενοποιημένη θεωρία μεροληψίας-διακύμανσης στην ταξινόμηση

Πάμε τώρα να επικεντρωθούμε στα μοντέλα ταξινόμησης. Για να το κάνουμε αυτό θα χρησιμοποιήσουμε μια ενοποιημένη θεωρία για την παραγοντοποίηση μεροληψίας-διακύμανσης (Domingos, 2000):

Θεωρούμε πάλι ένα σύνολο **D** από δεδομένα των μεταβλητών $X_1, X_2, \dots, X_d, Y$. Ένας ταξινομητής θα δημιουργήσει το μοντέλο $\hat{f}$ και για μία παρατήρηση $\boldsymbol{x}$ θα παραγάγει την πρόβλεψη $\hat{y} = \hat{f}(\boldsymbol{x})$. Έστω y η πραγματική τιμή της αντίστοιχης παρατήρησης. Πρακτικά, μια συνάρτηση κόστους είναι μια συνάρτηση $C(y, \hat{y})$ που μετράει το κόστος της πρόβλεψης $\hat{y}$ όταν η πραγματική τιμή της Y είναι y. Γενικά οι συναρτήσεις κόστους "τιμωρούν" τις κακές

προβλέψεις με μεγαλύτερα νούμερα, έτσι όσο καλύτερη είναι η πρόβλεψη τόσο μικρότερο αποτέλεσμα δίνει η συνάρτηση κόστους. Το γνωστό από τη γραμμική παλινδρόμηση **τετραγωνικό σφάλμα** $C(y,\hat{y}) = (y-\hat{y})^2$ είναι μια συνάρτηση κόστους. Για το πρόβλημα της ταξινόμησης μια πιο χρήσιμη συνάρτηση κόστους είναι η **μηδέν-ένα** (zero-one) συνάρτηση κόστους με τύπο:

$$C(y,\hat{y}) = \begin{cases} 0 \; \alpha\nu \; y = \hat{y} \\ 1 \; \alpha\nu \; y \neq \hat{y}. \end{cases}$$

Στόχος είναι να δημιουργηθεί ένα μοντέλο με το ελάχιστο κόστος, δηλαδή να ελαχιστοποιείται το μέσο $C(y,\hat{y})$ για όλα τα $\boldsymbol{x}$.

Η **βέλτιστη πρόβλεψη** $\hat{y}^*$ για κάποιο $\boldsymbol{x}$ είναι η πρόβλεψη που ελαχιστοποιεί το $E_Y[C(y,\hat{y}^*)]$ και ο δείκτης Y δηλώνει ότι η μέση τιμή είναι ως προς την τυχαία μεταβλητή $Y|\boldsymbol{X} = \boldsymbol{x}$.

Βέλτιστο μοντέλο είναι αυτό για το οποίο ισχύει $\hat{f}(\boldsymbol{x}) = \hat{y}^*$ για κάθε $\boldsymbol{x}$. Προφανώς για διαφορετικό σύνολο δεδομένων ο ίδιος ταξινομητής θα παραγάγει διαφορετικό μοντέλο οπότε η συνάρτηση κόστους $C(y,\hat{y})$ είναι συνάρτηση του συνόλου δεδομένων. Για να καταπολεμηθεί αυτό θα χρησιμοποιήσουμε τη μέση τιμή πάνω σε όλα τα σύνολα δεδομένων όπου χρειαστεί. Έστω $\mathbf{D}^*$ το σύνολο όλων των πιθανών δεδομένων. Τότε μπορεί να οριστεί το **αναμενόμενο κόστος** (expected cost) ως $E_{\mathbf{D}^*,Y}[C(y,\hat{y})]$. Ορίζουμε ως **βασική πρόβλεψη** (main prediction) για μια συνάρτηση κόστους C το:

$$\hat{y}_m = arg\min_{\hat{y}'} E_{\mathbf{D}^*}[C(\hat{y},\hat{y}')].$$

Αν δηλαδή φτιάξουμε το πολυσύνολο $\boldsymbol{Y}$ στο οποίο περιέχονται όλες οι προβλέψεις του ταξινομητή για καθένα από τα σύνολα δεδομένων του $\mathbf{D}^*$ (δηλαδή μια συγκεκριμένη πρόβλεψη y παρουσιάζεται πάνω από μια φορά στο $\boldsymbol{Y}$), η βασική πρόβλεψη θα είναι το $\hat{y}'$ για το οποίο το μέσο κόστος σε σχέση με όλα τα στοιχεία του $\boldsymbol{Y}$ είναι το ελάχιστο. Η βασική πρόβλεψη του τετραγωνικού σφάλματος είναι η μέση τιμή του $\boldsymbol{Y}$ και η βασική πρόβλεψη της μηδέν-ένα συνάρτησης κόστους είναι η πιο συχνή πρόβλεψη. Για παράδειγμα έστω ότι υπάρχουν k δυνατά σύνολα δεδομένων, δεδομένου μεγέθους και έστω ότι φτιάχνουμε έναν ταξινομητή για το καθένα. Αν $0.6k$ προβλέπουν την κλάση 1 και $0.4k$ την κλάση 0, τότε η βασική πρόβλεψη υπό την μηδέν-ένα συνάρτηση κόστους είναι η κλάση 1. Η βασική πρόβλεψη δεν χρειάζεται να είναι αναγκαστικά μέλος του $\boldsymbol{Y}$. Για παράδειγμα αν το $\boldsymbol{Y}$ είναι το $\{1,1,2,2\}$ η βασική πρόβλεψη υπό την τετραγωνική συνάρτηση κόστους είναι 1.5. Ορίζονται οι παρακάτω έννοιες:

Ορισμός: Μεροληψία ενός ταξινομητή (Bias) για μία παρατήρηση $\boldsymbol{x}$ καλείται η ποσότητα $B(\boldsymbol{x}) = C(\hat{y}^*,\hat{y}_m)$.

Δηλαδή η μεροληψία είναι το κόστος της βασικής πρόβλεψης σε σχέση με τη βέλτιστη πρόβλεψη.

Ορισμός: Διακύμανση ενός ταξινομητή (Variance) για μια παρατήρηση $\boldsymbol{x}$ καλείται η ποσότητα $V(\boldsymbol{x}) = E_{\mathbf{D}^*}[C(\hat{y}_m,\hat{y})]$.

Δηλαδή είναι το μέσο κόστος των προβλέψεων σε σχέση με τη βασική πρόβλεψη.

Ορισμός: Θόρυβος (Noise) μιας παρατήρησης $\boldsymbol{x}$ καλείται η ποσότητα $N(\boldsymbol{x}) = E_y[C(y,\hat{y}^*)]$.

Δηλαδή ο θόρυβος είναι ανεξάρτητος του ταξινομητή. Για τη μεροληψία και τη διακύμανση μπορούν να χρησιμοποιηθούν οι μέσες τιμές τους πάνω σε όλες τις παρατηρήσεις. Γράφουμε $E_{\boldsymbol{X}}[B(\boldsymbol{x})]$ και $E_{\boldsymbol{X}}[V(\boldsymbol{x})]$.

Έστω μια παρατήρηση $\boldsymbol{x}$ με πραγματική πρόβλεψη y. Έστω ταξινομητής που προβλέπει γι' αυτό το $\boldsymbol{x}$ μια τιμή $\hat{y}$ από ένα σύνολο δεδομένων μέσα στο $\mathbf{D}^*$. Θα δείξουμε ότι για ένα σύνολο συναρτήσεων κόστους C ισχύει η παρακάτω παραγοντοποίηση του αναμενόμενου κόστους:

$$\begin{aligned} E_{\mathbf{D}^*,Y}[C(y,\hat{y})] &= c_1 E_Y[C(y,\hat{y}^*)] + C(\hat{y}^*,\hat{y}_m) + c_2 E_{\mathbf{D}^*}[C(\hat{y}_m,\hat{y})] \\ &= c_1 N(\boldsymbol{x}) + B(\boldsymbol{x}) + c_2 V(\boldsymbol{x}), \end{aligned} \tag{1.1}$$

όπου τα c_1, c_2 κατάλληλες σταθερές που εξαρτώνται από τη συνάρτηση κόστους που χρησιμοποιούμε.

1.3.3. Το τετράγωνο των σφαλμάτων ως συνάρτηση κόστους

Αρχικά ας επαληθεύσουμε το αποτέλεσμα (1.1) που δείξαμε στην περίπτωση του τετραγωνικού σφάλματος ως συνάρτηση κόστους.

Αντικαθιστώντας $C(y,\hat{y}) = (y-\hat{y})^2$, $\hat{y}^* = E_Y[y]$, $\hat{y}_m = E_{\mathbf{D}^*}[\hat{y}]$ και $c_1 = c_2 = 1$ το 1.1 γίνεται:

$$\begin{aligned} E_{\mathbf{D}^*,Y}[(y-\hat{y})^2] &= E_Y[(y-E_Y[y])^2] + (E_Y[y] - E_{\mathbf{D}^*}[\hat{y}])^2 + E_{\mathbf{D}^*}[(E_{\mathbf{D}^*}[\hat{y}] - \hat{y})^2] \\ &= E_Y[C(y,E_Y[y])] + C(E_Y[y],E_{\mathbf{D}^*}[\hat{y}]) + E_{\mathbf{D}^*}[C(E_{\mathbf{D}^*}[\hat{y}],\hat{y})] \\ &= N(\boldsymbol{x}) + B(\boldsymbol{x}) + V(\boldsymbol{x}), \end{aligned}$$

Η οποία είναι η συνηθισμένη παραγοντοποίηση του τετραγωνικού κόστους που συναντάμε στη βιβλιογραφία, όπως για παράδειγμα στους Geman, Bienenstock, & Doursat (1992).

Το $\hat{y}^* = E_Y[y]$ ισχύει γιατί:

$$\begin{aligned} E_Y[(y-\hat{y})^2] &= Var_Y[(y-\hat{y})^2] + E_Y[y-\hat{y}]^2 = E_Y\big[(y-\hat{y}-E_y[y-\hat{y}])^2\big] + E_Y[y-\hat{y}]^2 \\ &= E_Y[(y-E_Y[y])^2] + E_Y[y-\hat{y}]^2, \end{aligned}$$

άρα το $E_Y[(y-\hat{y})^2]$ ελαχιστοποιείται για $\hat{y} = E_Y[y]$.

Να παρατηρήσουμε εδώ ότι στην ενοποιημένη θεωρία, το τετράγωνο στην μεροληψία $(E_Y[y] - E_{\mathbf{D}^*}[\hat{y}])^2$ προκύπτει λόγω της επιλογής του τετραγωνικού σφάλματος ως συνάρτηση κόστους. Έτσι όταν μιλάμε μέσω της θεωρίας αυτής, καλούμε όλη την ποσότητα μεροληψία ενώ με βάση την προηγούμενη ενότητα την καλούμε τετράγωνο της μεροληψίας. Αυτό συμβαίνει γιατί έχουμε διαφορετικό ορισμό για τη μεροληψία ενός ταξινομητή σε σχέση με τη μεροληψία ενός εκτιμητή της στατιστικής θεωρίας.

1.3.4. Η συνάρτηση κόστους μηδέν-ένα

Θα δείξουμε τώρα τη σχέση (1.1) για την μηδέν-ένα συνάρτηση κόστους σε προβλήματα με πολλές κλάσεις. Θα δείξουμε αρχικά ότι:

$$E_Y[C(y,\hat{y})] = C(\hat{y}^*,\hat{y}) + c_0 E_Y[C(y,\hat{y}^*)] \tag{1.2}$$

$$\mu\varepsilon\ c_0 = \begin{cases} 1 & \alpha\nu\ \ \hat{y}^* = \hat{y} \\ -P_Y(y = \hat{y}\,|\,\hat{y}^* \neq y) & \alpha\nu\ \ \hat{y}^* \neq \hat{y}. \end{cases}$$

Προφανώς αν $\hat{y}^* = \hat{y}$ τότε $c_0 = 1$ και η σχέση ισχύει γιατί $C(\hat{y}^*,\hat{y}^*) = 0$.

Έστω ότι $\hat{y}^* \neq \hat{y}$. Τότε $P_Y(y = \hat{y}\,|\,\hat{y}^* = y) = 0$ άρα

$$\begin{aligned} P_Y(y = \hat{y}) &= P_Y(\hat{y}^* \neq y)P_Y(y = \hat{y}\,|\,\hat{y}^* \neq y) + P_Y(\hat{y}^* = y)P_Y(y = \hat{y}\,|\,\hat{y}^* = y) \\ &= P_Y(\hat{y}^* \neq y)P_Y(y = \hat{y}\,|\,\hat{y}^* \neq y). \end{aligned}$$

Άρα

$$\begin{aligned} E_Y[C(y,\hat{y})] = P_Y(y \neq \hat{y}) &= 1 - P_Y(y = \hat{y}) = 1 - P_Y(\hat{y}^* \neq y)P_Y(y = \hat{y}\,|\,\hat{y}^* \neq y) \\ &= C(\hat{y}^*,\hat{y}) + c_0 E_Y[C(y,\hat{y}^*)]. \end{aligned}$$

Άρα δείξαμε την (1.2). Τώρα θα δείξουμε ότι:

$$E_{\mathbf{D}^*}[C(\hat{y}^*,\hat{y})] = C(\hat{y}^*,\hat{y}_m) + c_2 E_{\mathbf{D}^*}[C(\hat{y}_m,\hat{y})], \tag{1.3}$$

$$\mu\varepsilon\ c_2 = \begin{cases} 1 & \alpha\nu\ \ \hat{y}^* = \hat{y}_m \\ -P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}\,|\hat{y}\ \neq \hat{y}_m) & \alpha\nu\ \ \hat{y}^* \neq \hat{y}_m. \end{cases}$$

Αν $\hat{y}^* = \hat{y}_m$ τότε $c_2 = 1$ οπότε η σχέση (1.3) ισχύει.

Αν $\hat{y}^* \neq \hat{y}_m$ τότε $P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}\,|\hat{y}\ = \hat{y}_m) = 0$.

$$\begin{aligned} P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}) &= P_{\mathbf{D}^*}(\hat{y}\ \neq \hat{y}_m)P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}\,|\hat{y}\ \neq \hat{y}_m) + P_{\mathbf{D}^*}(\hat{y}\ = \hat{y}_m)P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}\,|\hat{y}\ = \hat{y}_m) \\ &= P_{\mathbf{D}^*}(\hat{y}\ \neq \hat{y}_m)P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}\,|\hat{y}\ \neq \hat{y}_m). \end{aligned}$$

Άρα

$$\begin{aligned} E_{\mathbf{D}^*}[C(\hat{y}^*,\hat{y})] = P_{\mathbf{D}^*}(\hat{y}^* \neq \hat{y}) &= 1 - P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}) \\ &= 1 - P_{\mathbf{D}^*}(\hat{y}\ \neq \hat{y}_m)P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}\,|\hat{y}\ \neq \hat{y}_m) \\ &= C(\hat{y}^*,\hat{y}_m) + c_2 E_{\mathbf{D}^*}[C(\hat{y}_m,\hat{y})]. \end{aligned}$$

Άρα δείξαμε την (1.3). Αντικαθιστώντας το (1.2) στο $E_{\mathbf{D}^*,Y}[C(y,\hat{y})]$:

$$\begin{aligned} E_{\mathbf{D}^*,Y}[C(y,\hat{y})] = E_{\mathbf{D}^*}\big[E_Y[C(y,\hat{y})]\big] &= E_{\mathbf{D}^*}\big[C(\hat{y}^*,\hat{y}) + c_0 E_Y[C(y,\hat{y}^*)]\big] \\ &= E_{\mathbf{D}^*}[c_0]E_Y[C(y,\hat{y}^*)] + E_{\mathbf{D}^*}[C(\hat{y}^*,\hat{y})], \end{aligned} \tag{1.4}$$

γιατί το $C(y,\hat{y}^*)$ δεν εξαρτάται από το $\mathbf{D}^*$. Έστω

$$c_1 = E_{\mathbf{D}^*}[c_0] = P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}) - P_{\mathbf{D}^*}(\hat{y}^* \neq \hat{y})P_{\boldsymbol{Y}}(\hat{y} = y\ |\ \hat{y}^* \neq y).$$

Αντικαθιστώντας στην (1.4) το c_1 και την (1.3) παίρνουμε:

$$\begin{aligned} E_{\mathbf{D}^*,Y}[C(y,\hat{y})] &= c_1 E_Y[C(y,\hat{y}^*)] + C(\hat{y}^*,\hat{y}_m) + c_2 E_{\mathbf{D}^*}[C(\hat{y}_m,\hat{y})] \\ &= c_1 N(\boldsymbol{x}) + B(\boldsymbol{x}) + c_2 V(\boldsymbol{x}), \end{aligned}$$

με

$$c_1 = P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}) - P_{\mathbf{D}^*}(\hat{y}^* \neq \hat{y})P_{\boldsymbol{Y}}(\hat{y} = y\ |\ \hat{y}^* \neq y)$$

και

$$c_2 = \begin{cases} 1 & \alpha\nu\ \ \hat{y}^* = \hat{y}_m\ \ \Leftrightarrow\ \ B(\boldsymbol{x}) = 0 \\ -P_{\mathbf{D}^*}(\hat{y}^* = \hat{y}\,|\hat{y}\ \neq \hat{y}_m) & \alpha\nu\ \ \hat{y}^* \neq \hat{y}_m\ \ \Leftrightarrow\ \ B(\boldsymbol{x}) = 1. \end{cases}$$

Από τον συντελεστή c_1 του θορύβου $N(\boldsymbol{x})$ παρατηρούμε ότι ο θόρυβος μπορεί να μειώσει το σφάλμα μόνο στην περίπτωση που η πρόβλεψη $\hat{y}$ είναι ίση με την παρατηρούμενη τιμή y αλλά ταυτόχρονα και τα δύο είναι διαφορετικά από τη βέλτιστη πρόβλεψη. Δηλαδή μπορεί να βελτιώσει το αποτέλεσμα "κατά λάθος". Από τον συντελεστή c_2 της διακύμανσης βλέπουμε πλέον ξεκάθαρα το δίπολο μεροληψίας-διακύμανσης. Αν το B είναι 0, το V έχει συντελεστή 1 και αυξάνει το σφάλμα. Αν το B είναι 1, ο συντελεστής του V γίνεται αρνητικός και μειώνει το σφάλμα.

1.4. Ο ταξινομητής Μηδενικού Κανόνα

Στόχος αυτής της ενότητας είναι να βρούμε ένα πρακτικό κάτω φράγμα για το πρόβλημα της στατιστικής ταξινόμησης, δηλαδή τη διαδικασία με την οποία βρίσκουμε τη χειρότερη λογική πρόβλεψη $\hat{y}$ από ένα σύνολο δεδομένων $\boldsymbol{D}$. Λέμε την χειρότερη λογική επειδή η χειρότερη δυνατή πρόβλεψη είναι η απόφαση μέσω της ρίψης νομίσματος. Στόχος είναι να θέσουμε μια βάση αξιολόγησης για τους μελλοντικούς μας ταξινομητές. Η απάντηση βρίσκεται στον **ταξινομητή μηδενικού κανόνα** (Zero Rule ή ZeroR) ο οποίος αγνοεί τις επεξηγηματικές μεταβλητές $X_1, X_2, \dots, X_d$ και επικεντρώνεται μόνο στη μεταβλητή απόκρισης Y. Ο ταξινομητής μηδενικού κανόνα κάνει έναν πίνακα συχνοτήτων της μεταβλητής Y και επιλέγει πάντα να ταξινομεί τις καινούριες παρατηρήσεις στην κατηγορία με τη μεγαλύτερη συχνότητα. Έστω ότι το Y παίρνει c τιμές, τις $1, 2, \dots, c$ και το δείγμα έχει n παρατηρήσεις. Τότε το μοντέλο που προκύπτει από τον ταξινομητή έχει τυπικά την παρακάτω μορφή:

$$\hat{y}_{ZeroR} = \underset{k}{\operatorname{argmax}} \left(\sum_{i=1}^{N} I(y_i = k) \right) \mu\varepsilon\ k = 1, 2, \dots, c\,, \qquad i = 1, 2, \dots, N$$

$$\kappa\alpha\iota\ I(y_i = k) = \begin{cases} 0\ \alpha\nu\ y_i \neq k \\ 1\ \alpha\nu\ y_i = k. \end{cases}$$

Μετά τον υπολογισμό του μοντέλου, μπορούμε να υπολογίσουμε το μέσο σφάλμα του με βάση τη μηδέν-ένα συνάρτηση κόστους και να το χρησιμοποιήσουμε ως μέτρο σύγκρισης για την αξιολόγηση των πραγματικών μοντέλων μας. Εν γένει δεν δεχόμαστε κανένα μοντέλο που να έχει μικρότερη ακρίβεια από το μοντέλο που προκύπτει από τον ταξινομητή μηδενικού κανόνα.

Από τον ταξινομητή μηδενικού κανόνα κατασκευάζεται με φυσικό τρόπο ο **ταξινομητής ενός κανόνα** (One Rule ή OneR) ο οποίος έχει ως στόχο να χρησιμοποιήσει μία από τις μεταβλητές $X_1, X_2, \dots, X_d$, συγκεκριμένα εκείνη με τη μεγαλύτερη επεξηγηματική ικανότητα. Έστω η προς ταξινόμηση παρατήρηση $\boldsymbol{x}^* = (x_1^*, x_2^*, \dots, x_d^*)$. Ο ταξινομητής αυτός κατασκευάζει d διασταυρωμένους πίνακες συχνοτήτων $[x_j, y]$ με $j = 1, 2, \dots, d$ μέσω των οποίων κατασκευάζονται d διαφορετικά μοντέλα με τύπο

$$\hat{y}_{x_j} = \underset{k}{\operatorname{argmax}} \left(\sum_{i=1}^{N} I\left(y_i = k | \, X_j = x_j^*\right) \right)$$

$$\mu\varepsilon\ j = 1, 2, \dots, d, \quad k = 1, 2, \dots, c \ \ \kappa\alpha\iota\ i = 1, 2, \dots, N.$$

και επιλέγει εκείνον με το μικρότερο μέσο σφάλμα, δηλαδή

$$\hat{y}_{OneR} = \underset{\hat{y}_{x_j}}{\operatorname{argmin}} E_y\left[C\left(y, \hat{y}_{x_j}\right)\right] \ με \ C\left(y, \hat{y}_{x_j}\right) = \begin{cases} 0 \ αν \ y = \hat{y}_{x_j} \\ 1 \ αν \ y \neq \hat{y}_{x_j} \end{cases}.$$

1.5. Ο ταξινομητής Bayes

Κατά αναλογία με την προηγούμενη ενότητα, στόχος είναι να βρούμε τη διαδικασία που θα μας δώσει το $\hat{y}^*$, δηλαδή το μοντέλο που θα κάνει τη βέλτιστη πρόβλεψη για κάθε $\boldsymbol{x}^*$, δεδομένου ότι μας ενδιαφέρει η περίπτωση της μηδέν-ένα συνάρτησης κόστους. Θέλουμε δηλαδή να βρούμε τη διαδικασία που ελαχιστοποιεί το:

$$E_Y[C(y, \hat{y})] \ με \ C(y, \hat{y}) = \begin{cases} 0 \ αν \ y = \hat{y} \\ 1 \ αν \ y \neq \hat{y} \end{cases}$$

και τότε, $\hat{y} = \hat{y}^*$. Αλλά

$$E_Y[C(y, \hat{y})] = P_Y(y \neq \hat{y}) = 1 - P_Y(y = \hat{y}).$$

Άρα για να ελαχιστοποιηθεί το $E_Y[C(y, \hat{y})]$ πρέπει να μεγιστοποιηθεί το $P_Y(y = \hat{y})$. Έστω ότι το Y παίρνει c τιμές, τις $1, 2, \dots, c$ και έστω μια παρατήρηση προς ταξινόμηση $\boldsymbol{x}^*$. Τότε πρέπει να υπολογιστεί το $\hat{y}$ μέσω της παρακάτω διαδικασίας που καλείται **ταξινομητής Bayes**:

$$\hat{y}^* = \hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}} P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) \ με \ k = 1, 2, \dots, c.$$

Πρέπει δηλαδή να βρεθεί για ποιο από τα c πιθανά $\hat{y}$ μεγιστοποιείται η πιθανότητα να ισχύει $y = \hat{y}$ όταν $\boldsymbol{X} = \boldsymbol{x}^*$. Έτσι φαίνεται ότι ο ταξινομητής Bayes ελαχιστοποιεί το μέσο σφάλμα για τη μηδέν-ένα συνάρτηση κόστους και το μέσο σφάλμα του $E_Y[C(y, \hat{y}^*)] = P_Y(y \neq \hat{y}^*)$ καλείται **πιθανότητα σφάλματος Bayes** (Bayes Probability of Error).

Προφανώς στην πράξη δε μπορούμε να γνωρίζουμε τις συναρτήσεις μάζας πιθανότητας $P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*)$ και ο ταξινομητής Bayes χρησιμοποιείται ως ένα θεωρητικό μέτρο σύγκρισης για τους υπόλοιπους ταξινομητές. Επειδή θα έχουμε ένα σύνολο από N δεδομένα $\boldsymbol{D}_N$, η πιθανότητα σφάλματος ενός ταξινομητή γράφεται τυπικότερα $P_Y(y \neq \hat{y}_N \mid \boldsymbol{D}_N)$, δηλαδή η πρόβλεψη αλλάζει καθώς μεταβάλλεται το μέγεθος του δείγματος. Το αναμενόμενο σφάλμα θα είναι $E_{\boldsymbol{D}}[P_Y(y \neq \hat{y}_N)]$.

Ορισμός: Ένας ταξινομητής θα καλείται **ασθενώς συνεπής** (weakly consistent) αν

$$E_{\boldsymbol{D}}[P_Y(y \neq \hat{y}_N)] \to P_Y(y \neq \hat{y}^*) \ καθώς \ N \to \infty.$$

Η ασθενής σύγκληση μπορεί εναλλακτικά να οριστεί ως σύγκληση κατά πιθανότητα, δηλαδή για κάθε $\varepsilon > 0$

$$\lim_{n \to \infty} P(P_Y(y \neq \hat{y}_N \mid \boldsymbol{D}_N) - P_Y(y \neq \hat{y}^*) > \varepsilon) = 0.$$

Ορισμός: Ένας ταξινομητής θα καλείται **ισχυρά συνεπής** (strongly consistent) αν

$$P_Y(y \neq \hat{y}_N \mid \boldsymbol{D}_N) \to P_Y(y \neq \hat{y}^*) \ σχεδόν \ βέβαια.$$

Επειδή η σχεδόν βέβαιη σύγκλιση συνεπάγει σύγκλιση κατά πιθανότητα, η ισχυρή συνέπεια συνεπάγει την ασθενή συνέπεια.

Ένας ασθενώς συνεπής ταξινομητής εγγυάται ότι αυξάνοντας την ποσότητα των δεδομένων, η πιθανότητα του να φτάσει η πιθανότητα σφάλματος πολύ κοντά στην βέλτιστη φτάνει οσοδήποτε κοντά στο 1. Ένας ισχυρά συνεπής ταξινομητής εγγυάται ότι αυξάνοντας την ποσότητα των δεδομένων η πιθανότητα σφάλματος φτάνει οσοδήποτε κοντά στην βέλτιστη για οποιαδήποτε ακολουθία παρατηρήσεων, αρκεί οι παρατηρήσεις να μην έχουν μηδενική πιθανότητα να παρατηρηθούν μαζί.

Ορισμός: Ένας ταξινομητής θα καλείται **γενικά (universally) ασθενώς συνεπής** (αντίστοιχα **γενικά ισχυρά συνεπής**) αν είναι ασθενώς συνεπής (αντίστοιχα ισχυρά συνεπής) για κάθε κατανομή των $(Y, \boldsymbol{X})$.

Υπάρχει επίσης μεγάλος αριθμός μεθόδων που προσπαθούν να προσεγγίσουν τον ταξινομητή Bayes υπολογίζοντας απευθείας το

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*).$$

Αυτοί οι ταξινομητές καλούνται **διαχωριστικοί ταξινομητές** (Discriminative Classifiers). Υπάρχει επίσης μεγάλος αριθμός ταξινομητών που υπολογίζουν τις πιθανότητες κάνοντας χρήση του θεωρήματος Bayes

$$\begin{aligned}\hat{y}_{Bayes} &= \underset{k}{\operatorname{argmax}}\, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) = \underset{k}{\operatorname{argmax}} \frac{P(\mathbf{X} = \boldsymbol{x}^* \mid Y = k)P(Y = k)}{P(\mathbf{X} = \boldsymbol{x}^*)} \\ &= \underset{k}{\operatorname{argmax}}\, P(\mathbf{X} = \boldsymbol{x}^* \mid Y = k)P(Y = k).\end{aligned}$$

Αυτοί οι ταξινομητές καλούνται **παραγωγικοί ταξινομητές** (Generative Classifiers). Να παρατηρήσουμε εδώ ότι αν αντικαταστήσουμε τις συναρτήσεις πυκνότητας/μάζας πιθανότητας με κάποιες εκτιμήσεις τους $\hat{P}(\mathbf{X} = \boldsymbol{x}^* \mid Y = k)$ και $\hat{P}(Y = k)$ καταλήγουμε στο:

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, \hat{P}(\boldsymbol{X} = \boldsymbol{x}^* \mid Y = k)\hat{P}(Y = k).$$

Για την εκτίμηση του $\hat{P}(\boldsymbol{X} = \boldsymbol{x}^* \mid Y = k)$ υπάρχουν πολλές προσεγγίσεις λόγω των ιδιαιτεροτήτων που μπορεί να έχει το σύνολο εκπαίδευσής μας κάθε φορά και στο επόμενο κεφάλαιο θα δούμε κάποιες τέτοιες διαδικασίες. Το Y όμως ακολουθεί πάντα κατηγορική κατανομή, άρα ο υπολογισμός των $\hat{P}(Y = k)$ είναι τετριμμένος. Χρησιμοποιούμε την εκτιμήτρια μέγιστης πιθανοφάνειας της κατηγορικής κατανομής που ταυτίζεται με τις σχετικές συχνότητες του κάθε k στο δείγμα. Μετά την απλοποίηση του $\frac{1}{N}$ που είναι σταθερά ανεξάρτητή της τιμής του Y ο ταξινομητής Bayes προσεγγίζεται ως:

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, \hat{P}(\boldsymbol{X} = \boldsymbol{x}^* \mid Y = k) \sum_{i=1}^{N} I(y_i = k), \quad \begin{matrix} i = 1, 2, \dots, N \\ k = 1, 2, \dots, c. \end{matrix}$$

Άρα η προσέγγιση του ταξινομητή Bayes μπορεί να θεωρηθεί ως μια διόρθωση του μοντέλου που προκύπτει από τον ταξινομητή μηδενικού κανόνα κατά τον παράγοντα $\hat{P}(\mathbf{X} = \boldsymbol{x}^* \mid Y = k)$. Οι επεξηγηματικές μεταβλητές εμφανίζονται μόνο σε αυτό τον παράγοντα, άρα ολόκληρη η διαδικασία της ταξινόμησης, υπό αυτή την προσέγγιση, μετασχηματίζεται στην εύρεση μιας εκτίμησης $\hat{P}(\boldsymbol{X} = \boldsymbol{x}^* \mid Y = k)$ που να βελτιώνει το μοντέλο που προκύπτει από τον ταξινομητή μηδενικού κανόνα μέσω των επεξηγηματικών μεταβλητών. Ο λόγος που

απορρίπτουμε κάθε μοντέλο με χειρότερη ακρίβεια από αυτό του ταξινομητή μηδενικού κανόνα είναι ότι σε αυτή την περίπτωση χρησιμοποιήσαμε τις επεξηγηματικές μεταβλητές με τόσο λανθασμένο τρόπο που το χειροτερέψαμε αντί να το βελτιώσουμε.

2. Η Στατιστική Προσέγγιση στην Ταξινόμηση

2.1. Εισαγωγικά

Στην εισαγωγή είδαμε δύο περιγραφές της παραμετρικότητας ως αφορμή για να μιλήσουμε για τις διαφορετικές κουλτούρες του κλάδου. Με βάση την εγκυκλοπαίδεια (Sammut & Webb, 2011) το λήμμα της μη παραμετρικής μεθόδου μας παραπέμπει στο λήμμα **μάθηση κατά παράδειγμα** (instance based learning) το οποίο μας δίνει τον παρακάτω τρίτο χαρακτηρισμό:

"Οι αλγόριθμοι μάθησης κατά παράδειγμα δεν δημιουργούν γενικά χαρακτηριστικά από τα παραδείγματα. Απλά αποθηκεύουν όλα τα δεδομένα και στην ώρα της εξέτασης η απάντηση προκύπτει από την εξέταση των κοντινότερων γειτόνων της εξεταζόμενης περίπτωσης"

Claude Sammut, Geoffrey I. Webb (2011) "Encyclopedia of machine learning" Springer σ. 549

Από το παραπάνω αντιλαμβανόμαστε άλλο ένα χαρακτηριστικό των μη παραμετρικών μοντέλων. Αυτό είναι ότι για να προβλέψουν κάθε τιμή πρέπει να επανεξετάσουν τα δεδομένα οπότε πρέπει να υπάρχουν διαθέσιμα στη μνήμη. Για αυτό η προσαρμογή τους καλείται και **μάθηση βασισμένη στη μνήμη** (memory based learning). Η όλη διαδικασία της ταξινόμησης λοιπόν έχει υπολογιστικό κόστος για κάθε πρόβλεψη αλλά η προσαρμογή τους έχει μηδενικό κόστος, απλά αποθηκεύουν το δείγμα εκπαίδευσης. Για αυτό αποκαλούνται και **τεμπέλικοι ταξινομητές** (lazy classifiers).

Δεδομένης της ασάφειας ως προς το τι είναι τελικά παραμετρικό και τι όχι, στην παρούσα εργασία θα αναφέρουμε ως μη παραμετρική μια μέθοδο μόνο στην περίπτωση που ικανοποιεί και τα τρία χαρακτηριστικά, δηλαδή αν δεν υποθέτει κατανομή, δεν έχει σταθερό αριθμό παραμέτρων και δεν δημιουργεί κάποιο στατιστικό μοντέλο. Θα καλούμε ως παραμετρική μέθοδο μια μέθοδο που δεν ικανοποιεί κανένα από τα παραπάνω. Τις μεθόδους που ικανοποιούν μέρος των παραπάνω χαρακτηριστικών δεν θα τις κατατάξουμε ως προς την παραμετρικότητά τους.

Δύο μη παραμετρικές μέθοδοι είναι ο ταξινομητής k-πλησιέστερων γειτόνων και η ταξινόμηση μέσω της μεθόδου των πυρήνων που θα δούμε παρακάτω. Στη συνέχεια θα μιλήσουμε για παραμετρική ταξινόμηση μέσω των παραδοσιακότερων ταξινομητών. Ο πρώτος είναι η λογιστική παλινδρόμηση από τη θεωρία των γενικευμένων γραμμικών μοντέλων. Ο δεύτερος είναι ο ταξινομητής Naïve Bayes, ένας απλός ταξινομητής που κάνει μια ισχυρή υπόθεση ανεξαρτησίας αλλά παρόλα αυτά πετυχαίνει μοντέλα με αξιοθαύμαστα αποτελέσματα. Ο τρίτος είναι ο ιστορικός ταξινομητής Γραμμικής Διαχωριστικής Ανάλυσης που όρισε ο Fisher. Τέλος, θα δούμε τις δύο βασικές προσεγγίσεις στην ταξινόμηση μέσω της Μπεϋζιανής στατιστικής.

2.2. Δύο μη παραμετρικοί ταξινομητές που ξεκινούν από τον Bayes

2.2.1. **Ο ταξινομητής K-πλησιέστερων γειτόνων**

Ένας μη παραμετρικός και μη γραμμικός ταξινομητής που προσεγγίζει τον ταξινομητή Bayes είναι ο ταξινομητής **K-πλησιέστερων γειτόνων** (K-Nearest neighbor). Σε αυτή την προσέγγιση θεωρούμε τις παρατηρήσεις μας ως σημεία του Ευκλείδειου χώρου $\mathbb{R}^d$. Έτσι, η τιμή x_j της μεταβλητής X_j των $X_1, X_2, \dots, X_d$ είναι για την j παρατήρηση η τιμή της j διάστασής της. Για το δείγμα μας προφανώς γνωρίζουμε τις αντίστοιχες τιμές της μεταβλητής απόκρισης Y. Θεωρούμε μία μετρική $\rho(\boldsymbol{a}, \boldsymbol{b})$, έστω την Ευκλείδεια μετρική $\rho(\boldsymbol{a}, \boldsymbol{b}) = \sqrt{\sum_{j=1}^{d}(a_j - b_j)^2}$ με $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^d$. Θεωρούμε επίσης έναν ακέραιο K. Λαμβάνοντας μια παρατήρηση προς ταξινόμηση $\boldsymbol{x}^* = (x_1, x_2, \dots, x_d)$ υπολογίζουμε τις K πλησιέστερες παρατηρήσεις του δείγματός μας σε σχέση με την $\boldsymbol{x}^*$ μέσω της ρ, έστω $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_K$. Η απόσταση

$$L = \max\big(\rho(\boldsymbol{x}_i, \boldsymbol{x}^*)\big) \, \mu\varepsilon \; i = 1, 2, \dots, K,$$

ορίζει μία σφαίρα με κέντρο το $\boldsymbol{x}^*$ και ακτίνα L στο $\mathbb{R}^d$, την

$$S_\rho(\boldsymbol{x}^*, L) = \{\boldsymbol{x} \in \mathbb{R}^d : \rho(\boldsymbol{x}, \boldsymbol{x}^*) \leq L\}.$$

Έστω ότι το Y παίρνει c τιμές, τις $1, 2, \dots, c$. Τότε, ο ταξινομητής K-πλησιέστερων γειτόνων προσεγγίζει τη δεσμευμένη πιθανότητα $P(Y = k \mid \boldsymbol{X} = \boldsymbol{x}^*)$ με $\boldsymbol{x}^* \in \mathbb{R}^d$ του ταξινομητή Bayes ως

$$P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) \approx \frac{1}{K} \sum_{\boldsymbol{x}_i \in S_\rho(\boldsymbol{x}^*, L)} I(y_i = k),$$

$$\mu\varepsilon \; k = 1, 2, \dots, c \;\; \kappa\alpha\iota \; i = 1, 2, \dots, N.$$

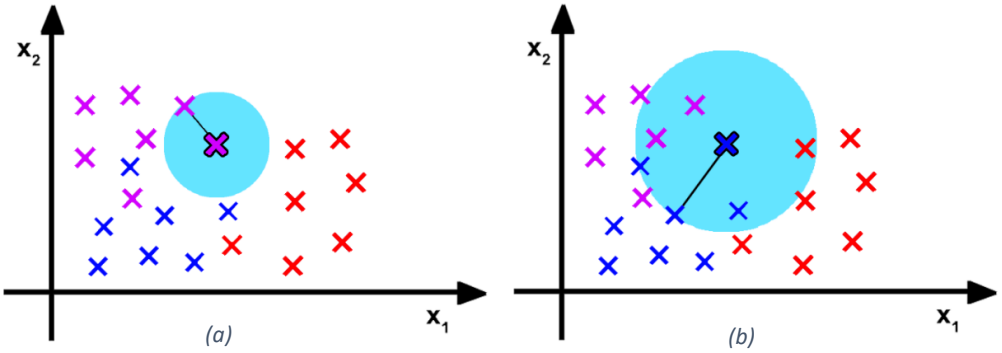
Οπότε ο ταξινομητής Bayes προσεγγίζεται ως:

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}} \, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) \approx \underset{k}{\operatorname{argmax}} \frac{1}{K} \sum_{x_i \in S_\rho(\boldsymbol{x}^*, L)} I(y_i = k)$$

$$= \underset{k}{\operatorname{argmax}} \sum_{\boldsymbol{x}_i \in S_\rho(\boldsymbol{x}^*, L)} I(y_i = k) \; = \hat{y}_{kNN.}$$

Ο ταξινομητής δηλαδή ταξινομεί την κάθε παρατήρηση με βάση την επικρατέστερη κλάση μεταξύ των παρατηρήσεων που βρίσκονται στη σφαίρα γύρω από το $\boldsymbol{x}^*$. Η σφαίρα S_ρ παίζει το ρόλο της δέσμευσης στην προς υπολογισμό δεσμευμένη πιθανότητα. Όπως γίνεται φανερό η επιλογή του K επηρεάζει το μοντέλο.

Στο Διάγραμμα 2.1 φαίνεται πως ο ταξινομητής K-πλησιέστερων γειτόνων προβλέπει διαφορετική κλάση για την ίδια παρατήρηση αν αλλάξει το K.



Διάγραμμα 2.1: Στο (a) βλέπουμε την πρόβλεψη του ταξινομητή 1-πλησιέστερου γείτονα και στο (b) τον ταξινομητή 6-πλησιέστερων γειτόνων για το ίδιο σημείο πρόβλεψης. Όπως είναι φανερό ο αριθμός των γειτόνων που θα επιλεχθεί επηρεάζει τις προβλέψεις.

Αρχικά να παρατηρήσουμε ότι στην περίπτωση δύο κλάσεων για σταθερό K, είναι καλή ιδέα να το επιλέξουμε περιττό αριθμό, μιας και έτσι θα αποφύγουμε την πιθανότητα ισοπαλιών. Επίσης, για σταθερό μέγεθος δείγματος, μικρότερα K δημιουργούν πιο ευέλικτα μοντέλα με μεγαλύτερο σφάλμα διακύμανσης και χαμηλότερο σφάλμα μεροληψίας. Στην περίπτωση πολύ μικρού K και δείγματος υπάρχει φόβος υπερπροσαρμογής. Για πολύ μεγάλα K δημιουργούνται πιο άκαμπτα μοντέλα με μεγαλύτερο σφάλμα μεροληψίας και χαμηλότερο σφάλμα διακύμανσης. Στην περίπτωση πολύ μεγάλου K και μικρού δείγματος υπάρχει φόβος υποπροσαρμογής επειδή καθώς το K τείνει στο N, το μοντέλο τείνει στον γραμμικό διαχωρισμό των παρατηρήσεων.

Στην πράξη επιλέγουμε το K να είναι συνάρτηση του μεγέθους των παρατηρήσεων N λόγω του παρακάτω θεωρήματος:

Θεώρημα: ο ταξινομητής K-πλησιέστερων γειτόνων είναι γενικά ασθενώς συνεπείς αν και μόνο αν

$$K \to \infty \; και \; \frac{K}{N} \to 0.$$

Για την απόδειξη του παραπάνω θεωρήματος και για της συνθήκες υπό τις οποίες ο ταξινομητής γίνεται γενικά ισχυρά συνεπής παραπέμπουμε στους Biau & Devroy (2015, pp. 242-249).

Μία συνηθισμένη επιλογή για το K είναι το $K = \log(N)$ ώστε να ικανοποιούνται και οι δύο συνθήκες του θεωρήματος καθώς:

$$K = \log(N) \to \infty \; καθώς \; το \; N \to \infty \quad και \quad \lim_{N\to\infty} \frac{K}{N} = \lim_{N\to\infty} \frac{\log(N)}{N} = \lim_{N\to\infty} \frac{1}{N} = 0.$$

Στη γλώσσα R ο ταξινομητής K-πλησιέστερων γειτόνων υπάρχει υλοποιημένος στη βιβλιοθήκη “class” από τους (Venables & Riplay, 2002) και τον καλούμε με την εντολή “knn”, όπως φαίνεται παρακάτω.

```
kNN3<-knn(train=dataset.train[,1:4], test=dataset.test[,1:4], cl= dataset.train[,5], k = 3)
```

Το όρισμα *train* είναι το δείγμα εκπαίδευσης σε μορφή πίνακα με μόνο τις επεξηγηματικές μεταβλητές, το *test* είναι το αντίστοιχο δείγμα ελέγχου, το *cl* είναι οι τιμές των κλάσεων του δείγματος εκπαίδευσης σε μορφή factor και το *k* ο αριθμός των πλησιέστερων γειτόνων. Ο ταξινομητής αυτός πέτυχε ακρίβεια 0.947 στο δείγμα Iris (βλ. Κεφάλαιο 5).

2.2.2. Ταξινόμηση μέσω της μεθόδου των πυρήνων

Μία άλλη μη παραμετρική και μη γραμμική προσέγγιση είναι η ταξινόμηση μέσω εκτίμησης της πυκνότητας πιθανότητας με τη **μέθοδο των πυρήνων** (Kernel). Ως μη παραμετρική μέθοδος, όπως και στον ταξινομητή k-πλησιέστερων γειτόνων δεν θα δημιουργηθεί κάποιο μαθηματικό μοντέλο αλλά κάθε φορά που θα θέλουμε να κάνουμε μια πρόβλεψη θα εξετάζουμε εκ νέου το δείγμα μας. Πιο συγκεκριμένα, ξεκινώντας από τον ταξινομητή Bayes:

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) = \underset{k}{\operatorname{argmax}}\, P(\mathbf{X} = \boldsymbol{x}^* \mid Y = k)P(Y = k).$$

Τις $P(Y = k)$ θα τις υπολογίσουμε ως τις σχετικές συχνότητες του κάθε k μιας και η μεταβλητή Y ακολουθεί κατηγορική κατανομή και η σχετική συχνότητα της κάθε κλάσης ως εκτιμήτρια της πιθανότητας εμφάνισής της προκύπτει από τη μεγιστοποίηση της συνάρτησης πιθανοφάνειας, άρα είναι συνεπής και ασυμπτωτικά αμερόληπτη εκτιμήτρια ελάχιστης διασποράς. Στόχος μας είναι να υπολογίσουμε τις c συναρτήσεις πυκνότητας πιθανότητας $P(\mathbf{X} = \boldsymbol{x}^* \mid Y = k)$ μόνο στο σημείο προς ταξινόμηση $\boldsymbol{x}^*$. Θα το κάνουμε αυτό σύμφωνα με το βιβλίο του Silverman (1998, pp. 11-19, 75-78).

Βασιζόμενοι στην ορισμό της συνάρτησης πυκνότητας πιθανότητας στη μονοδιάστατη περίπτωση, αν μια τυχαία μεταβλητή έχει συνάρτηση πυκνότητας πιθανότητας f, τότε:

$$f(x) = \lim_{h \to 0} \frac{P(x - h < X < x + h)}{2h}.$$

Για κάθε h, μπορούμε να υπολογίσουμε το $P(x - h < X < x + h)$ μέσω του ποσοστού των παρατηρήσεων που πέφτουν μέσα στο διάστημα $(x - h, x + h)$, άρα ένας εκτιμητής της συνάρτησης πυκνότητας πιθανότητας της f σε ένα x_0 που προκύπτει με φυσικό τρόπο είναι

$$\hat{f}(x_0) = \frac{1}{2nh}\ count\big(x :\ x \in (x_0 - h, x_0 + h)\big),$$

για κάποιο h μικρό. Προφανώς ένα στοιχείο x μετριέται μόνο αν

$$|x - x_0| < h \Leftrightarrow \left|\frac{x - x_0}{h}\right| < 1.$$

Οπότε ο παραπάνω εκτιμητής γίνεται:

$$\hat{f}(x_0) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{h} K\left(\frac{x_0 - x_i}{h}\right) \quad με \quad K(x) = \begin{cases} \frac{1}{2} & αν\ |x| < 1 \\ 0 & αλλιώς \end{cases}.$$

Αυτή η διαδικασία εκτίμησης της συνάρτησης πυκνότητας πιθανότητας στο σημείο x_0 λέγεται μέθοδος πυρήνων για κάποιον πυρήνα $K(x)$ και το h καλείται **εύρος ζώνης** (bandwidth). Ο συγκεκριμένος πυρήνας λέγεται ομοιόμορφος και δεν είναι ο μόνος δυνατός. Έχει το μειονέκτημα ότι οι συναρτήσεις πυκνότητας πιθανότητας που εκτιμά είναι

κλιμακωτές και δεν υπάρχουν παράγωγοι σε όλα τα σημεία. Επίσης κάθε παρατήρηση μέσα στο $(x_0 - h, x_0 + h)$ έχει το ίδιο βάρος όσο μακριά και να βρίσκεται από το x_0. Αυτό μας κάνει να χρησιμοποιούμε άλλους πυρήνες όπως:

$$Gaussian\colon\ K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}, x \in \mathbb{R}$$
$$Epanechnicov\colon\ K(x) = \frac{3}{4}(1 - x^2), |x| \leq 1$$
$$Biweight\colon\ K(x) = \frac{15}{16}(1 - x^2)^2, |x| \leq 1$$
$$Triweight\colon\ K(x) = \frac{35}{32}(1 - x^2)^3, |x| \leq 1.$$

Οι πυρήνες είναι επιλεγμένοι έτσι ώστε να ικανοποιούν τις παρακάτω ιδιότητες:

i. $K(x) \geq 0\ \forall x$
ii. $\int_{-\infty}^{\infty} K(x)dx = 1$
iii. $\int_{-\infty}^{\infty} xK(x)dx = 0$
iv. $0 < \int_{-\infty}^{\infty} x^2 K(x)dx < +\infty$

Από τα δύο πρώτα προκύπτει ότι ο εκτιμητής των πυρήνων είναι σ.π.π. γιατί:

$$\hat{f}(x) \geq 0\ \ \kappa\alpha\iota$$

$$\int_{-\infty}^{+\infty} \hat{f}(x)dx = \frac{1}{Nh}\int_{-\infty}^{+\infty} \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right) dx = \frac{1}{Nh}\sum_{i=1}^{N} h \int_{-\infty}^{+\infty} K(u)du = \frac{1}{N}N = 1.$$

Ας διαλέξουμε τον Gaussian και για λόγους απλότητας ας θεωρήσουμε το h σταθερό σε κάθε διάσταση. Τότε η εκτιμήτριά μας θα γίνει:

$$\hat{f}(x_0) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{h}\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x_0 - x_i}{h})^2}.$$

Ο ορισμός του εκτιμητή της συνάρτησης πυκνότητας πιθανότητας μέσω των πυρήνων γενικεύεται εύκολα στην περίπτωση d μεταβλητών ως:

$$\hat{f}(\boldsymbol{x}) = \frac{1}{Nh^d}\sum_{i=1}^{N} K\left\{\frac{1}{h}(\boldsymbol{x} - \boldsymbol{x}_i)\right\}, \qquad \mu\varepsilon \int_{R^d} K(\boldsymbol{x})d\boldsymbol{x} = 1.$$

Ο Gaussian πυρήνας d διαστάσεων είναι η συνάρτηση:

$$K(\boldsymbol{x}) = \frac{1}{2\pi^{d/2}} e^{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{x}}.$$

Άρα ο εκτιμητής στην πολυμεταβλητή περίπτωση για τον Gaussian πυρήνα είναι ο:

$$\hat{f}(\boldsymbol{x}) = \frac{1}{Nh^d(2\pi)^{\frac{d}{2}}}\sum_{i=1}^{N} e^{-\frac{1}{2h^2}(\boldsymbol{x} - \boldsymbol{x}_i)^2}.$$

Τελικά ο ταξινομητής των πυρήνων, ξεκινώντας από τον ταξινομητή Bayes, γίνεται:

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) = \underset{k}{\operatorname{argmax}}\, P(\mathbf{X} = \boldsymbol{x}^* \mid Y = k)P(Y = k) \approx$$

$$\approx \underset{k}{\operatorname{argmax}} \frac{1}{\sum_{i=1}^{N} I(y_i = k)\, h^d (2\pi)^{\frac{d}{2}}} \sum_{i=1}^{N} I(y_i = k) e^{-\frac{1}{2h^2}(\boldsymbol{x}^* - \boldsymbol{x}_i)^2} \frac{\sum_{i=1}^{N} I(y_i = k)}{N}$$

$$= \underset{k}{\operatorname{argmax}} \sum_{i=1}^{N} I(y_i = k) e^{-\frac{1}{2h^2}(\boldsymbol{x}^* - \boldsymbol{x}_i)^2} = \hat{y}_{Kernel}.$$

Τα N της δεσμευμένης συνάρτησης πυκνότητας πιθανότητας αντικαταστάθηκαν από $\sum_{i=1}^{N} I(y_i = k)$, δηλαδή το πλήθος παρατηρήσεων της κλάσης. Στην τελευταία εξίσωση απλοποιήσαμε τα $\sum_{i=1}^{N} I(y_i = k)$ σε αριθμητή και παρονομαστή και ότι δεν εξαρτάται από το k. Όπως φαίνεται από τον τελικό τύπο, κάθε φορά που θα επιχειρήσουμε να προβλέψουμε την κλάση μιας παρατήρησης θα πρέπει να κάνουμε c φορές την παραπάνω διαδικασία.

Στην R ο ταξινομητής της μεθόδου των πυρήνων βρίσκεται υλοποιημένος στη βιβλιοθήκη "kernlab" των Karatzoglou, Smola, Hornik, & Zeileis (2004). Η ταξινόμηση γίνεται με την εντολή "gausspr". Τρέξαμε την ταξινόμηση ως:

```
kernelCl<-gausspr(x= dataset.train[,1:4], y= dataset.train[,5], type="classification",
                  kernel="rbfdot")
```

Το όρισμα *x* είναι ένας πίνακας με τις επεξηγηματικές μεταβλητές του μοντέλου, το *y* ένα διάνυσμα τύπου factor με τις αντίστοιχες εξαρτημένες μεταβλητές, το *type* ορίζει τη διαδικασία που θα εκτελέσει η συνάρτηση "gausspr" και παίρνει τις τιμές "classification" για ταξινόμηση και "regression" για παλινδρόμηση. Τέλος το όρισμα *kernel* ορίζει το είδος πυρήνα που θα χρησιμοποιηθεί με "rbfdot" να είναι ο Gaussian πυρήνας. Οι παράμετροι του πυρήνα υπολογίζονται αυτόματα στη συνάρτηση "gausspr" για ταξινόμηση μέσω της προσέγγισης Laplace. Ο ταξινομητής αυτός πέτυχε ακρίβεια 0.941 στο δείγμα Iris (βλ. Κεφάλαιο 5).

2.2.3. Η κατάρα της διαστατικότητας

Τόσο η μέθοδος K-πλησιέστερων γειτόνων όσο και η ταξινόμηση μέσω πυρήνων δρουν τοπικά. Ο ταξινομητής K- πλησιέστερων γειτόνων εστιάζει σε μια σφαίρα ακτίνας

$$L = \max\big(\rho(\boldsymbol{a}_i, \boldsymbol{x}^*)\big)\, με\, i = 1, 2, \dots, K,$$

ενώ ο ταξινομητής μέσω πυρήνων όσο ορίζει το h και ο πυρήνας του. Το φαινόμενο που θα αναλυθεί παρουσιάζεται σύμφωνα με το βιβλίο των Hastie, Tibshirani, & Friedman (2013, p. 22). Έστω ότι τα δεδομένα μας είναι διάστασης d και είναι διασκορπισμένα ομοιόμορφα σε έναν μοναδιαίο υπερκύβο d διαστάσεων. Έστω τώρα ότι θέλουμε να φτιάξουμε μια γειτονιά, μέσω ενός μικρότερου υπερκύβου, που να περιέχει ένα ποσοστό r των παρατηρήσεων για τη μέθοδο των K-πλησιέστερων γειτόνων. Τότε το αναμενόμενο μήκος της ακμής του υπερκύβου θα είναι $e_d(r) = r^{\frac{1}{d}}$. Για 10 διαστάσεις, αν θέλουμε να συμπεριλάβουμε το 1%

των παρατηρήσεων θα πρέπει να χρησιμοποιήσουμε υπερκύβο ακμής $e_{10}(0.01) = 0.01^{\frac{1}{10}} = 0.63$ και επειδή ο συνολικός υπερκύβος θεωρήθηκε μοναδιαίος, για να συμπεριλάβουμε το 1% των δεδομένων θα χρειαστούμε για μήκος το 63% της κάθε διάστασης του συνολικού υπερκύβου. Προφανώς η τοπικότητα της μεθόδου χάνεται. Αν μειώσουμε το r, θα μειωθούν οι παρατηρήσεις σε κάθε γειτονιά και η διακύμανση του μοντέλου θα αυξηθεί.

Ανάλογα και για τη μέθοδο των πυρήνων: Η πυκνότητα των παρατηρήσεων σε κάθε πυρήνα θα είναι ανάλογη με το $N^{\frac{1}{d}}$ με N το μέγεθος του δείγματος. Αν θεωρήσουμε ότι με ένα δείγμα μεγέθους $N_1 = 100$ έχουμε πυκνές παρατηρήσεις σε έναν πυρήνα για ένα πρόβλημα μίας διάστασης για την εκτίμηση ενός σημείου, θα χρειαστούμε $N_{10} = 100^{10}$ παρατηρήσεις για να έχουμε έναν το ίδιο πυκνό πυρήνα στις 10 διαστάσεις. Αυτό το φαινόμενο καλείται **κατάρα της διαστατικότητας** (curse of dimensionality) και είναι ο λόγος που οι μη παραμετρικές μέθοδοι υπολειτουργούν στις υψηλές διαστάσεις. Γενικά τα μη παραμετρικά μοντέλα χρειάζονται περισσότερα δεδομένα από τα παραμετρικά για να μειώσουν το σφάλμα λόγω διακύμανσης. Η κατάρα της διαστατικότητας δεν εμφανίζεται σε τέτοιο βαθμό στα παραμετρικά μοντέλα, τα οποία όμως είναι πιο ευαίσθητα στα σφάλματα λόγω μεροληψίας μιας και κάνουν υποθέσεις για τη συμπεριφορά των δεδομένων.

2.3. Τρείς παραμετρικοί ταξινομητές που ξεκινούν από τον Bayes

Στην ενότητα αυτή θα δούμε τρεις παραμετρικούς ταξινομητές, που σημαίνει ότι για την υλοποίησή τους θα υποθέσουμε ότι οι μεταβλητές μας ακολουθούν συγκεκριμένη κατανομή. Για τον υπολογισμό των παραμέτρων των κατανομών αυτών επιλέγονται οι εκτιμήτριες που προκύπτουν από τη μεγιστοποίηση της συνάρτησης πιθανοφάνειας. Οι εκτιμήτριες αυτές προτιμώνται λόγω των ασυμπτωτικών ιδιοτήτων τους που αναφέρονται στο Παράρτημα *π.2*. Εδώ καλό είναι να τονιστεί ότι η συνέπεια εκτιμήτριας διαφέρει από τη συνέπεια ταξινομητή που ορίσαμε στην εισαγωγή.

2.3.1. Λογιστική παλινδρόμηση

Έστω ένα ζεύγος παρατηρήσεων x, y με το y να προέρχεται από μια δίτιμη διακριτή μεταβλητή Y με τιμές 0, 1. Επηρεασμένοι από τη θεωρία της παλινδρόμησης θα μπορούσαμε να κατασκευάσουμε ένα απλό γραμμικό μοντέλο πρόβλεψης της τιμής y δεδομένης της παρατήρησης x της μορφής

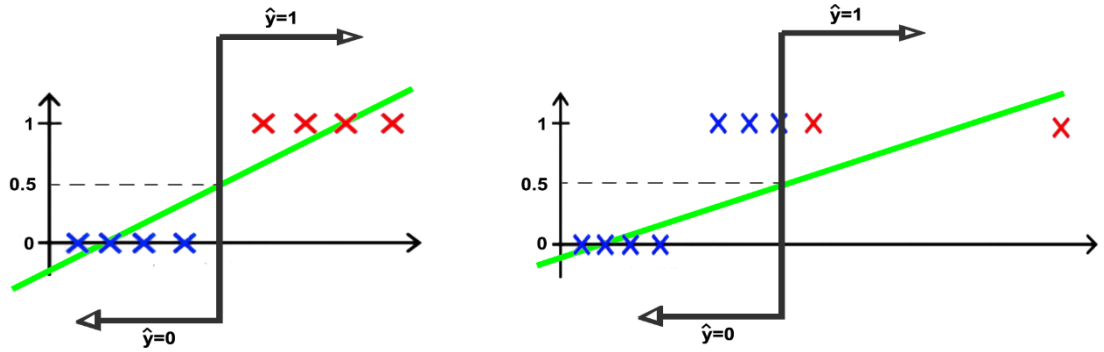
$$\hat{f} = \beta_0 + \beta_1 x.$$

Βέβαια αυτή η προσέγγιση μας δημιουργεί τα παρακάτω προβλήματα:

Πρώτων, δίνουμε την ελευθερία στην πρόβλεψή μας να πάρει τιμές στο $\mathbb{R}$, κάτι που μας δημιουργεί λογικά ζητήματα γιατί δεν μπορούμε να εξηγήσουμε το τι είναι το νούμερο που προβλέψαμε. Δεν είναι η τιμή του Y μιας και το Y παίρνει μόνο τις τιμές 0 και 1. Θα μπορούσαμε να ταξινομήσουμε την παρατήρηση στην κλάση $\hat{y} = 1$ αν $\hat{f} > 0.5$ και στην

κλάση $\hat{y} = 0$ αλλιώς. Όμως το ίδιο πρόβλημα εξακολουθεί να υπάρχει γιατί τότε υπονοούμε ότι το $\hat{f}$ είναι η πιθανότητα της κλάσης 1. Κάτι τέτοιο δεν ισχύει γιατί το $\hat{f}$ είναι ελεύθερο να πάρει τιμές τόσο μεγαλύτερες του 1 όσο και αρνητικές.

Δεύτερων η γραμμική παλινδρόμηση είναι πολύ ευαίσθητη σε ακραίες τιμές στη διακριτή περίπτωση όπως φαίνεται στο Διάγραμμα 2.2.



Διάγραμμα 2.2: Στο (a) παρατηρούμε την περίπτωση που η γραμμική παλινδρόμηση λύνει το πρόβλημα της ταξινόμησης ικανοποιητικά. Οι παρατηρήσεις όμως είναι ιδανικές δεδομένου ότι οι δύο κλάσεις, εκτός από πλήρως διαχωρίσιμες, φαίνονται να έχουν την ίδια διασπορά ως προς τη μεταβλητή x. Στο (b) βλέπουμε πόσο εύκολα με μία μόνο ακραία τιμή η γραμμική παλινδρόμηση ταξινομεί μη ικανοποιητικά.

Το πρόβλημα λύνεται μέσα από τη θεωρία παλινδρόμησης με τη χρήση της **λογιστικής παλινδρόμησης** (logistic regression) που αποτελεί ένα γενικευμένο γραμμικό μοντέλο και ως ταξινομητής είναι παραμετρικός και γραμμικός. Ορίζουμε τον **λόγο συμπληρωματικών πιθανοτήτων** (odds) ως:

$$odds = \frac{P(Y = 1 \mid \mathrm{X} = x^*)}{1 - P(Y = 1 \mid \mathrm{X} = x^*)}.$$

Στόχος μας είναι να συνδέσουμε την πιθανότητα $P(Y = 1 \mid \mathrm{X} = x^*)$ με το X. Η σχέση αυτή δε θα μπορούσε να είναι γραμμική όπως είδαμε παραπάνω, οπότε η σχέση τους θα πρέπει να τεθεί μέσω μιας **συνάρτησης σύνδεσης** (link function) δηλαδή μέσω κάποιου g τέτοιου ώστε $g(P(Y = 1 \mid \mathrm{X} = x^*)) = \beta_0 + \beta_1 x^*$. Επιλέγουμε για συνάρτηση σύνδεσης την συνάρτηση **λογαρίθμου λόγου συμπληρωματικών πιθανοτήτων** (logit) με τύπο:

$$logit(a) = \log\left(\frac{a}{1-a}\right), \qquad για\ a \in [0,1].$$

Η σχέση των $P(Y = 1 \mid \mathrm{X} = x^*),\ X$ γίνεται:

$$logit(P(Y = 1 \mid \mathrm{X} = x^*)) = \log\frac{P(Y = 1 \mid \mathrm{X} = x^*)}{1 - P(Y = 1 \mid \mathrm{X} = x^*)} = \beta_0 + \beta_1 x^*.$$

Με αυτό τον τρόπο εμφανίσαμε τον λόγο συμπληρωματικών πιθανοτήτων μέσα στη σχέση μας, κάτι που είναι καλό μιας και υπάρχει φυσικό νόημα. Η αντίστροφη συνάρτηση της logit είναι η **λογιστική** ή **σιγμοειδής συνάρτηση** (logistic ή sigmoid) με τύπο

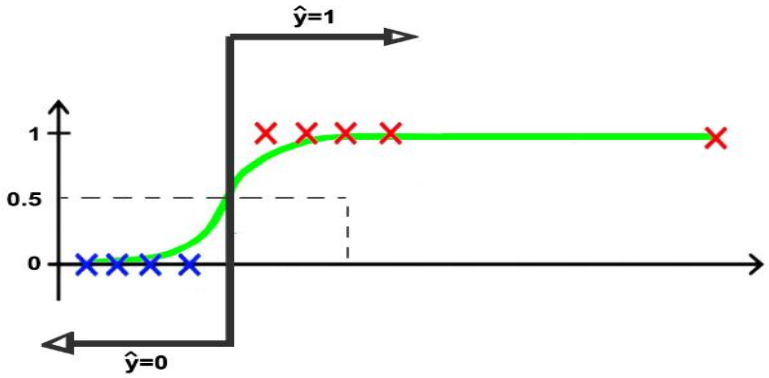
$$logistic(a) = \frac{1}{1 + e^{-a}} = \frac{e^a}{1 + e^a},$$

οπότε με τη χρήση της λογιστικής συνάρτησης λύνουμε ως προς την πιθανότητα $P(Y = 1 \mid \mathrm{X} = x^*)$

$$logistic(logit(P(Y = 1 \mid \mathrm{X} = x^*))) = logistic(\beta_0 + \beta_1 x^*)$$

$$\Rightarrow P(Y = 1 \mid \mathrm{X} = x^*) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x^*)}} = \frac{e^{(\beta_0 + \beta_1 x^*)}}{1 + e^{(\beta_0 + \beta_1 x^*)}}.$$

Η ιδιότητα της λογιστικής συνάρτησης να τείνει στο 1 όταν το x τείνει στο $+\infty$ και στο 0 όταν το x τείνει στο $-\infty$ την κάνει εξαιρετική επιλογή για να προσδιορίζει πιθανότητα, επίσης όπως φαίνεται και στο Διάγραμμα 2.3 διορθώνεται και η ευαισθησία στις ακραίες τιμές. Θα μπορούσαμε να ταξινομήσουμε την παρατήρηση στην κλάση $\hat{y} = 1$ αν $P(y = 1 \mid \mathrm{x} = x_0) > 0.5$ και στην κλάση $\hat{y} = 0$ αλλιώς. Το παράδειγμα του Διαγράμματος 2.2 φαίνεται προσαρμοσμένο στο Διάγραμμα 2.3 μέσω της λογιστικής παλινδρόμησης.



Διάγραμμα 2.3: Η λογιστική παλινδρόμηση, σε αντίθεση με τη γραμμική, δεν είναι ευαίσθητη σε ακραίες τιμές.

Ας γενικεύσουμε το πρόβλημα σε d διαστάσεις και c κλάσεις. Έστω σύνολο δεδομένων $\boldsymbol{D}$ με μεταβλητές $X_1, X_2, \dots, X_d, Y$ με το Y να παίρνει τις τιμές $1, 2, \dots, c$ και έστω $\boldsymbol{x_i} = (x_{i1}, x_{i2}, \dots, x_{id})$ η i παρατήρηση. Ξεκινάμε πάλι από τον ταξινομητή Bayes.

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*).$$

Σκοπός μας είναι να υπολογίσουμε τις πιθανότητες $P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*)$. Η λογιστική παλινδρόμηση για c κλάσεις θέτει ως βάση υπολογισμού τη δεσμευμένη πιθανότητα του $y = c$ και υποθέτει ότι ο λογάριθμος των λόγων της κάθε δεσμευμένης πιθανότητας $Y = k$ με την αντίστοιχη $Y = c$ είναι κάποιος γραμμικός συνδυασμός των ανεξάρτητων μεταβλητών $X_1, X_2, \dots, X_d$ δηλαδή:

$$log\frac{P(Y = 1 \mid \mathbf{X} = \boldsymbol{x}^*)}{P(Y = c \mid \mathbf{X} = \boldsymbol{x}^*)} = \beta_{10} + \beta_{11}\boldsymbol{x}^*{}_1 + \cdots + \beta_{1d}\boldsymbol{x}^*{}_d$$

$$log\frac{P(Y = 2 \mid \mathbf{X} = \boldsymbol{x}^*)}{P(Y = c \mid \mathbf{X} = \boldsymbol{x}^*)} = \beta_{20} + \beta_{21}\boldsymbol{x}^*{}_1 + \cdots + \beta_{2d}\boldsymbol{x}^*{}_d$$

$$\vdots$$

$$log\frac{P(Y = c-1 \mid \mathbf{X} = \boldsymbol{x}^*)}{P(Y = c \mid \mathbf{X} = \boldsymbol{x}^*)} = \beta_{(c-1)0} + \beta_{(c-1)1}\boldsymbol{x}^*{}_1 + \cdots + \beta_{(c-1)d}\boldsymbol{x}^*{}_d.$$

Λύνουμε ως προς $P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*)$:

$$P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) = \frac{e^{\beta_{k0} + \beta_{k1}\boldsymbol{x}^*{}_1 + \cdots + \beta_{kd}\boldsymbol{x}^*{}_d}}{1 + \sum_{l=1}^{c-1} e^{\beta_{l0} + \beta_{l1}\boldsymbol{x}^*{}_1 + \cdots + \beta_{ld}\boldsymbol{x}^*{}_d}}, \qquad k = 1, 2, \ldots, c-1$$

$$P(Y = c \mid \mathbf{X} = \boldsymbol{x}^*) = \frac{1}{1 + \sum_{l=1}^{c-1} e^{\beta_{l0} + \beta_{l1}\boldsymbol{x}^*{}_1 + \cdots + \beta_{ld}\boldsymbol{x}^*{}_d}}.$$

Ο ταξινομητής Bayes γίνεται:

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) \approx \underset{k}{\operatorname{argmax}} \begin{cases} \dfrac{e^{\boldsymbol{\beta}_k \boldsymbol{x}^*}}{1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta}_l \boldsymbol{x}^*}}, & k = 1, 2, \ldots, c-1 \\ \dfrac{1}{1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta}_l \boldsymbol{x}^*}}, & k = c. \end{cases} = \hat{y}_{LR},$$

χρησιμοποιώντας τον διανυσματικό συμβολισμό:

$$\beta_{k0} + \beta_{k1}\boldsymbol{x}^*{}_1 + \cdots + \beta_{kd}\boldsymbol{x}^*{}_d = \boldsymbol{\beta}_k \boldsymbol{x}^*.$$

Δουλειά μας τώρα είναι να εκτιμήσουμε τα $\boldsymbol{\beta}$ με βάση το σύνολο δεδομένων μας $\boldsymbol{D}$. Η μεταβλητή Y ακολουθεί κατηγορική κατανομή με συνάρτηση μάζας πιθανότητας:

$$f(Y|\boldsymbol{\beta}) = \prod_{k=1}^{c} P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*)^{I(Y=k)}.$$

Τα $\boldsymbol{\beta}$ θα υπολογιστούν μεγιστοποιώντας τη συνάρτηση πιθανοφάνειας της κατηγορικής κατανομής:

$$\begin{aligned} L(\boldsymbol{\beta}) &= \prod_{i=1}^{N} \left(\prod_{k=1}^{c} P(Y = k \mid \mathbf{X} = \boldsymbol{x_i})^{I(y_i=k)} \right) \\ &= \prod_{i=1}^{N} \left(\prod_{k=1}^{c-1} \left(\frac{e^{\boldsymbol{\beta}_k \boldsymbol{x_i}}}{1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta}_l \boldsymbol{x_i}}} \right)^{I(y_i=k)} \left(\frac{1}{1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta}_l \boldsymbol{x_i}}} \right)^{I(y_i=c)} \right). \end{aligned}$$

Τα $\boldsymbol{\beta}$ που μεγιστοποιούν την παραπάνω ποσότητα θα μεγιστοποιούν και το λογάριθμό της και το αντίστροφο άρα λαμβάνοντας τον λογάριθμο του L καταλήγουμε να ψάχνουμε τα $\boldsymbol{\beta}$ μέσω του τύπου της συνάρτησης λογαριθμο-πιθανοφάνειας:

$$\begin{aligned} l(\boldsymbol{\beta}) &= \sum_{i=1}^{N} \left(\sum_{k=1}^{c-1} \left(I(y_i = k) \left(\boldsymbol{\beta}_k \boldsymbol{x_i} - \ln\left(1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta}_l \boldsymbol{x_i}} \right) \right) \right) - I(y_i = c) \ln\left(1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta}_l \boldsymbol{x_i}} \right) \right) \\ &= \sum_{i=1}^{N} \left(\sum_{k=1}^{c-1} [I(y_i = k)(\boldsymbol{\beta}_k \boldsymbol{x_i})] - \left(\sum_{k=1}^{c} I(y_i = k) \right) \ln\left(1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta}_l \boldsymbol{x_i}} \right) \right) \\ &= \sum_{i=1}^{N} \left(\sum_{k=1}^{c-1} \left(I(y_i = k)(\boldsymbol{\beta}_k \boldsymbol{x_i}) \right) - \ln\left(1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta}_l \boldsymbol{x_i}} \right) \right), \end{aligned}$$

γιατί

$$\sum_{k=1}^{c} I(y_i = k) = 1$$

για κάθε παρατήρηση.

Τα $\boldsymbol{\beta}$ που μεγιστοποιούν την παραπάνω συνάρτηση βρίσκονται χρησιμοποιώντας τη μέθοδο Newton-Raphson η οποία περιγράφεται στο Παράρτημα *π.1.2*.

Στην R η λογιστική παλινδρόμηση για παραπάνω από δύο κλάσεις είναι υλοποιημένη στη βιβλιοθήκη "nnet" των Venables & Riplay (2002) και καλείται με την εντολή "multinom". Η παλινδρόμηση εκτελέστηκε ως εξής:

```
logistic<-multinom(formula=target~., data=dataset.train)
```

Η λογιστική παλινδρόμηση πέτυχε ακρίβεια 0.934 στο δείγμα Iris (βλ. Κεφάλαιο 5).

2.3.2. Naïve Bayes

Ο Naïve Bayes είναι ένας απλός παραμετρικός και γραμμικός ταξινομητής που ξεκινάει από τον ταξινομητή Bayes και κάνει την υπόθεση ότι οι επεξηγηματικές μεταβλητές $X_1,\ X_2, \dots, X_d$ είναι ανεξάρτητες μέσα στην κάθε κλάση k. Έστω $\boldsymbol{X} = (X_1,\ X_2, \dots, X_d)$ και $\boldsymbol{x}^* \in \mathbb{R}^d$ η προς ταξινόμηση παρατήρηση, τότε θεωρούμε την παρακάτω σχέση που καλείται **υπόθεση Naïve Bayes** (Naïve Bayes assumption):

$$P(\mathbf{X} = \boldsymbol{x}^* \mid Y = k) = \prod_{j=1}^{d} P(X_j = {x^*}_j \mid Y = k).$$

Άρα ο ταξινομητής Bayes γίνεται:

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) = \underset{k}{\operatorname{argmax}}\, P(\mathbf{X} = \boldsymbol{x}^* \mid Y = k)P(Y = k)$$
$$\approx \underset{k}{\operatorname{argmax}}\, P(Y = k) \prod_{j=1}^{d} P(X_j = x_j^* \mid Y = k) \ = \hat{y}_{NB.}$$

Ο ταξινομητής αυτός, κάνει μια τόσο ισχυρή υπόθεση αλλά απλοποιεί πολύ το πρόβλημα μιας και όλες οι παραπάνω πιθανότητες υπολογίζονται εύκολα από την εκτιμήτρια μέγιστης πιθανοφάνειας. Οι πιθανότητες $P(y = k)$ προκύπτουν από τις σχετικές συχνότητες των k μιας και το Y ακολουθεί κατηγορική κατανομή και η σχετική συχνότητα ως εκτιμήτρια της πιθανότητας εμφάνισης της κάθε κατηγορίας προκύπτει από τη μεγιστοποίηση της συνάρτησης πιθανοφάνειας. Το καθένα από τα $P(X_j = x_j^* \mid Y = k)$ προκύπτει από τη μεγιστοποίηση της συνάρτησης πιθανοφάνειας της κάθε μεταβλητής X_j του υποπληθυσμού του δείγματος για $Y = k$.

Ο ταξινομητής Naïve Bayes δεν προϋποθέτει ότι όλες οι επεξηγηματικές μεταβλητές ακολουθούν την ίδια κατανομή. Θα μπορούσε δηλαδή η μεταβλητή X_j να ακολουθεί κανονική κατανομή, η X_l κατανομή Bernoulli κ.λ.π. και να δημιουργηθεί το μοντέλο:

$$\hat{y} = \underset{k}{\operatorname{argmax}} \frac{\sum_{i=1}^{N} I(y_i = k)}{N} \cdot \dots \cdot \frac{1}{\sqrt{2\pi\hat{\sigma}_i{}^2}} e^{-\frac{1}{2}\left(\frac{x_j^* - \hat{\mu}_k}{\hat{\sigma}_k}\right)^2} \cdot \dots \cdot \frac{\sum_{i=1}^{N} I(x_{il} = x_l^* | y_i = k)}{\sum_{i=1}^{N} I(y_i = k)} \cdot \dots$$

$$με \quad \hat{\mu}_k = \frac{\sum_{i=1}^{N} I(y_i = k) x_{ik}}{\sum_{i=1}^{N} I(y_i = k)} \quad και \quad \hat{\sigma}_k{}^2 = \frac{\sum_{i=1}^{N} I(y_i = k)(x_{ik} - \hat{\mu}_k)^2}{\sum_{i=1}^{N} I(y_i = k)}.$$

Παρατηρούμε ότι για τη μεταβλητή που ακολουθεί κανονική κατανομή έχουμε αντικαταστήσει τη συνάρτηση πυκνότητας πιθανότητας, η οποία για την προς ταξινόμηση παρατήρηση δε μας δίνει πιθανότητα μιας και σε κάθε συνεχή κατανομή η πιθανότητα παρατήρησης μιας συγκεκριμένης τιμής είναι 0. Παρόλα αυτά, όπως αναφέρουν οι John & Langley (1995) αν πούμε $g(x; \mu, \sigma)$ τη συνάρτηση πυκνότητας πιθανότητας μιας συνεχούς κατανομής, η μεταβλητή βρίσκεται μέσα σε κάποιο διάστημα:

$$P(x \leq X \leq x + \Delta) = \int_{x}^{x+\Delta} g(x; \mu, \sigma) dx.$$

Από τον ορισμό της παραγώγισης

$$\lim_{\Delta \to 0} \frac{P(x \leq X \leq x + \Delta)}{\Delta} = g(x; \mu, \sigma).$$

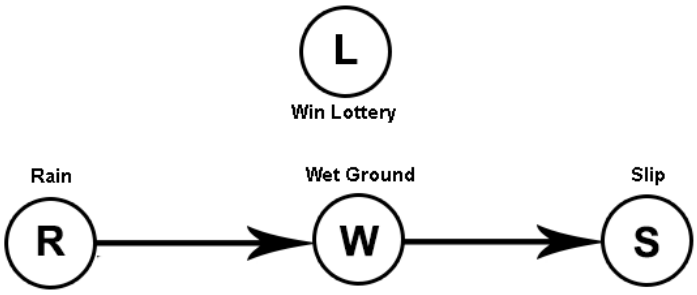
Οπότε για πολύ μικρή σταθερά Δ ισχύει

$$P(X = x) \approx \Delta g(x; \mu, \sigma).$$

Επειδή το Δ υπάρχει για όλες τις κλάσεις k δεν επηρεάζει τη βελτιστοποίηση και απαλείφεται. Παρόλη την ελευθερία στην επιλογή των κατανομών των μεταβλητών, έχουν επικρατήσει στη βιβλιογραφία συγκεκριμένες ορολογίες σε κάποιες περιπτώσεις που όλες οι μεταβλητές ακολουθούν την ίδια κατανομή λόγω των εφαρμογών της κάθε προσέγγισης.

- Στην περίπτωση που όλες οι επεξηγηματικές μεταβλητές ακολουθούν κανονική κατανομή, ο ταξινομητής καλείται Gaussian Naïve Bayes.
- Στην περίπτωση που όλες οι επεξηγηματικές μεταβλητές ακολουθούν πολυωνυμική κατανομή ή κατανομή Bernoulli ο ταξινομητής καλείται Multinomial ή Bernoulli Naïve Bayes αντίστοιχα.

Ο Naïve Bayes αποτελεί μια ειδική περίπτωση μιας οικογένειας ταξινομητών που καλούνται **ταξινομητές Μπεϋζιανών δικτύων** (Bayesian Network classifiers).



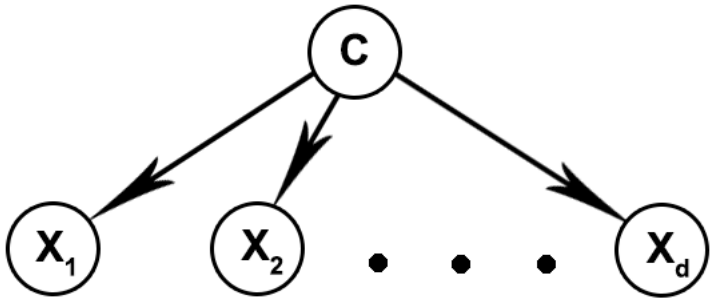
Διάγραμμα 2.4: Εδώ βλέπουμε ένα Μπεϋζιανό δίκτυο. Το οποίο μοντελοποιεί τις αιτιακές σχέσεις του ενδεχομένου να κερδίσουμε το λαχείο, να βρέξει, να είναι υγρό το πάτωμα και να γλιστρήσουμε. Όπως φαίνεται το Μπεϋζιανό δίκτυο είναι ένας κατευθυνόμενος και ακυκλικός γράφος με ακμές από την αιτία προς το αποτέλεσμα.

Ένα Μπεϋζιανό δίκτυο είναι ένα πιθανοθεωριτικό μοντέλο το οποίο εκφράζεται μέσω ενός κατευθυνόμενου και ακυκλικού γράφου με κόμβους τις μεταβλητές και ακμές τις αιτιακές σχέσεις των μεταβλητών.

Στο παράδειγμα του Διαγράμματος 2.4 έχουμε μοντελοποιήσει έτσι το πρόβλημά μας ώστε να θεωρούμε εκ των προτέρων ότι η μεταβλητή L που μας δείχνει αν κερδίσαμε ένα λαχείο είναι ανεξάρτητη από τις υπόλοιπες μεταβλητές (αν βρέχει, αν είναι υγρό το πάτωμα και αν γλιστρήσαμε). Αντιθέτως το αν γλιστρήσαμε το θεωρήσαμε εξαρτημένο από το αν ήταν υγρό το πάτωμα, το οποίο με τη σειρά του το θεωρούμε εξαρτημένο από το αν έβρεξε. Οπότε αν θέλουμε να υπολογίσουμε την από κοινού πιθανότητα των παραπάνω γράφουμε:

$$P(L, R, W, S) = P(L)P(R)P(W|R)P(S|W).$$

Αν χρησιμοποιήσουμε μια υπόθεση που προέρχεται από ένα Μπεϋζιανό δίκτυο στο πρόβλημα της ταξινόμησης (στο σημείο που χρησιμοποιήσαμε την υπόθεση ανεξαρτησίας Naïve Bayes) ο ταξινομητής καλείται γενικότερα ταξινομητής Μπεϋζιανών δικτύων. Στο Διάγραμμα 2.5 βλέπουμε το Μπεϋζιανό δίκτυο που χρησιμοποιεί ο Naïve Bayes.



Διάγραμμα 2.5: Η υπόθεση ανεξαρτησίας Naïve Bayes ως Μπεϋζιανό δίκτυο. Όλες οι επεξηγηματικές μεταβλητές έχουν αιτιακή σχέση με την εξαρτημένη αλλά όχι μεταξύ τους.

Στην R ο Naïve Bayes βρίσκεται υλοποιημένος στη βιβλιοθήκη "e1071" των (Meyer, Dimitriadou, Hornik, Weingessel, & Leisch, 2018) και καλείται με την εντολή "naiveBayes". Η υλοποίηση αυτή θεωρεί κανονικές κατανομές για όλες τις μεταβλητές άρα είναι η περίπτωση του Gaussian Naïve Baies. Η ταξινόμηση εκτελέστηκε ως:

```
GaussianNB<-naiveBayes(formula=target~., data=dataset.train, laplace = 0)
```

Το όρισμα *laplace = 0* απενεργοποιεί την **κανονικοποίηση Laplace** (Laplace smoothing) από τη διαδικασία Naïve Bayes ώστε η ταξινόμηση να γίνει με τον τρόπο που παρουσιάστηκε η θεωρία. Ο ταξινομητής Naïve Bayes πέτυχε ακρίβεια 0.962 στο δείγμα Iris (βλ. Κεφάλαιο 5).

2.3.3. Γραμμική και τετραγωνική διαχωριστική ανάλυση

Ο **ταξινομητής γραμμικής διαχωριστικής ανάλυσης** (Linear discriminant analysis classifier) είναι ένας παραμετρικός και γραμμικός ταξινομητής που αποτελεί γενίκευση του **γραμμικού διαχωρισμού του Fisher** και ένα από τα βασικότερα χαρακτηριστικά του είναι ότι μπορεί να χρησιμοποιηθεί για να μειώσει τις διαστάσεις του προβλήματος από τον αριθμό των μεταβλητών d σε $c-1$ όπου c ο αριθμός των κλάσεων του Y. Έστω $\boldsymbol{X}=(X_1,\ X_2,\ldots,X_d)$ και $\boldsymbol{x}^*\in\mathbb{R}^d$ η προς ταξινόμηση παρατήρηση. Ξεκινάμε από τον ταξινομητή Bayes

$$\hat{y}_{Bayes}=\underset{k}{\operatorname{argmax}}\,P(Y=k\mid \mathbf{X}=\boldsymbol{x}^*)=\underset{k}{\operatorname{argmax}}\,P(\mathbf{X}=\boldsymbol{x}^*\mid Y=k)P(Y=k),$$

θεωρούμε ότι το $\boldsymbol{X}=(X_1,\ X_2,\ldots,X_d)$ ακολουθεί μέσα σε κάθε κλάση k πολυδιάστατη κανονική κατανομή d διαστάσεων με διαφορετικές μέσες τιμές αλλά με κοινό πίνακα συνδιακύμανσης δηλαδή

$$\boldsymbol{X}_k\sim N(\boldsymbol{\mu}_k,\boldsymbol{\Sigma})\ \ \gamma\iota\alpha\ k=1,2,\ldots,c.$$

Όπως και στην ενότητα του Naïve Bayes μέσα στο $argmax$ θα θεωρούμε ότι για τη δεσμευμένη πιθανότητα ισχύει

$$P(\boldsymbol{X}=\boldsymbol{x}^*\mid Y=k)=g(\boldsymbol{x}^*;\boldsymbol{\mu}_k,\boldsymbol{\Sigma}),$$

δηλαδή ισούται με το ύψος της συνάρτησης πυκνότητας πιθανότητας στο σημείο $\boldsymbol{x}^*$. Συγκεκριμένα

$$P(\mathbf{X}=\boldsymbol{x}^*\mid Y=k)=\frac{1}{(2\pi)^{\frac{d}{2}}\left|\widehat{\boldsymbol{\Sigma}}\right|^{\frac{1}{2}}}e^{-\frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)},$$

με $\widehat{\boldsymbol{\mu}}_k=\frac{\sum_{i=1}^N I(y_i=k)\boldsymbol{x}_i}{\sum_{i=1}^N I(y_i=k)}$, $\widehat{\boldsymbol{\Sigma}}=\frac{\sum_{i=1}^N(\boldsymbol{x}_i-\widehat{\boldsymbol{\mu}})(\boldsymbol{x}_i-\widehat{\boldsymbol{\mu}})^T}{N}$ $\kappa\alpha\iota$ $\widehat{\boldsymbol{\mu}}=\frac{\sum_{i=1}^N\boldsymbol{x}_i}{N}$.

Ο ταξινομητής γίνεται

$$\begin{aligned}\hat{y}_{LDA}&=\underset{k}{\operatorname{argmax}}\,P(Y=k)\frac{1}{(2\pi)^{\frac{d}{2}}\left|\widehat{\boldsymbol{\Sigma}}\right|^{\frac{1}{2}}}e^{-\frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)}\\&=\underset{k}{\operatorname{argmax}}\,P(Y=k)e^{-\frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)}\\&=\underset{k}{\operatorname{argmax}}\log\left[P(Y=k)e^{-\frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)}\right]\\&=\underset{k}{\operatorname{argmax}}\log[P(Y=k)]-\frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)\\&=\underset{k}{\operatorname{argmax}}\ \log[P(Y=k)]-\frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\left(\widehat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{x}^*-\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_k\right)\\&=\underset{k}{\operatorname{argmax}}\log[P(Y=k)]-\frac{1}{2}\boldsymbol{x}^{*T}\widehat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{x}^*+\frac{1}{2}\boldsymbol{x}^{*T}\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_k+\frac{1}{2}\widehat{\boldsymbol{\mu}}_k{}^T\widehat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{x}^*\\&\quad-\frac{1}{2}\widehat{\boldsymbol{\mu}}_k{}^T\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_k.\end{aligned}$$

Ο $\widehat{\boldsymbol{\Sigma}}$ είναι συμμετρικός άρα και ο $\widehat{\boldsymbol{\Sigma}}^{-1}$ είναι συμμετρικός. Επίσης το $\boldsymbol{x}^{*T}\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_k$ είναι αριθμός γιατί $\boldsymbol{x}^{*T}\in\mathbb{R}^{1xd}$, $\widehat{\boldsymbol{\Sigma}}^{-1}\in\mathbb{R}^{dxd}$ και $\widehat{\boldsymbol{\mu}}_k\in\mathbb{R}^{dx1}$ άρα

$$\boldsymbol{x}^{*T}\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_k = (\boldsymbol{x}^{*T}\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_k)^T = \widehat{\boldsymbol{\mu}}_k{}^T(\boldsymbol{x}^{*T}\widehat{\boldsymbol{\Sigma}}^{-1})^T = \widehat{\boldsymbol{\mu}}_k{}^T(\widehat{\boldsymbol{\Sigma}}^{-1})^T\boldsymbol{x}^* = \widehat{\boldsymbol{\mu}}_k{}^T\widehat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{x}^*.$$

Επίσης η ποσότητα $\frac{1}{2}\boldsymbol{x}^{*T}\widehat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{x}^*$ είναι ανεξάρτητη της κλάσης k άρα δεν παίζει ρόλο στη βελτιστοποίηση οπότε εξαιρείται. Ο ταξινομητής παίρνει τη μορφή

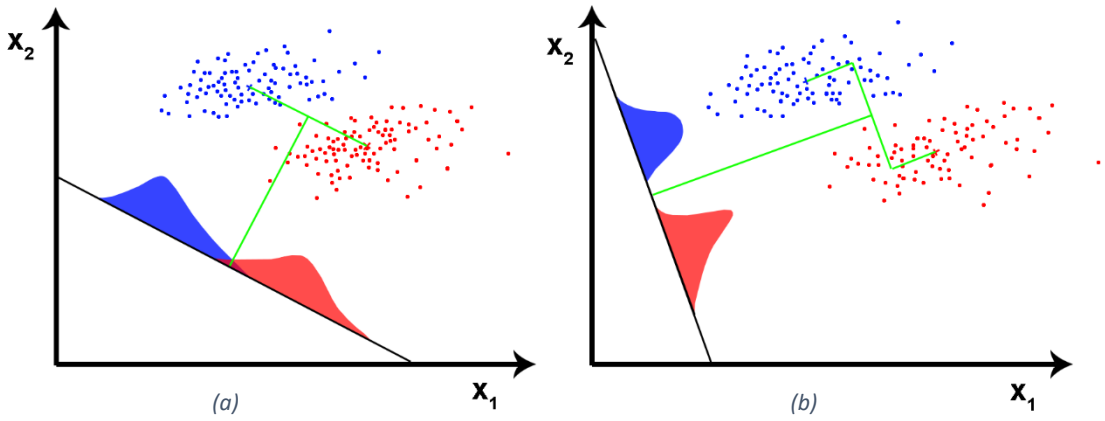
$$\hat{y}_{LDA} = \underset{k}{\operatorname{argmax}}\, log[P(Y=k)] + \boldsymbol{x}^{*T}\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_k - \frac{1}{2}\widehat{\boldsymbol{\mu}}_k{}^T\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_k,$$

με τους τύπους των $\widehat{\boldsymbol{\mu}}_k$, $\widehat{\boldsymbol{\mu}}$ *και* $\widehat{\boldsymbol{\Sigma}}$ να προκύπτουν από τη μεγιστοποίηση της συνάρτησης πιθανοφάνειας της πολυδιάστατης κανονικής κατανομής. Όπως και πριν η πιθανότητα της κάθε κλάσης θεωρείται η

$$\hat{P}(Y=k) = \frac{\sum_{i=1}^{N} I(y_i = k)}{N},$$

η οποία προκύπτει από τη μεγιστοποίηση της συνάρτησης πιθανοφάνειας της κατηγορικής κατανομής.

Οι μέσες τιμές των πολυδιάστατων κανονικών κατανομών όλων των κλάσεων c ορίζουν ένα υπερεπίπεδο $c-1$ διαστάσεων στο οποίο μπορούμε να προβάλουμε τα δεδομένα. Όμως, όπως βλέπουμε στο Διάγραμμα 2.6 (a) για το παράδειγμα δύο κλάσεων και δύο διαστάσεων, το υπερεπίπεδο δεν θα είναι βέλτιστο. Θέλουμε λοιπόν να βρούμε ένα υπερεπίπεδο που να μεγιστοποιεί την απόσταση μεταξύ των μέσων τιμών αλλά να ελαχιστοποιεί τη διασπορά της κάθε κλάσης όταν προβάλλουμε το δείγμα μας σε αυτό, θέλουμε δηλαδή να βρούμε το υπερεπίπεδο σύμφωνα με το Διάγραμμα 2.6 (b). Η διαδικασία γίνεται σύμφωνα με τα βιβλία των Bishop (2006, pp. 191-192) και Hastie, Tibshirani, & Friedman (2013, pp. 113-116).



Διάγραμμα 2.6: Ενώ η γραμμή που ενώνει τις μέσες τιμές ορίζει τη διεύθυνση της μέγιστης απόστασης μεταξύ των μέσων τιμών, τα δεδομένα δεν διαχωρίζονται καλά λόγω της συνδιακύμανσης (a). Το υπερεπίπεδο που ορίζεται από τη γραμμική διαχωριστική ανάλυση μεγιστοποιεί τη διαχωριστικότητα των δεδομένων (b).

Ορίζουμε τη **συνδιακύμανση εντός κλάσεων** (within-class covariance) ως το άθροισμα των επιμέρους συνδιακυμάνσεων κάθε κλάσης δηλαδή:

$$\boldsymbol{S}_W = \sum_{k=1}^{c}\sum_{i=1}^{N} I(y_i = k)(\boldsymbol{x}_i - \boldsymbol{m}_k)(\boldsymbol{x}_i - \boldsymbol{m}_k)^T$$

$$\boldsymbol{m}_k = \frac{\sum_{i=1}^{N} I(y_i = k)\boldsymbol{x}_i}{N_k} \quad και \quad N_k = \sum_{i=1}^{N} I(y_i = k).$$

Ορίζουμε τη **συνολική συνδιακύμανση** (total covariance) ως:

$$\boldsymbol{S}_T = \sum_{i=1}^{N} (\boldsymbol{x}_i - \boldsymbol{m})(\boldsymbol{x}_i - \boldsymbol{m})^T$$

$$\boldsymbol{m} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i = \frac{1}{N}\sum_{k=1}^{c} N_k \boldsymbol{m}_{k.}$$

Από τα παραπάνω προκύπτει η **συνδιακύμανση μεταξύ κλάσεων** (between-class covariance)

$$\boldsymbol{S}_B = \boldsymbol{S}_T - \boldsymbol{S}_W = \sum_{k=1}^{c} N_k (\boldsymbol{m}_k - \boldsymbol{m})(\boldsymbol{m}_k - \boldsymbol{m})^T.$$

Ορίζουμε $D' > 1$ στο πλήθος γραμμικές μεταβλητές ως $\boldsymbol{z}_j = \boldsymbol{w}_j{}^T \boldsymbol{x}$ $(j = 1, 2, \dots, D')$. Ορίζουμε επίσης και το διάνυσμα $\boldsymbol{z} = \boldsymbol{W}^T \boldsymbol{x}$ όπου οι στήλες του πίνακα $\boldsymbol{W}$ είναι τα διανύσματα $\boldsymbol{w}_j$.

Σε αυτό τον χώρο η συνδιακύμανση εντός κλάσεων και η συνδιακύμανση μεταξύ κλάσεων θα είναι

$$\boldsymbol{s}_W = \sum_{k=1}^{c}\sum_{i=1}^{N} I(y_i = k)(\boldsymbol{z}_i - \boldsymbol{\mu}_k)(\boldsymbol{z}_i - \boldsymbol{\mu}_k)^T$$

$$\boldsymbol{s}_B = \sum_{k=1}^{c} N_k (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T$$

$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^{N} I(y_i = k)\boldsymbol{z}_i}{N_k} \quad και \quad \boldsymbol{\mu} = \frac{1}{N}\sum_{k=1}^{c} N_k \boldsymbol{\mu}_k$$

Και

$$\boldsymbol{s}_W = \boldsymbol{W}^T \boldsymbol{S}_W \boldsymbol{W}$$

$$\boldsymbol{s}_B = \boldsymbol{W}^T \boldsymbol{S}_B \boldsymbol{W}.$$

Όπως είπαμε παραπάνω στόχος είναι να βρούμε ένα υπερεπίπεδο που να διαχωρίζει καλά τις κλάσεις δηλαδή οι προβολές του να έχουν μεγάλη συνδιακύμανση μεταξύ κλάσεων και μικρή συνδιακύμανση εντός των κλάσεων οπότε ψάχνουμε το $\boldsymbol{W}$ που θα μεγιστοποιεί το παρακάτω αριθμητικό μέτρο

$$J(W) = \frac{|\boldsymbol{s}_B|}{|\boldsymbol{s}_W|} = \frac{|\boldsymbol{W}^T \boldsymbol{S}_B \boldsymbol{W}|}{|\boldsymbol{W}^T \boldsymbol{S}_W \boldsymbol{W}|}.$$

Οι στήλες του βέλτιστου $\boldsymbol{W}$ είναι τα ιδιοδιανύσματα του γενικευμένου προβλήματος ιδιοτιμών

$$\boldsymbol{S}_B\boldsymbol{w}_i = \lambda_i \boldsymbol{S}_w\boldsymbol{w}_i$$

ή του τυπικού προβλήματος ιδιοτιμών

$$\boldsymbol{S}_w{}^{-1}\boldsymbol{S}_B\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i.$$

Το $\boldsymbol{S}_B$ έχει βαθμό το πολύ $c-1$ άρα έχει το πολύ $c-1$ μη μηδενικές ιδιοτιμές, οπότε με το $\boldsymbol{z} = \boldsymbol{W}^T\boldsymbol{x}$ πάμε από τον χώρο d διαστάσεων σε χώρο το πολύ $c-1$ διαστάσεων με τον βέλτιστο διαχωριστικά τρόπο. Δεν είμαστε υποχρεωμένοι να προβάλουμε τα δεδομένα σε ολόκληρο το $\boldsymbol{W}$. Συγκεκριμένα κάθε στήλη του $\boldsymbol{W}$ είναι τα ιδιοδιανύσματα σε φθίνουσα σειρά ως προς την απόλυτη τιμή της αντίστοιχης ιδιοτιμής. Έτσι αν προβάλουμε τα δεδομένα στο διάνυσμα $\boldsymbol{w}_1$ θα έχουμε τη βέλτιστη προβολή σε μια διάσταση, αν τα προβάλουμε στον πίνακα $[\boldsymbol{w}_1|\boldsymbol{w}_2]$ θα έχουμε τη βέλτιστη προβολή στις δύο διαστάσεις κ.ο.κ.

Στην περίπτωση που δεν θεωρήσουμε κοινό πίνακα συνδιακύμανσης τότε ο ταξινομητής καλείται **ταξινομητής τετραγωνικής διαχωριστικής ανάλυσης** (Quadratic discriminant analysis classifier) και δεν είναι πλέον γραμμικός. Έχει τη μορφή:

$$\begin{aligned}\hat{y}_{QDA} &= \underset{k}{\operatorname{argmax}}\, P(Y=k)\frac{1}{(2\pi)^{\frac{d}{2}}\left|\widehat{\boldsymbol{\Sigma}}_k\right|^{\frac{1}{2}}}e^{-\frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}_k{}^{-1}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)} \\ &= \underset{k}{\operatorname{argmax}}\, P(Y=k)\left|\widehat{\boldsymbol{\Sigma}}_k\right|^{-\frac{1}{2}}e^{-\frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}_k{}^{-1}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)} \\ &= \underset{k}{\operatorname{argmax}}\log[P(Y=k)] - \frac{1}{2}\left|\widehat{\boldsymbol{\Sigma}}_k\right| - \frac{1}{2}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}_k{}^{-1}(\boldsymbol{x}^*-\widehat{\boldsymbol{\mu}}_k).\end{aligned}$$

Από τη μεγιστοποίηση των συναρτήσεων πιθανοφάνειας της κατηγορικής και πολυμεταβλητής κανονικής κατανομής λαμβάνουμε

$$\widehat{\boldsymbol{\mu}}_k = \frac{\sum_{i=1}^N I(y_i=k)\boldsymbol{x}_i}{\sum_{i=1}^N I(y_i=k)}, \qquad \widehat{\boldsymbol{\Sigma}}_k = \frac{\sum_{i=1}^N I(y_i=k)(\boldsymbol{x}_i-\widehat{\boldsymbol{\mu}}_k)(\boldsymbol{x}_i-\widehat{\boldsymbol{\mu}}_k)^T}{\sum_{i=1}^N I(y_i=k)}$$

$$\kappa\alpha\iota\ \hat{P}(Y=k) = \frac{\sum_{i=1}^N I(y_i=k)}{N}.$$

Στην R οι ταξινομητές LDA και QDA είναι υλοποιημένοι στη βιβλιοθήκη "MASS" των (Venables & Riplay, 2002) και καλούνται με την εντολή "lda" και "qda" αντίστοιχα. Εκτελέστηκαν οι ταξινομήσεις ως:

```
LDA<-lda(formula=target~., data=dataset.train)
QDA<-qda(formula=target~., data=dataset.train)
```

Ο ταξινομητής LDA πέτυχε ακρίβεια 0.960 και ο QDA 0.966 στο δείγμα Iris (βλ. Κεφάλαιο 5).

2.4. Μπεϋζιανή Ταξινόμηση

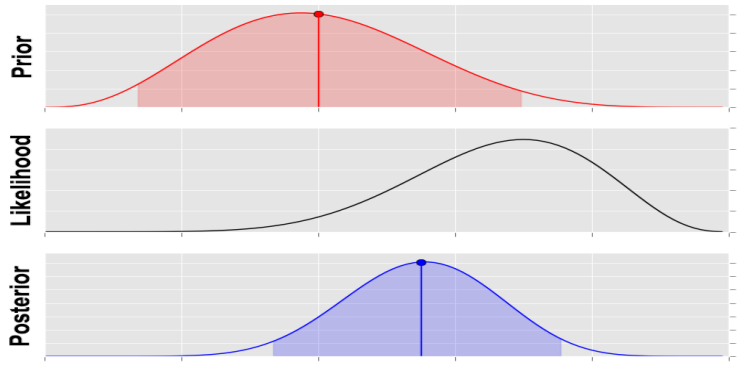
2.4.1. Εισαγωγικά για τη στατιστική κατά Bayes

Η Μπεϋζιανή ταξινόμηση δεν αποτελεί κάποια θεωρία που διαφοροποιείται από τις προηγούμενες ενότητες στο επίπεδο της δημιουργίας του ταξινομητή αλλά ξεκινά να προσεγγίζει διαφορετικά το θέμα σε στατιστικό επίπεδο, χρησιμοποιώντας **Μπεϋζιανή στατιστική** ή **στατιστική κατά Bayes**. Η στατιστική κατά Bayes είναι μια προσέγγιση της στατιστικής, εναλλακτική από την κλασσική. Ενώ η κλασική προσέγγιση θεωρεί ως μόνη πηγή γνώσης τη συνάρτηση πιθανοφάνειας, η Μπεϋζιανή προσέγγιση θεωρεί ότι η υποκειμενική αντίληψη του φαινομένου πρέπει να συμπεριληφθεί στη στατιστική διαδικασία. Θεωρεί επίσης ότι η μόνη ικανοποιητική περιγραφή της αβεβαιότητας επιτυγχάνεται μέσω της πιθανότητας, οπότε οι άγνωστοι σταθεροί παράγοντες μιας κατανομής που προσπαθούν να βρεθούν μέσω της πιθανοφάνειας στην κλασσική στατιστική, στη στατιστική κατά Bayes είναι τυχαίες μεταβλητές.

Έστω λοιπόν ότι έχουμε ένα δείγμα $\boldsymbol{D}$ και θεωρούμε ότι ακολουθεί μια κατανομή με συνεχή παράμετρο $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. Επειδή το $\boldsymbol{\theta}$ είναι πλέον τυχαία μεταβλητή, η συνάρτηση πυκνότητας πιθανότητας του θα είναι

$$f(\boldsymbol{\theta}|\boldsymbol{D}) = \frac{f(\boldsymbol{D}|\boldsymbol{\theta})f(\boldsymbol{\theta})}{f(\boldsymbol{D})} = \frac{f(\boldsymbol{D}|\boldsymbol{\theta})f(\boldsymbol{\theta})}{\int_{\Theta} f(\boldsymbol{D}|\boldsymbol{\theta})f(\boldsymbol{\theta})d\boldsymbol{\theta}}.$$

Χρησιμοποιήσαμε το θεώρημα Bayes υπό όρους πυκνότητας και στη συνέχεια το θεώρημα ολικής πιθανότητας στον παρονομαστή. Η δέσμευση ως προς $\boldsymbol{D}$ στο αριστερό μέλος υπάρχει προφανώς γιατί έχουμε παρατηρήσει το δείγμα μας. Άρα το $f(\boldsymbol{\theta}|\boldsymbol{D})$ είναι η γνώση μας για τη συνάρτηση πυκνότητας πιθανότητας του $\boldsymbol{\theta}$ αφότου συμβουλευτήκαμε τα δεδομένα μας. Το $f(\boldsymbol{\theta}|\boldsymbol{D})$ καλείται **ύστερη** ή **εκ των υστέρων** (posterior) κατανομή του $\boldsymbol{\theta}$. Στο δεξί μέλος παρατηρούμε ότι έχει προκύψει η πιθανοφάνεια $f(\boldsymbol{D}|\boldsymbol{\theta})$ και η $f(\boldsymbol{\theta})$ η οποία δεν έχει δέσμευση στα δεδομένα μας, οπότε πρόκειται για τη γνώση μας για τη συνάρτηση πυκνότητας πιθανότητας του $\boldsymbol{\theta}$ προτού να συμβουλευτούμε τα δεδομένα. Το $f(\boldsymbol{\theta})$ καλείται **πρότερη** ή **εκ των προτέρων** (prior) κατανομή του $\boldsymbol{\theta}$. Στον παρονομαστή παρατηρείται η ποσότητα $f(\boldsymbol{D})$ η οποία στην τελευταία μορφή του τύπου φαίνεται ότι είναι μια σταθερά κανονικοποίησης. Η σχέση πρότερης κατανομής, ύστερης κατανομής και πιθανοφάνειας φαίνεται στο Διάγραμμα 2.7.



Διάγραμμα 2.7: Βλέπουμε πως η ύστερη κατανομή προκύπτει ως συνδυασμός της πρότερης κατανομής και της πιθανοφάνειας μέσω των δεδομένων.

Όταν η πιθανοφάνεια πολλαπλασιασμένη με την πρότερη κατανομή δημιουργεί ύστερη ίδιου τύπου με την πρότερη λέμε ότι η πρότερη είναι **συζυγής** (conjugate) με την πιθανοφάνεια. Σε αυτή την περίπτωση ο υπολογισμός της ύστερης κατανομής μπορεί να γίνει αναλυτικά. Αντιθέτως, όταν οι κατανομές δεν είναι συζυγείς είμαστε υποχρεωμένοι να εξάγουμε τα συμπεράσματα που θέλουμε για την ύστερη κατανομή μέσω **προσομοίωσης** (simulation) τιμών της από τη συνάρτηση του δεύτερου μέλους του τύπου. Έχει αποδειχτεί ότι μέσω του προσομοιωμένου δείγματος μπορούμε να απαντήσουμε όλα τα ερωτήματα που χρειαζόμαστε για την ύστερη κατανομή. Η προσομοίωση γίνεται συνήθως με μεθόδους **Markov Chain Monte Carlo** (MCMC) (Gamerman & Lopes, 2006).

Την εκ των προτέρων κατανομή της παραμέτρου την θεωρεί υποκειμενικά ο κάθε αναλυτής και διαφορετικές επιλογές της καταλήγουν σε (συνήθως ελαφρώς) διαφορετικά αποτελέσματα. Αυτή είναι και η βασική επιχειρηματολογία της κριτικής της στατιστικής κατά Bayes. Βέβαια, σύμφωνα με το βιβλίο του Samaniego (2010, p. 66) έχει δειχτεί ότι οι Μπεϋζιανές διαδικασίες τείνουν να έχουν την ίδια ασυμπτωτική συμπεριφορά με τις καλύτερες κλασικές εναλλακτικές.

Η μη βεβαιότητα για την επιλογή της εκ των προτέρων κατανομής καταπολεμάται είτε με την επιλογή κατανομών με μεγάλη διασπορά είτε με την επιλογή **μη πληροφοριακών κατανομών** (uninformative ή non informative) όπως η ομοιόμορφη κατανομή.

2.4.2. Μπεϋζιανή ταξινόμηση με συζυγείς κατανομές

Όπως είπαμε και παραπάνω, στην Μπεϋζιανή ταξινόμηση δεν χρησιμοποιούμε κάποιο διαφορετικό ταξινομητή αλλά τους ήδη υπάρχοντες, με τη διαφορά ότι χρησιμοποιούμε Μπεϋζιανή συμπερασματολογία για τον υπολογισμό των παραμέτρων. Στην ενότητα αυτή θα παρουσιάσουμε τη Μπεϋζιανή ταξινόμηση χρησιμοποιώντας συζυγείς κατανομές μέσω του παραδείγματος του ταξινομητή Naïve Bayes.

Το συγκεκριμένο παράδειγμα είναι και μια ευκαιρία για να ξεκαθαριστούν όλες οι διαδικασίες που παρουσιάσαμε και έχουν πάρει το όνομά τους από τον Thomas Bayes:

$$\hat{y}_{Bayes} = \underset{k}{\operatorname{argmax}}\, P(Y = k \mid \mathbf{X} = \boldsymbol{x}^*) = \underset{k}{\operatorname{argmax}}\, P(\mathbf{X} = \boldsymbol{x}^* \mid Y = k) P(Y = k)$$
$$= \underset{k}{\operatorname{argmax}}\, P(Y = k) \prod_{j=1}^{d} P(X_j = x_j^* \mid Y = k) = \hat{y}_{NB}.$$

Η πρώτη ισότητα είναι ο ορισμός του ταξινομητή Bayes, ο οποίος είναι μια μαθηματική διαδικασία βελτιστοποίησης. Η δεύτερη ισότητα είναι το θεώρημα Bayes της θεωρίας πιθανοτήτων, το οποίο προφανώς αποτελεί μαθηματικό συλλογισμό. Η τρίτη ισότητα είναι η υπόθεση Naïve Bayes που αποτελεί εφαρμογή της γνώσης ανεξαρτησίας που έχουμε για το φαινόμενο που μελετάμε. Δεδομένου ότι είμαστε βέβαιοι για την υπόθεση ανεξαρτησίας, η ισότητα αυτή είναι απλά ένας μαθηματικός συλλογισμός.

Μέχρι αυτό το σημείο δεν χρησιμοποιήθηκε ακόμα στατιστική, άρα ο ταξινομητής Bayes ή ο ταξινομητής Naïve Bayes δεν είναι απαραίτητα Μπεϋζιανές διαδικασίες. Ο χαρακτηρισμός της στατιστικής διαδικασίας προκύπτει από τον τρόπο που θα υπολογιστεί μέσω του δείγματος η ποσότητα

$$P(Y = k)\prod_{j=1}^{d} P(X_j = x_j^*|\, Y = k).$$

Στο παράδειγμα που όλες οι μεταβλητές ακολουθούν κατηγορική κατανομή, με την κλασική προσέγγιση βρίσκονται όλες οι παράμετροι ως σχετικές συχνότητες, λόγω του ότι οι σχετικές συχνότητες προκύπτουν ως εκτιμήτριες από τη βελτιστοποίηση της συνάρτησης πιθανοφάνειας της κατηγορικής κατανομής. Στη Μπεϋζιανή προσέγγιση θα θεωρήσουμε τις παραμέτρους p_i της κατηγορικής κατανομής ως τυχαίες μεταβλητές και θα ψάξουμε την κατανομή τους. Έστω $\boldsymbol{p} = (p_1, p_2, \dots, p_k)$. Ακολουθώντας τον Tu (2014), η Μπεϋζιανή διαδικασία γράφεται

$$f(\boldsymbol{p}|D) = \frac{f(\boldsymbol{D}|\boldsymbol{p})f(\boldsymbol{p})}{\int_P f(\boldsymbol{D}|\boldsymbol{p})f(\boldsymbol{p})dp} = \frac{f(\boldsymbol{p})\prod_{y_i \in D} f(y_i|\boldsymbol{p})}{\int_P f(\boldsymbol{p})\prod_{y_i \in D} f(y_i|\boldsymbol{p})\, d\boldsymbol{p}},$$

όπου εδώ αναπτύξαμε την πιθανοφάνεια στα γινόμενά της. Ψάχνοντας μια πρότερη κατανομή τέτοια ώστε αν την πολλαπλασιάσουμε με την πιθανοφάνεια της κατηγορικής να λαμβάνουμε ίδιου τύπου ύστερη κατανομή, επιλέγεται η κατανομή Dirichlet με τύπο

$$f(p_1, p_2, \dots, p_c;\ a_1, a_2, \dots, a_c) = \frac{1}{Z(\alpha)}\prod_{k=1}^{c} p_k^{a_k - 1}, \qquad Z(\alpha) = \frac{\prod_{k=1}^{c}\Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{c} a_k)}.$$

Η $Z(\alpha)$ είναι η σταθερά κανονικοποίησης της κατανομής Dirichlet. Δεδομένου ότι διαλέξαμε Dirichlet κατανομή για την πρότερη, η ύστερη γίνεται:

$$f(\boldsymbol{p}|D) \propto f(\boldsymbol{p}|a)\prod_{y_i \in \boldsymbol{D}} f(y_i|\boldsymbol{p}) = f(p_1, p_2, \dots, p_c|\ a_1, a_2, \dots, a_c)\prod_{y_i \in \boldsymbol{D}} f(y_i|p_1, p_2, \dots, p_c)$$
$$\propto \prod_{k=1}^{c} p_k^{a_k - 1}\prod_{y_i \in \boldsymbol{D}}\prod_{k=1}^{c} p_k^{I(y_i = k)} = \prod_{k=1}^{c} p_k^{a_k - 1 + \sum_{y_i \in \boldsymbol{D}} I(y_i = k)},$$

που είναι ο πυρήνας της συνάρτησης πυκνότητας πιθανότητας της Dirichlet με

$$a_k' = a_k + \sum_{y_i \in \boldsymbol{D}} I(y_i = k).$$

Οπότε η ύστερη κατανομή ακολουθεί $Dir(a_1', a_2', \dots, a_c')$. Δεδομένου ότι η ύστερη κατανομή είναι γνωστή, το μόνο που μένει για τον υπολογισμό της είναι να βρεθούν τα a_j' και να κανονικοποιηθεί ώστε να ολοκληρώνει στη μονάδα. Άρα δείξαμε ότι η κατανομή Dirichlet είναι συζυγής με την κατηγορική. Οι παράμετροι $a_1, a_2, \dots, a_c$ καλούνται στη Μπεϋζιανή στατιστική **υπερπαράμετροι** (hyper-parameters). Σε περιπτώσεις που ξεφεύγουν από τα πλαίσια της εργασίας, θα μπορούσαμε να θεωρήσουμε κατανομές για τα α, θεωρώντας πρότερες κατανομές για αυτά, τις λεγόμενες **υπερπρότερες κατανομές** (hyper-priors). Στην περίπτωση αυτή ένα μοντέλο καλείται **ιεραρχικό Μπεϋζιανό μοντέλο** (hierarchical Bayesian model).

Οπότε υπολογίζεται αναλυτικά η κατανομή των $\boldsymbol{p}$ και αντικαθιστώνται στον τύπο του ταξινομητή Naïve Bayes. Η διαδικασία αυτή θα γίνει d φορές, μία για κάθε ανεξάρτητη μεταβλητή. Στην περίπτωση των συζυγών κατανομών με τη Μπεϋζιανή διαδικασία μπορεί κάθε ύστερη κατανομή να θεωρείται πρότερη για μελλοντικές παρατηρήσεις. Έτσι στην περίπτωση που ο στόχος μας είναι να κάνουμε μια μελλοντική πρόβλεψη, αντί για ένα

στατικό μοντέλο ταξινόμησης, μπορούμε να δημιουργήσουμε έναν "ζωντανό οργανισμό" που να προβλέπει, και όταν περάσει το χρονικό σημείο της πρόβλεψης (όταν δηλαδή ξέρουμε αν το μοντέλο προέβλεψε καλά ή όχι το φαινόμενο) να μαθαίνει και να προσαρμόζεται εκ νέου στα καινούρια δεδομένα. Λόγω της φύσης των συζυγών κατανομών, για να γίνει αυτό δεν χρειάζονται τα παλιά δεδομένα, μόνο η παλιά ύστερη κατανομή. Η διαδικασία αυτή καλείται **Μπεϋζιανή ενημέρωση** (Bayesian updating).

Η λογική μάθησης κατά αυτό τον τρόπο είναι πολύ κοντά στον τρόπο που μαθαίνει ο άνθρωπος. Όταν ερχόμαστε σε επαφή με μια νέα πληροφορία αναθεωρούμε την άποψη που έχουμε. Τελικά δεν θα χρειαστεί να θυμηθούμε κάθε πληροφορία που έχουμε συλλέξει σε όλη τη ζωή μας για να σχηματίσουμε τη νέα μας άποψη. Το μόνο που θα χρειαστούμε είναι την καινούρια πληροφορία και την παλιά μας άποψη. Εδώ όπου λέμε άποψη συμπεριλαμβάνουμε και το βαθμό βεβαιότητάς μας για αυτή, αντίστοιχα στο παράδειγμα της Μπεϋζιανής ενημέρωσης η βεβαιότητα αναπαρίσταται με πιο πληροφοριακές πρότερες κατανομές.

Στην R εκτελέστηκε ο κώδικας του Werner (2014) όπως φαίνεται στο Παράρτημα π.4 που είναι βασισμένος στον κώδικα MatLab του Barber (2014) όπως παρουσιάζεται στο βιβλίο του τελευταίου (Barber, 2012). Η μέθοδος υποθέτει ότι όλες οι πρότερες κατανομές είναι μη πληροφοριακές, θεωρώντας όλες τις παραμέτρους της πρότερης Dirichlet κατανομής να είναι $a_k = 1$. Η Μπεϋζιανή διαδικασία Naïve Bayes πέτυχε ακρίβεια 0.922 στο δείγμα Iris (βλ. Κεφάλαιο 5).

2.4.3. Μπεϋζιανή ταξινόμηση με μη συζυγείς κατανομές

Ας προσπαθήσουμε να κάνουμε την παραπάνω διαδικασία για τη λογιστική παλινδρόμηση. Στη βιβλιογραφία συνηθίζεται να χρησιμοποιείται κανονική κατανομή ως πρότερη κατανομή για τα βάρη β_{kj} της παλινδρόμησης άρα γράφουμε:

$$f(\beta_{kj}) = N\left(\beta_{\mu_{kj}}, {\sigma_{kj}}^2\right).$$

Μια συνηθισμένη τιμή για τη μέση τιμή κάθε β_{kj} είναι η $\beta_{\mu_{kj}} = 0$. Τελικά θα πρέπει να υπολογίσουμε την ύστερη κατανομή των β_{kj} ως

$$f\left(\beta_{kj}\middle|\boldsymbol{D}\right) = \frac{\prod_{i=1}^{N}\left(\prod_{k=1}^{c-1}\left(\frac{e^{\boldsymbol{\beta_k x_i}}}{1+\sum_{l=1}^{c-1}e^{\boldsymbol{\beta_l x_i}}}\right)^{I(y_i=k)}\left(\frac{1}{1+\sum_{l=1}^{c-1}e^{\boldsymbol{\beta_l x_i}}}\right)^{I(y_i=c)}\right)\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi{\sigma_{kj}}^2}}e^{-\frac{(\beta_{kj}-\beta_{\mu_{kj}})^2}{2{\sigma_{kj}}^2}}}{\int_{-\infty}^{+\infty}\prod_{i=1}^{N}\left(\prod_{k=1}^{c-1}\left(\frac{e^{\boldsymbol{\beta_k x_i}}}{1+\sum_{l=1}^{c-1}e^{\boldsymbol{\beta_l x_i}}}\right)^{I(y_i=k)}\left(\frac{1}{1+\sum_{l=1}^{c-1}e^{\boldsymbol{\beta_l x_i}}}\right)^{I(y_i=c)}\right)\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi{\sigma_{kj}}^2}}e^{-\frac{(\beta_{kj}-\beta_{\mu_{kj}})^2}{2{\sigma_{kj}}^2}}d\boldsymbol{\beta}}.$$

Παρατηρώντας την παραπάνω σχέση μπορούμε να συμπεράνουμε δύο πράγματα. Πρώτον η ύστερη κατανομή δεν είναι κανονική, οπότε δε μπορούμε να ακολουθήσουμε τη μεθοδολογία των συζυγών κατανομών. Δεύτερον το ολοκλήρωμα στον παρονομαστή είναι αρκετά πολύπλοκο ώστε να μη μπορούμε να το λύσουμε αναλυτικά. Η ανάγκη εύρεσης του ολοκληρώματος προκύπτει από το γεγονός ότι, σε αντίθεση με την προηγούμενη ενότητα,

δεν προκύπτει κάποια γνωστή κατανομή και άρα δεν γνωρίζουμε την σταθερά κανονικοποίησής της.

Μια ιδέα είναι να πάρουμε αυτή τη σχέση και να βελτιστοποιήσουμε ως προς β_{kj}, δηλαδή να υπολογίσουμε το

$$\underset{\boldsymbol{\beta}}{arg\max}\, f(\boldsymbol{\beta}|\boldsymbol{D}), \qquad \boldsymbol{\beta} = \begin{pmatrix} \beta_{10} & \cdots & \beta_{1d} \\ \vdots & \ddots & \vdots \\ \beta_{c0} & \cdots & \beta_{cd} \end{pmatrix}$$

δηλαδή να βρούμε το $\boldsymbol{\beta}$ που μεγιστοποιεί τον αριθμητή:

$$\underset{\boldsymbol{\beta}}{arg\max} \prod_{i=1}^{N} \left(\prod_{k=1}^{c-1} \left(\frac{e^{\boldsymbol{\beta_k} \boldsymbol{x_i}}}{1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta_l} \boldsymbol{x_i}}} \right)^{I(y_i = k)} \left(\frac{1}{1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta_l} \boldsymbol{x_i}}} \right)^{I(y_i = c)} \right) \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi {\sigma_{kj}}^2}} e^{-\frac{(\beta_{kj} - \beta_{\mu_{kj}})^2}{2{\sigma_{kj}}^2}}.$$

Αυτό γιατί ο παρονομαστής είναι μια σταθερά, οπότε δεν επηρεάζει τη βελτιστοποίηση. Η παραπάνω διαδικασία είναι μια εκτιμήτρια των β_{kj} και καλείται **εκτιμήτρια μέγιστης ύστερης κατανομής** (maximum a posteriori estimator ή MAP estimator). Βέβαια, ενώ έχουν θεωρηθεί οι παράμετροι β_{kj} ως τυχαίες μεταβλητές, η εκτίμησή τους με αυτό τον τρόπο θα γίνει σημειακά, κάτι που δεν ανήκει στο πνεύμα της Μπεϋζιανής συμπερασματολογίας. Για αυτό τον λόγο η εκτίμηση των β_{kj} μέσω της εκτιμήτριας μέγιστης ύστερης κατανομής καλείται και "ημι-Μπεϋζιανή" διαδικασία.

Ο Μπεϋζιανός τρόπος να λυθεί το παραπάνω πρόβλημα είναι μέσω της προσομοίωσης τιμών της ύστερης κατανομής, χρησιμοποιώντας τον αριθμητή του δεύτερου μέλους της εξίσωσης. Για να γίνει αυτό χρησιμοποιείται συνήθως κάποια διαδικασία Markov Chain Monte Carlo όπως ο αλγόριθμος Metropolis-Hastings.

Ο αλγόριθμος Metropolis-Hastings λαμβάνει μια κατανομή, στην περίπτωσή μας την ύστερη $f(\beta_{kj}|\boldsymbol{D})$, όχι απαραίτητα κανονικοποιημένη και δημιουργεί τιμές μέσω μιας Μαρκοβιανής αλυσίδας που για στάσιμη κατανομή έχει την εισαχθείσα κατανομή. Για να προσομοιωθούν οι τιμές χρειαζόμαστε μια κατανομή $g(.)$ την οποία γνωρίζουμε και μπορούμε να προσομοιώσουμε εύκολα τιμές από αυτή, πχ την κανονική κατανομή. Η κατανομή αυτή καλείται **κατανομή εισήγησης** (proposal distribution). Ο αλγόριθμος ξεκινάει από ένα σημείο εκκίνησης ${\beta_{kj}}^{(0)}$. Στο τυχαίο βήμα θα βρίσκεται στο σημείο ${\beta_{kj}}^{(i)}$. Παράγεται μια τιμή ${\beta_{kj}}^{*}$ από την κατανομή $g({\beta_{kj}}|{\beta_{kj}}^{(i)}, \boldsymbol{D})$ εδώ θέλουμε η κατανομή εισήγησης να εξαρτάται από το ${\beta_{kj}}^{(i)}$, για παράδειγμα αν η g είναι η κανονική κατανομή, το ${\beta_{kj}}^{(i)}$ μπορεί να είναι η μέση τιμή της. Την ${\beta_{kj}}^{*}$ τη δεχόμαστε με πιθανότητα

$$a_{MH}\left({\beta_{kj}}^{(i)}, {\beta_{kj}}^{*}|\boldsymbol{D}\right) = min\left\{1, \frac{f\left({\beta_{kj}}^{*}|\boldsymbol{D}\right)}{f\left({\beta_{kj}}^{(i)}|\boldsymbol{D}\right)} \frac{g\left({\beta_{kj}}^{(i)}|{\beta_{kj}}^{*}, \boldsymbol{D}\right)}{g\left({\beta_{kj}}^{*}|{\beta_{kj}}^{(i)}, \boldsymbol{D}\right)}\right\}.$$

Αν δεχτούμε την τιμή θέτουμε ${\beta_{kj}}^{(i+1)} = {\beta_{kj}}^{*}$ αλλιώς ${\beta_{kj}}^{(i+1)} = {\beta_{kj}}^{(i)}$.

Από τον τύπο της πιθανότητας αποδοχής καταλαβαίνουμε ότι η κατανομή που πάμε να προσομοιώσουμε δεν χρειάζεται να είναι κανονικοποιημένη επειδή ο τύπος της εμφανίζεται μόνο στο κλάσμα

$$\frac{f\left({\beta_{kj}}^{*}|\boldsymbol{D}\right)}{f\left({\beta_{kj}}^{(i)}|\boldsymbol{D}\right)},$$

άρα το ολοκλήρωμα κανονικοποίησης απαλείφεται. Μια συνήθης πρακτική μετά την προσομοίωση είναι το λεγόμενο **burn-in** των b πρώτων παραγόμενων τιμών, την απόρριψή τους δηλαδή ώστε να δώσουμε τον χρόνο στη Μαρκοβιανή αλυσίδα να σταθεροποιηθεί στη στάσιμη κατανομή. Επίσης, επειδή παράγουμε τιμές μέσω μιας Μαρκοβιανής αλυσίδας οι παραγόμενες τιμές θα έχουν αυτοσυσχέτιση, κάτι που στην περίπτωση χρήσης των τιμών για κάποια στατιστική συμπερασματολογία θα κάνει τις στατιστικές διαδικασίες να συγκλίνουν με πιο αργό ρυθμό. Το πρόβλημα της αυτοσυσχέτισης λύνεται εύκολα με **λέπτυνση** (thinning), δηλαδή με αποδοχή μόνο των σημείων σε θέση πολλαπλάσια ενός k.

Ο αλγόριθμος Metropolis-Hastings δίνει σε κάθε βήμα την πιθανότητα η αλυσίδα να παραμείνει στη θέση της, άρα η αλυσίδα είναι **απεριοδική**. Ο αλγόριθμος μπορεί να επιστρέψει σε οποιοδήποτε σημείο μιας και πάντα η κατανομή εισήγησης μπορεί να προτείνει οποιοδήποτε σημείο του χώρου καταστάσεων. Αν η κατανομή δεν είναι μηδενική σε αυτό το σημείο υπάρχει η πιθανότητα το σημείο να επιλεχθεί, άρα η Μαρκοβιανή αλυσίδα είναι **μη υποβιβάσιμη**. Ο χώρος καταστάσεων της Μαρκοβιανής αλυσίδας που παράγεται από τον αλγόριθμο Metropolis-Hastings είναι **διακριτός** μιας και η προσομοίωση γίνεται σε υπολογιστή.

Μέσω του Παραρτήματος π.3 έχουμε δείξει τα εξής:

- Από το Θεώρημα 1 η Μαρκοβιανή αλυσίδα είναι γνησίως επαναληπτική.
- Από το θεώρημα 2 η Μαρκοβιανή αλυσίδα έχει μοναδική αναλλοίωτη κατανομή.
- Από το θεώρημα 3 η κατανομή της Μαρκοβιανής αλυσίδας συγκλίνει στην αναλλοίωτη.

Μετά την προσομοίωση των τιμών της ύστερης κατανομής μπορεί να χρησιμοποιηθεί κάποιο μέτρο θέσης για να βρεθούν τα βέλτιστα β_{kj}, όπως ο δειγματικός μέσος. Στην περίπτωση που θέλουμε να υπολογίσουμε τη συνάρτηση πυκνότητας πιθανότητάς της μπορούμε να το κάνουμε μέσω της μεθόδου των πυρήνων.

Έχοντας υπολογίσει με Μπεϋζιανό τρόπο τα βέλτιστα β_{kj}, η ταξινόμηση γίνεται αντικαθιστώντας τα στον τύπο του ταξινομητή λογιστικής παλινδρόμησης:

$$\hat{y}_{LR} = \underset{k}{\operatorname{argmax}} \begin{cases} \dfrac{e^{\boldsymbol{\beta_k x^*}}}{1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta_l x^*}}}, & k = 1, 2, \dots, c-1 \\ \dfrac{1}{1 + \sum_{l=1}^{c-1} e^{\boldsymbol{\beta_l x^*}}}, & k = c, \end{cases}$$

που αναλύθηκε στην Ενότητα 2.3.1.
Στην R μια υλοποίηση της Μπεϋζιανής λογιστικής παλινδρόμησης βρίσκεται στη βιβλιοθήκη "brms" των (Bürkner, An R Package for Bayesian Multilevel Models, 2017), (Bürkner, 2018) και καλείται με την εντολή "brm". Ο ταξινομητής εκτελέστηκε ως εξής:

```
options (mc.cores=parallel::detectCores ())
BayesianLR<- brm (target~., data=dataset.train, family="categorical", chains=3,
                  iter=3000,  warmup =1500, thin = 1, prior=c(set_prior ("normal (0, 8)")))
```

Η εντολή *"options (mc.cores=parallel::detectCores ())"* ελέγχει τους διαθέσιμους πυρήνες του επεξεργαστή. Το όρισμα *family="categorical"* προσδιορίζει ότι η διαδικασία που θα

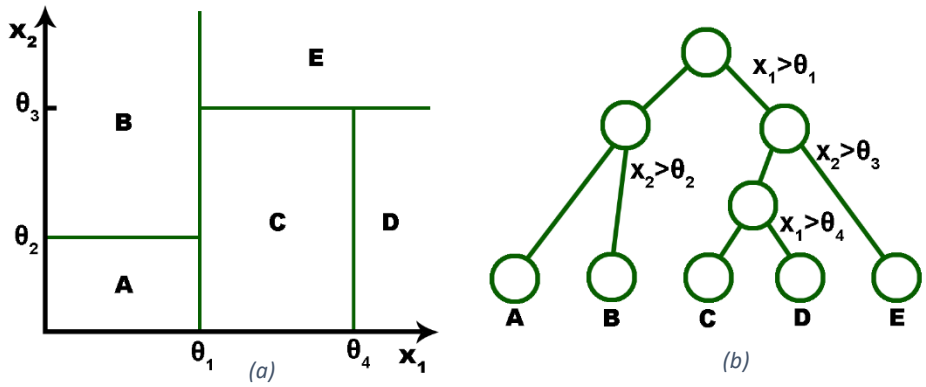
εκτελεστεί είναι λογιστική παλινδρόμηση. Το *chains* ορίζει το πόσες Μαρκοβιανές αλυσίδες θα δημιουργηθούν και, αν υπάρχουν διαθέσιμοι πυρήνες στον επεξεργαστή, η συνάρτηση υποστηρίζει την παράλληλη εκτέλεσή τους. Το *iter* είναι ο αριθμός προσομοιώσεων που θα γίνουν ανά αλυσίδα, το *thin* είναι το thinning, το *warmup* είναι το burn-in και μέσω του *prior* ορίζουμε την πρότερη κατανομή των συντελεστών της παλινδρόμησης που στην προκειμένη επιλέχθηκαν να είναι κανονικές κατανομές με μέση τιμή 0 και διασπορά 8. Η Μπεϋζιανή λογιστική παλινδρόμηση πέτυχε ακρίβεια 0.933 στο δείγμα Iris (βλ. Κεφάλαιο 5).

3. Η Αλγοριθμική Προσέγγιση στην Ταξινόμηση

3.1. Δέντρα απόφασης

Η πρακτική της ταξινόμησης μέσω δέντρων απόφασης αποτελείται από μια οικογένεια αλγορίθμων και είναι η απλούστερη και διαισθητικότερη προσέγγιση στην ταξινόμηση της αλγοριθμικής κουλτούρας που είδαμε στην εισαγωγή. Στόχος των διαδικασιών ταξινόμησης με δέντρα είναι η δημιουργία τμημάτων στον χώρο των επεξηγηματικών μεταβλητών χρησιμοποιώντας μία μεταβλητή κάθε φορά. Έπειτα, κάθε τέτοιο τμήμα αντιστοιχίζεται με μια τιμή για την προς ταξινόμηση μεταβλητή που βρίσκεται στο εσωτερικό του. Εδώ θα παρουσιάσουμε τη διαδικασία **δέντρων ταξινόμησης και παλινδρόμησης** (Classification And Regression Tree ή CART) που παρουσιάστηκε πρώτα από τους Breiman, Friedman, Olsen, & Stone (1984). Θα ακολουθήσουμε τη διαδικασία του Bishop (2006, pp. 663-666).

Όπως βλέπουμε και στο Διάγραμμα 3.1, στο πρώτο βήμα ο αλγόριθμος CART χωρίζει όλο τον χώρο σε δύο τμήματα σύμφωνα με το αν ισχύει $x_1 > \theta_1$ ή $x_1 \leq \theta_1$ με θ_1 την πρώτη παράμετρο του μοντέλου επιλεγμένη όπως θα δούμε παρακάτω. Αυτό δημιουργεί δύο υπό-χώρους οι οποίοι θα επαναδιαχωριστούν ανεξάρτητα. Για παράδειγμα ο υπό-χώρος $x_1 \leq \theta_1$ θα επαναδιαχωριστεί σύμφωνα με το αν ισχύει $x_2 \leq \theta_2$ ή $x_2 > \theta_2$ κ.ο.κ. Οι αλγόριθμοι που σε κάθε βήμα λαμβάνουν μια τοπικά βέλτιστη λύση καλούνται **άπληστοι** (greedy) αλγόριθμοι, οπότε ο CART είναι ένας άπληστος αλγόριθμος. Στο τέλος θα αντιστοιχήσουμε τον κάθε υπο-χώρο με μια κατηγορία. Προφανώς αφού θα διχοτομούμε σε κάθε βήμα τον χώρο, το δέντρο που θα προκύψει θα είναι δυαδικό, δηλαδή κάθε κόμβος θα έχει το πολύ δύο παιδιά.



Διάγραμμα 3.1: Στο (a) βλέπουμε πως ο αλγόριθμος CART διακριτοποιεί τον χώρο των ανεξάρτητων μεταβλητών. Στο (b) φαίνεται το δέντρο απόφασης που κάνει αυτή τη δουλειά.

Σε κάθε βήμα η επιλογή της μεταβλητής διαχωρισμού και της αντίστοιχης παραμέτρου θ_i γίνεται με βάση την ελαχιστοποίηση του **κριτηρίου Gini** που ορίζεται ως

$$\mathrm{Q}_\tau(\mathrm{T}) = \sum_{k=1}^{C} p_{\tau k}(1 - p_{\tau k}).$$

Έχουμε θεωρήσει τα μέχρι τώρα φύλλα του δέντρου αριθμημένα ως $\tau = 1, \dots, |T|$ και $p_{\tau k}$ το ποσοστό των δεδομένων κλάσης k στο τμήμα R_τ, το οποίο είναι το τμήμα που ορίζεται από το φύλλο τ. Στο πρώτο βήμα η διαδικασία γίνεται σε όλο τον χώρο του δείγματος.

Η διαδικασία ελαχιστοποίησης γίνεται εξαντλητικά, δηλαδή υπολογίζονται τα κριτήρια Gini όλων των δυνατών διαχωρισμών για καθεμία από τις μεταβλητές και επιλέγεται ο διαχωρισμός με την ελάχιστη τιμή. Στην περίπτωση συνεχών μεταβλητών επιλέγεται ένα βήμα διακριτοποίησης το οποίο θα μπορούσε να είναι η διαφορά των κοντινότερων παρατηρήσεων για κάθε μεταβλητή.

Για να σταματήσει η διαδικασία χρησιμοποιούμε ένα κριτήριο παύσης που το ορίζουμε με βάση τον αριθμό των δεδομένων που απομένουν σε κάθε τμήμα. Η διαδικασία δηλαδή μπορεί να περιγραφεί ως εξής:

- Αν ισχύει το κριτήριο παύσης σταμάτα
- Για κάθε μεταβλητή διακριτοποίησε
- Για κάθε διακριτοποίηση κάθε μεταβλητής
    - Θεώρησέ την ως διαχωριστική υπερεπιφάνεια
    - Υπολόγισε το κριτήριο Gini
- Διάλεξε ως διαχωρισμό αυτόν με το ελάχιστο Gini
- Επανάλαβε τη διαδικασία για κάθε υποχώρο

Με αυτό τον τρόπο δημιουργούμε μεγάλα δέντρα και στη συνέχεια τα μικραίνουμε με μια διαδικασία που καλείται **κλάδεμα** (pruning). Θεωρούμε ως κριτήριο κλαδέματος το

$$C(T) = \sum_{\tau=1}^{|T|} E_\tau(T) + \lambda|T|,$$

με $E_\tau(T)$ το σφάλμα λάθος ταξινόμησης για δεδομένο δέντρο και $|T|$ το πλήθος φύλλων του δέντρου. Το λ είναι μια σταθερά ομαλοποίησης που υπολογίζεται με τη **διασταυρωμένη επικύρωση** (cross validation) του μοντέλου. Το $|T|$ είναι η συνάρτηση ομαλοποίησης που είπαμε στην εισαγωγή και τιμωρεί το μοντέλο με μεγάλα νούμερα όταν είναι πολύπλοκο, όταν δηλαδή έχει πολλά φύλλα. Έτσι δημιουργείται ένα δίπολο σφάλματος-πολυπλοκότητας του μοντέλου. Τελικά επιλέγουμε ένα υπο-δέντρο με χαμηλό κριτήριο κλαδέματος. Το κλάδεμα γίνεται από τα φύλλα στη ρίζα.

Η διαδικασία αυτή δημιουργεί ως τελικό μοντέλο ένα δέντρο, οπότε για την ταξινόμηση μιας παρατήρησης ακολουθούμε το δέντρο από τη ρίζα μέχρι να καταλήξουμε σε ένα φύλλο, το οποίο είναι αντιστοιχισμένο με την πλειοψηφική κατηγορία του τμήματος που ορίζει.

Διαφορετικές μέθοδοι δέντρων απόφασης δημιουργούν διαφορετικού τύπου δέντρα. Σε αντίθεση με τον CART, ο αλγόριθμος C4.5 δεν δημιουργεί αναγκαστικά δυαδικά δέντρα και χρησιμοποιεί ως κριτήριο επιλογής υπερεπιπέδου σε κάθε βήμα την ελαχιστοποίηση της εντροπίας. Ως διαδικασίες όμως βρίσκονται πολύ κοντά στην υλοποίησή τους. Οι αλγόριθμοι

δέντρων απόφασης έχουν ως κύριο θετικό χαρακτηριστικό την διαισθητικότητα που τους χαρακτηρίζει, δηλαδή ένας ανθρώπινος παρατηρητής μπορεί να δει και να θεωρήσει λογικά τα βήματα ακολουθώντας ένα δέντρο απόφασης. Έχει παρατηρηθεί παρόλα αυτά ότι ο CART είναι πολύ ευαίσθητος σε μικρές αλλαγές του δείγματος. Η κύρια κριτική των μεθόδων αυτών είναι ότι κάθε διαχωρισμός πρέπει να είναι παράλληλος στους $d-1$ άξονες, δηλαδή η διαδικασίες αυτές θα υπολειτουργούσαν σε ένα δείγμα δύο διαστάσεων που θα έπρεπε να χωριστεί με μια ευθεία 45 μοιρών μεταξύ των αξόνων.

Στην R μια υλοποίηση του αλγορίθμου CART βρίσκεται στη βιβλιοθήκη "rpart" των (Therneau & Atkinson, 2018) και καλείται με την εντολή "rpart". Εκτελέστηκε ως:

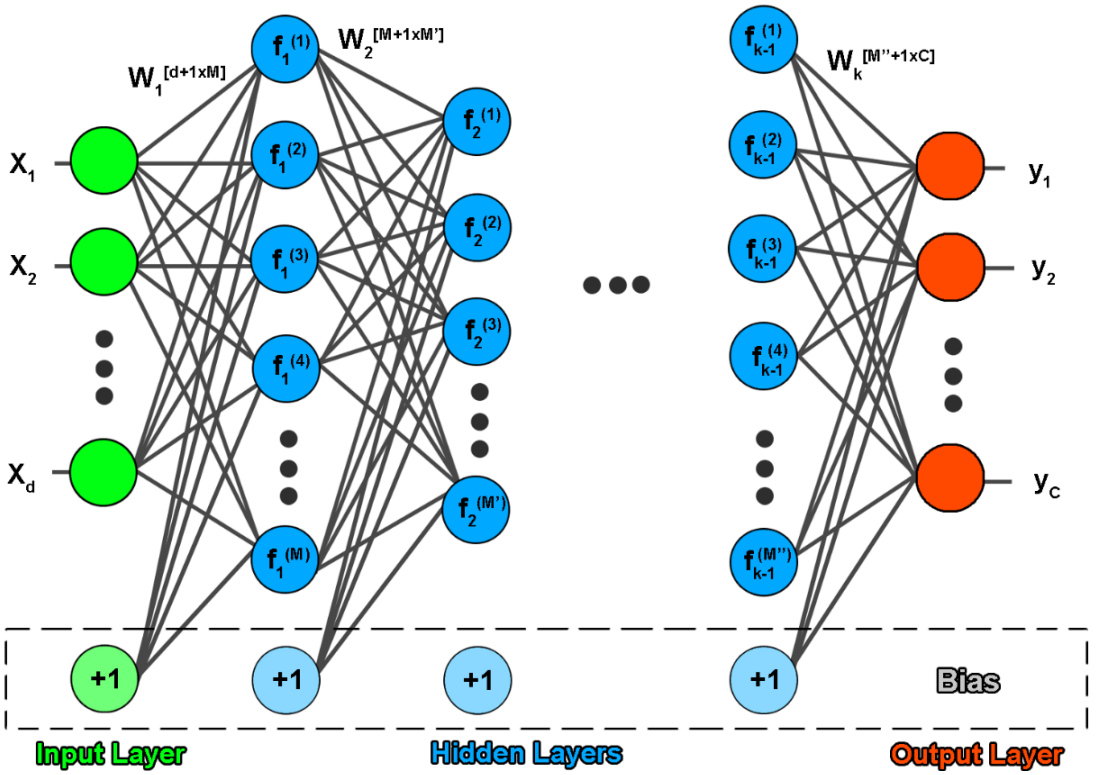
```
CART<-rpart(target~., data= dataset.train, method="class")
```

Ο ταξινομητής CART πέτυχε ακρίβεια 0.928 στο δείγμα Iris (βλ. Κεφάλαιο 5).

3.2. Νευρωνικά Δίκτυα

3.2.1. Εισαγωγικά για τα νευρωνικά δίκτυα

Ο όρος "Νευρωνικά δίκτυα" έχει εξελιχθεί ώστε να περιλαμβάνει μια πληθώρα μοντέλων και μεθόδων μάθησης. Οι διαδικασίες αυτές είναι εμπνευσμένες από τη βιολογία και συγκεκριμένα αποτελούν μια απλουστευτική προσέγγιση της λειτουργείας των νευρώνων του εγκεφάλου. Θα μελετήσουμε τα νευρωνικά δίκτυα ως διαδικασίες μέσα από το παράδειγμα της ταξινόμησης. Έστω ότι έχουμε ένα σετ δεδομένων D μεγέθους N, με $\boldsymbol{X} = (X_1, X_2, \dots, X_d)$ επεξηγηματικές μεταβλητές και μια εξαρτημένη μεταβλητή Y που παίρνει c τιμές. Μέχρι τώρα στη γενική περίπτωση ψάχναμε να βρούμε μια $\hat{f}(\boldsymbol{x})$ τέτοια ώστε να ελαχιστοποιεί μια συνάρτηση κόστους $C\big(\hat{f}(\boldsymbol{x}), y\big)$. Στις περισσότερες περιπτώσεις η συνάρτηση κόστους ήταν μια κυρτή συνάρτηση οπότε δεν είχαμε κάποια δυσκολία στη βελτιστοποίησή της. Στην προσέγγιση του προβλήματος μέσα από τη θεωρία νευρωνικών δικτύων αντί να θεωρήσουμε μια σχετικά απλή $\hat{f}(\boldsymbol{x})$ μπορούμε να αφήσουμε το πρόβλημα πιο ελεύθερο στον χώρο των δυνατών λύσεων και να θεωρήσουμε μια σύνθεση συναρτήσεων της μορφής $\hat{\boldsymbol{f}}(\boldsymbol{x}) = \hat{\boldsymbol{f}}_k\left(\hat{\boldsymbol{f}}_{k-1}(\dots \hat{\boldsymbol{f}}_1(\boldsymbol{x}))\right)$. Η κάθε $\hat{\boldsymbol{f}}_i$ καλείται i **στρώμα** (layer) του δικτύου. Το $\boldsymbol{x}$ καλείται **στρώμα εισόδου** (input layer) και το και το $\hat{\boldsymbol{f}}_k$ στρώμα εξόδου (output layer). Τα υπόλοιπα $k-1$ στρώματα καλούνται **κρυφά στρώματα** (hidden layers). Ο αριθμός των στρωμάτων ορίζει το **βάθος** (depth) του δικτύου, και ένα δίκτυο μπορεί να χαρακτηριστεί ως **δίκτυο $\boldsymbol{k}$ στρωμάτων** (k layer network) ή ως **δίκτυο $\boldsymbol{k-1}$ κρυφών στρωμάτων** ($k-1$ hidden layer network). Ένα νευρωνικό δίκτυο με μεγάλο βάθος καλείται **βαθύ νευρωνικό δίκτυο** (Deep neural network). Τη δομή ενός νευρωνικού δικτύου μπορούμε να τη δούμε στο Διάγραμμα 3.2:



Διάγραμμα 3.2: Η δομή ενός νευρωνικού δικτύου ταξινόμησης για d ανεξάρτητες μεταβλητές και c κλάσεις. Με πράσινο βλέπουμε τις μονάδες εισαγωγής που ταυτίζονται με τις ανεξάρτητες μεταβλητές. Οι μπλε μονάδες είναι οι μονάδες που ανήκουν σε κρυφά στρώματα και ταυτίζονται με συναρτήσεις προς εκπαίδευση. Με πορτοκαλί βλέπουμε την έξοδο της διαδικασίας που είναι η πιθανότητα η κάθε παρατήρηση να ανήκει στην κάθε κλάση

Κάθε κόμβος του Διαγράμματος 3.2 καλείται **μονάδα** (unit). Η πρώτη στήλη μονάδων αναπαριστούν τις ανεξάρτητες μεταβλητές του προβλήματός μας και είναι το στρώμα εισόδου του δικτύου. Η δεύτερη στήλη είναι το πρώτο κρυφό στρώμα του δικτύου και κάθε μονάδα της αναπαριστά μια $\hat{\boldsymbol{f}}_1(\boldsymbol{x}) = (\hat{f}_1^{(1)}(\boldsymbol{x}), \hat{f}_1^{(2)}(\boldsymbol{x}), \dots, \hat{f}_1^{(M)}(\boldsymbol{x}))$. Μπορούμε να χρησιμοποιήσουμε αυθαίρετο αριθμό μονάδων M σε κάθε βάθος του νευρωνικού δικτύου αλλά όλες οι συναρτήσεις στο ίδιο βάθος θα είναι της ίδιας μορφής με δυνατότητα να διαφοροποιούνται μόνο ως προς τους σταθερούς όρους, στα **βάρη** (weights) δηλαδή των τιμών εισαγωγής από το προηγούμενο επίπεδο. Η τελευταία στήλη αναπαριστά το στρώμα εξόδου και αποτελείται από c μονάδες. Η καθεμία επιστρέφει την πιθανότητα η παρατήρηση να ανήκει στην αντίστοιχη κλάση. Οι μονάδες που συμβολίζονται με "$+1$" σε κάθε βάθος καλούνται **μεροληψία** (Bias) και προσφέρουν μια επιπλέον ευελιξία στο μοντέλο, συγκεκριμένα προσφέρουν σε κάθε συνάρτηση τη δυνατότητα να έχει και έναν σταθερό όρο. Η ύπαρξη ακμής από μια μονάδα στρώματος i σε μια στρώματος $i+1$ δείχνει ότι η συνάρτηση της μονάδας στρώματος $i+1$ παίρνει ως όρισμα την τιμή της μονάδας στρώματος i. Ένα νευρωνικό δίκτυο στο οποίο κάθε μονάδα προηγούμενου στρώματος συνδέεται με κάθε μονάδα του επόμενου (εκτός από τη μεροληψία) καλείται **πλήρως συνδεδεμένο νευρωνικό δίκτυο** (Fully connected neural network).

Κάθε $\hat{f}_i^{(j)}$ καλείται **συνάρτηση ενεργοποίησης** (activation function). Οι βασικότερες συναρτήσεις ενεργοποίησης είναι οι:

- Ταυτοτική: $f(x) = x$
- Binary step ή Heaviside: $f(x) = \begin{cases} 0 \ \alpha\nu \ x < 0 \\ 1 \ \alpha\nu \ x \geq 0 \end{cases}$
- Σιγμοειδής ή λογιστική: $f(x) = \frac{1}{1+e^{-x}}$
- Υπερβολική εφαπτομένη: $f(x) = \tanh(x)$
- ReLU: $f(x) = \begin{cases} 0 \ \alpha\nu \ x < 0 \\ x \ \alpha\nu \ x \geq 0 \end{cases}$.

Στην περίπτωση που το πρόβλημά μας είναι η ταξινόμηση σε δυο κλάσεις, έχουμε μια μόνο κρυφή μονάδα και η συνάρτηση ενεργοποίησης της είναι η binary step ο ταξινομητής καλείται **ταξινομητής νευρώνα** (Perceptron classifier)

Βλέποντας ένα νευρωνικό δίκτυο με μια πιο στατιστική ματιά αντιλαμβανόμαστε ότι κάθε στρώμα του δικτύου αποτελεί ένα σύνολο νέων επεξηγηματικών μεταβλητών, οι οποίες προκύπτουν από τον συνδυασμό των μεταβλητών που ορίζονται από το προηγούμενο στρώμα, κάτι που κάνει το μοντέλο πολύ ευέλικτο και άρα ευπαθές στην υπερπροσαρμογή. Στην πράξη στα νευρωνικά δίκτυα πάντα χρησιμοποιούμε την τεχνική της ομαλοποίησης που εξηγήθηκε στην εισαγωγή.

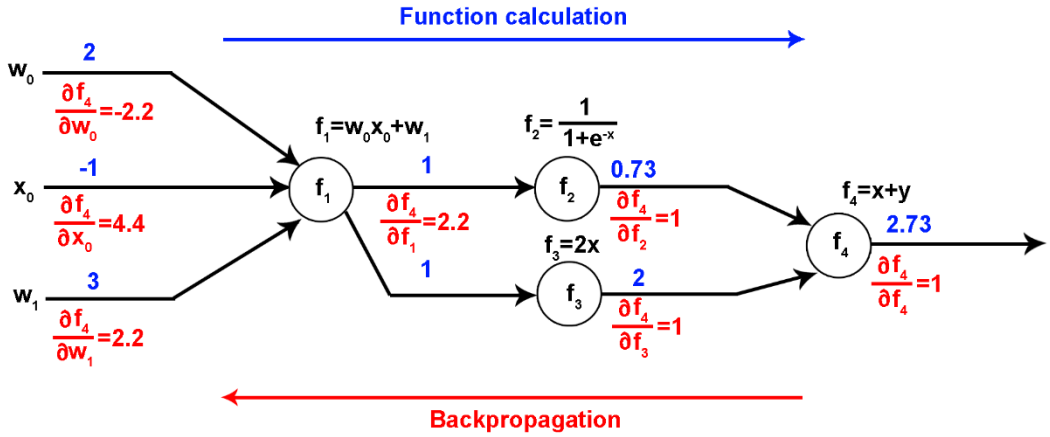
Η σχηματική παρουσίαση της δομής ενός νευρωνικού δικτύου για δεδομένα βάρη W αποτελεί έναν σχεδιασμό ενός υπολογιστικού συστήματος το οποίο λαμβάνει ως εισαγωγή τις d μεταβλητές μιας παρατήρησης και μέσω των συναρτήσεων πολλαπλασιασμένων με τα ανάλογα βάρη επιστρέφει την πιθανότητα η παρατήρηση να ανήκει σε καθεμία από τις κλάσεις. Όπως είναι φανερό, για να κατασκευάσουμε ένα τέτοιο μοντέλο πρέπει να υπολογίσουμε τα W και αυτό θα γίνει μέσω της ελαχιστοποίησης μιας συνάρτησης κόστους ως προς τα W. Η συνάρτηση αυτή είναι συνήθως η αρνητική συνάρτηση πιθανοφάνειας ομαλοποιημένη με τρόπο ώστε να ελαχιστοποιείται το άθροισμα των τετραγώνων των βαρών. Στην περίπτωση του νευρωνικού δικτύου με ένα κρυφό στρώμα και αγνοώντας για ευκολία τις σταθερές μεροληψίας μια συνάρτηση κόστους είναι η

$$J = J_{MLE} + \lambda \left(\sum_{j'=1}^{j'=M} \sum_{i'=1}^{i'=d} \left(W_1^{(i'j')} \right)^2 + \sum_{j''=1}^{j''=c} \sum_{i''=1}^{i''=M'} \left(W_2^{(i''j'')} \right)^2 \right).$$

Για να ελαχιστοποιήσουμε τη συνάρτηση κόστους χρησιμοποιούμε τη μέθοδο κατάβασης βαθμίδας όπως περιγράφεται στο Παράρτημα *π.1.1*. Πρέπει λοιπόν να υπολογίσουμε σε κάθε βήμα της διαδικασίας τα $\nabla_{\boldsymbol{W}_1} J$ και $\nabla_{\boldsymbol{W}_2} J$ ώστε να κινούμαστε προς την αντίθετη κατεύθυνση από τη βαθμίδα. Το πρόβλημα τόσο εδώ όσο και στη γενική περίπτωση είναι ότι η J_{MLE} εμπεριέχει την $\hat{\boldsymbol{f}}(\boldsymbol{x})$ που είναι μια μη κυρτή σύνθεση συναρτήσεων, και τα βάρη είναι πολλά, άρα το να βρεθούν όλες οι μερικές παράγωγοί της J ως προς τα βάρη με εξαντλητικό τρόπο έχει απαγορευτικά υψηλό κόστος. Το πρόβλημα λύνεται με τη μέθοδο της **οπισθοδρομικής διάδοσης** (Backpropagation) που μέσω του κανόνα της αλυσίδας, χρησιμοποιεί μερικές παραγώγους που έχουν ήδη υπολογιστεί για να υπολογιστούν οι επόμενες.

3.2.2. Η μέθοδος της οπισθοδρομικής διάδοσης

Η οπισθοδρομική διάδοση είναι μια μέθοδος υπολογισμού των μερικών παραγώγων $\frac{\partial f_k}{\partial x}$ μιας σύνθεσης συναρτήσεων $f(x) = f_k(f_{k-1}(\dots f_1(x)))$ υπολογίζοντας τις ενδιάμεσες $\frac{\partial f_k}{\partial f_i}$ μέσω του κανόνα της αλυσίδας και χρησιμοποιώντας σε κάθε βήμα υπολογισμούς που έγιναν σε προηγούμενα βήματα. Οι αλγόριθμοι που έχουν το παραπάνω χαρακτηριστικό καλούνται **δυναμικοί αλγόριθμοι**.



Διάγραμμα 3.3: Το υπολογιστικό γράφημα μιας σύνθεσης συναρτήσεων και πως σε δύο χρόνους υπολογίζονται όλες οι μερικές παράγωγοι μέσω της οπισθοδρομικής διάδοσης.

Το Διάγραμμα 3.3 αποτελεί το **υπολογιστικό γράφημα** (computational graph) μιας σύνθεσης συναρτήσεων

$$f(w_0, x_0, w_1) = f_4\left(f_2(f_1(w_0, x_0, w_1)), f_3(f_1(w_0, x_0, w_1))\right).$$

Στόχος είναι να υπολογιστούν για τη θέση στην οποία βρίσκεται η συνάρτηση $(w_0 = 2, x_0 = -1, w_1 = 3)$ οι μερικές παράγωγοι της f_4 ως προς τα w_0, x_0, w_1. Το πρώτο βήμα του αλγορίθμου είναι να υπολογίσει τις τιμές των συναρτήσεων σε αυτή τη θέση (μπλε). Έπειτα πηγαίνοντας από το τέλος προς την αρχή υπολογίζει κάθε μερική παράγωγο της f_4 ως προς τις μεταβλητές που ορίζουν όλες οι προηγούμενες συναρτήσεις (κόκκινο), ξεκινώντας από τις κοντινότερες στην f_4, μέχρι να φτάσει στις αρχικές μεταβλητές w_0, x_0, w_1.

i) Αρχικά γράφουμε το τετριμμένο

$$\frac{\partial f_4}{\partial f_4} = 1.$$

ii) Έπειτα υπολογίζουμε τα

$$\frac{\partial f_4}{\partial f_2} = \frac{\partial (f_2 + f_3)}{\partial f_2} = 1$$

$$\frac{\partial f_4}{\partial f_3} = \frac{\partial (f_2 + f_3)}{\partial f_3} = 1.$$

iii) Στον επόμενο υπολογισμό, θέλουμε να χρισημοποιήσουμε τους υπολογισμούς του προηγούμενου βήματος για να γλιτώσουμε πράξεις, αυτό θα γίνει μέσω του κανόνα της αλυσίδας. Θα σπάσουμε προσθετικά γιατί η f_1 φτάνει στην f_4 από δύο δρόμους.

$$\frac{\partial f_4}{\partial f_1} = \frac{\partial f_4}{\partial f_2}\frac{\partial f_2}{\partial f_1} + \frac{\partial f_4}{\partial f_3}\frac{\partial f_3}{\partial f_1} = 1\frac{\partial f_2}{\partial f_1} + 1\frac{\partial f_3}{\partial f_1} = 1(1-f_2)f_2 + 1\frac{\partial (2f_1)}{\partial f_1} = 2.2.$$

iv) Συνεχίζοντας στην ίδια λογική, εμφανίζουμε με τον κανόνα της αλυσίδας τον όρο $\frac{\partial f_4}{\partial f_1}$ για να υπολογίσουμε τις τελευταίες μερικές παραγώγους.

$$\frac{\partial f_4}{\partial w_0} = \frac{\partial f_4}{\partial f_1}\frac{\partial f_1}{\partial w_0} = 2.2\frac{\partial (w_0 x_0 + w_1)}{\partial w_0} = 2.2x_0 = -2.2,$$
$$\frac{\partial f_4}{\partial x_0} = \frac{\partial f_4}{\partial f_1}\frac{\partial f_1}{\partial x_0} = 2.2\frac{\partial (w_0 x_0 + w_1)}{\partial x_0} = 2.2w_0 = 4.4,$$
$$\frac{\partial f_4}{\partial w_1} = \frac{\partial f_4}{\partial f_1}\frac{\partial f_1}{\partial w_1} = 2.2\frac{\partial (w_0 x_0 + w_1)}{\partial w_1} = 2.2.$$

Παρατηρούμε λοιπόν ότι για κάθε έξοδο της συνάρτησης ως προς την οποία θέλουμε να παραγωγίσουμε την f_4 πρέπει να υπολογίσουμε μόνο μια άμεση μερική παράγωγο και ότι όλους τους υπόλοιπους υπολογισμούς, όσο πολύπλοκοι και αν είναι τους έχουμε κάνει σε προηγούμενα βήματα. Έχοντας υπολογίσει τις μερικές παραγώγους της f_4 ως προς τις αρχικές μεταβλητές, μπορούμε να προχωρήσουμε στο επόμενο βήμα της κατάβασης βαθμίδας. Φυσικά σε κάθε βήμα της κατάβασης βαθμίδας θα υπολογίζουμε τις μερικές παραγώγους με αυτό τον τρόπο.

Να σημειώσουμε σε αυτό το σημείο ότι χρησιμοποιήσαμε τη σιγμοειδή συνάρτηση λόγω της παρακάτω ιδιότητας της παραγώγου της:

$$\frac{\partial \sigma(x)}{\partial x} = \frac{\partial (1+e^{-x})^{-1}}{\partial x} = (1+e^{-x})^{-2}e^{-x} = \frac{1+e^{-x}-1}{1+e^{-x}}\frac{1}{1+e^{-x}}$$
$$= \left(\frac{1+e^{-x}}{1+e^{-x}} - \frac{1}{1+e^{-x}}\right)\frac{1}{1+e^{-x}} = \big(1-\sigma(x)\big)\sigma(x).$$

Άρα η παράγωγος της $\sigma(x)$ είναι συνάρτηση της εξόδου της. Αυτή την ιδιότητα χρησιμοποιήσαμε στο βήμα (iii).

Στην περίπτωση μεταβλητών με περισσότερες διαστάσεις η διαδικασία παραμένει ίδια

Ακολουθώντας το παράδειγμα του Διαγράμματος 3.4 αρχικά υπολογίζουμε τη συνάρτηση (μπλε) στη θέση

$$W = \begin{bmatrix} 0.1 & 0.5 \\ -0,3 & 0.8 \end{bmatrix}, \qquad x = \begin{bmatrix} 0.2 \\ 0.4 \end{bmatrix}.$$

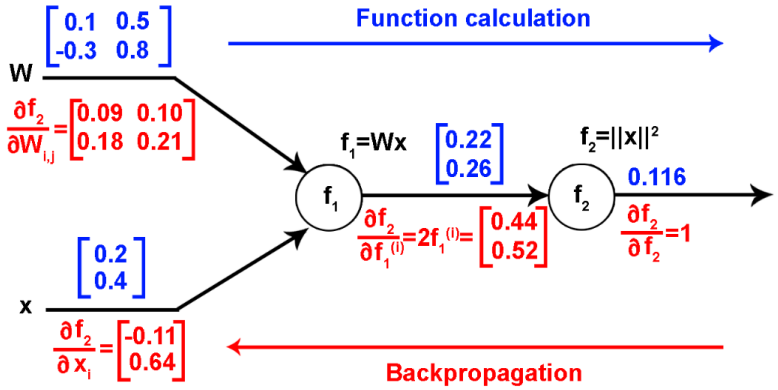
Έπειτα από το τέλος προς την αρχή υπολογίζουμε τις μερικές παραγώγους (κόκκινο)

$$\frac{\partial f_2}{\partial f_2} = 1,$$

$$\frac{\partial f_2}{\partial f_1{}^{(i)}} = 2f_1 = \begin{bmatrix} 0.44 \\ 0.52 \end{bmatrix},$$

$$\frac{\partial f_2}{\partial W_{i,j}} = \sum_K \frac{\partial f_2}{\partial f_1^{(k)}} \frac{\partial f_1^{(k)}}{\partial W_{i,j}} = \sum_K \left(2f_1^{(k)}\right)\left(1_{k=i}x_j\right) = 2f_1 x_j = \begin{bmatrix} 0{,}09 & 0{,}10 \\ 0{,}18 & 0{,}21 \end{bmatrix},$$

$$\frac{\partial f_2}{\partial x_i} = \sum_K \frac{\partial f_2}{\partial f_1^{(k)}} \frac{\partial f_1^{(k)}}{\partial x_i} = \sum_K \left(2f_1^{(k)}\right)\left(W_{k,i}\right) = \begin{bmatrix} -0.11 \\ 0.64 \end{bmatrix}.$$

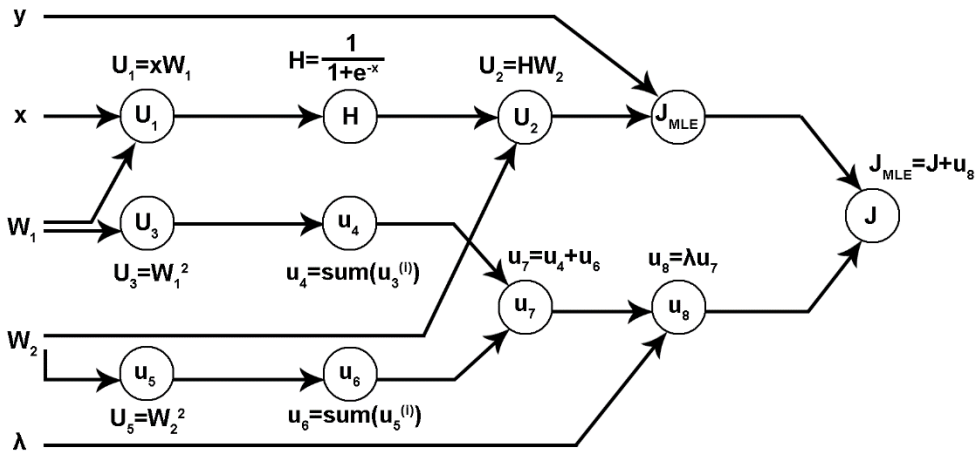


Διάγραμμα 3.4: Το υπολογιστικό γράφημα σύνθεσης συναρτήσεων πολλών μεταβλητών και ο υπολογισμός των μερικών παραγώγων σε δύο χρόνους μέσω της οπισθοδρομικής διάδοσης.

Άρα για να υπολογίσουμε τα βάρη στο πρόβλημα της προσαρμογής του νευρωνικού δικτύου πρέπει να εκφράσουμε τη συνάρτηση κόστους που επιλέξαμε ως ένα υπολογιστικό γράφημα και να εφαρμόσουμε οπισθοδρομική διάδοση. Το υπολογιστικό γράφημα της

$$J = J_{MLE} + \lambda \left(\sum_{j'=1}^{j'=M} \sum_{i'=1}^{i'=d} \left(W_1^{(i'j')}\right)^2 + \sum_{j''=1}^{j''=c} \sum_{i''=1}^{i''=M'} \left(W_2^{(i''j'')}\right)^2 \right)$$

φαίνεται στο Διάγραμμα 3.5.



Διάγραμμα 3.5: Το υπολογιστικό γράφημα της συνάρτησης κόστους ενός νευρωνικού δικτύου.

Όπως είπαμε και παραπάνω έχουμε αγνοήσει τους παράγοντες μεροληψίας και έχουμε θεωρήσει νευρωνικό δίκτυο με ένα κρυφό στρώμα με M συναρτήσεις ενεργοποίησης.

Επειδή τα δείγματα εκπαίδευσης στην πράξη, σε περιπτώσεις όπως η όραση υπολογιστών, έχουν μέγεθος τάξης δισεκατομμυρίων, οι χρόνοι υπολογισμού της βαθμίδας είναι απαγορευτικοί ακόμα και με τη βοήθεια της οπισθοδρομικής διάδοσης. Χρησιμοποιείται λοιπόν η μέθοδος της στοχαστικής κατάβασης βαθμίδας με τη χρήση προσαρμοσμένης δέσμης B μεγέθους N' με παρατηρήσεις της τάξης των εκατοντάδων σε κάθε βήμα υπολογισμού, όπως περιγράφεται στο Παράρτημα *π.1.3*.

Το x είναι ο $N' \times d$ πίνακας των επεξηγηματικών δεδομένων με κάθε γραμμή μια παρατήρηση της προσαρμοσμένης δέσμης και κάθε στήλη μια μεταβλητή. Το y είναι ο $N' \times C$ πίνακας με κάθε γραμμή μια παρατήρηση και κάθε στήλη 1 αν ανήκει στην κλάση που ορίζει η στήλη και 0 αλλιώς. Το W_1 είναι ο $d \times M$ πίνακας με τα βάρη των d μεταβλητών για καθεμία από τις M συναρτήσεις ενεργοποίησης. Το W_2 είναι ο $M \times C$ πίνακας με τα βάρη των M συναρτήσεων ενεργοποίησης για καθεμία από τις C κλάσεις. Το λ είναι ο παράγοντας ομαλοποίησης. Μέσω της οπισθοδρομικής διάδοσης βρίσκονται τα $\frac{\partial J}{\partial w}$.

Τα νευρωνικά δίκτυα που μελετήσαμε σε αυτή την ενότητα καλούνται νευρωνικά δίκτυα **πρόσθιας τροφοδότησης** (feedforward neural networks) ή αλλιώς **πολυστρωματικοί νευρώνες** (multilayer perceptron). Τα νευρωνικά δίκτυα αυτής της αρχιτεκτονικής έχουν μονάδες που στέλνουν τα αποτελέσματά τους μόνο σε μονάδες μεγαλύτερου βάθους. Αν δηλαδή μια μονάδα βάθους 2 παραγάγει ένα αποτέλεσμα, στα νευρωνικά δίκτυα πρόσθιας τροφοδότησης, το αποτέλεσμα αυτό θα σταλεί σε κάποιες μονάδες βάθους μεγαλύτερου του 2. Τα νευρωνικά δίκτυα που δεν έχουν αυτό τον περιορισμό καλούνται **αναδραστικά** ή **νευρωνικά δίκτυα με ανατροφοδότηση** (Recurrent ή Feedback neural networks). Η μελέτη αναδραστικών νευρωνικών δικτύων ξεφεύγει από τους στόχους της παρούσας εργασίας.

Στην R μια υλοποίηση νευρωνικών δικτύων βρίσκεται στη βιβλιοθήκη "nnet" των (Venables & Riplay, 2002) με την εντολή "nnet". Εκτελέστηκε η εντολή ως εξής:

```
NN9<-nnet(target~., data=dataset.train, size=9)
```

Η συνάρτηση δημιουργεί ένα νευρωνικό δίκτυο με ένα κρυφό στρώμα. Το όρισμα *size* καθορίζει τον αριθμό των μονάδων του κρυφού στρώματος και επιλέχθηκε να είναι 9. Ο συγκεκριμένος ταξινομητής πέτυχε ακρίβεια 0.933 στο δείγμα Iris (βλ. Κεφάλαιο 5).

3.3. Μηχανές διανυσμάτων υποστήριξης

3.3.1. Ταξινομητής μέγιστου περιθωρίου

Έστω ένα υπερεπίπεδο d διαστάσεων $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_d x_d = 0$. Και έστω ένα διάνυσμα $\boldsymbol{x} = (x_1, x_2, \dots, x_d)$. Φυσικά αν το $\boldsymbol{x}$ βρίσκεται πάνω στο υπερεπίπεδο ικανοποιεί την εξίσωση του υπερεπιπέδου. Στην περίπτωση που δεν βρίσκεται πάνω στο υπερεπίπεδο, το αριστερό μέλος της εξίσωσης δεν θα ισούται με 0 αλλά με κάτι ή αρνητικό ή θετικό. Δηλαδή:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_d x_d > 0$$

ή

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_d x_d < 0.$$

Το πρόσημο του $f(\boldsymbol{x}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_d x_d$ ορίζει την πλευρά που βρίσκεται το $\boldsymbol{x}$ ως προς το υπερεπίπεδο. Μπορούμε λοιπόν να χρησιμοποιήσουμε ένα υπερεπίπεδο ως μοντέλο ταξινόμησης με μόνο κριτήριο το πρόσημο του αριστερού μέλους της εξίσωσής του.

Έστω τώρα ότι έχουμε ένα γραμμικώς διαχωρίσιμο πρόβλημα ταξινόμησης δύο μεταβλητών με το y να παίρνει τις τιμές -1 και 1 και έστω ότι τις τιμές του y τις αντιστοιχίζουμε έτσι στις δύο κατηγορίες ώστε για ένα υπερεπίπεδο διαχωρισμού:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_d x_d > 0 \ \ \alpha\nu \ y_i = 1$$

Και

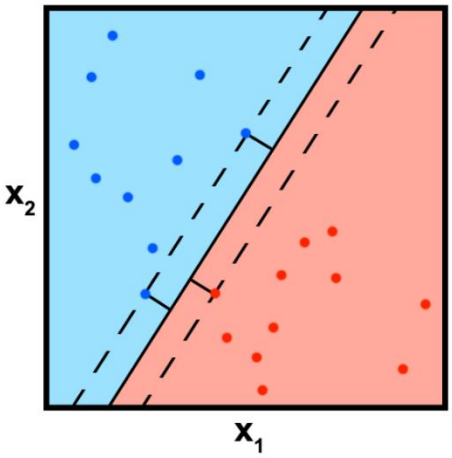
$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_d x_d < 0 \ \ \alpha\nu \ y_i = -1.$$

Τότε το υπερεπίπεδο διαχωρισμού έχει για κάθε $i = 1, 2, \dots, n$ την ιδιότητα:

$$y_i(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_d x_d) > 0.$$

Αν ένα πρόβλημα είναι γραμμικώς διαχωρίσιμο τότε θα υπάρχουν άπειρα υπερεπίπεδα που θα διαχωρίζουν τα δεδομένα. Η ποιο διαισθητική επιλογή υπερεπιπέδου γίνεται με το κριτήριο το υπερεπίπεδο να απέχει όσο το δυνατών περισσότερο από τα δεδομένα. Αυτό το υπερεπίπεδο καλείται **υπερεπίπεδο μέγιστου περιθωρίου** (Maximal margin hyperplane) και ο ταξινομητής που το δημιουργεί καλείται **ταξινομητής μέγιστου περιθωρίου** (Maximal margin classifier). Η απόσταση του υπερεπιπέδου από το κοντινότερο σημείο των δεδομένων καλείται **περιθώριο** (Margin) και το συμβολίζουμε με M. Φυσικά για να πετύχουμε το μέγιστο περιθώριο θα πρέπει το υπερεπίπεδο να απέχει απόσταση M από τουλάχιστον δύο σημεία, ένα από κάθε κλάση. Τα σημεία που απέχουν απόσταση M από το υπερεπίπεδο καλούνται **διανύσματα υποστήριξης** (Support vectors) μιας και είναι διανύσματα d διαστάσεων και είναι τα μόνα που επηρεάζουν (υποστηρίζουν) το υπερεπίπεδο.

Το Διάγραμμα 3.6 δείχνει ένα υπερεπίπεδο δημιουργημένο από τον ταξινομητή μέγιστου περιθωρίου και τα τρία διανύσματα υποστήριξής του. Παρατηρούμε ότι αν αφαιρούσαμε οποιαδήποτε παρατήρηση πέραν των διανυσμάτων υποστήριξης, το υπερεπίπεδο θα ήταν το ίδιο.



Διάγραμμα 3.6: Το υπερεπίπεδο μέγιστου περιθωρίου, σε ένα γραμμικώς διαχωρίσιμο δείγμα, δημιουργημένο από τον ταξινομητή μέγιστου περιθωρίου μέσω των τριών διανυσμάτων υποστήριξης.

Έστω $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_d)$. Για να βρούμε το υπερεπίπεδο μέγιστου περιθωρίου πρέπει να λύσουμε το πρόβλημα μεγιστοποίησης:

$$\underset{\boldsymbol{\beta},\, \beta_0}{arg\max}\, M\,, \qquad \text{υπό τις συνθήκες } \begin{cases} \|\boldsymbol{\beta}\| = 1 \\ y_i(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0) \geq M, \end{cases}$$

δηλαδή ψάχνουμε τα $\boldsymbol{\beta}$, β_0 που μεγιστοποιούν το M αλλά ταυτόχρονα θέτουν κάθε παρατήρηση τουλάχιστον M μακριά από το υπερεπίπεδο. Για να απαλείψουμε την πρώτη συνθήκη αρκεί να μετασχηματίσουμε τη δεύτερη ως:

$$\frac{1}{\|\boldsymbol{\beta}\|} y_i(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0) \geq M$$

και η παραπάνω πράξη επαναπροσδιορίζει το β_0. Γράφουμε ισοδύναμα:

$$y_i(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0) \geq M\|\boldsymbol{\beta}\|.$$

Βέβαια, για κάθε $\boldsymbol{\beta}$, β_0 που ικανοποιεί τις παραπάνω ανισότητες, τις ικανοποιεί και κάθε θετικό πολλαπλάσιό του. Άρα μπορούμε να θεωρήσουμε αυθαίρετα ότι $\|\boldsymbol{\beta}\| = \frac{1}{M}$.

Το πρόβλημα μετασχηματίζεται στο παρακάτω:

$$\underset{\boldsymbol{\beta},\, \beta_0}{arg\min}\frac{1}{2}\|\boldsymbol{\beta}\|^2\,, \qquad \text{υπό τη συνθήκη } \; y_i(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0) \geq 1,$$

γιατί η μεγιστοποίηση του $\frac{1}{\|\boldsymbol{\beta}\|}$ ισοδυναμεί με την ελαχιστοποίηση του $\|\boldsymbol{\beta}\|$. Η σταθερά $\frac{1}{2}$ και η ύψωση στο τετράγωνο δεν επηρεάζουν την ελαχιστοποίηση και έγιναν για λόγους μαθηματικής ευκολίας.

Το πρόβλημα είναι κυρτό ως τετραγωνικό πρόβλημα με γραμμική ανισότητα ως περιορισμό. Για να το λύσουμε θα χρησιμοποιήσουμε τη μέθοδο των πολλαπλασιαστών Lagrange και θα μετασχηματίσουμε το πρόβλημα στο Wolfe δυϊκό σύμφωνα με το Παράρτημα π.1.4:

$$L_P = \frac{1}{2}\|\boldsymbol{\beta}\|^2 - \sum_{i=1}^{N} a_i\big[\, y_i\big(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0\big) - 1\big].$$

Θέτοντας τις πρώτες παραγώγους με 0 παίρνουμε:

$$\boldsymbol{\beta} = \sum_{i=1}^{N} a_i y_i \boldsymbol{x_i} \tag{3.2}$$

$$\sum_{i=1}^{N} a_i y_i = 0. \tag{3.3}$$

Αντικαθιστώντας στην L_P καταλήγουμε στην Wolfe δυϊκή αντικειμενική συνάρτηση προς μεγιστοποίηση:

$$L_D = \sum_{i=1}^{N} a_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j y_i y_j \boldsymbol{x_i}^T \boldsymbol{x_j}, \tag{3.4}$$

υπό τη συνθήκη:

$$a_i \geq 0. \tag{3.5}$$

Η λύση πρέπει να ικανοποιεί τις συνθήκες Karush–Kuhn–Tucker που εκτός από τις $(3.2), (3.3), (3.5)$ είναι και η:

$$a_i\big[\, y_i\big(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0\big) - 1\big] = 0 \;\forall i. \tag{3.6}$$

Το πρόβλημα παραμένει κυρτό και υπολογίζεται είτε με τη μέθοδο των κλίσεων ή με τη μέθοδο Newton-Raphson που αναλύονται στο Παράρτημα π.1.

Από τη συνθήκη (3.6) παρατηρούμε ότι αν το $a_i > 0$ τότε $y_i\big(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0\big) = 1$ άρα το $\boldsymbol{x_i}$ βρίσκεται πάνω στο περιθώριο. Έτσι ορίζονται τυπικά τα διανύσματα υποστήριξης. Αν το $y_i\big(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0\big) > 1$ τότε το $\boldsymbol{x_i}$ δεν βρίσκεται πάνω στο περιθώριο και $a_i = 0$.

Από τη συνθήκη (3.2) βλέπουμε ότι το $\boldsymbol{\beta}$ ορίζεται ως γραμμικός συνδυασμός μόνο διανυσμάτων υποστήριξης.

Φυσικά, όπως είπαμε στην αρχή, όταν βρεθούν τα $\boldsymbol{\beta}, \beta_0$ ο ταξινομητής είναι ο:

$$\hat{y}_{MMC} = sign\big(\boldsymbol{x}^T\boldsymbol{\beta} + \beta_0\big).$$

3.3.2. Ταξινομητής διανυσμάτων υποστήριξης

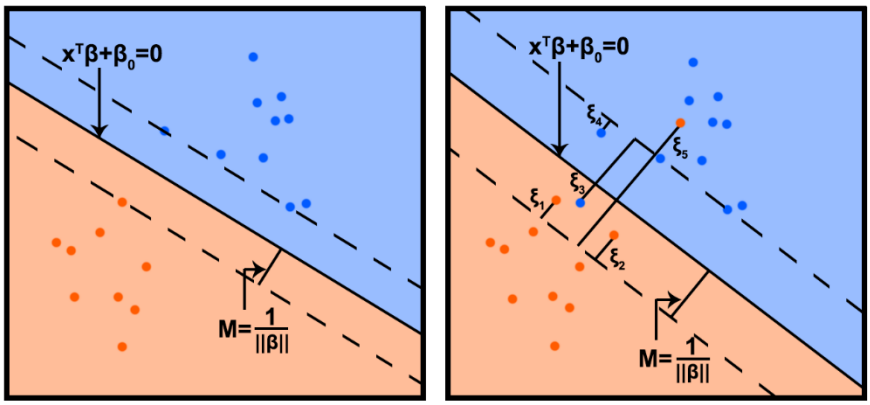
Στην περίπτωση που το πρόβλημα δεν είναι γραμμικώς διαχωρίσιμο αλλά θεωρούμε ότι το μοντέλο θα πρέπει να είναι γραμμικό, ο ταξινομητής μέγιστου περιθωρίου γενικεύεται στον **ταξινομητή διανυσμάτων υποστήριξης** (Support vector classifier).

Όπως και πριν, θα μεγιστοποιήσουμε ως προς M αλλά θα δώσουμε τη δυνατότητα σε κάποιες παρατηρήσεις να ταξινομηθούν στη λάθος πλευρά του περιθωρίου όπως φαίνεται

στο Διάγραμμα 3.7. Ορίζουμε τις μεταβλητές χαλάρωσης $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_N)$ και η συνθήκη του αρχικού προβλήματος βελτιστοποίησης γίνεται:

$$y_i(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0) \geq M(1 - \xi_i).$$

Θεωρούμε ότι $\xi_i \geq 0$ για κάθε i και $\sum_{i=1}^{N} \xi_i \leq c$ για κάποια σταθερά c. Το ξ_i είναι τόσο μεγάλο όσο πιο κακά το προβλέπει το μοντέλο μας, είναι δηλαδή ανάλογο της απόστασης του σημείου από το περιθώριο της κλάσης του, εφόσον το σημείο βρίσκεται από τη λανθασμένη πλευρά του περιθωρίου. Έτσι φράσοντας το άθροισμα $\sum_{i=1}^{N} \xi_i \leq c$ φράσουμε και τη συνολική αναλογία των προβλέψεων να πέσουν στη λάθος πλευρά του περιθωρίου.



(a) *(b)*

Διάγραμμα 3.7: Στο (a) βλέπουμε το υπερεπίπεδο του ταξινομητή μέγιστου περιθωρίου της γραμμικώς διαχωρίσιμης περίπτωσης και στο (b) βλέπουμε το υπερεπίπεδο του ταξινομητή διανυσμάτων υποστήριξης της μη γραμμικώς διαχωρίσιμης περίπτωσης.

Όπως και πριν θεωρούμε $\|\boldsymbol{\beta}\| = \frac{1}{M}$ και η βελτιστοποίηση γίνεται:

$$\underset{\boldsymbol{\beta},\, \beta_0}{arg\min} \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C\sum_{i=1}^{N} \xi_i\,, \qquad \mu\varepsilon\ \sigma\upsilon\nu\theta\acute{\eta}\kappa\varepsilon\varsigma\ \begin{cases} y_i(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0) \geq 1 - \xi_i \\ \xi_i \geq 0, \end{cases}$$

Με την παράμετρο κόστους C να παίζει το ρόλο της σταθεράς c. Η γραμμικά διαχωρίσιμη περίπτωση λαμβάνεται για $C = \infty$.

Το πρόβλημα κι εδώ είναι κυρτό ως τετραγωνικό με γραμμική ανισότητα για περιορισμό. Σύμφωνα πάλι με το Παράρτημα π.1.4, χρησιμοποιούμε πολλαπλασιαστές Lagrange και παίρνουμε τη συνάρτηση:

$$L_P = \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C\sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} a_i\big[\, y_i(\boldsymbol{x_i}^T\boldsymbol{\beta} + \beta_0) - (1 - \xi_i)\big] - \sum_{i=1}^{N} \mu_i\xi_i.$$

Θέτοντας τις παραγώγους με 0 λαμβάνουμε:

(3.7)

$$\boldsymbol{\beta} = \sum_{i=1}^{N} a_i\, y_i \boldsymbol{x_i} \tag{3.8}$$

$$\sum_{i=1}^{N} a_i\, y_i = 0 \tag{3.10}$$

$$a_i = C - \mu_i, \forall i.$$

Για $a_i, \mu_i, \xi_i \geq 0, \forall i$. Αντικαθιστώντας στην L_P παίρνουμε τη Wolfe δυϊκή αντικειμενική συνάρτηση:

$$L_D = \sum_{i=1}^{N} a_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j y_i y_j \boldsymbol{x}_i{}^T \boldsymbol{x}_j, \tag{3.11}$$

η οποία δίνει ένα κάτω φράγμα της αντικειμενικής συνάρτησης (3.7) για κάθε δυνατό σημείο. Μεγιστοποιούμε την L_D με συνθήκες $0 \leq a_i \leq C$ και $\sum_{i=1}^{N} a_i\, y_i = 0$. Εκτός από τις $(3.8), (3.9), (3.10)$ οι συνθήκες Karush–Kuhn–Tucker συμπεριλαμβάνουν τους περιορισμούς:

$$a_i\big[\, y_i\big(\boldsymbol{x_i^T}\boldsymbol{\beta} + \beta_0\big) - (1 - \xi_i)\big] = 0 \tag{3.12}$$

$$\mu_i \xi_i = 0 \tag{3.13}$$

$$y_i\big(\boldsymbol{x_i^T}\boldsymbol{\beta} + \beta_0\big) - (1 - \xi_i) \geq 0. \tag{3.14}$$

Οι εξισώσεις $(3.8) - (3.14)$ χαρακτηρίζουν μοναδικά τη λύση του προβλήματος. Το πρόβλημα παραμένει κυρτό τετραγωνικό. Μεγιστοποιούμε την (3.11) με κατάβαση βαθμίδας ή Newton-Raphson σύμφωνα με το Παράρτημα π.1.

Από τη συνθήκη (3.12) παρατηρούμε ότι το a_i είναι μη μηδενικό μόνο αν

$$\big[\, y_i\big(\boldsymbol{x_i^T}\boldsymbol{\beta} + \beta_0\big) - (1 - \xi_i)\big] = 0,$$

αν δηλαδή στη συνθήκη (3.14) ισχύει η ισότητα. Οι παρατηρήσεις που έχουν μη μηδενικό a_i καλούνται σε αυτή την περίπτωση διανύσματα υποστήριξης και είναι αυτά που είτε είναι πάνω στο περιθώριο (που έχουν μηδενικό ξ_i) είτε αυτά που είναι από τη λάθος μεριά του περιθωρίου της κλάσης τους. Αυτό φαίνεται από τη μορφή του $\boldsymbol{\beta}$ όπως υποδεικνύει η συνθήκη (3.8). Το $\boldsymbol{\beta}$ δηλαδή θα είναι της μορφής:

$$\widehat{\boldsymbol{\beta}} = \sum_{i=1}^{N} \hat{a}_i\, y_i \boldsymbol{x_i}.$$

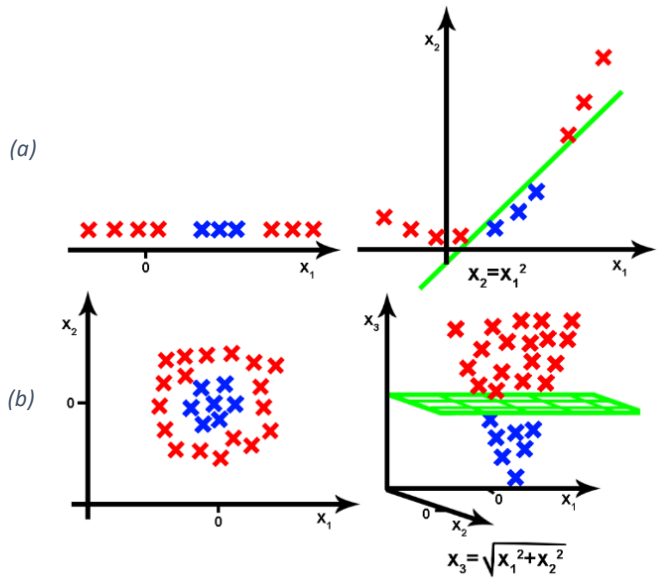
Φυσικά, όταν βρεθούν τα $\boldsymbol{\beta}$, β_0 ο ταξινομητής είναι ο:

$$\hat{y}_{SVC} = sign\big(\boldsymbol{x^T}\boldsymbol{\beta} + \beta_0\big).$$

3.3.3. Ταξινομητής μηχανής διανυσμάτων υποστήριξης

Στην περίπτωση που θέλουμε να ξεφύγουμε από τη γραμμική ταξινόμηση, μπορούμε ακόμα και μέσα στο πλαίσιο αυτής της μεθοδολογίας. Η λογική είναι ότι αν ένα πρόβλημα δεν είναι γραμμικό, του προσθέτουμε κάποιες διαστάσεις και το μετασχηματίζουμε σε ένα γραμμικό πρόβλημα στον καινούριο χώρο. Δηλαδή το σύνορο απόφασης στον καινούριο χώρο είναι ένα υπερεπίπεδο. Στον παλιό χώρο φυσικά το σύνορο απόφασης θα είναι μη γραμμικό. Στο Διάγραμμα 3.8 βλέπουμε δύο τέτοια παραδείγματα στα οποία η διάσταση αυξήθηκε κατά μία. Βέβαια για τη δημιουργία της κάθε διάστασης χρειάστηκε να συμβουλευτούμε τα αρχικά διαγράμματα και να πάρουμε την απόφαση ως προς τη μορφή της παραπάνω

διάστασης. Αυτό δεν είναι εφικτό στην περίπτωση που βρισκόμαστε ήδη σε πολλές διαστάσεις.



Διάγραμμα 3.8: Στο (a) βλέπουμε ένα μη γραμμικώς διαχωρίσιμο πρόβλημα μιας διάστασης που μετασχηματίζεται σε ένα γραμμικώς διαχωρίσιμο πρόβλημα δύο διαστάσεων και στο (b) βλέπουμε ένα μη γραμμικώς διαχωρίσιμο πρόβλημα δύο διαστάσεων που μετασχηματίζεται σε ένα γραμμικώς διαχωρίσιμο πρόβλημα τριών διαστάσεων.

Η μέθοδος των μηχανών διανυσμάτων υποστήριξης προσφέρει μια διαδικασία στην οποία η αύξηση των διαστάσεων γίνεται αυτόματα. Έστω ότι έχουμε ένα πρόβλημα με δυο επεξηγηματικές μεταβλητές $\boldsymbol{X} = (X_1, X_2)$. Τότε θα μπορούσαμε να ορίσουμε κάποιες συναρτήσεις βάσεις $h_m(\boldsymbol{x})$, $m = 1, 2, \dots, M$, με M τον αριθμό των διαστάσεων του καινούριου χώρου. Κάθε συνάρτηση $h_m(\boldsymbol{x})$ παίρνει μια τιμή του αρχικού χώρου και την προβάλει στην h_m διάσταση του καινούριου χώρου. Έστω ένα σημείο του παλιού χώρου:

$$\boldsymbol{x}^{(0)} = \left(x_1^{(0)}, x_2^{(0)}\right)$$

αντιστοιχίζεται στο σημείο του καινούριου χώρου ως:

$$h\left(\boldsymbol{x}^{(0)}\right) = \left(h_1\left(\boldsymbol{x}^{(0)}\right), h_2\left(\boldsymbol{x}^{(0)}\right), \dots, h_M\left(\boldsymbol{x}^{(0)}\right)\right).$$

Έστω ότι επιλέγουμε έναν καινούριο χώρο με $M = 6$, συγκεκριμένα: $h_1(\boldsymbol{x}) = 1$, $h_2(\boldsymbol{x}) = \sqrt{2}x_1$, $h_3(\boldsymbol{x}) = \sqrt{2}x_2$, $h_4(\boldsymbol{x}) = x_1^2$, $h_5(\boldsymbol{x}) = x_2^2$, $h_6(\boldsymbol{x}) = \sqrt{2}x_1x_2$. Έστω τώρα ότι μας ενδιαφέρει το εσωτερικό γινόμενο δυο σημείων $\boldsymbol{x}^{(0)}$, $\boldsymbol{x}^{(1)}$ σε αυτό τον χώρο:

$$\begin{aligned}\left\langle h\left(\boldsymbol{x}^{(0)}\right), h\left(\boldsymbol{x}^{(1)}\right)\right\rangle = \\ = 1 + 2x_1^{(0)}x_1^{(1)} + 2x_2^{(0)}x_2^{(1)} + \left(x_1^{(0)}x_1^{(1)}\right)^2 + \left(x_2^{(0)}x_2^{(1)}\right)^2 \\ + 2x_1^{(0)}x_1^{(1)}x_2^{(0)}x_2^{(1)} = \left(1 + x_1^{(0)}x_1^{(1)} + x_2^{(0)}x_2^{(1)}\right)^2 = \left(1 + \left\langle \boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}\right\rangle\right)^2 \\ = K\left(\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}\right).\end{aligned}$$

Η συνάρτηση αυτή είναι μια συνάρτηση πυρήνα και υπολογίζει το εσωτερικό γινόμενο των στοιχείων της σε έναν άλλο χώρο περισσότερων διαστάσεων. Να παρατηρήσουμε εδώ ότι αν υπολογίζαμε κατευθείαν το :

$$K\left(\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}\right) = \left(1 + \langle \boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)} \rangle\right)^2,$$

δεν θα χρειαζόταν να βρούμε τις συναρτήσεις βάσεις του χώρου, θα υπολογίζαμε απευθείας το εσωτερικό γινόμενο $\langle \boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)} \rangle$ στον χώρο $h(\boldsymbol{x})$.

Στην προηγούμενη ενότητα είδαμε τόσο τη συνάρτηση βελτιστοποίησης του ταξινομητή διανυσμάτων υποστήριξης όσο και τη μορφή του τελικού μοντέλου:

$$L_D = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_i a_j y_i y_j \boldsymbol{x}_i{}^T \boldsymbol{x}_j$$

και

$$sign\left(\boldsymbol{x_i}^T \boldsymbol{\beta} + \beta_0\right), \qquad \mu\varepsilon\ \boldsymbol{\beta} = \sum_{i=1}^{N} a_i\ y_i \boldsymbol{x_i},$$

άρα

$$sign\left(\sum_{i=1}^{N} a_i\ y_i \boldsymbol{x}^T \boldsymbol{x_i} + \beta_0\right).$$

Παρατηρούμε λοιπόν ότι οι επεξηγηματικές μεταβλητές εμφανίζονται μόνο ως εσωτερικά γινόμενα. Αν θέλαμε να υπολογίσουμε το μοντέλο του παραπάνω ταξινομητή σε έναν καινούριο χώρο περισσότερων διαστάσεων $h(\boldsymbol{x})$ οι σχέσεις γίνονται:

$$L_D = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_i a_j y_i y_j \langle h(\boldsymbol{x}_i), h\left(\boldsymbol{x}_j\right) \rangle$$

$$sign\left(\sum_{i=1}^{N} a_i\ y_i \langle h(\boldsymbol{x}), h(\boldsymbol{x}_i) \rangle + \beta_0\right).$$

Άρα για κάποιον πυρήνα K οι παραπάνω σχέσεις παίρνουν την τελική τους μορφή:

$$L_D = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_i a_j y_i y_j K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$$

$$\hat{y}_{SVM} = sign\left(\sum_{i=1}^{N} a_i\ y_i K(\boldsymbol{x}, \boldsymbol{x}_i) + \beta_0\right).$$

Κατά τα άλλα η διαδικασία παραμένει ίδια με την προηγούμενη ενότητα. Ο ταξινομητής που δημιουργεί ένα μη γραμμικό σύνορο απόφασης προβάλλοντας τα δεδομένα σε έναν χώρο μεγαλύτερης διάστασης μέσω κάποιου πυρήνα και μετά ακολουθεί τη διαδικασία του ταξινομητή διανυσμάτων υποστήριξης καλείται **ταξινομητής μηχανής διανυσμάτων υποστήριξης** (Support vector machine classifier ή SVM classifier). Όπως είπαμε και παραπάνω δεν χρειάζεται να υπολογίσουμε εμείς τις συναρτήσεις βάσης του καινούριου χώρου. Την καινούρια βάση την ορίζει ο κάθε πυρήνας και υπολογίζει αυτόματα το εσωτερικό γινόμενο στον χώρο αυτό. Ο πυρήνας που χρησιμοποιήσαμε στο παράδειγμα καλείται πυρήνας

πολυωνύμου δεύτερου βαθμού και φυσικά δεν είναι ο μόνος. Οι κλασικότεροι πυρήνες στη βιβλιογραφία των ταξινομητών μηχανής διανυσμάτων υποστήριξης είναι:

$$Πολυωνύμου\ d-βαθμού\text{: } K\left(\boldsymbol{x}^{(0)},\boldsymbol{x}^{(1)}\right)=\left(1+\langle\boldsymbol{x}^{(0)},\boldsymbol{x}^{(1)}\rangle\right)^{d}.$$

$$Ακτινωτής\ βάσης\ (Radial\ Basis)\text{: } K\left(\boldsymbol{x}^{(0)},\boldsymbol{x}^{(1)}\right)=e^{-\gamma\left\|\boldsymbol{x}^{(0)}-\boldsymbol{x}^{(1)}\right\|^{2}}.$$

$$Νευρωνικού\ δικτύου\ (Neural\ network)\text{: } K\left(\boldsymbol{x}^{(0)},\boldsymbol{x}^{(1)}\right)=\tanh\left(k_1\langle\boldsymbol{x}^{(0)},\boldsymbol{x}^{(1)}\rangle+k_2\right).$$

Την Ενότητα 2.2.2. είδαμε τη μέθοδο των πυρήνων για τον υπολογισμό μιας συνάρτησης πυκνότητας πιθανότητας. Χρησιμοποιήσαμε τότε τους πυρήνες ως συναρτήσεις που μετράνε τοπικότητα και είχαν οριστεί ως συναρτήσεις μιας μεταβλητής επειδή θεωρούσαμε ότι η άλλη είναι πάντα το σημείο προς ταξινόμηση. Στην παρούσα ενότητα είδαμε και τις δυνατότητές τους στον υπολογισμό εσωτερικών γινομένων σε μεγάλες διαστάσεις.

Οι ταξινομητές μέγιστου περιθωρίου, διανυσμάτων υποστήριξης και μηχανής διανυσμάτων υποστήριξης καλούνται πολλές φορές στη βιβλιογραφία **μηχανές διανυσμάτων υποστήριξης** (Support vector machines).

Στην R μια υλοποίηση του ταξινομητή μηχανής διανυσμάτων υποστήριξης βρίσκεται στη βιβλιοθήκη "e1071" των (Meyer, Dimitriadou, Hornik, Weingessel, & Leisch, 2018) και καλείται με την εντολή "svm". Η συνάρτηση εκτελέστηκε ως:

```
SVM<-svm(target~., data=dataset.train)
```

Ο ταξινομητής μηχανής διανυσμάτων υποστήριξης πέτυχε ακρίβεια 0.945 στο δείγμα Iris (βλ. Κεφάλαιο 5).

3.4. Συνδυαστικοί Ταξινομητές

3.4.1. Γενικά στοιχεία και η διαδικασία του ψηφίσματος

Οι **συνδυαστικοί ταξινομητές** ή **επιτροπές** (Ensemble classifiers ή committees) είναι διαδικασίες που χρησιμοποιούν ένα σύνολο από μοντέλα δημιουργημένα από τους κλασικούς ταξινομητές που είδαμε παραπάνω για να κάνουν μια νέα καλύτερη πρόβλεψη. Είναι γνωστό ότι κανένας ταξινομητής δε μπορεί να δημιουργήσει το καλύτερο μοντέλο για όλα τα προβλήματα. Κάθε διαδικασία είναι άμεσα ή έμμεσα μεροληπτική, προτιμώντας κάποιες γενικεύσεις έναντι άλλων και η διαδικασία είναι πετυχημένη αν η μεροληψία αυτή ταιριάζει στα χαρακτηριστικά του εκάστοτε προβλήματος. Η παρακάτω πρόταση είναι γνωστή ως **no free lunch theorem**: *"Αν ένας ταξινομητής είναι καλύτερος από έναν άλλο σε κάποια προβλήματα, τότε υπάρχουν αναγκαστικά άλλα προβλήματα στα οποία η σχέση είναι αντεστραμμένη."* Οι συνδυαστικοί ταξινομητές ξεπερνούν το πρόβλημα αυτό συνδυάζοντας τις προβλέψεις πολλών διαφορετικών ταξινομητών.

Η απλούστερη διαδικασία συνδυασμού ταξινομητών καλείται **ψήφισμα** (voting) και δημιουργεί, από το ίδιο δείγμα, V οποιουδήποτε τύπου μοντέλα $\hat{y}^1, \hat{y}^2, \dots, \hat{y}^V$. Έπειτα ταξινομεί την παρατήρηση στην πλειοψηφική πρόβλεψη των V μοντέλων. Δηλαδή:

$$\hat{y}_{Vote} = arg\max_k \sum_{v=1}^{V} I(\hat{y}^v = k).$$

Η διαδικασία αυτού του τύπου ψηφίσματος καλείται **πλειοψηφικό ψήφισμα** (Majority voting). Υπάρχει η επιλογή να τεθεί ένα διαφορετικό βάρος στην ψήφο του κάθε μοντέλου, ανάλογα με το ποιο μοντέλο θεωρείται πιο αξιόπιστο. Σε αυτή την περίπτωση δημιουργούνται V βάρη $w^1, w^2, \dots, w^V$ και το μοντέλο γίνεται:

$$\hat{y}_{Vote} = arg\max_k \sum_{v=1}^{V} w^v I(\hat{y}^v = k).$$

Η διαδικασία ψηφίσματος τότε καλείται **πλειοψηφικό ψήφισμα με βάρη** (Weighted majority voting). Η πιο κλασική επιλογή των βαρών γίνεται μέσω της υπόθεσης ότι σε κάθε μοντέλο, οι επεξηγηματικές μεταβλητές είναι ανεξάρτητες για δεδομένη κλάση. Κάνουμε δηλαδή σε κάθε μοντέλο την υπόθεση Naïve Bayes. Τότε αποδεικνύεται ότι τα βέλτιστα βάρη είναι τα:

$$w^v \propto \frac{p^v}{1 - p^v},$$

με $p^1, p^2, \dots, p^V$ τις ακρίβειες των μοντέλων.

Γενικά κάθε συνδυαστική μέθοδος εφαρμόζεται για να μικρύνει το σφάλμα λανθασμένης ταξινόμησης, το οποίο όπως είδαμε στην εισαγωγή αναλύεται σε σφάλμα λόγω μεροληψίας και σε σφάλμα λόγω διακύμανσης. Παρακάτω θα δούμε κάποιες πιο σύνθετες διαδικασίες συνδυασμού μοντέλων που εξειδικεύονται στο να ελαχιστοποιούν κάποιον από τους δύο παράγοντες σφάλματος. Η διαδικασία του πλειοψηφικού ψηφίσματος πέτυχε ακρίβεια 0.941 στο δείγμα Iris.

3.4.2. Η διαδικασία του Bagging και τα Τυχαία Δάση

Το **Bagging** (Bootstrap aggregation) είναι ένας συνδυαστικός ταξινομητής δημιουργημένος για να βελτιώσει ένα σύνολο μοντέλων με μεγάλο σφάλμα διακύμανσης. Στην περίπτωση των δέντρων απόφασης, το σφάλμα λόγω διακύμανσης αυξάνεται καθώς αυξάνεται το βάθος των δέντρων. Άρα τα μεγάλα δέντρα χωρίς κλάδεμα είναι καλοί υποψήφιοι για τη διαδικασία του Bagging. Η λογική του είναι η εξής:

- Δημιουργούμε B δείγματα $D^{*1}, D^{*2}, \dots, D^{*B}$ χρησιμοποιώντας την μέθοδο επαναδειγματοληψίας Bootstrap.
- Δημιουργούμε B μοντέλα $\hat{y}^{*1}, \hat{y}^{*2}, \dots, \hat{y}^{*B}$ ίδιου τύπου (συνήθως μεγάλα δέντρα απόφασης).
- Ταξινομούμε κάθε παρατήρηση στην πλειοψηφική κατηγορία των B μοντέλων.

Δηλαδή:

$$\hat{y}_{Bag} = arg\max_{k} \sum_{b=1}^{B} I(\hat{y}^{*b} = k).$$

Για τον υπολογισμό του σφάλματος του μοντέλου από τον ταξινομητή Bagging δεν χρειάζεται να χωρίσουμε το δείγμα σε δείγμα εκπαίδευσης και δείγμα ελέγχου ούτε να χρησιμοποιήσουμε τη διαδικασία του cross-validation. Αποδεικνύεται ότι καθένα από τα B μοντέλα χρησιμοποιούν περίπου τα $2/3$ των παρατηρήσεων του αρχικού δείγματος μιας και η μέθοδος Bootstrap κάνει δειγματοληψία με επανατοποθέτηση. Μια παρατήρηση που δεν χρησιμοποιήθηκε σε ένα από τα B μοντέλα καλείται **out-of-bag παρατήρηση** (OOB) για αυτό το μοντέλο. Και περιμένουμε κάθε παρατήρηση να είναι OOB σε $B/3$ μοντέλα. Προβλέπουμε το σφάλμα λανθασμένης ταξινόμησης χρησιμοποιώντας ως πρόβλεψη για κάθε παρατήρηση την πλειοψηφία μόνο των μοντέλων στα οποία η παρατήρηση αυτή είναι OOB. Το πλεονέκτημα αυτής της διαδικασίας είναι ότι κοστίζει υπολογιστικά πολύ λιγότερο από τη διαδικασία cross-validation, μιας και ήδη η δημιουργία των B μοντέλων κοστίζει πολύ.

Δυστυχώς με τη διαδικασία αυτή χάνονται δύο πολύ θετικά χαρακτηριστικά των δεντρικών μοντέλων. Αρχικά χάνεται η διαισθητικότητα που χαρακτηρίζει τα δέντρα, μιας και πλέον έχουμε να εξετάσουμε μερικές εκατοντάδες δέντρα. Δεν μπορούμε δηλαδή να αναπαραστήσουμε το μοντέλο μας σε μια μορφή κατανοητή από τον άνθρωπο. Δεύτερον, σε κάθε πρόβλεψη έχουμε πλέον υπολογιστικό κόστος. Να παρατηρήσουμε εδώ ότι ενώ έχουμε υπολογιστικό κόστος σε κάθε πρόβλεψη, το Bagging δεν αποτελεί μάθηση βασισμένη στη μνήμη.

Στην R μια υλοποίηση του Bagging βρίσκεται στη βιβλιοθήκη “ipred” των (Peters & Hothorn, 2017) και καλείται με την εντολή “bagging”. Η εντολή εκτελέστηκε ως:

```
Bagging<-bagging(target~., data=dataset.train, nbagg=10000)
```

Με το όρισμα *nbagg* να προσδιορίζει τον αριθμό των δειγμάτων bootstrap της μεθόδου. Η μέθοδος πέτυχε 0.928 ακρίβεια στο δείγμα Iris (βλ. Κεφάλαιο 5).

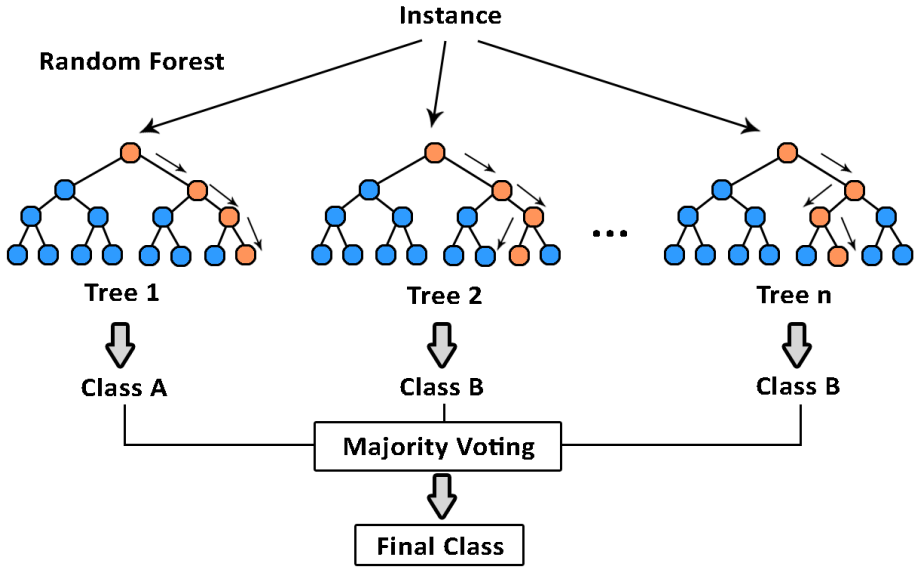
Μια βελτίωση της διαδικασίας του Bagging που αποσυσχετίζει τα επιμέρους δέντρα είναι η διαδικασία των **τυχαίων δασών** (Random forests). Τόσο το Bagging όσο και τα τυχαία δάση ορίστηκαν αρχικά από τον Leo Breiman για το συνδυασμό δέντρων απόφασης στα Breiman (1996) και (2001). Το Bagging έχει τη δυνατότητα να συνδυάσει μοντέλα που δεν είναι δέντρα, η διαδικασία των τυχαίων δασών όμως περιορίζεται σε αυτά. Όπως και στο Bagging δημιουργούμε B Bootstrap δείγματα και εκπαιδεύουμε ένα δέντρο από το καθένα με την εξής διαφορά:

Στη διαδικασία της δημιουργίας του κάθε δέντρου, σε κάθε βήμα επιλογής της μεταβλητής που θα διχοτομήσει τον χώρο, επιλέγουμε m στο πλήθος μεταβλητές και αφήνουμε τη διαδικασία να επιλέξει μόνο μια από αυτές. Οι m διαθέσιμες μεταβλητές αλλάζουν σε κάθε βήμα διχοτόμησης του χώρου και για κάθε δέντρο. Αν όλες οι μεταβλητές είναι d, η ποιο συνηθισμένη επιλογή για το m είναι $m = \lfloor \sqrt{d} \rfloor$. Γενικά η επιλογή μικρού m βοηθάει στις περιπτώσεις που το δείγμα μας έχει πολύ συσχετισμένες επεξηγηματικές μεταβλητές.

Με άλλα λόγια, όταν δημιουργούμε ένα τυχαίο δάσος, σε κάθε βήμα της δημιουργίας του κάθε δέντρου ο αλγόριθμος δεν μπορεί να επιλέξει την πλειοψηφία των μεταβλητών, πράγμα αρκετά αντιδιαισθητικό αλλά με την εξής λογική:

Έστω ότι υπάρχει μια μεταβλητή με πολύ μεγάλη επεξηγηματική ισχύ. Τότε τα περισσότερα δέντρα θα έθεταν αυτή τη μεταβλητή ως πρώτη διχοτόμηση και θα έμοιαζαν πολύ μεταξύ τους, άρα τα δέντρα θα είχαν πολύ μεγάλη συσχέτιση. Ο συνδυασμός πολύ συσχετισμένων μοντέλων δεν μειώνει πολύ το σφάλμα λόγω διακύμανσης. Με τη διαδικασία των τυχαίων δασών κατά μέσο όρο $(d-m)/d$ διαχωρισμοί δεν θα μπορούν να επιλέξουν την ισχυρή μεταβλητή και έτσι τα δέντρα έχουν πολύ διαφορετική δομή. Προφανώς για $m = d$ η διαδικασία των τυχαίων δασών συμπίπτει με το Bagging.

Η υπόλοιπη διαδικασία είναι πανομοιότυπη με το Bagging. Η ταξινόμηση γίνεται σύμφωνα με την πλειοψηφική πρόβλεψη και ο υπολογισμός του σφάλματος λανθασμένης ταξινόμησης μέσω της διαδικασίας out-of-bag. Τόσο στο Bagging όσο και στα τυχαία δάση, τα μεγάλα B δεν υπερπροσαρμόζουν τα δεδομένα. Η διαδικασία των τυχαίων δασών συνοψίζεται στο Διάγραμμα 3.9.



Διάγραμμα 3.9: Η μέθοδος των τυχαίων δασών. Δημιουργεί ένα σύνολο δέντρων δημιουργημένο από τυχαίο υποδείγμα παρατηρήσεων και μεταβλητών και συνδυάζει τα αποτελέσματά τους μέσω ψηφοφορίας.

Στην R μια υλοποίηση του Random Forest βρίσκεται στη βιβλιοθήκη "randomForest" των (Liaw & Wiener, 2002) και καλείται με την εντολή "randomForest". Η εντολή εκτελέστηκε ως:

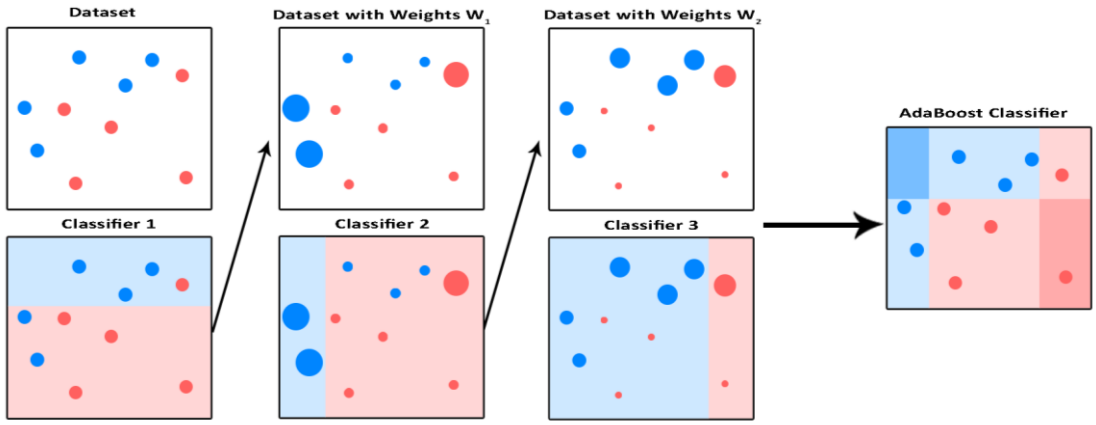
```
RF<-randomForest(target~., data=dataset.train, ntree=10000)
```

Με το όρισμα *ntree* να προσδιορίζει τον αριθμό των δέντρων της μεθόδου. Η μέθοδος πέτυχε 0.934 ακρίβεια στο δείγμα Iris (βλ. Κεφάλαιο 5).

3.4.3. Η διαδικασία του Boosting

Το boosting είναι μια τεχνική που συνδυάζει **ασθενείς ταξινομητές** (Weak classifiers), δηλαδή ταξινομητές ελαφρώς καλύτερους από τη ρίψη νομίσματος, ώστε να δημιουργηθεί ένας ισχυρός ταξινομητής. Τα αρχικά μοντέλα είναι συνήθως απλά, όπως το δέντρο απόφασης με μια τομή και πάσχουν από υψηλό σφάλμα λόγω μεροληψίας. Το boosting εξειδικεύεται στη μείωση του σφάλματος λόγω μεροληψίας. Θα δούμε το boosting μέσω του αλγορίθμου **AdaBoost** (Adaptive Boosting) που είναι ο δημοφιλέστερος αλγόριθμος για boosting.

Η βασική διαφορά του boosting από το bagging είναι ότι τα αρχικά μοντέλα προσαρμόζονται σειριακά και κάθε μοντέλο προσαρμόζεται χρησιμοποιώντας μια μορφή του αρχικού δείγματος αλλά με προσαρμοσμένα βάρη στην κάθε παρατήρηση. Το κάθε βάρος εξαρτάται από το πόσο καλή πρόβλεψη έγινε για την αντίστοιχη παρατήρηση από το προηγούμενο μοντέλο. Η διαδικασία παρουσιάζεται στο Διάγραμμα 3.10.



Διάγραμμα 3.10: Ο Αλγόριθμος AdaBoost. Σε κάθε βήμα δημιουργεί ένα δέντρο απόφασης ενός κόμβου. Έπειτα αναπροσαρμόζει τα δεδομένα προσδίδοντάς μεγάλα βάρη στις λανθασμένες ταξινομήσεις και μικρά στις ορθές και συνεχίζει να προσαρμόζει τα νέα δεδομένα όσες φορές του οριστεί. Η τελική απόφαση είναι η πλειοψηφία των προηγούμενων αποφάσεων.

Έστω ένα πρόβλημα ταξινόμησης δύο κλάσεων, με $\boldsymbol{x} = (x_1, x_2, \dots, x_d)$ τις επεξηγηματικές μεταβλητές και N παρατηρήσεις. Σε κάθε παρατήρηση αντιστοιχούμε ένα βάρος w_n και στην αρχή θέτουμε όλα τα βάρη να είναι ίσα με $1/N$. Θεωρούμε επίσης ότι έχουμε τη δυνατότητα σε κάθε βήμα να δημιουργούμε ένα μοντέλο που λαμβάνει τα δεδομένα με τα βάρη τους και επιστρέφει μια συνάρτηση πρόβλεψης $y(\boldsymbol{x}) \in \{-1, 1\}$. Σε κάθε βήμα ο AdaBoost δημιουργεί ένα νέο μοντέλο στο οποίο τα βάρη προσαρμόζονται ανάλογα με τις προβλέψεις των προηγούμενων μοντέλων. Στα λάθος ταξινομημένα σημεία αντιστοιχεί μεγαλύτερα βάρη και στα σωστά ταξινομημένα αντιστοιχεί μικρότερα. Τέλος, όταν δημιουργηθούν τα προκαθορισμένα μοντέλα, γίνεται η ψηφοφορία με βάρη για να δημιουργηθεί το τελικό μοντέλο.

Ακολουθεί η αναλυτική διαδικασία:

- Αντιστοιχούμε όλα τα αρχικά βάρη $\{w_n\}$ με $w_n^{(1)} = 1/N$ για $n = 1, 2, \dots, N$.
- Για $m = 1, 2, \dots, M$:
  - Δημιουργούμε το μοντέλο $y_m(x)$ ελαχιστοποιώντας τη συνάρτηση κόστους:

$$J_m = \sum_{n=1}^{N} w_n^{(m)} I(y_m(x_n) \neq t_n)$$

  - Υπολογίζουμε τις ποσότητες:

$$\varepsilon_m = \frac{\sum_{n=1}^{N} w_n^{(m)} I(y_m(x_n) \neq t_n)}{\sum_{n=1}^{N} w_n^{(m)}}$$

$$a_m = \ln\left\{\frac{1-\varepsilon_m}{\varepsilon_m}\right\}$$

  - Υπολογίζουμε τα βάρη του επόμενου βήματος:

$$w_n^{(m+1)} = w_n^{(m)} e^{a_m I(y_m(x_n) \neq t_n)}$$

- Κάνουμε τις προβλέψεις μας με τη χρήση του τελικού μοντέλου:

$$\hat{y}_{Boost} = sign\left(\sum_{m=1}^{M} a_m y_m(x)\right).$$

Από το βήμα υπολογισμού των βαρών του επόμενου βήματος παρατηρούμε ότι το βάρος αυξάνεται αν η παρατήρηση ταξινομήθηκε σωστά και μειώνεται αν ταξινομήθηκε λάθος. Οι επόμενοι ταξινομητές δηλαδή δίνουν έμφαση στις λάθος ταξινομημένες παρατηρήσεις. Οι ποσότητες ε_m αναπαριστούν τα αντιστοιχισμένα με βάρη μέτρα του σφάλματος κάθε αρχικού μοντέλου. Άρα τα a_m που είναι τα βάρη της ψηφοφορίας των μοντέλων δίνουν μεγαλύτερη επιρροή στην κρίση των πιο εύστοχων μοντέλων.

Στην R μια υλοποίηση του αλγορίθμου AdaBoost βρίσκεται στη βιβλιοθήκη “adabag” των (Alfaro, Gamez, & Garcia, 2013) και καλείται με την εντολή “boosting”. Η συνάρτηση εκτελέστηκε ως:

```
AdaBoost<- boosting(target~., data=dataset.train, mfinal=100)
```

Με *mfinal* τον αριθμό επαναλήψεων της μεθόδου. Η μέθοδος πέτυχε ακρίβεια 0.934 στο δείγμα Iris (βλ. Κεφάλαιο 5).

3.4.4. Μέτα-ταξινομητές

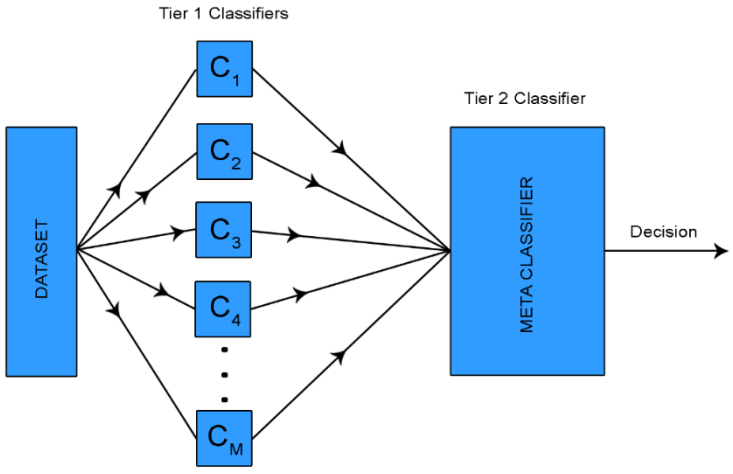
Οι **μέτα-ταξινομητές** (meta-classifiers) είναι μια κατηγορία συνδυαστικών ταξινομητών που αποτελούνται από δύο ή περισσότερες φάσεις. Στην πρώτη φάση προσαρμόζεται ένα σύνολο

από ταξινομητές. Στη δεύτερη φάση οι προβλέψεις αυτών των ταξινομητών γίνονται επεξηγηματικές μεταβλητές και προσαρμόζουν ένα ή περισσότερα νέα μοντέλα. Το **stacking** ή **stacked generalization** είναι ίσως η δημοφιλέστερη τέτοια διαδικασία. Χρησιμοποιεί έναν μετα-ταξινομητή και προσπαθεί να συνδυάσει μοντέλα τα οποία εξειδικεύονται σε διαφορετικά χωρία του προβλήματος όπως στο Διάγραμμα 3.11.

Κατά τη διαδικασία αυτή:

- Χωρίζουμε το δείγμα σε τρία ίσα μέρη.
- Με το πρώτο δημιουργούμε τους αρχικούς ταξινομητές.
- Οι αρχικοί ταξινομητές προβλέπουν τις τιμές των δεδομένων του δεύτερου μέρους.
- Οι προβλέψεις του δεύτερου μέρους χρησιμοποιούνται ως επεξηγηματικές μεταβλητές για να δημιουργηθεί το τελικό μοντέλο.
- Το τρίτο μέρος του δείγματος αποτελεί το δείγμα ελέγχου.

Η διαδικασία μπορεί να βελτιωθεί αν αντί για απλή πρόβλεψη, οι αρχικοί ταξινομητές επιστρέφουν την πιθανότητα της κάθε κλάσης. Σε αυτή την περίπτωση ο αριθμός των επεξηγηματικών μεταβλητών του μέτα-ταξινομητή είναι ίσος με τον αριθμό των κλάσεων του προβλήματος. Έχει δειχθεί ότι η παραπάνω διαδικασία δίνει τα βέλτιστα αποτελέσματα αν χρησιμοποιηθεί **γραμμική παλινδρόμηση πολλαπλής απόκρισης** (multi-response linear regression) ως μέτα-ταξινομητής. Υπό αυτές τις συνθήκες έχει δειχθεί ότι το stacking δίνει καλύτερα αποτελέσματα από ότι το καλύτερο αρχικό μοντέλο.



Διάγραμμα 3.11: Το stacking δημιουργεί ένα σύνολο από μοντέλα και χρησιμοποιεί τις προβλέψεις τους ως επεξηγηματικές μεταβλητές.

Η διαδικασία υλοποιήθηκε στην R χωρίζοντας το δείγμα σε τρία υποσύνολα. Από το πρώτο δημιουργήθηκαν τα αρχικά μοντέλα και πρόβλεψαν τις τιμές του δεύτερου. Από το δεύτερο και τις προβλεπόμενες τιμές των μοντέλων δημιουργήθηκε ο μέτα-ταξινομητής μέσω λογιστικής παλινδρόμησης. Ο μέτα-ταξινομητής εκτίμησε τις τιμές του τρίτου υποσυνόλου,

από το οποίο και κρίθηκε. Η παραπάνω διαδικασία πέτυχε ακρίβεια 0.969 στο δείγμα Iris που είναι η μέγιστη της παρούσας εργασίας (βλ. Κεφάλαιο 5).

4. Αριθμητικά Μέτρα Επίδοσης Μοντέλων

Στη διαδικασία δημιουργίας μοντέλων πρόβλεψης για ταξινόμηση η επικρατούσα πρακτική είναι να χωρίζεται το δείγμα σε δυο υποδείγματα. Το πρώτο καλείται **δείγμα εκπαίδευσης** (train set) και συνήθως επιλέγεται να είναι το 80% του συνολικού δείγματος. Το δεύτερο καλείται **δείγμα ελέγχου** (test set) που παραδοσιακά είναι το υπόλοιπο 20%. Ο διαχωρισμός γίνεται με απλή τυχαία δειγματοληψία χωρίς επανατοποθέτηση. Το μοντέλο δημιουργείται με τη χρήση μόνο του δείγματος εκπαίδευσης. Με αυτό τον τρόπο μπορούμε να δούμε πως συμπεριφέρεται το μοντέλο τόσο στις παρατηρήσεις που το δημιούργησαν όσο και σε νέες παρατηρήσεις. Έχοντας κάνει απλό τυχαίο διαχωρισμό των δεδομένων μας έχουμε εξασφαλίσει για μεγάλο δείγμα ότι τα υποδείγματα που δημιουργήσαμε προκύπτουν από την ίδια κατανομή.

Έστω ότι έχουμε δημιουργήσει δύο μοντέλα πρόβλεψης και θέλουμε να επιλέξουμε το ένα έναντι του άλλου. Χρησιμοποιώντας το δείγμα ελέγχου δημιουργούμε για καθένα από τα μοντέλα αυτά τον **πίνακα σύγχυσης** (confusion matrix), που είναι ο πίνακας με στήλες τον αριθμό των παρατηρήσεων που ταξινομήθηκαν σε κάθε κλάση και γραμμές τον αριθμό των παρατηρήσεων που ανήκουν σε κάθε κλάση. Δηλαδή στο κελί C_{ij} του πίνακα σύγχυσης αθροίζονται οι παρατηρήσεις που ενώ ανήκουν στην κλάση i ταξινομήθηκαν στην κλάση j. Προφανώς οι ορθές ταξινομήσεις βρίσκονται στη διαγώνιο του πίνακα.

Έστω ότι μελετάμε την κλάση k. Μέσω του πίνακα σύγχυσης ορίζονται οι **αληθώς θετικές** (true positive, συμβολίζουμε TP) προβλέψεις της κλάσης που είναι ο αριθμός των παρατηρήσεων που και ανήκουν και ταξινομήθηκαν στην κλάση k. Οι αληθώς θετικές προβλέψεις ταυτίζονται με το στοιχείο της διαγωνίου που αντιστοιχεί στη γραμμή της κλάσης δηλαδή στο παράδειγμά μας το κελί C_{kk}. Δηλαδή:

$$TP(k) = C_{kk}.$$

Ορίζονται επίσης οι **ψευδώς αρνητικές** (false negative, συμβολίζουμε FN) προβλέψεις της κλάσης που είναι ο αριθμός των παρατηρήσεων που ενώ ανήκουν στην κλάση k ταξινομήθηκαν σε κάποια άλλη κλάση. Μέσω του πίνακα σύγχυσης οι ψευδώς αρνητικές προβλέψεις υπολογίζονται ως το άθροισμα της k γραμμής εκτός του στοιχείου C_{kk}, δηλαδή:

$$FN(k) = \sum_{\substack{j \\ j \neq k}} C_{kj}.$$

Ορίζονται οι **αληθώς αρνητικές** (true negative, συμβολίζουμε TN) προβλέψεις της κλάσης k, δηλαδή το άθροισμα όλων των παρατηρήσεων που προβλέφθηκαν ως όχι k και δεν ήταν k. Μέσω του πίνακα σύγχυσης οι αληθώς αρνητικές προβλέψεις υπολογίζονται ως το άθροισμα όλου του πίνακα εκτός από τη γραμμή k και τη στήλη k, δηλαδή:

$$TN(k) = \sum_{\substack{i \\ i \neq k}} \sum_{\substack{j \\ j \neq k}} C_{ij}.$$

Τέλος ορίζουμε για μια πρόβλεψη k τις **ψευδώς θετικές** παρατηρήσεις (false positive, συμβολίζουμε FP) που είναι το σύνολο των παρατηρήσεων που ταξινομήθηκαν ως k ενώ δεν ανήκουν στην κλάση k. Μέσω του πίνακα σύγχυσης οι ψευδώς θετικές παρατηρήσεις υπολογίζονται ως το άθροισμα της στήλης k εκτός του στοιχείου C_{kk}, δηλαδή:

$$FP(k) = \sum_{\substack{i \\ i \neq k}} C_{ik}.$$

Οι παραπάνω έννοιες συνοψίζονται στο Διάγραμμα 4.1. Φυσικά ο πίνακας σύγχυσης είναι μια διαμέριση όλων των παρατηρήσεων και κατά συνέπεια ισχύει:

$$\sum_i \sum_j C_{ij} = N.$$

Predicted Class →

Actual Class ↓

	Class 1	Class 2	Class 3	Class 4
Class 1	21	1	0	2
Class 2	0	15	0	3
Class 3	2	0	25	0
Class 4	1	0	2	18

True Positive True Negative False Positive False Negative

Διάγραμμα 4.1: Στο διάγραμμα φαίνονται, πάνω στον πίνακα σύγχυσης, με χρωματική κωδικοποίηση οι true positive, true negative, false positive και false negative παρατηρήσεις του δείγματος ως προς την κλάση 1.

Με βάση τα παραπάνω, το πρώτο και απλούστερο κριτήριο επιλογής μοντέλου που μπορούμε να ορίσουμε είναι η **ακρίβεια** (accuracy) του μοντέλου που ορίζεται ως:

$$Accuracy = \frac{\sum_{k=1}^{c} TP(k)}{N} = \frac{\sum_{k=1}^{c} C_{kk}}{N}. \quad (4.1)$$

Δηλαδή η ακρίβεια είναι το ποσοστό ορθών ταξινομήσεων. Ισοδύναμο κριτήριο με την ακρίβεια είναι ο **ρυθμός σφάλματος** (error rate) που ορίζεται ως το ποσοστό λανθασμένων ταξινομήσεων, δηλαδή:

$$Error\ Rate = 1 - Accuracy.$$

Θα μπορούσε λοιπόν κανείς, αν έχει να επιλέξει μεταξύ δύο μοντέλων, να δει ποιο έχει την μεγαλύτερη ακρίβεια (ή αντιστοίχως το μικρότερο ρυθμό σφάλματος) και να κρατήσει αυτό. Το πρόβλημα σε αυτή την πρακτική είναι ότι, στην περίπτωση που οι κλάσεις δεν είναι ισορροπημένες ως προς τον αριθμό των παρατηρήσεων, η ακρίβεια μπορεί να γίνει τελείως παραπλανητική. Έστω ότι το 90% των παρατηρήσεων ανήκουν στην κλάση 1 και το άλλο 10% είναι μοιρασμένο στις υπόλοιπες κλάσεις. Τότε ένα μοντέλο που προβλέπει πάντα την κλάση 1 έχει κατευθείαν ακρίβεια 90% και ας προβλέπει λανθασμένα όλες τις άλλες παρατηρήσεις.

Δημιουργείται λοιπόν η ανάγκη να οριστούν παραπάνω κριτήρια που να καταπολεμούν το παραπάνω φαινόμενο.

Ορίζουμε την **ακρίβεια προσέγγισης** (precision ή positive predictive value) της κάθε κλάσης ως:

$$Percision(k) = \frac{TP(k)}{TP(k) + FP(k)},$$

που είναι το ποσοστό των παρατηρήσεων κλάσης k που ταξινομήθηκαν σωστά. Άμεσα ορίζεται και η **μέση ακρίβεια προσέγγισης** (average precision) ως:

$$Average\ Precision = \frac{\sum_{k=1}^{c} Precision(k)}{c}.$$

Ορίζουμε επίσης την **ευαισθησία** ή **ανάκληση** (sensitivity ή recall) της κάθε κλάσης ως:

$$Recall(k) = \frac{TP(k)}{TP(k) + FN(k)},$$

που μας δείχνει σε τι ποσοστό οι παρατηρήσεις που προβλέφθηκαν ως κλάσης k προβλέφθηκαν σωστά. Άμεσα ορίζεται και η **μέση ευαισθησία** ή **μέση ανάκληση** (average sensitivity ή average recall) ως:

$$Average\ Recall = \frac{\sum_{k=1}^{c} Recall(k)}{c}.$$

Η μέση ακρίβεια προσέγγισης και η μέση ανάκληση μπορούν να μελετηθούν ως δίπολο διότι είναι εύκολο να παρατηρηθεί υψηλό το ένα από τα δύο και χαμηλό το άλλο. Για να δημιουργηθεί ένα κριτήριο που να συνδυάζει και τα δύο αλλά να τείνει περισσότερο στο χαμηλότερο ώστε να είμαστε βέβαιοι ότι και τα δύο είναι ικανοποιητικά χρησιμοποιούμε τον αρμονικό τους μέσο. Ορίζουμε λοιπόν το **κριτήριο F_1** ως:

$$F_1 = 2\frac{Average\ Precision \times Average\ Recall}{Average\ Precision + Average\ Recall} = \frac{2}{1/Average\ Precision + 1/Average\ Recall}.$$

Εκτός από τη σύγκριση δύο μοντέλων, μέσω της ακρίβειας και του κριτηρίου F_1, μπορούμε να ελέγξουμε και την ποιότητα ενός μοντέλου ως προς το δίπολο μεροληψίας-διακύμανσης. Δημιουργούμε δυο πίνακες σύγχυσης, έναν με το δείγμα ελέγχου και ένα με το δείγμα εκπαίδευσης. Επιλέγουμε ένα κριτήριο, π.χ. την ακρίβεια:

- Στην περίπτωση που και οι δύο ακρίβειες είναι χαμηλές, είναι ένδειξη υπο-προσαρμογής μιας και είναι πιθανό το φαινόμενο να περιγράφεται από ένα πιο σύνθετο μοντέλο από αυτό που κατασκευάστηκε.
- Στην περίπτωση που η μια ακρίβεια είναι αρκετά μεγαλύτερη από την άλλη, είναι ένδειξη υπερ-προσαρμογής, μιας και το μοντέλο εξηγεί καλά έναν πληθυσμό αλλά όχι τόσο το φαινόμενο.
- Στην περίπτωση χασματικών διαφορών στις ακρίβειες, αν δηλαδή η μια είναι πολύ μεγάλη και η άλλη πολύ μικρή, είναι ένδειξη κακής δειγματοληψίας μιας και υπάρχει η πιθανότητα τα δύο δείγματα να μην είναι αντιπροσωπευτικά της κατανομής από την οποία προέρχονται.

Στις παραπάνω περιπτώσεις θεωρούμε ότι οι ανεξάρτητες μεταβλητές έχουν την αναγκαία προβλεπτική ικανότητα για την εξαρτημένη.

Φυσικά για να επιλεχθεί ένα μοντέλο θα πρέπει να εξεταστούν όλα τα παραπάνω. Δεν αρκεί δηλαδή η υπεροχή ενός μοντέλου στα κριτήρια που ορίσαμε έναντι κάποιου άλλου με χειρότερες επιδώσεις, πρέπει να ελεγχθεί η περίπτωση το μοντέλο να έχει καλύτερες επιδώσεις λόγω υπερ-προσαρμογής. Τα υπερ-προσαρμοσμένα μοντέλα έχουν καλύτερες επιδώσεις στα κριτήρια που παρουσιάσαμε συνήθως σε αυτά που προκύπτουν από το δείγμα εκπαίδευσης. Δεν σημαίνει όμως ότι είναι αδύνατο να υπερ-προσαρμώσει κάποιος ένα μοντέλο ως προς το δείγμα ελέγχου.

5. Εφαρμογή στο δείγμα Iris

5.1. Το Δείγμα Iris και Μεθοδολογία Ανάλυσης

Το δείγμα που χρησιμοποιήθηκε για την εφαρμογή των μεθόδων ταξινόμησης είναι το δείγμα “Iris” ή “Iris flower” ή “ Fisher’s Iris”. Πρόκειται για ένα δείγμα με τέσσερις επεξηγηματικές μεταβλητές και με τρείς κλάσεις στη μεταβλητή απόκρισης. Το δείγμα αυτό χρησιμοποιήθηκε πρώτη φορά το 1936 στο paper “The use of multiple measurements in taxonomic problems” του Ronald Fisher (Fisher, 1936). Αποτελείται από 50 παρατηρήσεις για καθένα από τα τρία είδη κρίνου “iris serosa”, “iris varginica”, “iris versicolor” και το συνολικό του μέγεθος είναι 150 παρατηρήσεις. Οι τέσσερις επεξηγηματικές μεταβλητές είναι το μήκος και το πλάτος των σεπάλων και των πετάλων του κρίνου σε εκατοστά.

Στην R είναι εκχωρημένο το δείγμα αυτό επειδή λόγω της ιστορικότητάς του είναι ένα από τα δημοφιλέστερα δείγματα για τον έλεγχο μεθόδων ταξινόμησης. Καλείται με την εντολή:

```
dataset<-iris
```

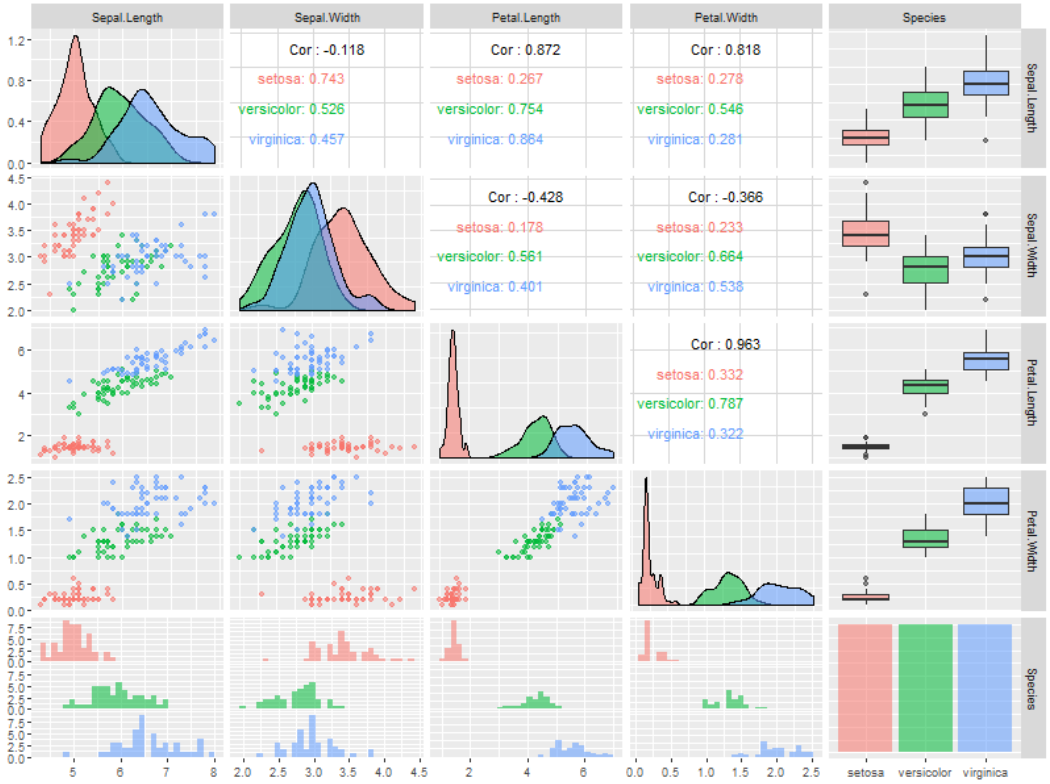
Για εποπτικούς λόγους κάνουμε τα ανά δύο διαγράμματα διασποράς, τα θηκογραφήματα ανά κλάση, τα ιστογράμματα ανά κλάση, τα διαγράμματα των εκτιμήσεών μας για τις συναρτήσεις πυκνότητας πιθανότητας ανά κλάση και τις ανά δύο συσχετίσεις κατά Pearson μέσω της εντολής “ggpairs” της βιβλιοθήκης “GGally” των Schloerke, et al. (2018) που είναι μια επέκταση της βιβλιοθήκης “ggplot2” του Wickham (2016). Η εντολή εκτελέστηκε ως:

```
ggpairs(dataset, mapping = aes(color = Species, alpha = 0.5))
```

Εδώ ορίσαμε τα χρώματα των διαγραμμάτων να αλλάζουν σύμφωνα με την τιμή της μεταβλητής Species μέσω του ορίσματος *color*. Θέσαμε 50% ημιδιαφάνεια στα στοιχεία των διαγραμμάτων μέσω του ορίσματος *alpha*.

Από το Διάγραμμα 5.5 παρατηρούμε ότι η κλάση “setosa” είναι πλήρως διαχωρίσιμη από τις άλλες δύο με τη χρήση μιας μόνο μεταβλητής, είτε της Petal.Length είτε της Petal.Width άρα η μόνη πρόκληση στο συγκεκριμένο δείγμα είναι ο διαχωρισμός των άλλων δύο κλάσεων.

Το δείγμα μας είναι πλήρως ισορροπημένο, δηλαδή κάθε κλάση έχει ίσο αριθμό παρατηρήσεων άρα δεν υπάρχει λόγος χρήσης του κριτηρίου F1. Επίσης δεν υπάρχει λόγος να προτιμήσουμε κάποιο είδος σφάλματος έναντι κάποιου άλλου. Λόγω της φύσης του προβλήματος κάθε σφάλμα μας κοστίζει το ίδιο οπότε θα θεωρήσουμε ως βέλτιστο κατώφλι να είναι το 0.5. Από τα παραπάνω καταλήγουμε ότι η ακρίβεια του μοντέλου αρκεί για την αξιολόγηση των ταξινομήσεων.



Διάγραμμα 5.1: Γραφική αναπαράσταση του δείγματος Iris. Αποτελείται από τα διαγράμματα διασποράς ανά δύο επεξηγηματικές μεταβλητές, εκτιμήσεις των διαγραμμάτων πυκνότητας πιθανότητας της κάθε μεταβλητής, θηκογραφήματα, ιστογράμματα και συσχετίσεις κατά Pearson με χρωματική διαφοροποίηση σύμφωνα με τη μεταβλητή απόκρισης Spieces.

Για τη δημιουργία και την αξιολόγηση των μεθόδων εκτός του stacking δημιουργήθηκαν δύο υποδείγματα του αρχικού δείγματος. Το δείγμα εκπαίδευσης αποτελείται από το 80% των παρατηρήσεων και το δείγμα ελέγχου από το υπόλοιπο 20%. Κάθε μοντέλο δημιουργήθηκε μέσω του δείγματος εκπαίδευσης και αξιολογήθηκε ως προς την ακρίβειά του μέσω του δείγματος ελέγχου. Κάθε διαδικασία εκτελέστηκε πέντε φορές και κάθε φορά επαναλαμβανόταν η τυχαία επαναδειγματοληψία για το διαχωρισμό δείγματος εκπαίδευσης και δείγματος ελέγχου. Η ακρίβεια που παρουσιάζεται είναι η μέση ακρίβεια των πέντε εκτελέσεων της κάθε μεθόδου. Η διαδικασία του ψηφίσματος έγινε πάνω στα αποτελέσματα όλων των διαδικασιών εκτός του stacking.

Για το stacking το δείγμα χωρίστηκε σε τρία ίσα υποδείγματα. Μέσω του πρώτου ξαναδημιουργήθηκαν όλα τα υπόλοιπα μοντέλα. Τα μοντέλα εκτίμησαν τις κλάσεις του δεύτερου υποδείγματος και το μοντέλο stacking δημιουργήθηκε μέσω λογιστικής παλινδρόμησης από το δεύτερο υποδείγμα με επεξηγηματικές μεταβλητές μόνο τις εκτιμήσεις των άλλων μοντέλων. Η αξιολόγηση του stacking έγινε μέσω του τρίτου υποδείγματος. Η διαδικασία αυτή εκτελέστηκε πέντε φορές και η ακρίβεια που παρουσιάζεται είναι η μέση ακρίβεια των εκτελέσεων.

5.2. Αποτελέσματα και Συμπεράσματα

Τα αποτελέσματα των μοντέλων που εκτελέστηκαν συνοψίζονται στον Πίνακα 5.1:

Accuracy							
3-NN	Kernel	Logistic Regression	Gaussian Naïve Bayes	LDA	QDA	Bayesian NB	Bayesian LR
0.947	0.941	0.934	0.962	0.960	0.966	0.922	0.933
Neural Network	SVM	CART	Bagging	Random Forests	AdaBoost	Voting	Stacking
0.933	0.947	0.928	0.928	0.934	0.934	0.941	0.969

Πίνακας 5.1: Οι ακρίβειες των ταξινομητών που δημιουργήθηκαν με το δείγμα Iris. Τις καλύτερες επιδόσεις πέτυχαν οι μέθοδοι που υπέθεσαν κανονικές κατανομές για τα δεδομένα (Gaussian Naïve Bayes, LDA, QDA) και ο μετα-ταξινομητής stacking

Η ακρίβεια υπολογίστηκε σύμφωνα με τον τύπο (4.1). Παρατηρούμε ότι χειρότερη επίδοση έχουν οι διαδικασίες που συσχετίζονται με δέντρα απόφασης, οι Μπεϋζιανές διαδικασίες, η λογιστική παλινδρόμηση και το νευρωνικό δίκτυο ενός κρυφού στρώματος με εννιά μονάδες.

Παρατηρώντας τα διαγράμματα διασποράς των δεδομένων βλέπουμε ότι σε πολλές περιπτώσεις υπάρχουν γραμμικά σύνορα απόφασης με κλίση περίπου 45° που θα διαχώριζαν καλά τις κλάσεις “iris varginica”, “iris versicolor”. Δεδομένης αυτής της παρατήρησης οι ταξινομητές που συσχετίζονται με δέντρα απόφασης ξεκινάνε με ένα μειονέκτημα. Φυσικά δεν περιμέναμε πολύ υψηλές επιδώσεις από τον αλγόριθμο CART μιας και η αξία του βρίσκεται στην εποπτικότητά του και στο ότι είναι η βάση για άλλους συνδυαστικούς ταξινομητές. Όμως έτσι μπορεί να εξηγηθεί η κακή επίδοση τον υπόλοιπων τέτοιων διαδικασιών. Το Bagging απέτυχε να διορθώσει τον αλγόριθμο CART, πετυχαίνοντας την ίδια ακρίβεια με αυτόν αλλά η βελτίωσή του, ο Random Forests, πέτυχε ελαφρώς καλύτερη ακρίβεια. Ο αλγόριθμος AdaBoost πέτυχε την ίδια σχετικά χαμηλή επίδοση με τον Random Forests.

Οι Μπεϋζιανές διαδικασίες έχουν χαμηλή επίδοση στο συγκεκριμένο δείγμα επειδή δεν γνωρίζαμε κάποια πληροφορία για τις πρότερες κατανομές. Στη Μπεϋζιανή λογιστική παλινδρόμηση για παράδειγμα θεωρήσαμε ότι οι παράγοντες της λογιστικής παλινδρόμησης έχουν μέση τιμή 0 και διασπορά 8. Ακόμα και ως μη πληροφορική πρότερη, δίνεται κάποια λανθασμένη πληροφορία για τους παράγοντες της λογιστικής παλινδρόμησης και ρίχνει την ακρίβεια ελαφρώς σε σχέση με την αντίστοιχη κλασική λογιστική παλινδρόμηση. Στον Μπεϋζιανό Naïve Bayes, εκτός από την υπόθεση ανεξαρτησίας Naïve Bayes, γίνεται και η υπόθεση ότι οι επεξηγηματικές μεταβλητές ακολουθούν κατηγορική κατανομή. Μέσω των διαγραμμάτων διασποράς αλλά και από τη φύση του προβλήματος αντιλαμβανόμαστε ότι και οι τέσσερις επεξηγηματικές μεταβλητές είναι συνεχείς μεταβλητές. Η διαδικασία διακριτοποίησής των συνεχών μεταβλητών είναι αιτία μείωσης της πληροφορίας που θα έχουν οι νέες ομαδοποιημένες μεταβλητές για την μεταβλητή απόκρισης. Η διακριτοποίηση είναι επίσης ο λόγος που οι δύο λογιστικές παλινδρομήσεις πέτυχαν σχετικά χαμηλές επιδόσεις.

Το νευρωνικό δίκτυο φαίνεται ότι χρειαζόταν παραπάνω δεδομένα για να πετύχει καλύτερη ακρίβεια. Το ότι το δημιουργήσαμε με εννιά μονάδες το έκανε πολύ ευέλικτο, άρα με μεγαλύτερο σφάλμα διακύμανσης. Ο περιορισμένος αριθμός παρατηρήσεων του δείγματος δεν ενδείκνυται για τόσο ευέλικτα μοντέλα.

Μέτρια σχετική απόδοση πέτυχαν οι μη παραμετρικές μέθοδοι 3-NN και kernel, όπως επίσης ο SVM και το voting. Ενώ τις βέλτιστες επιδόσεις πετυχαίνουν οι μέθοδοι που υποθέτουν κανονικές κατανομές στα δεδομένα, δηλαδή ο Gaussian Naïve Bayes, η LDA και η QDA όπως επίσης και η διαδικασία stacking.

Παρατηρούμε ότι όλες οι κλάσεις σε όλα τα διαγράμματα διασποράς σχηματίζουν ελλειψοειδή συμπλέγματα, κάτι που μας θυμίζει δεδομένα που προέρχονται από πολυμεταβλητή κανονική κατανομή. Οι υποθέσεις δηλαδή των Gaussian Naïve Bayes, LDA και QDA είναι εύλογες για το συγκεκριμένο δείγμα, και ως παραμετρικές και καθόλου ευέλικτες μέθοδοι συγκλίνουν γρήγορα στα αποτελέσματα. Η βάσιμη υπόθεση δηλαδή κάνει τις προβλέψεις τους εύστοχες και η μη ευέλικτη φύση τους κάνει την εύστοχη πρόβλεψη να έρχεται με πολύ λίγα δεδομένα.

Η διαδικασία του stacking σημείωσε την καλύτερη επίδοση και δείχνει να αντιλήφθηκε το ποιες μέθοδοι λειτούργησαν καλύτερα και σε ποια χωρία των λύσεων.

Παράρτημα

π.1. Βελτιστοποίηση

π.1.1. Κατάβαση βαθμίδας

Θα θεωρούμε ως βελτιστοποίηση μόνο την ελαχιστοποίηση συναρτήσεων. Στην περίπτωση που μας ενδιαφέρει η μεγιστοποίηση μιας $f(x)$, την αντιμετωπίζουμε ως τη διαδικασία ελαχιστοποίησης της $-f(x)$. Ακολουθείται η παρουσίαση του θέματος σύμφωνα με το βιβλίο των Goodfellow, Bengio, & Courville (2016, pp. 82-93, 150-152).

Όπως είναι γνωστό, η παράγωγος μιας συνάρτησης $f: \mathbb{R} \to \mathbb{R}$ δείχνει την κλίση της συνάρτησης, άρα μας υποδεικνύει το πόσο πρέπει να αυξομειώσουμε μια μικρή αλλαγή ε του x ώστε να πάρουμε την αντίστοιχη αλλαγή της f. Δηλαδή:

$$f(x + \varepsilon) \approx f(x) + \varepsilon f'(x).$$

Τα σημεία για τα οποία ισχύει $f'(x) = 0$ ονομάζονται κρίσιμα σημεία και υποδεικνύουν τοπικό ελάχιστο, τοπικό μέγιστο ή κάποιο **σαγματικό σημείο** (saddle point). Αν η παράγωγος είναι αρνητική γνωρίζουμε ότι για να βρούμε κάποιο τοπικό ελάχιστο πρέπει να κινηθούμε δεξιά και αν είναι θετική αριστερά. Οπότε γνωρίζουμε ότι για μικρό ε ισχύει:

$$f\left(x - \varepsilon\, sign\big(f'(x)\big)\right) < f(x).$$

Στην περίπτωση συναρτήσεων $f: \mathbb{R}^n \to \mathbb{R}$ η μερική παράγωγος $\frac{\partial f(x)}{x_i}$ δείχνει πως μεταβάλλεται η f μόνο ως προς τη μεταβλητή x_i. Ορίζεται η **βαθμίδα** ή **κλίση** (gradient) της f ως το διάνυσμα που περιέχει όλες τις μερικές παραγώγους της και είναι η γενίκευση της έννοιας της παραγώγου στη διανυσματική περίπτωση. Συμβολίζεται ως

$$\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = \begin{bmatrix} \frac{\partial f(\boldsymbol{x})}{x_1} \\ \vdots \\ \frac{\partial f(\boldsymbol{x})}{x_d} \end{bmatrix}.$$

Εδώ ένα κρίσιμο σημείο είναι το σημείο που έχει κάθε στοιχείο της βαθμίδας μηδενικό. Η **παράγωγος κατά κατεύθυνση** (directional derivative) μιας συνάρτησης f για μια κατεύθυνση $\boldsymbol{u}$ (για $\boldsymbol{u}$ μοναδιαίο) ορίζεται να είναι η κλίση της συνάρτησης στην κατεύθυνση $\boldsymbol{u}$. Δηλαδή είναι η παράγωγος της συνάρτησης $f(\boldsymbol{x} + \alpha \boldsymbol{u})$ ως προς το α. άρα

$$\frac{\partial f(\boldsymbol{x} + \alpha \boldsymbol{u})}{\partial \alpha} = \boldsymbol{u}^T \nabla_{\boldsymbol{x}} f(\boldsymbol{x}).$$

Για να ελαχιστοποιηθεί η f πρέπει να βρούμε τη διεύθυνση στην οποία η f μειώνεται γρηγορότερα, άρα ψάχνουμε το:

$$\min_{\boldsymbol{u},\, \boldsymbol{u}^T u = 1} \boldsymbol{u}^T \nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = \min_{\boldsymbol{u},\, \boldsymbol{u}^T u = 1} \|\boldsymbol{u}\|_2 \|\nabla_{\boldsymbol{x}} f(\boldsymbol{x})\|_2 \cos\theta.$$

Με θ τη γωνία μεταξύ του $\boldsymbol{u}$ και της βαθμίδας. Αντικαθιστώντας $\|\boldsymbol{u}\|_2 = 1$ και αγνοώντας το $\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x})\|_2$ που δεν εξαρτάται από το $\boldsymbol{u}$, απλοποιούμε στην εύρεση του $\min_{\boldsymbol{u}} \cos\theta$, το οποίο ελαχιστοποιείται όταν το $\boldsymbol{u}$ είναι αντίρροπο στη βαθμίδα. Άρα για να ελαχιστοποιήσουμε την f κινούμαστε κάθε φορά προς τη διεύθυνση $-\nabla_{\boldsymbol{x}} f(\boldsymbol{x})$ οπότε κάθε καινούριο σημείο θα υπολογίζεται μέσω του τύπου:

$$\boldsymbol{x}' = \boldsymbol{x} - \varepsilon \nabla_{\boldsymbol{x}} f(\boldsymbol{x}).$$

Το ε καλείται **ρυθμός μάθησης** (learning rate). Η διαδικασία που βρίσκει τοπικό ακρότατο επαναλαμβάνοντας την παραπάνω διαδικασία μέχρι να υπολογιστεί μηδενική βαθμίδα ονομάζεται μέθοδος **κατάβασης βαθμίδας** ή **ελάττωση κατά κλίση** (gradient descent ή steepest descent).

π.1.2. Πέρα από τη βαθμίδα

Για τις συναρτήσεις $f: \mathbb{R}^m \to \mathbb{R}^n$ ορίζεται ο **ιακωβιανός πίνακας** (Jacobian matrix) ως:

$$J = \begin{bmatrix} \dfrac{\partial \boldsymbol{f}(\boldsymbol{x})}{x_1} & \cdots & \dfrac{\partial \boldsymbol{f}(\boldsymbol{x})}{x_m} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f_1(\boldsymbol{x})}{\partial x_1} & \cdots & \dfrac{\partial f_1(\boldsymbol{x})}{\partial x_m} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial f_n(\boldsymbol{x})}{\partial x_1} & \cdots & \dfrac{\partial f_n(\boldsymbol{x})}{\partial x_m} \end{bmatrix}.$$

Στα προβλήματα με $f: \mathbb{R}^n \to \mathbb{R}$ που μας ενδιαφέρουν, η βαθμίδα είναι συνάρτηση $f: \mathbb{R}^n \to \mathbb{R}^n$ και μπορεί να οριστεί ο Ιακωβιανός πίνακάς της:

$$J\big(\nabla_{\boldsymbol{x}} f(\boldsymbol{x})\big) = \boldsymbol{H}(f)(\boldsymbol{x})_{i,j} = \frac{\partial^2}{\partial x_i \partial x_j} f(\boldsymbol{x}) = \begin{bmatrix} \dfrac{\partial^2 f(\boldsymbol{x})}{\partial x_1 \partial x_1} & \cdots & \dfrac{\partial^2 f(\boldsymbol{x})}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial^2 f(\boldsymbol{x})}{\partial x_n \partial x_1} & \cdots & \dfrac{\partial^2 f(\boldsymbol{x})}{\partial x_n \partial x_n} \end{bmatrix}.$$

Ο παραπάνω πίνακας καλείται **εσσιανός πίνακας** (Hessian matrix) και είναι ο πίνακας με όλες τις δεύτερες μερικές παραγώγους της f. Στα σημεία που οι δεύτερης τάξης μερικές παράγωγοι είναι συνεχείς, οι διαφορικοί τελεστές είναι αντιμεταθετικοί, δηλαδή:

$$\frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} = \frac{\partial^2 f(\boldsymbol{x})}{\partial x_j \partial x_i}.$$

Άρα ο εσσιανός πίνακας είναι συμμετρικός στα σημεία αυτά. Επειδή είναι πραγματικός και συμμετρικός, μπορούμε να εφαρμόσουμε **παραγοντοποίηση ιδιοτιμών** (eigenvalue decomposition) με πραγματικές ιδιοτιμές και ορθογώνια βάση ιδιοδιανυσμάτων. Η δεύτερη παράγωγος κατά κατεύθυνση $\boldsymbol{d}$ ($\boldsymbol{d}$ μοναδιαίο) είναι η ποσότητα $\mathbf{d}^{\boldsymbol{T}}\boldsymbol{H}\boldsymbol{d}$. Όταν το $\boldsymbol{d}$ είναι ιδιοδιάνυσμα, λαμβάνουμε την αντίστοιχη ιδιοτιμή, αλλιώς έναν γραμμικό συνδυασμό των ιδιοτιμών με συντελεστές στο $(0, 1)$. Η μέγιστη ιδιοτιμή μας δείχνει τη μέγιστη δεύτερη παράγωγο και η ελάχιστη ιδιοτιμή την ελάχιστη δεύτερη παράγωγο.

Η δεύτερη παράγωγος κατά κατεύθυνση μας δείχνει πόσο καλά πρέπει να περιμένουμε να λειτουργήσει το επόμενο βήμα της κατάβασης βαθμίδας. Κάνοντας προσέγγιση με

ανάπτυγμα Taylor δεύτερου βαθμού στη συνάρτηση $f(\boldsymbol{x})$ γύρω από το σημείο $\boldsymbol{x^{(0)}}$ του βήματος που βρισκόμαστε λαμβάνουμε:

$$f(\boldsymbol{x}) \approx f\left(\boldsymbol{x^{(0)}}\right) + \left(\boldsymbol{x} - \boldsymbol{x^{(0)}}\right)^T \boldsymbol{g} + \frac{1}{2}\left(\boldsymbol{x} - \boldsymbol{x^{(0)}}\right)^T \boldsymbol{H}\left(\boldsymbol{x} - \boldsymbol{x^{(0)}}\right),$$

με $\boldsymbol{g}$ τη βαθμίδα και $\boldsymbol{H}$ τον εσσιανό πίνακα. Το καινούριο σημείο της διαδικασίας θα είναι το $\boldsymbol{x} = \boldsymbol{x^{(0)}} - \varepsilon \boldsymbol{g}$ άρα ο τύπος θα γίνει:

$$f\left(\boldsymbol{x^{(0)}} - \varepsilon \boldsymbol{g}\right) \approx f\left(\boldsymbol{x^{(0)}}\right) - \varepsilon \boldsymbol{g}^T \boldsymbol{g} + \frac{1}{2}\varepsilon^2 \boldsymbol{g}^T \boldsymbol{H} \boldsymbol{g}.$$

Στον παραπάνω τύπο παρατηρούμε τρεις όρους: Την αρχική τιμή της συνάρτησης, την αναμενόμενη βελτίωση λόγω της κλίσης της συνάρτησης και τη διόρθωση που πρέπει να εφαρμόσουμε λόγω της καμπυλότητας της συνάρτησης. Παρατηρούμε βέβαια ότι αν η ποσότητα $\boldsymbol{g}^T \boldsymbol{H} \boldsymbol{g}$ είναι μηδενική ή αρνητική η προσέγγιση Taylor προβλέπει ότι όσο αυξάνουμε το ε θα μειώνεται η συνάρτηση για πάντα. Στην πραγματικότητα όμως γνωρίζουμε ότι η μέθοδος δεν θα είναι ακριβής για μεγάλα ε άρα θα πρέπει κάποιος να καταφύγει σε ευρετικές μεθόδους υπολογισμού του ε στην περίπτωση αυτή. Όταν το $\boldsymbol{g}^T \boldsymbol{H} \boldsymbol{g}$ είναι θετικό, λύνοντας ως προς το ε που θα κάνει την προσέγγιση Taylor να ελαχιστοποιηθεί περισσότερο λαμβάνουμε:

$$\varepsilon^* = \frac{\boldsymbol{g}^T \boldsymbol{g}}{\boldsymbol{g}^T \boldsymbol{H} \boldsymbol{g}}.$$

Στη χειρότερη περίπτωση το $\boldsymbol{g}$ θα έχει την ίδια κατεύθυνση με το ιδιοδιάνυσμα του $\boldsymbol{H}$ που του αντιστοιχεί η μέγιστη ιδιοτιμή. Τότε το βέλτιστο βήμα θα είναι $\frac{1}{\lambda_{max}}$.

Στις συναρτήσεις μιας μεταβλητής η δεύτερη παράγωγος χρησιμοποιείται για να προσδιοριστεί ο τύπος ενός κρίσιμου σημείου. Όταν δηλαδή η $f'(x) = 0$ τότε αν $f''(x) > 0$ σημαίνει ότι η $f'(x)$ αυξάνεται όσο κινούμαστε προς τα δεξιά και μειώνεται όσο κινούμαστε προς τα αριστερά. Άρα για μικρά ε ισχύει ότι $f'(x - \varepsilon) < 0$ και $f'(x + \varepsilon) > 0$. Αυτός ο τρόπος μας εξασφαλίζει ότι όταν $f'(x) = 0$, αν $f''(x) > 0$ βρισκόμαστε σε τοπικό ελάχιστο και αν $f''(x) < 0$ βρισκόμαστε σε τοπικό μέγιστο. Αυτή η διαδικασία είναι γνωστή ως το τεστ της δεύτερης παραγώγου. Όταν και η δεύτερη παράγωγος είναι μηδέν το τεστ δεν είναι ικανό να μας δώσει απάντηση μιας και μπορεί να βρισκόμαστε είτε σε σαγματικό σημείο είτε σε κάποιο επίπεδο τμήμα της συνάρτησης.

Στις πολλαπλές διαστάσεις χρειάζεται να εξετάσουμε όλες τις δεύτερες παραγώγους της συνάρτησης. Μπορούμε να γενικεύσουμε την παραπάνω διαδικασία χρησιμοποιώντας την παραγοντοποίηση κατά ιδιοτιμές του εσσιανού πίνακα της συνάρτησης. Σε ένα κρίσιμο σημείο, δηλαδή σε ένα σημείο που ισχύει $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = 0$ μπορούμε να εξετάσουμε τις ιδιοτιμές του εσσιανού πίνακα. Αν ο εσσιανός πίνακας είναι θετικά ορισμένος (έχει όλες τις ιδιοτιμές θετικές) βρισκόμαστε σε τοπικό ελάχιστο. Αν είναι αρνητικά ορισμένος (έχει όλες τις ιδιοτιμές αρνητικές) βρισκόμαστε σε τοπικό μέγιστο. Αν ο πίνακας έχει ταυτόχρονα και θετικές και αρνητικές ιδιοτιμές τότε βρισκόμαστε σε σαγματικό σημείο. Το τεστ δεν είναι ικανό να μας δώσει απάντηση στην περίπτωση που όλες οι μη μηδενικές ιδιοτιμές έχουν το ίδιο πρόσημο και υπάρχει έστω και μια μηδενική ιδιοτιμή.

Ο **δείκτης κατάστασης** (condition number) ενός πίνακα ορίζεται ως:

$$\kappa(A) = \|A\|\|A^{-1}\|$$

και αν είναι κανονικός, αν δηλαδή $A^*A = AA^*$ τότε $\kappa(A) = \left|\frac{\lambda_{max}}{\lambda_{min}}\right|$. Αν ο δείκτης κατάστασης είναι μεγάλος τότε ο πίνακας είναι ευάλωτος σε σφάλματα.

Στην περίπτωση του εσσιανού πίνακα (που είναι συμμετρικός άρα και κανονικός), ο δείκτης κατάστασης δείχνει πόσο διαφέρουν οι δεύτερες παράγωγοι της συνάρτησης. Όταν ο δείκτης κατάστασης είναι μεγάλος η μέθοδος κατάβασης βαθμίδας δεν συμπεριφέρεται καλά. Αυτό ισχύει γιατί η παράγωγος προς μια κατεύθυνση μπορεί να αυξάνεται γρήγορα ενώ η παράγωγος προς μια άλλη κατεύθυνση μπορεί να αυξάνεται πολύ αργά. Η μέθοδος κατάβασης βαθμίδας δεν μπορεί να αντιληφθεί αυτή την αλλαγή στην παράγωγο οπότε δε μπορεί να επιλέξει την κατεύθυνση στην οποία η παράγωγος παραμένει αρνητική για το μέγιστο διάστημα. Αυτή η κατάσταση μας δυσκολεύει και στην επιλογή του βήματος. Το βήμα σε αυτές τις περιπτώσεις πρέπει να είναι αρκετά μικρό ώστε να μην προσπεράσει το ελάχιστο σημείο. Αυτό σημαίνει όμως ότι σε άλλα σημεία με μικρότερη καμπυλότητα, λόγω του μικρού βήματος, η διαδικασία θα αργεί πολύ.

Αυτό το πρόβλημα μπορεί να καταπολεμηθεί χρησιμοποιώντας πληροφορία από τον εσσιανό πίνακα ώστε να καθοδηγηθεί η έρευνα του ελάχιστου. Η απλούστερη μέθοδος που το κάνει αυτό είναι η μέθοδος Newton-Raphson. Ξεκινώντας από το ανάπτυγμα Taylor δευτέρου βαθμού στο αρχικό σημείο $\boldsymbol{x^{(0)}}$:

$$f(\boldsymbol{x}) \approx f\left(\boldsymbol{x^{(0)}}\right) + \left(\boldsymbol{x} - \boldsymbol{x^{(0)}}\right)^T \boldsymbol{g} + \frac{1}{2}\left(\boldsymbol{x} - \boldsymbol{x^{(0)}}\right)^T \boldsymbol{H}\left(\boldsymbol{x} - \boldsymbol{x^{(0)}}\right).$$

Λύνουμε ως προς το κρίσιμο σημείο της συνάρτησης και λαμβάνουμε:

$$\boldsymbol{x^*} = \boldsymbol{x^{(0)}} - \frac{\boldsymbol{g}}{\boldsymbol{H}}.$$

Όταν η f είναι μια θετικά ορισμένη τετραγωνική συνάρτηση η μέθοδος Newton-Raphson μεταπηδά στο ελάχιστο της συνάρτησης σε ένα βήμα. Όταν η f δεν είναι πραγματικά τετραγωνική αλλά προσεγγίζεται τοπικά μέσω μιας θετικά ορισμένης τετραγωνικής συνάρτησης η Newton-Raphson εφαρμόζει τον παραπάνω τύπο επαναληπτικά. Η επαναληπτική διαδικασία μετάβασης στο ελάχιστο σημείο της προσέγγισης και η επαναπροσέγγιση της συνάρτησης συγκλίνει συνήθως γρηγορότερα από τη μέθοδο κατάβασης βαθμίδας αλλά είναι πιο επικίνδυνη κοντά σε σαγματικά σημεία. Σε αυτές τις περιπτώσεις η μέθοδος κατάβασης βαθμίδας λειτουργεί καλύτερα μιας και δεν ελκύεται από τα σαγματικά σημεία εκτός και αν η κλίση της συνάρτησης δείχνει προς σαγματικό σημείο.

Οι μέθοδοι βελτιστοποίησης που χρησιμοποιούν μόνο την κλίση, όπως η μέθοδος κατάβασης βαθμίδας, καλούνται **μέθοδοι βελτιστοποίησης πρώτης τάξης**. Μέθοδοι όπως η Newton-Raphson που χρησιμοποιούν και τον εσσιανό πίνακα καλούνται **μέθοδοι βελτιστοποίησης δεύτερης τάξης**.

Φυσικά όλα τα προβλήματα λύνονται όταν η συνάρτηση προς βελτιστοποίηση είναι κυρτή, μιας και σε αυτές τις συναρτήσεις δεν υπάρχουν σαγματικά σημεία.

π.1.3. Στοχαστική κατάβαση βαθμίδας

Ένα μόνιμο πρόβλημα της μηχανικής μάθησης είναι ότι χρειάζονται πολλά δεδομένα ώστε ένα μοντέλο να γενικεύει σωστά. Το πλήθος των δεδομένων μεταφράζεται σε υπολογιστικό κόστος. Μια συνάρτηση κόστους προς ελαχιστοποίηση συχνά αποδομείται σε ένα άθροισμα συναρτήσεων κόστους ανά παρατήρηση. Στο παράδειγμα της αρνητικής συνάρτησης λογαριθμο-πιθανοφάνειας γράφουμε:

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m} L\left(\boldsymbol{x}^{(i)}, y^{(i)}, \boldsymbol{\theta}\right),$$

όπου L η συνάρτηση κόστους ανά παρατήρηση, συγκεκριμένα:

$$L(\boldsymbol{x}, y, \boldsymbol{\theta}) = -logp(y|\boldsymbol{x}; \boldsymbol{\theta}).$$

Για αυτές τις αθροιστικές συναρτήσεις κόστους η διαδικασία κατάβασης βαθμίδας χρειάζεται τον υπολογισμό των:

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m} \nabla_{\boldsymbol{\theta}} L\left(\boldsymbol{x}^{(i)}, y^{(i)}, \boldsymbol{\theta}\right).$$

Το υπολογιστικό κόστος αυτής της διαδικασίας είναι της τάξης $O(m)$. Καθώς όμως το δείγμα αυξάνεται στο επίπεδο των δισεκατομμυρίων παρατηρήσεων ο χρόνος που χρειάζεται ένα μόνο βήμα της κατάβασης βαθμίδας γίνεται απαγορευτικός.

Η λογική της **στοχαστικής κατάβασης βαθμίδας** (Stochastic gradient descent) είναι να θεωρούμε τη βαθμίδα ως πρόβλεψη. Αυτή η πρόβλεψη μπορεί να γίνει χρησιμοποιώντας ένα μικρό υποσύνολο του αρχικού δείγματος. Σε κάθε βήμα λοιπόν κάνουμε μια επαναδειγματοληψία μεγέθους m' τάξης μεγέθους μερικών εκατοντάδων. Το κάθε υποσύνολο $B = \{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(m')}\}$ καλείται **προσαρμοσμένη δέσμη** (minibatch). Επιλέγεται ομοιόμορφα από το αρχικό μας δείγμα και μέσω αυτού προσεγγίζουμε τη βαθμίδα του βήματος ως:

$$\boldsymbol{g} = \frac{1}{m'}\nabla_{\boldsymbol{\theta}} \sum_{i=1}^{m'} L\left(\boldsymbol{x}^{(i)}, y^{(i)}, \boldsymbol{\theta}\right).$$

Η στοχαστική κατάβαση βαθμίδας προχωράει κανονικά στο επόμενο σημείο όπως και η κλασική κατάβαση βαθμίδας:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \varepsilon \boldsymbol{g},$$

με ε το ρυθμό μάθησης.

π.1.4. Πολλαπλασιαστές Lagrange-Δυισμός

Έστω το πρόβλημα βελτιστοποίησης:

$$\underset{\boldsymbol{x}}{\operatorname{argmin}}\big(f_0(\boldsymbol{x})\big) \ (1)$$

με τους περιορισμούς:

$$f_i(\boldsymbol{x}) \leq 0, \qquad i = 1, 2, \dots, m$$

$$h_i(\boldsymbol{x}) = 0, \qquad i = 1, 2, \dots, p$$

και $\boldsymbol{x} \in \mathbb{R}^n$. Θεωρούμε ότι το πεδίο ορισμού $D = \bigcap_{i=0}^{m} dom\, f_i \cap \bigcap_{i=1}^{p} dom\, h_i$ δεν είναι το κενό και συμβολίζουμε τη βέλτιστη τιμή του προβλήματος με p^*. Μπορούμε να μετασχηματίσουμε το πρόβλημα σε ένα χωρίς περιορισμούς με τη χρήση της συνάρτησης Lagrange. Ορίζουμε τη **συνάρτηση Lagrange** $L: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ που αντιστοιχεί στο πρόβλημα ως:

$$L(\boldsymbol{x}, \lambda, v) = f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i f_i(\boldsymbol{x}) + \sum_{i=1}^{p} v_i h_i(\boldsymbol{x}),$$

με πεδίο ορισμού $domL = D \times \mathbb{R}^m \times \mathbb{R}^p$. Το λ_i (αντίστοιχα v_i) είναι ο **πολλαπλασιαστής Lagrange** (Lagrange multiplier) της i-οστής ανισότητας $f_i(\boldsymbol{x}) \leq 0$ (αντίστοιχα $h_i(\boldsymbol{x}) = 0$) .

Ορίζουμε τη **δυϊκή συνάρτηση Lagrange** ή **δυϊκή συνάρτηση** (Lagrange dual function ή Dual function) $g: \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ ως την ελάχιστη τιμή της συνάρτησης Lagrange ως προς $\boldsymbol{x}$:

$$g(\boldsymbol{\lambda}, \boldsymbol{v}) = \inf_{\boldsymbol{x} \in D} L(\boldsymbol{x}, \lambda, v) = \inf_{\boldsymbol{x} \in D} \left(f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i f_i(\boldsymbol{x}) + \sum_{i=1}^{p} v_i h_i(\boldsymbol{x}) \right)$$

$$\boldsymbol{\lambda} \in \mathbb{R}^m, \boldsymbol{v} \in \mathbb{R}^p.$$

Όταν η συνάρτηση Lagrange δεν έχει κάτω φράγμα ως προς $\boldsymbol{x}$, η δυϊκή παίρνει την τιμή $-\infty$. Η δυϊκή συνάρτηση μας δίνει ένα κάτω φράγμα της βέλτιστης τιμής p^* του αρχικού προβλήματος. Για να βρούμε το καλύτερο κάτω φράγμα ορίζουμε το πρόβλημα βελτιστοποίησης:

$$\underset{\boldsymbol{\lambda}, \boldsymbol{v}}{\operatorname{argmax}}\big(g(\boldsymbol{\lambda}, \boldsymbol{v})\big) \ (2)$$

με τον περιορισμό:

$$\lambda_i \geq 0.$$

Αυτό το πρόβλημα καλείται το **δυϊκό πρόβλημα** (dual problem) του (1). Το (1) καλείται **πρωτογενές πρόβλημα** (primal problem).

Η βέλτιστη τιμή d^* ενός δυϊκού προβλήματος Lagrange είναι εξ' ορισμού το καλύτερο κάτω φράγμα της βέλτιστης τιμής p^* του πρωτογενούς προβλήματος που μπορεί να βρεθεί από τη δυϊκή συνάρτηση Lagrange. Πάντα ισχύει η παρακάτω ανισότητα που καλείται **ασθενής δυισμός** (Weak duality).

$$d^* \leq p^*.$$

Η διαφορά $p^* - d^*$ καλείται **βέλτιστο δυϊκό χάσμα** (Optimal duality gap).

Στην περίπτωση που ισχύει η ισότητα, δηλαδή όταν το δυϊκό χάσμα είναι μηδέν λέμε ότι ισχύει **ισχυρός δυισμός** (Strong duality).

Έστω $f_0, f_1, \dots, f_m, h_1, h_2, \dots, h_p$ διαφορίσιμες. Έστω επίσης $\boldsymbol{x}^*$ ένα πρωτογενές βέλτιστο σημείο και $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ ένα δυϊκό βέλτιστο σημείο με μηδενικό δυϊκό χάσμα. Επειδή το σημείο $\boldsymbol{x}^*$ ελαχιστοποιεί την $L(\boldsymbol{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ ως προς $\boldsymbol{x}$, η βαθμίδα πρέπει να μηδενίζεται στο x^* άρα:

$$\nabla f_0(\boldsymbol{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla f_i(\boldsymbol{x}^*) + \sum_{i=1}^{p} \nu_i^* \nabla h_i(\boldsymbol{x}^*) = 0 \quad (3)$$

άρα

$$f_i(\boldsymbol{x}^*) \leq 0, \qquad i = 1, \dots, m \quad (4)$$

$$h_i(\boldsymbol{x}^*) = 0, \qquad i = 1, \dots, p \quad (5)$$

$$\lambda_i^* \geq 0, \qquad i = 1, \dots, m \quad (6)$$

$$\lambda_i^* f_i(\boldsymbol{x}^*) = 0, \qquad i = 1, \dots, m. \quad (7)$$

Οι συνθήκες $(3) - (7)$ καλούνται **συνθήκες Karush-Kuhn-Tucker** (**KKT**) και είναι αναγκαίες συνθήκες για να είναι ένα σημείο βέλτιστο. Στην περίπτωση που το πρόβλημα είναι κυρτό, οι παραπάνω συνθήκες είναι και ικανές, δηλαδή το $\boldsymbol{x}^*$ και το $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ που τις ικανοποιούν είναι αντίστοιχα πρωτογενώς και δυϊκά βέλτιστα και έχουν μηδενικό δυϊκό χάσμα.

Έστω ένα δυϊκό πρόβλημα της μορφής:

$$\underset{\boldsymbol{\lambda}}{\operatorname{argmin}} \inf_{\boldsymbol{x} \in D} \left(f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i f_i(\boldsymbol{x}) \right)$$

με τον περιορισμό:

$$\lambda_i \geq 0, \iota = 1, 2, \dots, m.$$

Έστω $f_0, f_1, \dots, f_m$ κυρτές και συνεχώς διαφορίσιμες. Τότε το infimum προκύπτει όταν η βαθμίδα μηδενίζεται άρα το δυϊκό πρόβλημα (2) γίνεται:

$$\underset{\boldsymbol{\lambda}, \boldsymbol{\nu}}{\operatorname{argmax}} \left(f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i f_i(\boldsymbol{x}) \right),$$

με τους περιορισμούς:

$$\nabla f_0(\boldsymbol{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla f_i(\boldsymbol{x}^*) = 0,$$

$$\lambda_i \geq 0, i = 1, 2, \dots, m.$$

Το πρόβλημα αυτό καλείται δυϊκό Wolfe πρόβλημα. Στους περιορισμούς περιλαμβάνονται και οι συνθήκες Karush-Kuhn-Tucker. Ο περιορισμός αυτών των προβλημάτων μπορεί να είναι μη γραμμικός άρα το πρόβλημα μπορεί να γίνει μη κυρτό. Ο ασθενής δυισμός όμως ισχύει πάντα στα δυϊκά προβλήματα Wolfe.

π.2. Εκτιμήτρια μέγιστης πιθανοφάνειας

Οι παρακάτω συνθήκες και ιδιότητες παρουσιάζονται σύμφωνα με (Greene, 2011, pp. 554-555).

Ορίζουμε τις **Συνθήκες Κανονικής Εκτίμησης** (Regularity Conditions) για ένα τυχαίο δείγμα με συνάρτηση πυκνότητας πιθανότητας $f(y_i, \theta_0)$:

1. Οι τρείς πρώτες παράγωγοι του $lnf(y_i, \theta)$ ως προς θ είναι συνεχείς και πεπερασμένες για σχεδόν όλα τα y_i και για όλα τα θ. Αυτή η συνθήκη διασφαλίζει την ύπαρξη κάποιας προσέγγισης με σειρά Taylor και την πεπερασμένη διακύμανση των παραγώγων του lnL.
2. Οι αναγκαίες συνθήκες την εύρεση της αναμενόμενης πρώτης και δεύτερης παραγώγου του $lnf(y_i, \theta)$ ικανοποιούνται.
3. Για κάθε θ, το $|\frac{\partial^3 lnf(y_i,\theta)}{\partial\theta_j\partial\theta_k\partial\theta_l}|$ είναι μικρότερο από μια συνάρτηση που έχει πεπερασμένη αναμενόμενη τιμή. Αυτή η συνθήκη μας επιτρέπει να κόψουμε τη σειρά Taylor.

Υπό τις συνθήκες κανονικής εκτίμησης η εκτιμήτρια μέγιστης πιθανοφάνειας έχει τις παρακάτω ασυμπτωτικές ιδιότητες:

1. Είναι **συνεπής** (consistent). Δηλαδή $\hat{\theta} \xrightarrow{p} \theta_0$ καθώς $n \to \infty$.
2. Είναι **ασυμπτωτικά κανονική** (asymptotically normal). Δηλαδή $\hat{\theta} \xrightarrow{D} N(\theta_0, I^{-1})$ καθώς $n \to \infty$ όπου $I(\theta_0) = -E_0[\frac{\partial^2 lnL}{\partial\theta_0\partial\theta_0'}]$ ο **πίνακας πληροφορίας Fisher**.
3. Είναι **ασυμπτωτικά αποδοτική** (asymptotically efficient). Δηλαδή είναι συνεπής, ασυμπτωτικά κανονική και ο ασυμπτωτικός πίνακας συνδιακύμανσής της δεν είναι μεγαλύτερος από τον ασυμπτωτικό πίνακα συνδιακύμανσης καμίας άλλης συνεπούς και ασυμπτωτικά κανονικής εκτιμήτριας.
4. Είναι **Αναλλοίωτη** (invariant): Δηλαδή η εκτιμήτρια μέγιστης πιθανοφάνειας του $\gamma_0 = c(\theta_0)$ είναι η $c(\hat{\theta})$ αν η $c(\theta_0)$ είναι συνεχής και συνεχώς διαφορίσιμη συνάρτηση.

Η ιδιότητα της συνέπειας διασφαλίζει ότι καθώς το δείγμα μεγαλώνει η εκτίμησή πλησιάζει την πραγματική τιμή οσοδήποτε κοντά. Από 2, 3 η εκτιμήτρια μέγιστης πιθανοφάνειας είναι ασυμπτωτικά αμερόληπτη εκτιμήτρια ελάχιστης διασποράς μιας και η μέση τιμή της τείνει στην προς εκτίμηση ποσότητα λόγω της ιδιότητας 2 και πετυχαίνει ασυμπτωτικά το κάτω φράγμα Cramer-Rao από το 3. Ενδεικτικά αναφέρουμε ότι η εκτιμήτρια μέγιστης πιθανοφάνειας για τη μέση τιμή και τη διασπορά της κανονικής κατανομής είναι

$$\hat{\mu} = \frac{\sum_{i=1}^{n} x_i}{n} = \bar{x}_n \; και \; \hat{\sigma}^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x}_n)^2}{n}.$$

Η εκτιμήτρια μέγιστης πιθανοφάνειας της πιθανότητας εμφάνισης της κάθε τιμής της πολυωνυμικής κατανομής (και κατ' επέκταση της διωνυμικής, κατηγορικής και Bernoulli) είναι η σχετική συχνότητά της

$$\hat{p}_i = \frac{n_i}{n} \quad για \; κάθε \; i = 1, 2, \dots, c.$$

π.3. Στοχαστικές Ανελίξεις

Ορισμός 1: Έστω ένας χώρος πιθανότητας $(\Omega, \mathcal{F}, P)$, ένα σύνολο Χ που το καλούμε **χώρο καταστάσεων** (state space) και ένα άπειρο σύνολο $T \subset \mathbb{R}$. Έστω επίσης ότι για κάθε $t \in T$ υπάρχει μια τυχαία μεταβλητή $X_t: \Omega \to \mathbb{R}$ ορισμένη στο $(\Omega, \mathcal{F}, P)$. Η συνάρτηση $X: T \times \Omega \to \mathbb{R}$ που ορίζεται ως $X(t, \omega) = X_t(\omega)$ καλείται στοχαστική ανέλιξη. Γράφουμε: $X = \{X_t, t \in T\}$.

Παρατήρηση 1: Αν το T επιλεχθεί να είναι το $\mathbb{Z}^+$ η στοχαστική ανέλιξη καλείται **διακριτού χρόνου**. Αν το T επιλεχθεί να είναι το $\mathbb{R}^+$ η στοχαστική ανέλιξη καλείται **συνεχούς χρόνου**.

Ορισμός 2: Μια στοχαστική ανέλιξη καλείται **Μαρκοβιανή αλυσίδα** (Markov chain) αν για κάθε $n \in \mathbb{N}$ και κάθε $\upsilon_0, \dots, \upsilon_{n-1}, x, y \in X$ ισχύει:

$$P[X_{n+1} = y | X_0 = \upsilon_0, \dots, X_{n-1} = \upsilon_{n-1}, X_n = x] = P[X_{n+1} = y |\, X_n = x].$$

Ορισμός 3: Η συλλογή $P = \{p(x,y)\}_{x,y \in X}$ με $p(x,y) = P[X_{n+1} = y |\, X_n = x]$ ονομάζεται **πίνακας πιθανοτήτων μετάβασης** (transition matrix) της αλυσίδας.

Ορισμός 4: Θα λέμε ότι η κατάσταση $y \in X$ είναι **προσβάσιμη** από την κατάσταση $x \in X$ και θα συμβολίζουμε $x \to y$, αν υπάρχει $n \geq 0$ τέτοιο ώστε $p^{(n)}(x,y) > 0$.

Ορισμός 5: Θα λέμε ότι δύο καταστάσεις $x, y \in X$ **επικοινωνούν** και θα συμβολίζουμε $x \leftrightarrow y$, αν $x \to y$ και $y \to x$.

Πρόταση 1: Η σχέση $\leftrightarrow$ είναι μια σχέση ισοδυναμίας άρα διαμερίζει τον χώρο X σε κλάσεις που τις ονομάζουμε κλάσεις επικοινωνίας.

Ορισμός 6: Μια κλάση επικοινωνίας C θα λέγεται **ανοιχτή**, αν υπάρχουν καταστάσεις $x \in C$ και $y \notin C$ τέτοιες ώστε $x \to y$. Αν μια κλάση δεν είναι ανοιχτή, θα λέγεται **κλειστή**.

Ορισμός 7: Μια Μαρκοβιανή αλυσίδα λέγεται **μη υποβιβάσιμη** (irreducible), αν ολόκληρος ο χώρος καταστάσεων είναι μια (κλειστή αναγκαστικά) κλάση.

Ορισμός 8: Για μια κατάσταση $x \in X$ ορίζουμε το **σύνολο των δυνατών χρόνων επιστροφής** στο x ως:

$$R(x) = \{n \in \mathbb{N}: p^{(n)}(x,x) > 0\}, \qquad p^{(n)}(x,x) = P[X_n = x | X_0 = x]$$

Ο μέγιστος κοινός διαιρέτης του $R(x)$ ονομάζεται **περίοδος** της κατάστασης x και συμβολίζεται με $d(x)$. Στην περίπτωση που $d(x) = 1$ λέμε ότι η κατάσταση x είναι **απεριοδική** (aperiodic).

Ορισμός 9: Ορίζεται ο **χρόνος πρώτης επιστροφής** στη x ως $T_x^+ = \inf\{k > 0: X_k = x\}$.

Ορισμός 10: Θα λέμε μια κατάσταση $x \in X$ **γνησίως επαναληπτική** (positive recurrent) αν $E_x[T_x^+] < +\infty$.

Θεώρημα 1: Μια μη υποβιβάσιμη Μαρκοβιανή αλυσίδα σε έναν πεπερασμένο χώρο καταστάσεων είναι γνησίως επαναληπτική.

Θεώρημα 2: Μια μη υποβιβάσιμη και γνησίως επαναληπτική αλυσίδα έχει μοναδική αναλλοίωτη κατανομή.

Θεώρημα 3: Έστω μια μη υποβιβάσιμη, γνησίως επαναληπτική και απεριοδική αλυσίδα στον χώρο καταστάσεων X. Αν $\pi_n = P[X_n = x]$ για x ∈ X είναι η κατανομή της αλυσίδας μετά από n βήματα και π είναι η αναλλοίωτη κατανομή της αλυσίδας τότε $\lim_{n \to +\infty} \pi_n = \pi$.

π.4. Κώδικας Μπεϋζιανού Naïve Bayes

Ο παρακάτω κώδικας παρατίθεται σύμφωνα με τον Werner (Werner, 2014).

```
# bayesian naive bayes classifier for categorical inputs
# loosely based on some matlab code from the BRML toolbox
# http://web4.cs.ucl.ac.uk/staff/D.Barber/pmwiki/pmwiki.php?n=Brml.Software
# (c) Peter Werner July 2014
#
# as input it expects a set of features (as R factors) and targets
# prediction returns the probabilities and class labels
#
# the main differences are:
# 1) supports an arbitrary number of states per feature
# 2) "vectorized" prediction function which can take a table of observations
# 3) R-ification, expects features/targets to be factors
# see dbayes-test.R for examples.

condp <- function(pin) {
   p <- pin/max(pin)
   pnew <- p / sum(p)
   return(pnew)
}

logZdirichlet <- function(u) {
   lz <- sum(lgamma(u)) - lgamma(sum(u))
   return(lz)
}

nbd_train <- function(ytrain, xtrain) {

   classlabs <- unique(ytrain)
   nclass <- length(classlabs)
   nfeat <- ncol(xtrain)

   classcnt <- vector("numeric")
   upost <- list()

   #count occurances for each class
   for (cl in classlabs) {
         classcnt[cl] <- sum(ytrain == cl)
   }

   #for each feature
   for (feat in colnames(xtrain)) {
      #get how many states it has
      nstate <- length(levels(xtrain[,feat]))
      #our flat prior
      prior <- matrix(1, nrow=nclass, ncol=nstate, dimnames=list(levels(ytrain),
levels(xtrain[,feat])))
```

```
      #the posterior we wish to learn from the data
post <-  matrix(0, nrow=nclass, ncol=nstate, dimnames=list(levels(ytrain), levels(xtrain[,feat])))

      #for each class
      for (cl in classlabs) {
         #for each state in the feature
         for (st in levels(xtrain[,feat])) {
            #count how often this (class, feature) is in state st
            post[cl,st] <- prior[cl, st] + sum(xtrain[which(ytrain==cl),feat] == st)
         }
      }
      upost[[feat]] <- post
   }
   #prob. of each class
   cml <- condp(unlist(classcnt))
   return(list(upost=upost, classML=cml, labs=classlabs))
}

nbd_single <- function(xtest, nbc) {

   nbcp <- nbc$upost
   cml <- nbc$classML
   nclass <- length(cml)
   logclasspost <- vector("numeric")

   for (cl in nbc$labs) {
      logclasspost[cl] <- log(cml[cl])
   }

   for (feat in names(nbcp)) {
      nstate <- ncol(nbcp[[feat]])
      utest <- matrix(0, nrow=nclass, ncol=nstate,
                  dimnames=list(rownames(nbcp[[feat]]), colnames(nbcp[[feat]])))
      for (cl in nbc$labs) {
         for (st in colnames(nbcp[[feat]])) {
            utest[cl, st] <- nbcp[[feat]][cl, st] + sum(xtest[feat] == st)
         }
         lztest <- logZdirichlet(utest[cl,])
         lzpost <- logZdirichlet(nbcp[[feat]][cl,])
         logclasspost[cl] <- logclasspost[cl] + lztest - lzpost
      }
   }
   return(condp(exp(logclasspost)))
}
```

```
nbd_predict <- function(xtest, nbc) {

  nobs <- nrow(xtest)
  if (is.null(dim(xtest)) || nobs == 1) {
    return(nbd_single(xtest, nbc))
  }

  nbcp <- nbc$upost
  cml <- nbc$classML
  nclass <- length(cml)
  logclasspost <- matrix(log(cml), nrow=nobs, nc=nclass, byrow=T)
  colnames(logclasspost) <- names(cml)

  #for each feature
  for (feat in names(nbcp)) {
    #count how many states it has
    nstate <- ncol(nbcp[[feat]])
    #create an array to store class, state count matrix for each input observation
    utmat <- array(0, dim=c(nobs, nclass, nstate))
    dimnames(utmat) <- list(c(), rownames(nbcp[[feat]]), colnames(nbcp[[feat]]))
    #for each class
    for (cl in names(cml)) {
      #for each state in this feature
      for (st in colnames(nbcp[[feat]])) {
        #record the posterior + if this obs is in state st for feature feat
        utmat[,cl, st] <- nbcp[[feat]][cl, st] + ifelse(xtest[,feat] == st, 1, 0)
      }
      #work out the dirichlet numerator
      lztest <- apply(utmat[,cl,], 1, logZdirichlet)
      #and denominator
      lzpost <- logZdirichlet(nbcp[[feat]][cl,])
      #update the class posterior
      logclasspost[,cl] <- logclasspost[,cl] + lztest - lzpost
    }
  }
  #the class probabilities
  #we run through exp to get out of log space
  #then condp just makes sure they are legitimate probabilities
  classProbs <- t(apply(exp(logclasspost), 1, condp))
  #also return the class labels
  mlc <- apply(classProbs, 1, which.max)
  return(list(prob=classProbs, class=nbc$labs[mlc]))
}
```

Βιβλιογραφία

Α) Διεθνής Βιβλιογραφία

Alfaro, E., Gamez, M., & Garcia, N. (2013). adabag: An R Package for Classification with Boosting and Bagging. *Journal of Statistical Software, **54***, 1-35.

Alpaydin, E. (2010). *Introduction to Machine Learning.* MIT Press. London.

Barber, D. (2012). *Bayesian Reasoning and Machine Learning.* Cambridge University Press. Cambridge.

Barber, D. (2014). *BRMLtoolbox (matlab)*. Ανάκτηση από http://www.cs.ucl.ac.uk/staff/d.barber/brml/

Biau, G., & Devroy, L. (2015). *Lectures on the Nearest Neighbor Method.* Springer. London.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning.* Springer. Singapore.

Boyd, S., & Vandenberghe, l. (2004). *Convex Optimization.* Cambrodge University Press. Cambridge.

Breiman, L. (1996). Bagging Predictors. *Machine Learning, **24***, 123-140.

Breiman, L. (2001). Random Forests. *Machine Learning, **45***, 5-32.

Breiman, L. (2001). Statistical Modeling: The Two Cultures. *Statistical Science, **16***, 199-215.

Breiman, L., Friedman, J. H., Olsen, R. A., & Stone, C. J. (1984). *Classification and Regression Trees.* Chapman & Hall. New York.

Bürkner, P.-C. (2017). An R Package for Bayesian Multilevel Models. *Journal of Statistical Software, **80***, 1-28.

Bürkner, P.-C. (2018). Advanced Bayesian Multilevel Modeling with the R Package brms. *The R Journal, **10***, 395-411.

Collins, M. (2012). *The Naive Bayes Model, Maximum-Likelihood Estimation, and the EM Algorithm.* Ανάκτηση από http://www.cs.columbia.edu/~mcollins/em.pdf

Czepiel, S. A. (2002). *Maximum Likelihood Estimation of Logistic Regression Models: Theory and Implemendation.* Ανάκτηση από https://czep.net/stat/mlelr

Devroye, L., Gyorfi, L., & Lugosi, G. (1996). *A Probabilistic Theory of Pattern Recognition.* Springer. New York.

Domingos, P. (2000). A Unified Bias-Variance Decomposition for Zero-One and Squared Loss. *AAAI/IAAI, **2000***, 564-569.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics, **7***, 179-188.

Gamerman, D., & Lopes, H. F. (2006). *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference.* Chapman and Hall-CRC. Boca Raton.

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural Networks and the Bias/Variance Dilemma. *Neural Computation, **4***, 1-58.

Gentle, J. E. (2009). *Computational Statistics.* Springer-Velrag New York. New York.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning.* MIT Press. Massachusetts.

Greene, W. H. (2011). *Econometric Analysis.* Prentice Hall. New Jersey.

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining Concepts and Techniques.* Morgan Kaufman. New York.

Hastie, T., Tibshirani, R., & Friedman, J. (2013). *The Elements of Statistical Learning.* Springer. California.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2014). *An Introduction to Statistical Learning with applications in R.* Springer. New York.

Japkowicz, N., & Shah, M. (2011). *Evaluating Learning Algorithms: A Classification Perspective.* Cambridge University Press. Cambridge.

John, G. H., & Langley, P. (1995). Estimating Continuous Distributions in Bayesian Classifiers. (σσ. 338-345). Morgan Kaufmann Publishers Inc.

Karatzoglou, A., Smola, A., Hornik, K., & Zeileis, A. (2004). kernlab - An S4 Package for Kernel Methods in R. *Journal of Statistical Software, **11***, 1-20.

Li, F.-F., Jason, J., & Yeung, S. (2017). *CS231n: Convolutional Neural Networks for Visual Recognition. Spring 2017. Standford University*. Ανάκτηση από http://cs231n.stanford.edu/syllabus

Liaw, A., & Wiener, M. (2002). Classification and Regression by randomForest. *R News, **2***, 18-22.

Mamunur, R. (2008). *Inference on Logistic Regression Models.* Graduate College of Bowling Green. Ohio.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., & Leisch, F. (2018). *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien*. Ανάκτηση από http://www.stats.ox.ac.uk/pub/MASS4

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective.* MIT Press. London.

Negnevitsky, M. (2004). *Artificial Intelligence: A Guide to Intelligent Systems.* Addison Wesley. London.

Peters, A., & Hothorn, T. (2017). *ipred: Improved Predictors*. Ανάκτηση από https://CRAN.R-project.org/package=ipred

Quinlan, R. J. (1986). Induction of Decision Trees. *Machine Learning, **1***, 81-106.

Quinlan, R. J. (1993). *C4.5: Programs for machine learning.* Morgan Kaufmann. San Francisco.

Rennie, J. D. (2001). *Improving Multi-Class Text Classification with Naive Bayes.* Massachusetts: Massachusetts Institute of Technology-Artificial Intelligence Laboratory.

Rokach, L. (2010). *Pattern Classification Using Ensemble Methods.* World Scientific Publishing Company. Singapore.

Russell, S., & Norvig, P. (2010). *Artificial Intelligence a Modern Approach.* Prentice Hall. New Jersey.

Samaniego, F. J. (2010). *A Comparison of the Bayesian and Frequentist Approaches to Estimation.* Springer. New York.

Sammut, C., & Webb, G. I. (2011). *Encyclopedia of Machine Learning.* Springer. New York.

Sayad, S. (2010). *Classification-Basic Methods.* Ανάκτηση από University of Toronto: http://chem-eng.utoronto.ca/~datamining/Presentations/Basic_Methods

Schloerke, B., Crowley, J., Cook, D., Briatte, F., Marbach, M., Thoen, E., . . . Larmarange, J. (2018). *GGally: Extension to 'ggplot2'*. Ανάκτηση από https://CRAN.R-project.org/package=GGally

Silverman, B. W. (1998). *Density Estimation for Statistics and Data Analysis.* Chapman and Hall/CRC. Boca Raton.

Therneau, T., & Atkinson, B. (2018). *rpart: Recursive Partitioning and Regression Trees*. Ανάκτηση από https://CRAN.R-project.org/package=rpart

Tu, S. (2014). *The Dirichlet-Multinomial and Dirichlet-Categorical Models for Bayesian Inference*. Ανάκτηση 10 1, 2017, από https://people.eecs.berkeley.edu/~stephentu/writeups/dirichlet-conjugate-prior

Venables, W. N., & Riplay, B. D. (2002). *Modern Applied Statistics with S.* Springer. New York.

Werner, P. (2014). *Dirichlet-Bayes.R*. Ανάκτηση από https://github.com/petewerner/ml-examples/blob/master/dbayes/dirichlet-bayes.R

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis.* Springer. New York.

Wild, C. (2006). The Concept of Distribution. *Statistics Educational Research Journal, **5***.

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data Mining Practical Machine Learning Tools and Techniques.* Morgan Kaufman. Amsterdam.

Wolfe, P. (1961). A Duality Theorem for Non-Linear Programming. *Quarterly of Applied Mathematics, **19***, 239-244.

Β) Ελληνική Βιβλιογραφία

Λουλάκης, Μ. (2015). *Στοχαστικές Διαδικασίες.* [ηλεκτρ. βιβλ.] Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών. Αθήνα. Διαθέσιμο στο: http://hdl.handle.net/11419/6003.