Ατομική Διπλωματική Εργασία

ΣΥΣΤΗΜΑ ΑΞΙΟΛΟΓΗΣΗΣ ΤΗΣ ΑΞΙΟΠΙΣΤΙΑΣ ΤΩΝ ΠΕΛΑΤΩΝ ΧΡΗΜΑΤΟΟΙΚΟΝΟΜΙΚΩΝ ΙΔΡΥΜΑΤΩΝ

Γιάννος Καραολής

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάϊος 2009

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΣΥΣΤΗΜΑ ΑΞΙΟΛΟΓΗΣΗΣ ΤΗΣ ΑΞΙΟΠΙΣΤΙΑΣ ΤΩΝ ΠΕΛΑΤΩΝ ΧΡΗΜΑΤΟΟΙΚΟΝΟΜΙΚΩΝ ΙΔΡΥΜΑΤΩΝ

Γιάννος Καραολής

Επιβλέπων Καθηγητής
Ανδρέας Ανδρέου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάϊος 2009

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον καθηγητή κύριο Ανδρέα Ανδρέου, για την πολύτιμη βοήθεια που μου πρόσφερε κατά τη διάρκεια της υλοποίησης της διπλωματικής μου εργασίας αλλά και για την αμέριστη υπομονή που έδειξε μαζί μου. Επίσης, ένα μεγάλο ευχαριστώ στο Τμήμα Δανείων της υπηρεσίας IBU της Τράπεζας Κύπρου στην οποία και εργάζομαι, για τις πολύτιμες πληροφορίες που μου παρείχαν γύρω από το θέμα του Credit Scoring.

Περίληψη

Με την εκπόνηση της εργασίας αυτής, σκοπός ήταν να υλοποιηθεί ένα σύστημα credit scoring το οποίο με έξυπνο και γρήγορο τρόπο να προβλέπει με όσο το δυνατόν μεγαλύτερη επιτυχία και ακρίβεια την συμπεριφορά ενός δανειζόμενου κατά τη διάρκεια της αποπληρωμής του δανείου του.

Υλοποιήθηκε 1 μέθοδος που καλύπτει και την τεχνική των Decision Trees και την τεχνική των Classification Trees αφού οι δύο θεωρίες έτσι κι αλλιώς έχουν πολλές ομοιότητες και ελάχιστες διαφορές μεταξύ τους ενώ παράλληλα βασίζονται στους ίδιους αλγόριθμους οπότε η υλοποίηση δεν μπορούσε να είναι διαφορετική για την κάθε τεχνική.

Χρησιμοποιώντας σαν βάση τον διάσημο αλγόριθμο C4.5 (Quinlan, 1993) ο οποίος αποτελεί συνέχεια του αλγόριθμου ID3, καταφέραμε να κτίσουμε το δέντρο, με διάφορες τεχνικές να μικρύνουμε σε μέγεθος του δέντρου ενώ παράλληλα να αυξήσουμε την αποδοτικότητα του. Πήραμε διάφορες μετρήσεις για κάθε περίπτωση και συγκρίναμε τα αποτελέσματα.

Γενικά, τα αποτελέσματα ήταν αρκετά καλά αφού πλησίαζαν και μερικές φορές ξεπερνούσαν αυτά των ερευνών που μελετήθηκαν κατά τη θεωρία και στο τέλος αποδείχτηκε ότι η τεχνική των Decision/Classification Trees αποτελεί ένα γρήγορο και αποδοτικό τρόπο για την πρόβλεψη μιας συμπεριφοράς.

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

1.1 Ορισμός του Credit Scoring

Είναι ένα εξελιγμένο στατιστικά εργαλείο ή σύστημα που χρησιμοποιείται ευρέως από χρηματοοικονομικά ιδρύματα (κυρίως από Τράπεζες) στην προσπάθεια τους να κρίνουν αν θα εγκρίνουν ή να απορρίψουν ένα προσωπικό ή εταιρικό δάνειο. Καθορίζεται έτσι το ρίσκο που υπάρχει κατά τη διάρκεια της χορήγησης ενός δανείου.

Για να κατασκευαστεί ένα τέτοιο μοντέλο, θα πρέπει να αναλυθούν εις βάθος ιστορικά δεδομένα από πιο παλιά δάνεια που τυχόν να έκανε ο πελάτης αλλά και άλλες χρήσιμες πληροφορίες για τον πελάτη-δανειζόμενο οι οποίες μπορούν να παρθούν από την αίτηση δανείου την οποία θα συμπληρώσει. Στοιχεία όπως το μηνιαίο εισόδημα του πελάτη, τα συνολικά έξοδα του, πόσο καιρό ο πελάτης εργάζεται στην ίδια δουλειά, αν ο πελάτης είναι ιδιοκτήτης σπιτιού ή ενοικιάζει, αν κατέχει δικό του αυτοκίνητο, αν οι λογαριασμοί που διατηρεί σε Τράπεζες δεν είναι χρεωστικοί είναι μερικά από τα στοιχεία που οι περισσότερες Τράπεζες ζητούν να μάθουν έτσι ώστε να δημιουργήσουν ένα «προφίλ» για τον δανειζόμενο.

Στα περισσότερα μοντέλα (scoring systems), ένα ψηλό σκορ συνεπάγεται χαμηλό ρίσκο ενώ η Τράπεζα θέτει ένα όριο (το λεγόμενο cut off) βασιζόμενη στο βαθμό ρίσκου που είναι διαθετημένη να ανεχτεί. Είναι βέβαια στην δική της αρμοδιότητα και θέληση να

αποδεχτεί πελάτες με σκορ πιο ψηλό από το όριο που έθεσε και να απορρίψει πελάτες με σκορ μικρότερο από το όριο.

1.2 Πρόβλημα που καλείται να λύσει το Credit Scoring

Εδώ και αρκετά χρόνια διάφοροι οργανισμοί και εταιρείες, στην πλειοψηφία τους τράπεζες, προσπαθούν να υιοθετήσουν διάφορες αυτοματοποιημένες τεχνικές, οι οποίες θα τους βοηθήσουν τόσο στις αποφάσεις τους σε θέματα έγκρισης δανείων, όσο και στην εξοικονόμηση χρόνου και κόστους. Το μεγαλύτερο πρόβλημα που υπάρχει εδώ και χρόνια, είναι τα χρέη που δημιουργούνται στους χρηματοοικονομικούς οργανισμούς, λόγω της αδυναμίας διάφορων πελατών να ξεπληρώσουν τα δάνεια που τους παραχωρήθηκαν. Τρανό παράδειγμα αποτελεί το πρόσφατο θέμα χρεοκοπίας δύο μεγάλων τραπεζών της Αμερικής, που οφειλόταν στη συνεχή και απερίσκεπτη παραχώρηση δανείων, χωρίς την απαιτούμενη μελέτη, όσον αφορά την αξιοπιστία και δυνατότητα των πελατών να ξεπληρώσουν το δανειζόμενο ποσό. Για το σκοπό αυτό έχουν αναπτυχθεί διάφορα συστήματα τα οποία διαχωρίζουν τους πελάτες ως κακοπληρωτές ή καλοπληρωτές, βασισμένα σε παλαιότερες συναλλαγές τους. Με τον τρόπο αυτό, τα συστήματα παρέχουν περισσότερες πληροφορίες στους αρμόδιους για τα θέματα δανεισμού, βελτιώνοντας έτσι την ορθότητα των αποφάσεων τους.

1.3 Συμπερασματικά

Ένα καλό scoring system δεν θα προβλέψει με ακρίβεια την συμπεριφορά του δανειζόμενου κατά τη διάρκεια της αποπληρωμής του δανείου του αλλά θα δώσει στην Τράπεζα μια πολύ καλή εικόνα για αυτή την συμπεριφορά. Τουλάχιστον όμως, όσο είναι δυνατόν, το σύστημα θα μειώσει το ρίσκο που παίρνει η Τράπεζα για να δανείσει χρήματα.

Λαμβάνοντας υπόψη και τις σύγχρονες τάσεις της αγοράς που αναφέρουν ότι ο πλανήτης μπήκε ήδη σε κλοιό παγκόσμιας κρίσης, οι Τράπεζες και γενικότερα οι Χρηματοοικονομικοί οργανισμοί, θα πρέπει να είναι πολύ προσεκτικοί που και σε ποιον δανείζουν χρήματα. Ως εκ τούτου, η χρήση του συστήματος του credit scoring κρίνεται επιτακτική!!

Κεφάλαιο 2

Credit Scoring

2.1 Ιστορική Αναδρομή

Το σύστημα του Credit Scoring αρχικά γεννήθηκε σαν Credit Reporting και χρησιμοποιείτο από mail order organizations κυρίως για υποστήριξη πωλήσεων και από εκδότες πιστωτικών και επαγγελματικών καρτών αφού οι υπηρεσίες και τα προϊόντα τους είχαν τα εξής χαρακτηριστικά:

- Μεγάλος όγκος λειτουργιών
- Ελάχιστη επαφή με τον πελάτη
- Συγκεντρωμένες διαδικασίες
- Χαμηλής αξίας συναλλαγές

Ακολούθως, χρησιμοποιείτο ευρέως από τους λιανικούς έμπορους (retail merchants) της εποχής οι οποίοι στην προσπάθεια τους να δυναμώσουν την σχέση και την συνεργασία τους, αντάλλαζαν πληροφορίες για τους πελάτες τους. Έτσι είχαμε και τη δημιουργία των πρώτων credit bureaus.

Μετά το τέλος της Βιομηχανικής Επανάστασης, η ιδέα του Credit Reporting άρχισε να παίρνει την σημερινή του μορφή (Credit Scoring) και να εισάγεται δειλά-δειλά στις Τράπεζες αφού επικρατούσαν τα εξής χαρακτηριστικά:

- Μικρός όγκος λειτουργιών
- Πολλές συναλλαγές με μεγάλα ποσά

- Σημαντική η επαφή με τον πελάτη

Τα τελευταία 15 χρόνια όμως οι σύγχρονες τάσεις της αγοράς που κάνει τις ανάγκες των ανθρώπων για δανεισμό από Χρηματοοικονομικά Ιδρύματα τεράστιες, έχει καταστήσεις την εξάπλωση του credit scoring ραγδαία!!

2.2 Τεχνικές που χρησιμοποιούνται

Πολλές τεχνικές χρησιμοποιούνται ευρέως για την υλοποίηση συστημάτων credit scoring. Από τις πιο διάσημες, είναι αυτές των νευρωνικών δικτύων και των δέντρων αποφάσεων και ταξινόμησης (decision/classification trees) οι οποίες παρουσιάζουν αρκετά καλά αποτελέσματα. Άλλες κοινές τεχνικές οι οποίες χρησιμοποιούνται ήδη, είναι οι στατιστικές μέθοδοι Discriminate analysis και Logistic regression, οι οποίες είναι αρκετά αξιόπιστες.
Τεχνικές λιγότερο διάσημες και με διαδεδομένες με πιο μικρά ποσοστά επιτυχίας είναι οι γενετικοί αλγόριθμοι, γενετικός προγραμματισμός καθώς και data mining. Δειλά-δειλά κάνουν την εμφάνιση τους και συστήματα βασισμένα στη Θεωρία της Ασαφούς Λογικής τα οποία όμως βρίσκονται ακόμη σε πιλοτικό στάδιο.

2.3 Θεωρητικό Υπόβαθρο

Κατά τη διάρκεια του 1^{ου} εξαμήνου και στα πλαίσια της εκμάθησης της θεωρίας βάση της οποίας θα υλοποιηθεί το σύστημα, μελετήθηκαν κάποια άρθρα τα οποία αναφέρονται στο θέμα του Credit scoring, με στόχο την καλύτερη κατανόηση του θέματος αλλά και τη μελέτη διάφορων ιδεολογιών. Παρακάτω παραθέτω κάποιες περιπτώσεις υλοποίησης συστήματος Credit scoring, βασισμένες σε Decision/Classification trees:

I) *Analyzing Credit Risk Data: A comparison of Logistic Discrimination, Classification Tree Analysis and Feedforward Networks*, Gerhard Arminger, Daniel Enache, Thorsten Bonne, Germany

Το άρθρο αυτό υλοποιεί την τεχνική των Classification Trees και χρησιμοποιεί δεδομένα από μια μεγάλη Τράπεζα της Γερμανίας για όλα τα δάνεια που δόθηκαν σε πελάτες την χρονιά 1991-1992. Το αρχικό σύνολο των δεδομένων αριθμεί 8163 περιπτώσεις πελατών και χωρίζεται σε 3 κατηγορίες:

- Training sample – 2684 cases
- Cross Validation sample – 2734 cases
- Test Sample – 2745 cases

Η κάθε κατηγορία εξυπηρετεί και ένα σκοπό: η πρώτη καθορίζει τις παραμέτρους του μοντέλου, η δεύτερη για να μειωθεί το πρόβλημα του overfitting και η τρίτη για να ελεγχθεί η δυνατότητα πρόβλεψης του προγράμματος.

Για την καλύτερη πρόβλεψη της συμπεριφοράς ενός δανειζόμενου χρησιμοποιήθηκε η εξαρτώμενη μεταβλητή. Αυτή η μεταβλητή είναι binary και παίρνει την τιμή 1 αν σύμφωνα με το αποτέλεσμα του προγράμματος ΚΑΝΕΝΑ πρόβλημα δεν θα υπάρξει κατά την αποπληρωμή όλου του δανείου, ενώ αντίθετα παίρνει την τιμή 0 όταν σύμφωνα πάντα με την πρόβλεψη θα συμβεί το αντίθετο.

TYPE	0	1	∑
Training	1294 (48.2%)	1390 (51.8%)	2684
Cross Validation	1320 (48.3%)	1414 (51.7%)	2734
Test	1299 (47.3%)	1446 (52.7%)	2745
∑	3913 (47.9%)	4250 (52.1%)	8163

Πίνακας 2.1 – Συχνότητες των εξόδων του προγράμματος

Χρησιμοποιήθηκαν συνολικά 6 χαρακτηριστικά. Η Τράπεζα παρείχε στους ερευνητές πολλά άλλα χαρακτηριστικά αλλά τελικώς επιλέχθηκαν τα πιο κάτω που κρίθηκαν ότι θα μπορούσαν να βοηθήσουν περισσότερο από τα υπόλοιπα στην εκπόνηση του προγράμματος. Πιο κάτω παρουσιάζονται και οι κατηγορίες στις οποίες χωρίζονται αυτά τα χαρακτηριστικά:

1. “Sex” (SEX)

SEX	1	2
Training Set	489	2195
Cross Validation Set	503	2231
Test Set	508	2237

2. “Starting year of job” (SINCE)

SINCE	1	2	3	4
Training Set	899	621	507	657
Cross Validation Set	844	665	563	662
Test Set	892	695	537	621

3. “Year of Birth” (BIRTH)

BIRTH	1	2	3	4	5
Training Set	221	361	685	596	821
Cross Validation Set	192	383	711	604	844
Test Set	189	414	699	599	844

4. “Ownership of a car” (CAR)

CAR	1	2
Training Set	1932	752
Cross Validation Set	1955	779
Test Set	1965	780

5. “Ownership of a telephone” (TEL)

TEL	1	2
Training Set	1943	741
Cross Validation Set	1961	773
Test Set	1949	796

6. “Marital Status” (FAM)

FAM	1	2	3	4	5	6
Training Set	1059	1406	24	150	19	26
Cross Validation Set	1095	1465	23	130	8	13
Test Set	1079	1479	20	137	12	18

Το δύσκολο στην όλη μελέτη είναι να βρεθεί πιο από τα 6 χαρακτηριστικά θα πρέπει να διασπαστεί πρώτο. Το χαρακτηριστικό με την ψηλότερη ομοιογένεια

(highest measure of purity) θα διασπαστεί πρώτο. Για να υπολογιστεί η ομοιογένεια χρησιμοποιήθηκε η τεχνική Gini Index με τον εξής τύπο:

$$i(t) = 2p(1|t)(1 \quad p(1|t))$$

όπου t είναι ο εκάστοτε κόμβος, το p(1/t) είναι η πιθανότητα της κατηγορίας 1 και το 1- p(1/t) είναι η πιθανότητα της κατηγορίας 2.

Υπολογίστηκε το Gini Index για όλα τα χαρακτηριστικά καθώς επίσης και το ποσοστό λάθους ταξινόμησης με τον εξής τύπο:

$$R^*(T) = \sum_{t \in \tilde{T}} (1 - p(j(t)|t))C(j(t))p(t)$$

όπου T είναι το σύνολο των φύλλων του δέντρου, το C(j(t)) τα κόστη των λανθασμένων ταξινομήσεων, το p(t) είναι η πιθανότητα του κόμβου t και τέλος το j(t) είναι η κατηγορία στην οποία ανήκει το φύλλο.

Αφού εκτελέστηκαν όλοι οι απαραίτητοι υπολογισμοί, δημιουργήθηκαν οι κανόνες (rules) και κτίστηκε το δέντρο.

Οι κανόνες για τους κόμβους μη-φύλλα είναι οι εξής:

node	splitting	-	rule
1	FAM	∈	(1,4,5,6)
2	BIRTH	≤	1969.5
3	CAR	=	1
4	SINCE	≤	1986.5
5	FAM	∈	(2,3,6)
6	FAM	∈	(2,3,4,5)
7	TEL	=	1
8	BIRTH	≤	1954.5
9	BIRTH	≤	1968.5
10	SINCE	≤	1990.5
11	TEL	=	1
12	BIRTH	≤	1966.5
13	SINCE	≤	1988.5
14	SINCE	≤	1984.5

Πίνακας 2.2 – Κανόνες του Δέντρου (Μη-φύλλα)

Παράλληλα, τα αποτελέσματα για τα φύλλα του δέντρου που παράχθηκαν συνοψίζονται στον Πίνακα 2.3. Παρουσιάζονται ο αριθμός του φύλλου, σε ποια κατηγορία ανήκει το φύλλο (0 ή 1), αριθμός των δεδομένων που περνούν από το φύλλο και τέλος το ποσοστό λανθασμένης ταξινόμησης (probability of misclassification). Προφανώς, όσο πιο κοντά στο 0.5 είναι η πιθανότητα, τόσο το χειρότερο αφού τα δεδομένα δεν ταξινομήθηκαν με τον πιο βέλτιστο τρόπο.

node	category	n	prob. of misclassification
1	0	4	0.750
2	1	129	0.318
3	0	57	0.369
4	0	18	0.445
5	1	177	0.378
6	0	71	0.465
7	0	49	0.368
8	0	119	0.278
9	0	297	0.347
10	0	325	0.286
11	1	927	0.303
12	1	36	0.361
13	0	108	0.398
14	1	103	0.349
15	0	314	0.398

Πίνακας 2.3 – Αποτελέσματα φύλλων

Τέλος, ο Πίνακας 2.4 δείχνει χαρακτηριστικά την απόδοση (performance) του συστήματος ταξινομώντας τις τιμές του cross validation set.

	predicted		
observed	0	1	
0	882	438	0.6682
1	480	934	0.6605
	1362	1372	0.6642

Πίνακας 2.4 – Απόδοση συστήματος

II) *Analyzing Credit Risk Data: A comparison of Logistic Discrimination, Classification Tree Analysis and Feedforward Networks,* Gerhard Arminger, Daniel Enache, Thorsten Bonne, Department of Economics, Bergische Universitat, Germany

Στο δεύτερο άρθρο οι μελετητές χρησιμοποίησαν δεδομένα από 1001 ανθρώπους οι οποίοι είχα πιστωτική κάρτα, την είχαν χρησιμοποιήσει και το ιστορικό των συναλλαγών τους ήταν γνωστό. Σε περίπτωση που κάποιος από αυτούς είχε χάσει μια ή περισσότερες πληρωμές για εξόφληση του χρέους του, τότε αυτή χαρακτηριζόταν ως ‘slow’, αλλιώς ως ‘good’. Ένας πελάτης χαρακτηρίζεται

κακοπληρωτής αν αδυνατεί να πληρώσει 3 ή περισσότερες δόσεις. Χρησιμοποίησαν 14 χαρακτηριστικά, τα οποία φαίνονται πιο κάτω.

TABLE 1
Characteristics variables

Applicant's employment status
Years at bank
Home mortgage value
Number of children
Years at present employment
Residential status
Types of account
Other cards held
Outgoings
Estimated value of home
Home phone
Applicant's income
Spouse's income
Major credit cards held

Πίνακας 2.5 – Λίστα Χαρακτηριστικών

Όπως και στο προηγούμενο άρθρο, έτσι και τώρα οι μελετητές έπρεπε να υπολογίσουν την ομοιογένεια του κάθε χαρακτηριστικού. Η τεχνική που χρησιμοποιήθηκε αυτή τη φορά ήταν η Εντροπία και υπολογίζεται με τον τύπο:

$$I(T_k) = -\sum_{j=1}^{J} P_i(C_j) \cdot \log_2(P_i(C_j))$$

Επίσης, με τη βοήθεια του Gain Ratio, οι μελετητές κατόρθωσαν να υπολογίσουν το κέρδος διάσπασης για κάθε χαρακτηριστικό. Το Gain Ratio υπολογίζεται ως:

$$\text{Gain Ratio}_x = G_x / S_x$$

όπου $G_x = I(T) - I_x(T)$ και $S_x = -\sum_{k=1}^{n} p_k \cdot \log_2 p_k$

Κατασκευάστηκαν 10 διαφορετικά δέντρα βάση των 901 περιπτώσεων ενώ ελέχθησαν με τα εναπομείναντα 100 δεδομένα. Το μέγεθος των δέντρων ήταν από 421-470 κόμβους. Μετά την εφαρμογή του pruning, το μέγεθος μειώθηκε σε Στα 8 από τις 10 δέντρα η τεχνική του pruning μείωσε το ποσοστό λανθασμένης ταξινόμησης στο test set.

Κεφάλαιο 3

Τεχνικό υπόβαθρο

3.1 Δέντρα Ταξινόμησης

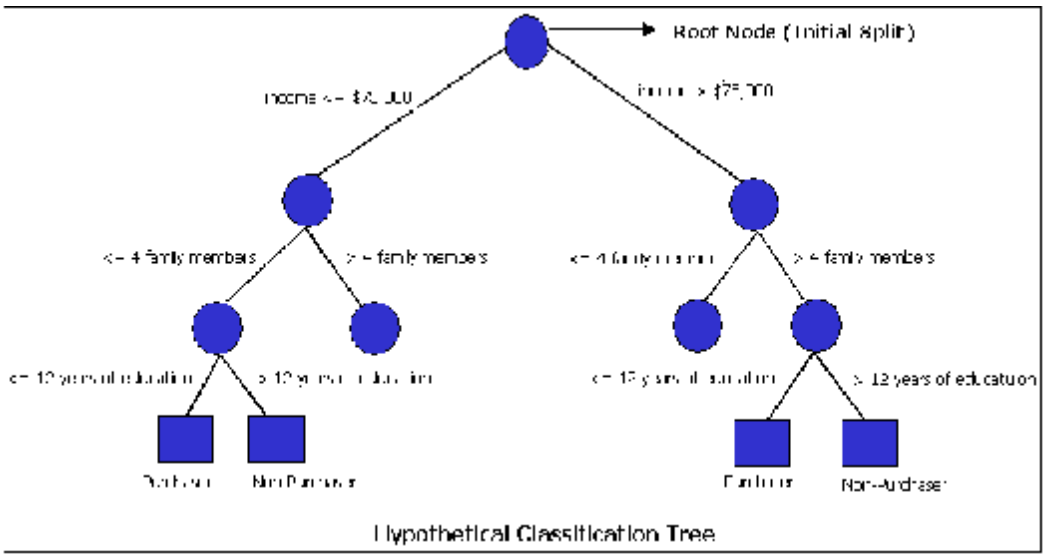
Τα δέντρα ταξινόμησης είναι πολύ καλή λύση όταν το ζήτημα είναι η ταξινόμηση ή η πρόβλεψη της εξόδου και σκοπός είναι να δημιουργηθούν κανόνες εύκολοι στην κατανόηση και επεξήγηση.

3.1.1 Κατασκευή Δέντρου Ταξινόμησης

Ένα δέντρο ταξινόμησης κατασκευάζεται βάση μιας πολύ διάσημης μεθόδου, της γνωστής binary recursive partitioning. Είναι μια επαναληπτική μέθοδος διαίρεσης του αρχικού συνόλου (root) των δεδομένων για μια εξαρτώμενη μεταβλητή (dependent categorical variable) σε 2 υποσύνολα-απογόνους. Το πρόβλημα είναι να βρεθεί η λεγόμενη explanatory μεταβλητή η οποία διαιρεί το σύνολο (sample) των δεδομένων. Τα υποσύνολα που δεν διαιρούνται περαιτέρω, ονομάζονται terminal nodes.

3.1.2 Παράδειγμα Δέντρου Ταξινόμησης

Το πιο κάτω παράδειγμα χρησιμοποιεί σαν classification label το purchaser και το non-purchaser. Δηλαδή θα πρέπει το κάθε φύλλο του δέντρου να περιέχει μόνο μία κατηγορία από τις πιο πάνω. Αρχικά όλα τα δεδομένα είναι μαζεμένα στην ρίζα του δέντρου. Έπειτα, ο αλγόριθμος επαναληπτικά χωρίζει τα δεδομένα σε 2 μέρη, εξετάζοντας κάθε φορά μια μεταβλητή λαμβάνοντας υπόψη τον κανόνα της μεταβλητής. Στο παράδειγμα, το αρχικό σύνολο χωρίζεται σε 2 κατηγορίες, χρησιμοποιώντας την μεταβλητή «income» και τον κανόνα <=$75,000 και >=$75,000. Ακολούθως, την μεταβλητή «family members» και τον κανόνα >4 και <=4, όπως φαίνεται και στο πιο κάτω σχήμα. Σκοπός της διάσπασης είναι το εκάστοτε καινούργιο σύνολο που δημιουργείται να περιέχει όσο το δυνατόν πιο ομοιογενή δεδομένα (πάντα από τις κατηγορίες purchaser και non-purchaser). Η διαίρεση συνεχίζεται ως ότου δεν υπάρχουν βοηθητικές διασπάσεις. Το πιο σημαντικό είναι να βρεθεί ο κανόνας βάση του οποίου γίνεται η αρχική διάσπαση.



Σχήμα 3.1 Παράδειγμα Classification Tree

3.2 Δέντρα Απόφασης (Decision Trees)

Ένα δέντρο απόφασης (decision tree) αναπαριστά μια διαδικασία λήψης απόφασης όπου κάθε πιθανό "σημείο απόφασης" ή κατάσταση αναπαρίσταται από έναν κόμβο ενώ κάθε πιθανή επιλογή που μπορεί να γίνει σε ένα σημείο απόφασης αναπαρίσταται με μια ακμή προς έναν κόμβο-παιδί.

3.2.1 Κατασκευή του δέντρου:

1. Ξεκινούμε έχοντας ένα κόμβο που περιέχει όλες τις εγγραφές.

2. Ακολουθεί η διάσπαση του κόμβου (μοίρασμα των εγγραφών) με βάση μια συνθήκη-διαχωρισμού σε κάποιο από τα γνωρίσματα.

3. Αναδρομική κλήση του βήματος 2 σε κάθε κόμβο (Recursive Partitioning Algorithm –RPA) έως ότου οι εγγραφές ενός τελικού κόμβου (φύλλο-leaf) να ανήκουν σε ένα μόνο σύνολο (goods ή bads).

4. Αφού κατασκευαστεί το δέντρο, γίνονται κάποιες βελτιστοποιήσεις (tree pruning)

3.2.2 Πλεονεκτήματα

- Μη παραμετρική προσέγγιση: Δε στηρίζεται σε υπόθεση εκ των προτέρων γνώσης σχετικά με τον τύπο της κατανομής πιθανότητας που ικανοποιεί η κλάση ή τα άλλα γνωρίσματα.
- Η κατασκευή του βέλτιστου δέντρου απόφασης είναι ένα NP-complete πρόβλημα.
- Αφού το δέντρο κατασκευαστεί, η ταξινόμηση νέων εγγραφών είναι πολύ γρήγορη
- Εύκολα στην κατανόηση (ιδιαίτερα τα μικρά δέντρα)

3.2.3 Μειονεκτήματα

- Δεν μπορεί να χειριστεί περίπλοκες σχέσεις μεταξύ των γνωρισμάτων
- Απλά όρια απόφασης (decision boundaries)
- Προβλήματα όταν λείπουν πολλά δεδομένα

3.3 Εξαρτώμενη Μεταβλητή (Dependent Variable) - Class

Για να «μετρήσουμε» ή να προβλέψουμε καλύτερα την συμπεριφορά ενός δανειζόμενου θα χρησιμοποιήσουμε την επονομαζόμενη εξαρτώμενη μεταβλητή. Αυτή η μεταβλητή θα είναι δίτιμη (binary) και θα παίρνει την τιμή 1 αν σύμφωνα με το αποτέλεσμα του προγράμματος ΚΑΝΕΝΑ πρόβλημα δεν θα υπάρξει κατά την αποπληρωμή όλου του δανείου, ενώ αντίθετα θα παίρνει την τιμή 0 όταν σύμφωνα πάντα με την πρόβλεψη θα συμβεί το αντίθετο.

3.4 Δεδομένα Εισόδου

Οι παρακάτω μεταβλητές αποφασίστηκαν όπως είναι οι είσοδοι στο πρόγραμμα μας. Αξίζει να σημειωθεί ότι πολλές άλλες μεταβλητές χρησιμοποιούνται στην Τράπεζα αλλά μερικές απ' αυτές είτε χρησιμοποιώντας τες δεν θα καλυτερεύσουν την απόδοση-πρόγνωση του συστήματος είτε δεν μπορούν καθόλου να χρησιμοποιηθούν λόγω των αυστηρών κανονισμών των Τραπεζών για προστασία των Προσωπικών Δεδομένων. Έτσι, αφού δεν κατέστη δυνατόν να εξασφαλίσουμε πραγματικά δεδομένα από κάποια Κυπριακή Τράπεζα, αποφασίστηκε όπως χρησιμοποιηθούν αυτούσια δεδομένα από την επίσημη ιστοσελίδα του Πανεπιστημίου του Μονάχου έτσι ώστε να μπορέσουμε να εκπονήσουμε την εργασία. Πιο κάτω φαίνεται αυτούσιος ο πίνακας όπως πάρθηκε από το Πανεπιστήμιο του Μονάχου και φαίνονται οι κατηγορίες κάθε γνωρίσματος και οι τιμές που παίρνουν.

Description	Categories	Value
Balance of current account	No balance or debit	2
	Less than 200 Euros	3
	More than 200 Euros	4
	No running account	1
Duration in months	<=6	10
	6 < ... <= 12	9
	12 < ... <= 18	8
	18 < ... <= 24	7
	24 < ... <= 30	6
	30 < ... <= 36	5
	36 < ... <= 42	4
	42 < ... <= 48	3
	48 < ... <= 54	2
	> 54	1
Payment of previous credits	No previous credits	2
	Paid back previous credits at this bank	4
	No problems with current credits at this bank	3
	Hesitant payment of previous credits	0
	There are further credits running but at other banks	1

Purpose of credit	new car	1
	used car	2
	items of furniture	3
	radio / television	4
	household appliances	5
	repair	6
	education	7
	vacation	8
	retraining	9
	business	10
	other	0
Amount of credit	<=500	10
	500 < ... <= 1000	9
	1000 < ... <= 1500	8
	1500 < ... <= 2500	7
	2500 < ... <= 5000	6
	5000 < ... <= 7500	5
	7500 < ... <= 10000	4
	10000 < ... <= 15000	3
	15000 < ... <= 20000	2
	> 20000	1
Value of savings	Less than 1000 Euros	2
	1000 <= ... < 5000 Euros	3
	5000 <= ... < 10000 Euros	4
	More than 10000 Euros	5
	No savings	1

Has been employed by current employer for	Unemployed	1
	<= 1 year	2
	1 <= ... < 4 years	3
	4 <= ... < 7 years	4
	>= 7 years	5
Installment in % of available income	>= 35	1
	25 <= ... < 35	2
	20 <= ... < 25	3
	< 20	4
Marital Status / Sex	male: divorced / living apart	1
	female: divorced / living apart / married & Male: single	2
	male: single	2
	male: married / widowed	3
	female: single	4
Further debtors / Guarantors	none	1
	Co-Applicant	2
	Guarantor	3
Living in current household for	< 1 year	1
	1 <= ... < 4 years	2
	4 <= ... < 7 years	3
	>= 7 years	4

Most valuable available assets	Ownership of house or land	4
	Savings contract with a building society / Life insurance	3
	Car / Other	2
	Not available / no assets	1
Age in years	0 <= ... <= 25	1
	26 <= ... <= 39	2
	40 <= ... <= 59	3
	60 <= ... <= 64	5
	>= 65	4
Further running credits	At other banks	1
	At department store or mail order house	2
	No further running credits	3
Type of apartment	Rented flat	2
	Owner-occupied flat	3
	Free apartment	1
Number of previous credits at this bank	One	1
	Two or three	2
	Four or five	3
	Six or more	4
Occupation	Unemployed / unskilled with no permanent residence	1
	Unskilled with permanent residence	2
	Skilled worker / skilled employee / minor civil servant	3
	Executive / self-employed / higher civil servant	4

Number of persons entitled to maintenance	0 to 2	2
	3 and more	1
Telephone	No	1
	Yes	2
Foreign worker	Yes	1
	No	2

Πίνακας 3.2 Δεδομένα Εισόδου (Dataset)

Κεφάλαιο 4

Μεθοδολογία

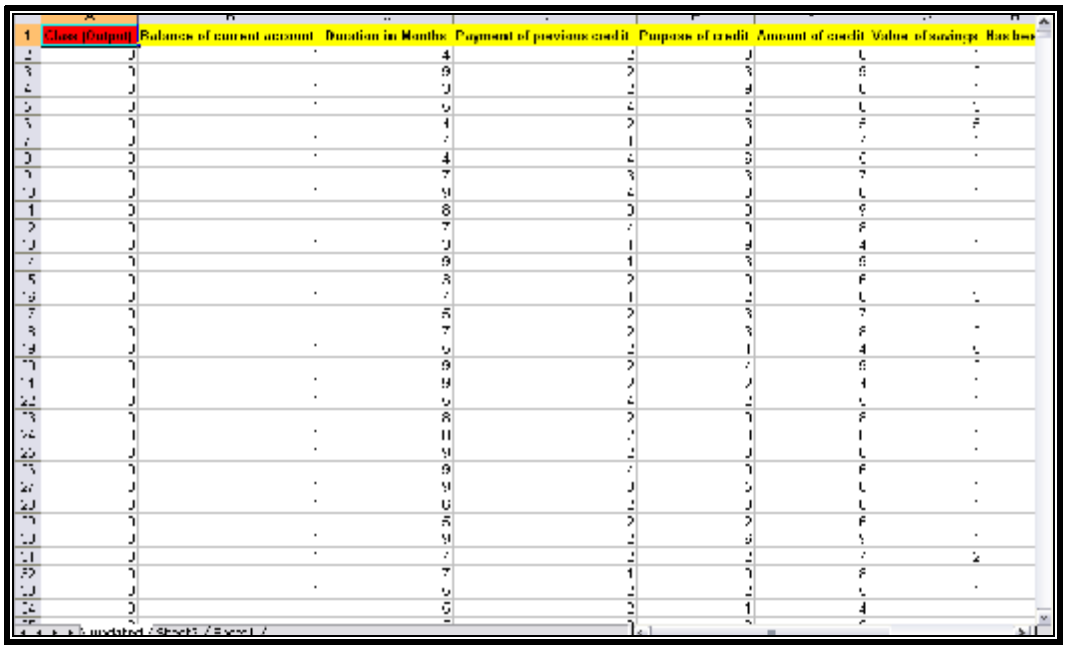
4.1. Σκοπός του συστήματος

Όπως έχει ήδη αναφερθεί, σκοπός της όλη εργασίας ήταν να υλοποιηθεί ένα έξυπνο σύστημα που έχει στόχο την επίτευξη της μέγιστης δυνατής επιτυχίας πρόβλεψης, όσον αφορά την αξιοπιστία ενός πελάτη.
Ως εκ τούτου, υλοποιήθηκαν δύο διαφορετικές μέθοδοι, οι οποίες στοχεύουν τόσο στην επίτευξη καλού αποτελέσματος, όσο και στην εξεύρεση των σημαντικότερων παραμέτρων του δικτύου αλλά και των σημαντικότερων παραγόντων που επηρεάζουν την εκπαίδευση του δικτύου, με βάση πάντα το dataset που αναφέρθηκε πιο πάνω.
Οι δύο αυτοί μέθοδοι αναλύονται πιο κάτω.

4.2 Υλοποίηση Συστήματος με τη μέθοδο Decision/Classification Trees

Τώρα, θα εξηγήσουμε βήμα-βήμα πως εργαστήκαμε για να υλοποιήσουμε το σύστημα χρησιμοποιώντας την μέθοδο των δέντρων απόφασης και πως πετύχαμε τα επιθυμητά αποτελέσματα.
Γενικά, τα δέντρα απόφασης (decision trees) θεωρείται μια ταξινομημένη δομή δέντρου που χρησιμοποιείται για να ταξινομήσει classes (τάξεις) με βάση κάποιους κανόνες (rules) για τα χαρακτηριστικά (attributes) της τάξης (class). Με άλλα λόγια, δεδομένης μιας σειράς από χαρακτηριστικά μαζί με τις τάξεις τους, ένα δέντρο απόφασης μπορεί να δημιουργήσει μια σειρά από κανόνες έτσι ώστε να «αναγνωριστεί» ή να «χαρακτηριστεί» η τάξη (class).

Στην δική μας περίπτωση, θα έχουμε 20 χαρακτηριστικά (βλέπε πίνακα 3.2) και 2 τάξεις (1 και 0). Ένα μέρος του πίνακα φαίνεται πιο κάτω. Για σκοπούς παρουσίασης το dataset εισάχθηκε στο Microsoft Excel.



Σχήμα 4.1. Παρουσίαση Δεδομένων Εισόδου

Παράλληλα, ένα δέντρο απόφασης μπορεί να χρησιμοποιηθεί για να βρεθεί σε ποια τάξη ανήκει μια σειρά από δεδομένα. Πολύ απλά, ακολουθώντας τα σωστό μονοπάτι στο δέντρο και φτιάχνοντας έτσι κανόνες, κάθε φορά που περνούμε από κάποιο κόμβο του δέντρου, μπορούμε εύκολα να καταλήξουμε στο φύλλο το οποίο θα μας δώσει και την σωστή τάξη (στην περίπτωση μας, 1=καλοπληρωτής, 0=κακοπληρωτής).

Τώρα, θα εξηγήσουμε το πώς εργαστήκαμε για να παράξουμε το δέντρο απόφασης με βάση τα δεδομένα μας, το οποίο αποτελεί και το πιο δύσκολο κομμάτι. Η διαδικασία αυτή είναι γνωστή και με το όνομα *decision tree inductive.* Χρησιμοποιήθηκε ο αλγόριθμος C4.5 ο οποίος αποτελεί επέκταση του γνωστού αλγόριθμου ID3 που κατασκευάστηκε αρχικά από τον Ross Quinlan.

Μερικές βελτιστοποιήσεις του C4.5 έναντι του ID3 και τις οποίες φυσικά εκμεταλλευτήκαμε είναι:

- Χειρίζεται και continuous και discrete δεδομένα με τη χρήση threshold

- Χειρίζεται training data ακόμα και με ελλιπή τιμές στα χαρακτηριστικά
- Χειρίζεται χαρακτηριστικά με διαφορετικά κόστη
- Η τεχνική του pruning μπορεί να εφαρμοστεί και μετά το ολοκληρωτικό κτίσιμο του δέντρου

Τα βήματα που ακολουθούν αποτελούν στην πράξη τα βήματα για υλοποίηση του αλγόριθμου.

Αφού έχουμε στη διάθεση μας ένα πίνακα (Table) που αποτελείται από χαρακτηριστικά και την τάξη τους, μπορούμε να μετρήσουμε την ομοιογένεια (homogeneity) του. Ένας τέτοιος πίνακας, ονομάζεται ομοιογενής (pure) αν αποτελείται μόνο από μία τάξη ενώ ονομάζεται ανομοιογενής (impure) αν αποτελείται από περισσότερες από μια τάξεις. Προφανώς, στην περίπτωση μας, ο δικός μας πίνακας (dataset) είναι impure αφού αποτελείται από 2 τάξεις (1 και 0).

Υπάρχουν διάφοροι τρόποι για να μετρήσει κάποιος την ανομοιογένεια (impurity degree) κάποιου πίνακα. Τρεις απ'αυτούς είναι:

1. *Εντροπία (Entropy)*

Υπολογίζεται ως εξής:

$$Entropy = \sum_j -p_j log_2 p_j$$

2. *Gini Index*

Υπολογίζεται ως εξής:

$$Gini\ Index = 1 - \sum_j p_j^2$$

3. *Classification Error*

Υπολογίζεται ως εξής:

$$Classification\ Error = 1 - max\{p_j\}$$

Όπου, p_j είναι η τιμής της πιθανότητας της τάξης j.

Δεν είναι ανάγκη να χρησιμοποιηθούν και οι 3 τεχνικές. Η μία είναι αρκετή και πιο διαδεδομένη είναι η εντροπία, η οποία επιλέχθηκε να χρησιμοποιηθεί και στη δική μας μελέτη.

Τα δικά μας δεδομένα είναι στο σύνολο 1000. Αυτά χωρίζονται 700 σε «καλοπληρωτές-1» και 300 σε «κακοπληρωτές-0».

Άρα, υπολογίζουμε την πιθανότητα της κάθε τάξης:

Prob (1) = 700/1000 = 0.7

Prob (0) = 300/1000 = 0.3

Ας σημειωθεί ότι για τον υπολογισμό της πιθανότητας, λαμβάνουμε υπόψη τις τάξεις και όχι τα χαρακτηριστικά.

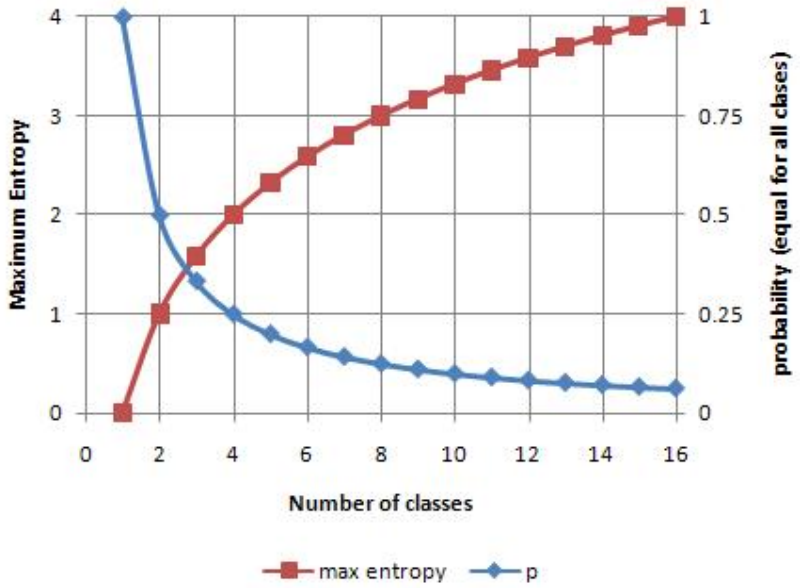
Έχοντας τώρα δεδομένες τις πιθανότητες, υπολογίζουμε το impurity του πίνακα μας χρησιμοποιώντας τις 3 πιο πάνω τεχνικές.

1. Εντροπία

Entropy = – 0.7 lg (0.7) – 0.3 lg (0.3) = 0.886

Η εντροπία ενός pure πίνακα, είναι 0 αφού η πιθανότητα είναι 1 και lg (1) = 0.

Επίσης, η εντροπία φτάνει την μέγιστη τιμή της όταν όλες οι τάξεις του πίνακα έχουν την ίδια πιθανότητα.



Σχήμα 4.2 Εντροπία vs. Αριθμός Κλάσεων vs. Πιθανότητες

Στην πιο πάνω γραφική παράσταση παρουσιάζονται οι τιμές της εντροπίας σε σύγκριση με τον αριθμό των τάξεων (n) και την πιθανότητα που είναι ίση με 1/n. Σε αυτή την περίπτωση, η μέγιστη εντροπία είναι ίση με –n*p*lgp.

Χαρακτηριστικά, η πιο πάνω γραφική παράσταση μας δείχνει ότι η τιμή της εντροπίας είναι μεγαλύτερη από 1 όταν οι τάξεις είναι περισσότερες από 2.

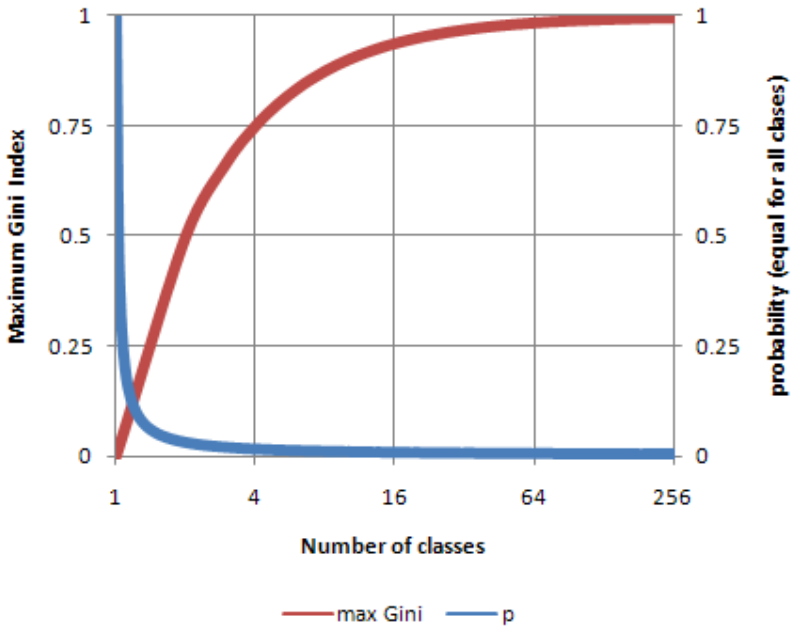
Επίσης, αξίζει να σημειωθεί ότι όταν ο αριθμός των groups της κλάσης είναι 2 (όπως και στη δική μας περίπτωση), η μέγιστη τιμή της εντροπίας είναι 1. Άρα, στους υπολογισμούς που θα ακολουθήσουν, θα πρέπει η τιμές των εντροπιών να κυμαίνονται από 0-1.

2. Gini Index

Υπολογίζουμε πάλι το impurity του πίνακα με την τεχνική Gini Index:

Gini Index = 1 – (0.7^2 + 0.3^2) = 0.42

Το Gini Index ενός pure πίνακα είναι 0 αφού η πιθανότητα είναι 1 και 1-(1^2)=0. Όπως και η εντροπία, το Gini Index φτάνει την μέγιστη τιμή του όταν όλες οι τάξεις του πίνακα έχουν την ίδια πιθανότητα, δηλαδή είναι ίσες με 0.5, ενώ παίρνει την ελάχιστη τιμή του όταν μια από τις τάξεις έχει πιθανότητα ίση με 0.



Σχήμα 4.3 Gini Index vs. Αριθμός Κλάσεων vs. Πιθανότητες

Στην πιο πάνω γραφική παράσταση παρουσιάζονται οι τιμές του Gini Index σε σύγκριση με τον αριθμό των τάξεων (n) και την πιθανότητα που είναι ίση με 1/n. Επιτυγχάνουμε μέγιστη τιμή στο Gini Index όταν: 1-n*(1/n)^2 = 1-1/n. Αξίζει να σημειωθεί ότι η τιμή του Gini Index είναι πάντα μεταξύ 0 και 1 ανεξάρτητα από τον αριθμό των τάξεων.

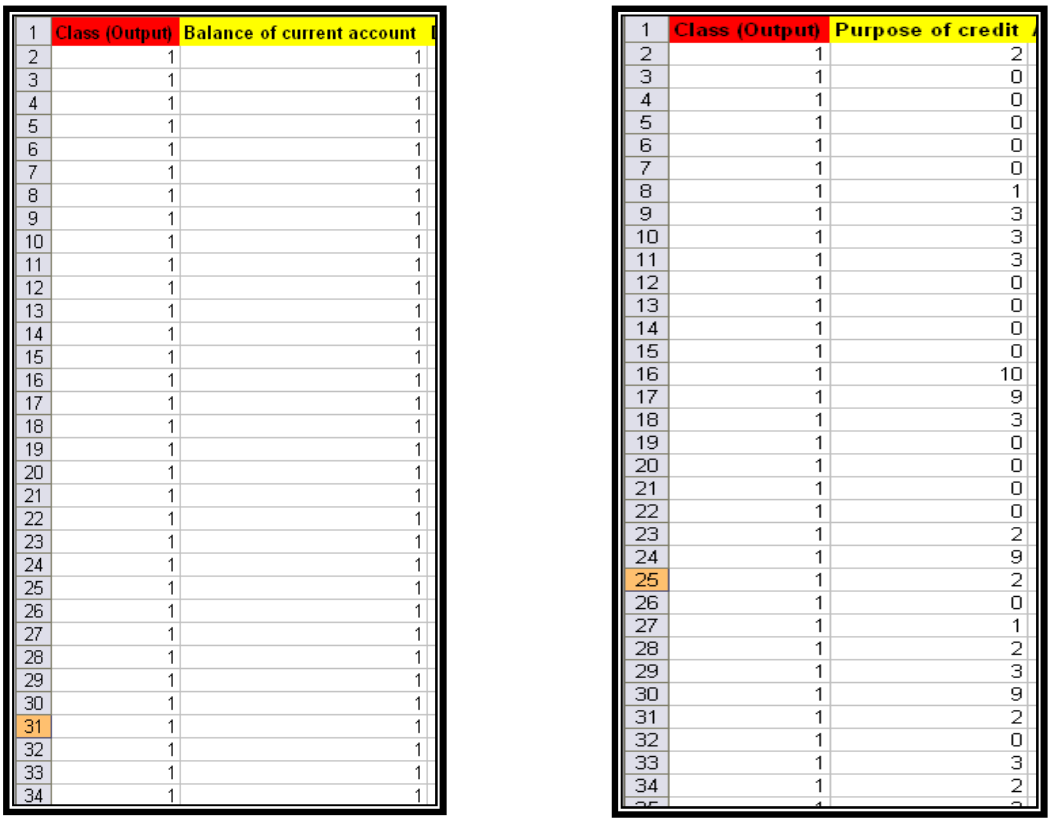
3. Classification Error

Ο τρίτος και τελευταίος τρόπος που χρησιμοποιήθηκε για τον υπολογισμό του impurity του πίνακα είναι η τεχνική classification error.

Classification Error Index = 1 – Max{0.7,0.3} = 1 - 0.7 = 0.3

Και σε αυτή την τεχνική, το Classification Error είναι 0 όταν ο πίνακας είναι pure. Η τιμή του Classification Error είναι πάντα μεταξύ 0 και 1 ενώ επιτυγχάνουμε μέγιστη τιμή όταν: 1-max{1/n} =1-1/n.

Αφού κατορθώσαμε να μετρήσουμε το impurity του πίνακα μας, προχωρήσαμε στην παραγωγή του δέντρου. Υπάρχουν πολλοί αλγόριθμοι που το επιτυγχάνουν αυτό όπως είναι ο ID3, CART, C4.5 και άλλοι. Όλοι έχουν κοινό ότι είναι επαναληπτικοί και όπως προαναφέρθηκε, εμείς χρησιμοποιήσαμε τον C4.5. Να πως συνεχίζει να δουλεύει ο αλγόριθμος: από τον αρχικό μας πίνακας που περιέχει τα 20 χαρακτηριστικά (20 στήλες) συν ακόμα μια στήλη που είναι η τάξη, απομονώνουμε κάθε χαρακτηριστικό (attribute) με την τάξη του. Δημιουργούμε δηλαδή 20 διαφορετικούς πίνακες με 2 στήλες (το attribute και το class) – Σχήμα 4.4.



1	Class (Output)	Balance of current account
2	1	1
3	1	1
4	1	1
5	1	1
6	1	1
7	1	1
8	1	1
9	1	1
10	1	1
11	1	1
12	1	1
13	1	1
14	1	1
15	1	1
16	1	1
17	1	1
18	1	1
19	1	1
20	1	1
21	1	1
22	1	1
23	1	1
24	1	1
25	1	1
26	1	1
27	1	1
28	1	1
29	1	1
30	1	1
31	1	1
32	1	1
33	1	1
34	1	1

1	Class (Output)	Purpose of credit
2	1	2
3	1	0
4	1	0
5	1	0
6	1	0
7	1	0
8	1	1
9	1	3
10	1	3
11	1	3
12	1	0
13	1	0
14	1	0
15	1	0
16	1	10
17	1	9
18	1	3
19	1	0
20	1	0
21	1	0
22	1	0
23	1	2
24	1	9
25	1	2
26	1	0
27	1	1
28	1	2
29	1	3
30	1	9
31	1	2
32	1	0
33	1	3
34	1	2

Σχήμα 4.4 Παραδείγματα 2 πινάκων μετά τη διάσπαση

Ταξινομούμε (sort) τον κάθε πίνακα ξεχωριστά βάση του εκάστοτε χαρακτηριστικού. Ακολούθως, διασπούμε περαιτέρω τους πίνακες αφού σπάζουμε τον κάθε ένα στις κατηγορίες από τις οποίες αποτελείται. Για παράδειγμα, το χαρακτηριστικό *Value of Savings*, μπορεί να πάρει τιμές από 1-5, ανάλογα από τις αποταμιεύσεις που έχει προφανώς ο πελάτης. Έτσι ο πίνακας που περιέχει το χαρακτηριστικό αυτό και την κλάση (τάξη-class) θα διασπαστεί σε 5 υπό-πίνακες. Υπολογίζουμε ακολούθως τις εντροπίες του κάθε υπό-πίνακα. Το ίδιο γίνεται για όλα τα χαρακτηριστικά του αρχικού dataset.

Το επόμενο βήμα είναι να υπολογιστεί το λεγόμενο Information Gain. Είναι η μέτρηση της διαφοράς της εντροπίας του αρχικού dataset και των εντροπιών των υπό-πινάκων. Έτσι, θα δούμε πιο θα είναι το κέρδος μας αν διασπάσουμε το αρχικό dataset βάση κάποιου χαρακτηριστικού και θα μπορέσουμε να συγκρίνουμε το impurity degree του πίνακα πριν και μετά τη διάσπαση.

Ο μαθηματικός τύπος που μας υπολογίζει το Information Gain είναι:

Information gain (i) = **Entropy of parent table D – Sum (n k /n * Entropy of each value k of subset table Si).**

Για κάθε χαρακτηριστικό του πίνακα δεδομένων υπολογίζεται το Information Gain. Μόλις γίνουν όλοι οι υπολογισμοί, βρίσκουμε το χαρακτηριστικό με το πιο ψηλό Information Gain (i* = argmax {information gain of attribute i}). Αυτό θα είναι και το χαρακτηριστικό που θα διασπαστεί πρώτο. Με το να επιλέξουμε το χαρακτηριστικό με το πιο ψηλό information gain ως το πρώτο που θα διασπαστεί, επιτυγχάνουμε τη γρήγορη διάσπαση των χαρακτηριστικών που θα μας μοιράσουν τα δεδομένα μας όσο πιο καλύτερα ποιοτικά και ποσοτικά γίνεται, κάνοντας έτσι το χτίσιμο του δέντρου πιο ομαλό και με μεγαλύτερη ακρίβεια.

Το πρόγραμμα μας, μετά από τους υπολογισμούς που ήδη έχουμε περιγράψει, αποφάσισε όπως το χαρακτηριστικό που θα διασπαστεί πρώτο είναι το *Balance of current account.*

Παράλληλα, η όλη διαδικασία επαναλαμβάνεται χωρίς τώρα να περιλαμβάνεται το πιο πάνω χαρακτηριστικό στον πίνακα δεδομένων έτσι ώστε να βρεθεί το επόμενο προς διάσπαση χαρακτηριστικό.

Κεφάλαιο 5

Συμπεράσματα

5.1. Αποτελέσματα προγράμματος

Αρχικά, σαν είσοδο του προγράμματος είχαμε τον πίνακα δεδομένων ακριβώς όπως ήταν, δηλαδή το 100% του dataset.
Από το πρόγραμμα, παράχθηκαν τα εξής αποτελέσματα:

	Predicted		
Observed	**0**	**1**	
0	282	18	300
1	36	664	700
	318	682	1000

Πίνακας 5.1 Πρώτα αποτελέσματα

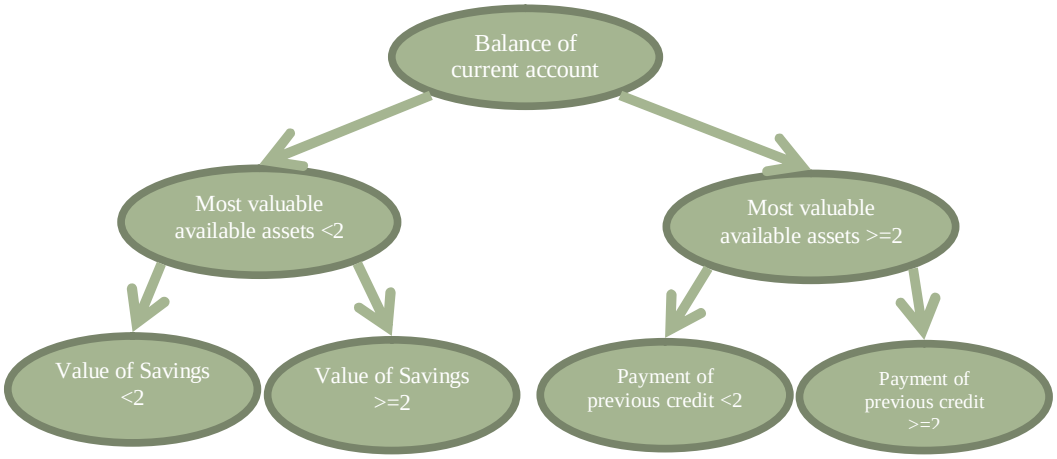
Το σημαντικό μήνυμα που παίρνουμε από τον πιο πάνω πίνακα είναι το ποσοστό της λάθος ταξινόμησης (misclassification). Αυτό το ποσοστό μας το δείχνουν οι αριθμοί με κόκκινο χρώμα (36 και 18). Το άθροισμα τους είναι 54 και το ολικό δείγμα ως γνωστό 1000. Άρα το ποσοστό της λάθος ταξινόμησης είναι 54/1000*100 = 5.4%.
Το ποσοστό δεν είναι και το χείριστο αλλά σίγουρα μπορεί να βελτιωθεί.
Άλλη σημαντική παρατήρηση από την πρώτη μας μέτρηση είναι και το μέγεθος του δέντρου που κτίστηκε. Αποτελείται από 320 κόμβους εκ των οποίων 161 είναι

φύλλα. Το σίγουρο είναι ότι το δέντρο είναι τεράστιο και θα πρέπει να γίνει πιο αποδοτικό μικραίνοντας παράλληλα σε μέγεθος!

Αξίζει εδώ να αναφέρουμε και την σειρά με την οποία το πρόγραμμα διάσπασε τα χαρακτηριστικά, ξεκινώντας από πάνω (ρίζα) προς τα κάτω:

1. Balance of current account
2. Most valuable available assets
3. Has been employed by current employer for
4. Value of Savings
5. Payment of previous credit
6. Duration in Months
7. Further running credits
8. Instalment in % of available income

Ένα μέρος του δέντρου που παράχθηκε από το πρόγραμμα μας, φαίνεται και πιο κάτω.



Σχήμα 5.1 Μέρος του δέντρου

Μια καλή τεχνική για να πετύχουμε καλύτερη αποδοτικότητα και μικρότερο μέγεθος του δέντρου είναι να διασπάσουμε το αρχικό dataset μας σε 2 μέρη: το training set και το test set.

Το training set θα χρησιμοποιείται για το κτίσιμο του δέντρου ενώ το test set για την μέτρηση της ακρίβειας (accuracy).

Πήραμε συνολικά 10 μετρήσεις με διαφορετικά ποσοστά στο training και test set.

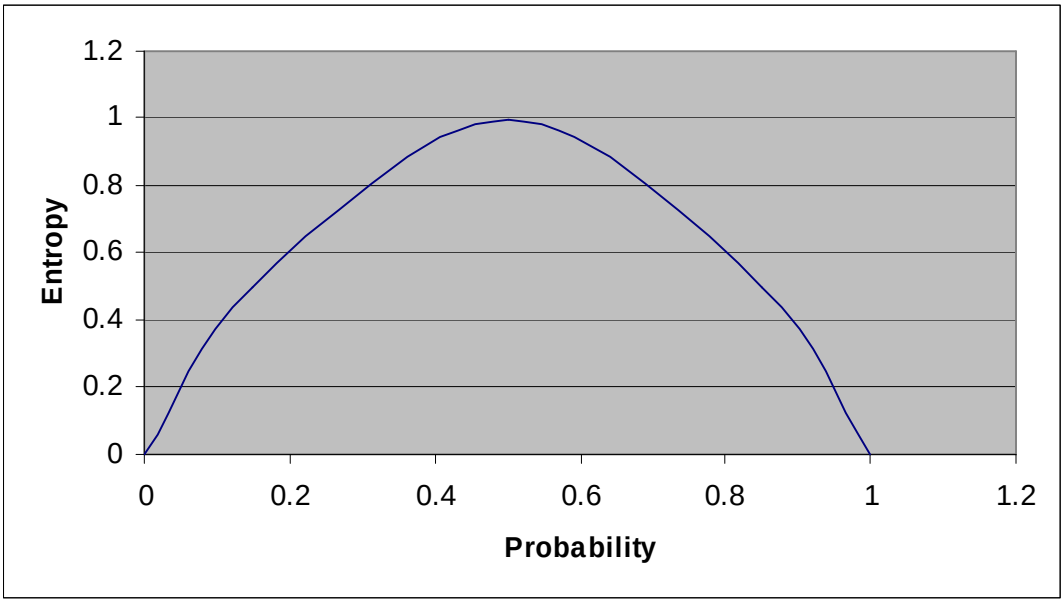
Παρακάτω φαίνεται αναλυτικά ο πίνακας με τις μετρήσεις:

RESULTS						
Training Data (%)	Test Data (%)	Number of nodes	Number of leaves	Average accuracy of the rules	missclassification on training data (%)	missclassification on test data (%)
99	1	310	156	71.2	6.27	36.36
95	5	310	156	70.4	5.6	35.85
90	10	286	144	70.5	5.39	21.82
85	15	280	141	72.5	4.74	28.21
80	20	280	141	70.6	5.65	32.26
73	27	242	122	71.1	5.14	32.14
70	30	242	122	70.8	5.42	29.77
65	35	224	113	70.9	4.4	34.9
60	40	192	97	70.7	6.3	27.2
50	50	166	84	73.2	4.55	28.95

Πίνακας 5.2 Μετρήσεις μετά τη διάσπαση του dataset

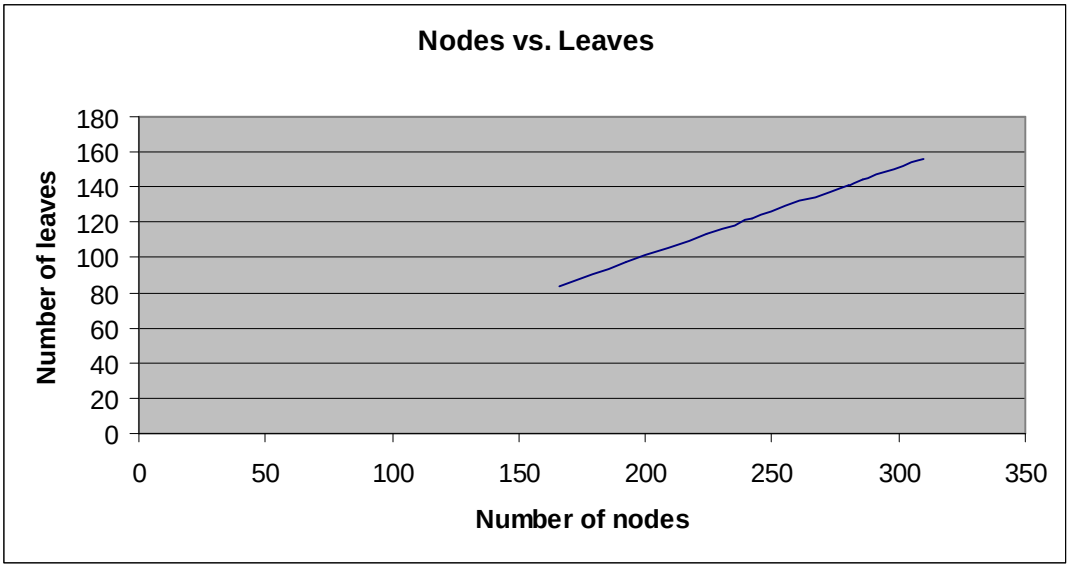
Στον πίνακα φαίνονται καθαρά οι μετρήσεις για κάθε περίπτωση ξεχωριστά. Φαίνονται ο αριθμός των συνολικών κόμβων και φύλλων του εκάστοτε δέντρου που κατασκευαζόταν, ο μέσος όρος της ακρίβειας των κανόνων (rules) του δέντρου και τέλος τα ποσοστά της λάθος ταξινόμησης για το training και το test set.

Για να είμαστε σίγουροι ότι προχωρούμε σωστά με τις μετρήσεις και τους όλους υπολογισμούς του προγράμματος αποφασίσαμε να ρίξουμε μια ματιά στους υπολογισμούς της Εντροπίας. Όπως αναφέρθηκε στο Κεφάλαιο 4, αφού η κλάση (τάξη-class) αποτελείται από 2 groups, η μέγιστη τιμή που πρέπει να πάρει η εντροπία είναι 1.



Σχήμα 5.2 Εντροπία vs. Πιθανότητα

Όπως βλέπουμε και στην πιο πάνω γραφική παράσταση η οποία μας δείχνει τις εντροπίες όλων των χαρακτηριστικών σε σχέση με τις πιθανότητες, πράγματι η θεωρία επαληθεύεται αφού η μέγιστη τιμή εντροπίας κάποιου χαρακτηριστικού είναι 1 ενώ παίρνει την μέγιστη τιμή της όταν η πιθανότητα είναι 0.5 (=50%).
Παράλληλα, ελέγξαμε αν πράγματι οι μετρήσεις που πήραμε πιο πάνω και φαίνονται στον Πίνακα 5.2, ήταν λογικές και όσο το δυνατόν πιο σωστές. Έτσι πήραμε τις δύο στήλες που δείχνουν τον συνολικό αριθμό των κόμβων και των φύλλων και δημιουργήσαμε την πιο κάτω γραφική παράσταση.



Σχήμα 5.3 Αριθμός φύλλων vs. Αριθμός κόμβων

Όπως χαρακτηριστικά φαίνεται, όσο αυξάνεται ο αριθμός των συνολικών κόμβων του δέντρου τόσο αυξάνεται και ο αριθμός των φύλλων – σύμφωνα πάντα με τις μετρήσεις μας – το οποίο είναι απόλυτα φυσιολογικό και σαφώς είναι το αποτέλεσμα που περιμέναμε. Οι δύο πιο πάνω γραφικές παραστάσεις αποτελούν μια πρώτη ένδειξη ότι πράγματι βαδίζουμε σε σωστό δρόμο!
Από όλες όμως τις μετρήσεις και τις παρατηρήσεις που κάναμε ως τώρα, είδαμε ότι τα δέντρα που κατασκευάζονται είναι μεγάλα σε μέγεθος και όχι με την καλύτερη αποδοτικότητα. Άρα σκεφτήκαμε, θα πρέπει να βρούμε την αιτία αυτού του προβλήματος και λύσεις για να το περιορίσουμε.

5.2. Το πρόβλημα του Overfitting

Η αιτία είναι το φαινόμενο του *overfitting*, το οποίο θα μπορούσε να αποδοθεί ως το υπερβολικό ταίριασμα με τα δεδομένα εκπαίδευσης. Μία υπόθεση h λέγεται πως *υπερταιριάζει (overfits)* με τα δεδομένα εκπαίδευσης αν υπάρχει μια άλλη υπόθεση h' τέτοια ώστε η h να έχει μικρότερο σφάλμα από την h' για τα δεδομένα εκπαίδευσης, αλλά η h' να έχει μικρότερο σφάλμα από την h για τη συνολική κατανομή των στιγμιότυπων. Η h' δηλαδή είναι καλύτερη προσέγγιση του πραγματικού μοντέλου από την h . Οι κύριοι λόγοι εμφάνισης του overfitting είναι οι εξής:

- Ο μεγάλος αριθμός παραμέτρων του μοντέλου, ή πιο γενικά η ικανότητα του αλγορίθμου μάθησης να κατασκευάζει ιδιαίτερα πολύπλοκα μοντέλα.
- Η μη κατάλληλη επιλογή των χαρακτηριστικών αναπαράστασης.
- Ο θόρυβος (noise) των δεδομένων εκπαίδευσης, δηλαδή τα τυχαία λάθη που είναι δυνατόν να περιέχονται στα δεδομένα. Αν και θα θέλαμε να είχαμε απολύτως αξιόπιστα δεδομένα τα οποία να χρησιμοποιούσαμε για την κατασκευή του ταξινομητή, στην πράξη αυτό δεν είναι πάντα εφικτό. Αξίζει να σημειωθεί πως η πιο κοινή πηγή θορύβου είναι ο "ανθρώπινος παράγοντας", π.χ. στην εισαγωγή των δεδομένων. Είναι επομένως λογικό πως ένας ταξινομητής προσαρμοσμένος απόλυτα ή πολύ κοντά στα (θορυβώδη) δεδομένα εκπαίδευσης, δεν αναμένεται να διατηρήσει την υψηλή του απόδοση σε νέα μη παρατηρηθέντα δεδομένα, ή όπως λέγεται δε θα έχει μεγάλη *ακρίβεια γενίκευσης (generalization accuracy)*.
- Οι τυχαίες κανονικότητες που είναι δυνατόν να εμφανιστούν, σε μικρά κυρίως σύνολα εκπαίδευσης, και οι οποίες μπορούν να οδηγήσουν στη δημιουργία ταξινομητών που έχουν κάνει λανθασμένες, στην πραγματικότητα, γενικεύσεις.

Το overfitting είναι μια σημαντική πρακτική δυσκολία για πολλούς αλγορίθμους μάθησης. Για τη μετρίασή του – επειδή ακόμα δεν έχει βρεθεί μέθοδος που θα το εκλείψει εντελώς - έχουν επινοηθεί μέθοδοι, τόσο προσαρμοσμένες σε καθέναν από αυτούς, όσο και ανεξάρτητες αλγορίθμου.

5.3. Τεχνική Pruning

Η πιο διαδεδομένη τεχνική για μείωση του overfitting είναι το pruning (κλάδεμα). Υπάρχουν 2 τεχνικές:

- Κλάδεμα κατά το κτίσιμο του δέντρου
- Ολοκληρωτικό κτίσιμο του δέντρου και μετά κλάδεμα (Post-Pruning)

Η τεχνική που χρησιμοποιείται συνήθως είναι η δεύτερη και είναι η τεχνική που χρησιμοποιήθηκε και στο δικό μας πρόγραμμα.

Να πως δουλεύει θεωρητικά και επιγραμματικά ο αλγόριθμος:

- Για κάθε εσωτερικό κόμβο:
    1. Έλεγχος για το αν η απομάκρυνσή του από το δένδρο, μαζί με το υποδένδρο του οποίου αποτελεί ρίζα, και η ανάθεση της συχνότερα εμφανιζόμενης κατηγορίας σε αυτό δε βλάπτει την ακρίβεια που μετράται σε κάποιο ανεξάρτητο σώμα επικύρωσης.
    2. Επιλογή του κόμβου που υπόσχεται την καλύτερη απόδοση, και αποκοπή του υποδένδρου του.
    3. Επιστροφή στο βήμα 1, όσο το δένδρο επιδέχεται βελτίωση, σύμφωνα με κάποιο δεδομένο κατώφλι.

Αφού εφαρμόσαμε στην πράξη τον αλγόριθμο, επαναλάβαμε τις μετρήσεις για να δούμε αν πράγματι η τεχνική φέρνει αποτελέσματα. Τα αποτελέσματα των μετρήσεων φαίνονται στον πίνακα 5.3.

RESULTS

Training Data (%)	Test Data (%)	Number of nodes	Number of leaves	Average accuracy of the rules	misclassification on training data (%)	misclassification on test data (%)
[illegible]	1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
90	10	254	[illegible]	[illegible]	5.72	[illegible]
[illegible]	15	[illegible]	[illegible]	[illegible]	5.52	20.2
80	20	230	113	63.6	5.24	25.03
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	4.44	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	40	152	[illegible]	[illegible]	[illegible]	[illegible]
50	50	146	74	75.3	5.73	31.12

Πίνακας 5.3 Αποτελέσματα μετά το Pruning

Πάμε τώρα να αναλύσουμε ενδεικτικά ένα από τα αποτελέσματα για να δούμε τα συμπεράσματα μπορούμε να βγάλουμε.

Ας πιάσουμε την περίπτωση όπου το Training Set είναι 85% του αρχικού dataset και το Test Set το 15%.

Βλέπουμε χαρακτηριστικά ότι ο αριθμός των συνολικών κόμβων του δέντρου αλλά και των φύλλων μειώθηκε. Πριν την εφαρμογή της τεχνικής του pruning το δέντρο περιείχε 280 κόμβους με 141 φύλλα, ενώ μετά την εφαρμογή της τεχνικής, οι κόμβοι του δέντρου μειώθηκαν σε 238 και τα φύλλα σε 120.

Η τεχνική δούλεψε!

Πιο κάτω φαίνονται τα αποτελέσματα (σωστές-λάθος προβλέψεις) και όπως και πριν, με κόκκινο χρώμα εμφανίζονται οι αριθμοί της λανθασμένης ταξινόμησης.

Στον πρώτο πίνακα φαίνονται τα αποτελέσματα για το Training Set και στον δεύτερο τα αποτελέσματα για το Test Set.

	Predicted		
Observed	**0**	**1**	
0	224	33	257
1	14	580	594
	238	613	851

Πίνακας 5.4 Αποτελέσματα Training Set για 85% με 15%

	Predicted		
Observed	**0**	**1**	
0	20	23	43
1	22	84	106
	42	107	149

Πίνακας 5.5 Αποτελέσματα Test Set για 85% με 15%

Ενδιαφέρον παρουσιάζει η μελέτη για τα στοιχεία των κόμβων του δέντρου που κτίστηκε. Για σκοπούς παρουσίασης πάλι τα στοιχεία εισήχθησαν στο Microsoft Excel και ένα μέρος τους παρατίθεται πιο κάτω. Βλέπουμε στην πρώτη στήλη τον

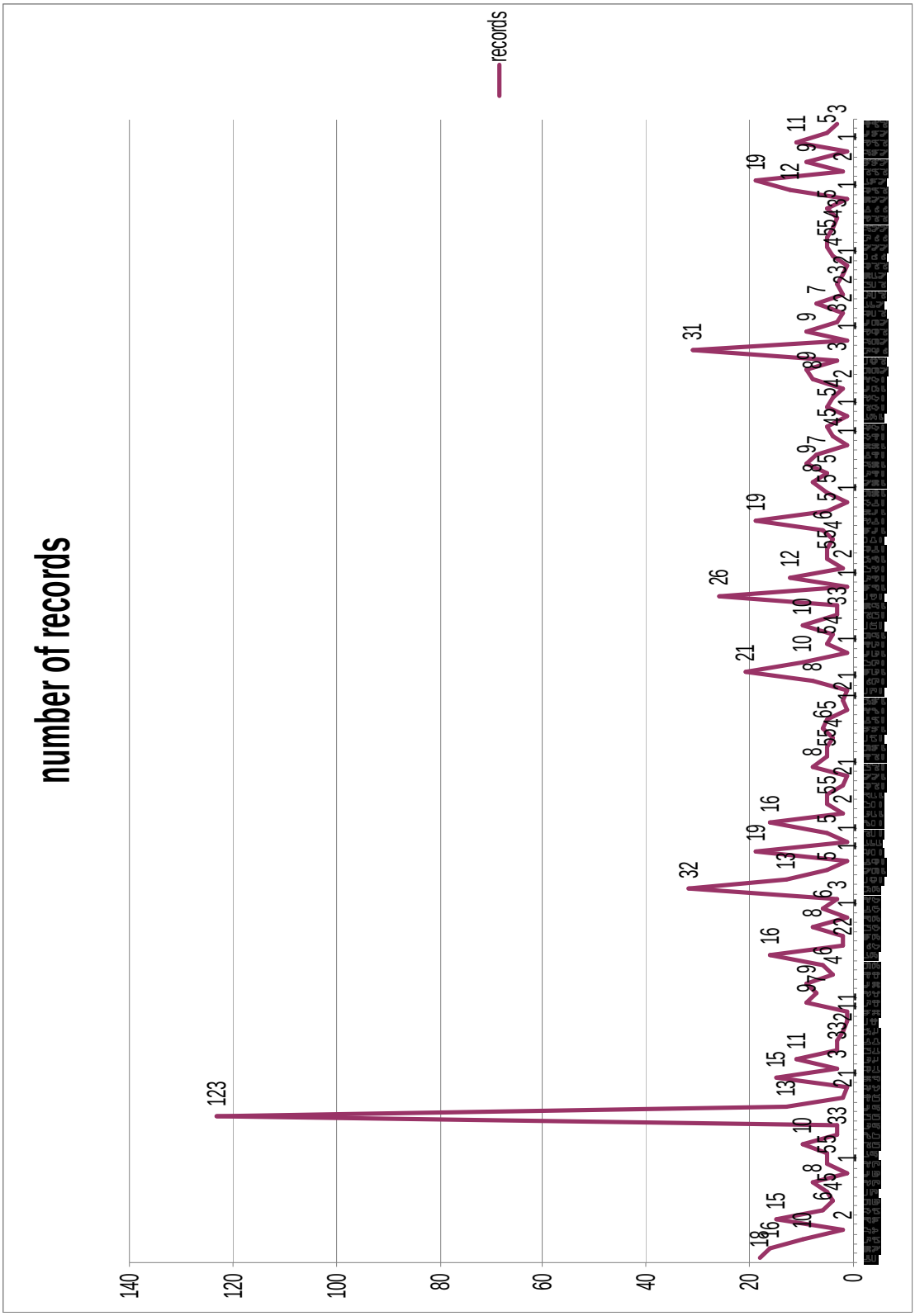
αριθμό του κόμβου, στην δεύτερη τον αριθμό των στοιχείων (records) που πέρασαν από εκείνον τον κόμβο και τέλος στην τρίτη στήλη το ποσοστό της λανθασμένης ταξινόμησης (misclassification) για τον κάθε κόμβο. Με κίτρινο χρώμα διακρίνονται οι κόμβοι που είναι φύλλα στο δέντρο.

Node Number	n	% miscalssification
17	18	0
28	16	12.5
34	10	0
35	2	0
36	15	6.67
39	6	0
40	4	0
41	5	0
46	8	0
47	1	0
48	5	0
51	5	20
52	10	20
53	3	0
54	3	33.33
55	123	0.81
57	13	7.69
58	2	0
60	1	0
68	15	0
70	3	0
74	11	0
75	3	0
77	3	33.33
79	2	0
81	1	0
83	1	0
84	9	0
86	7	0
87	9	11.11
88	4	25
90	6	0
91	16	0

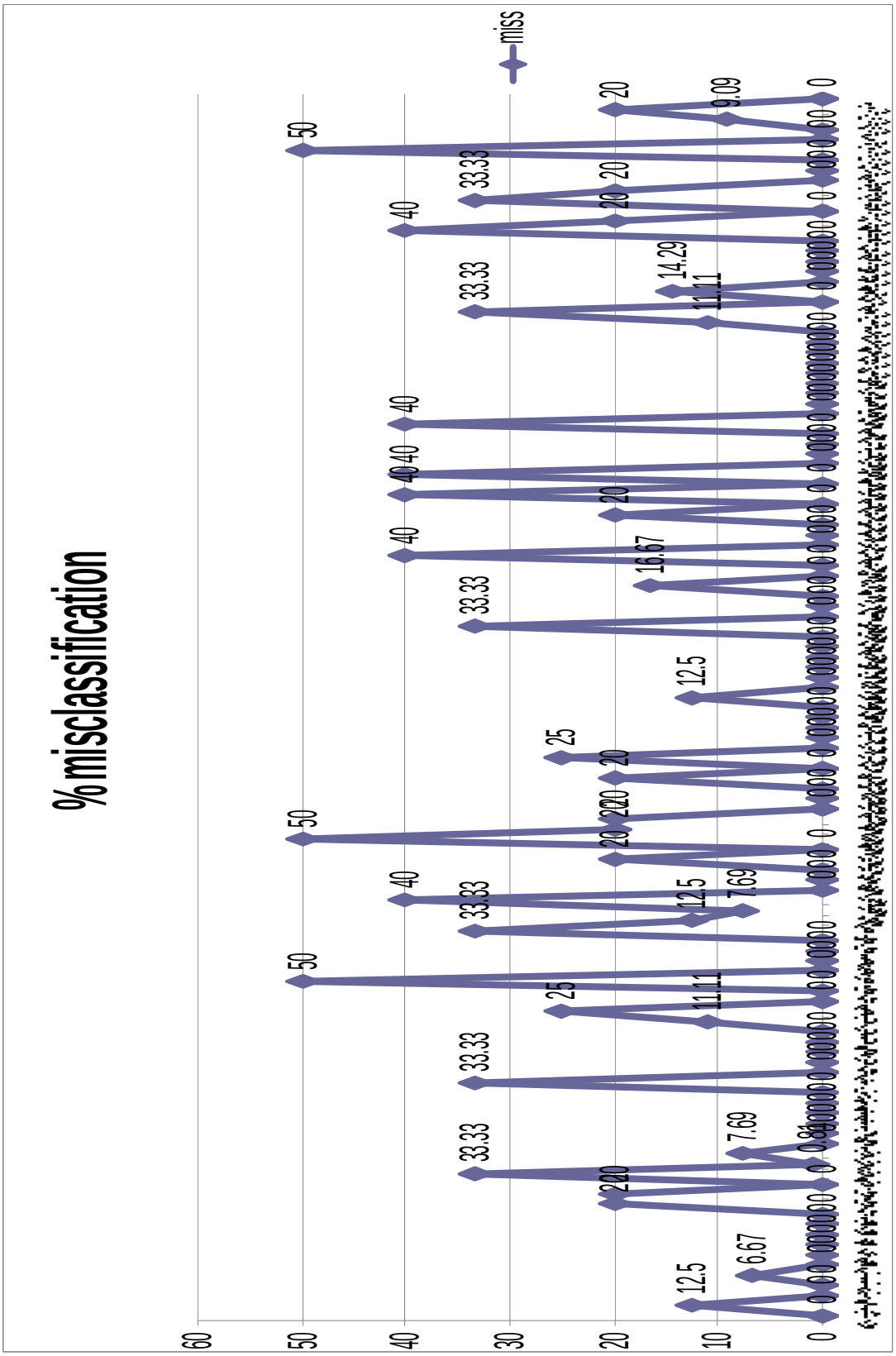
235	1	0
236	11	9.09
237	5	20
238	3	0
0	851	30.2
1	467	44.54
2	384	12.76
3	131	28.24
4	336	49.11
5	205	18.05
6	179	6.7
7	96	35.42
8	35	8.57
9	56	28.57
10	280	46.79
11	33	36.36
12	172	14.53
13	22	27.27
14	157	3.82
15	41	43.9
16	55	20
18	17	17.65
19	46	21.74
20	10	40
21	71	35.21
22	209	40.67
23	27	25.93
24	6	16.67
25	157	12.1
26	15	40
27	6	33.33
29	142	2.11
30	15	20
31	11	18.18
32	30	46.67

Σχήμα 5.4 Στοιχεία των κόμβων του Δέντρου

Βάση των πιο πάνω στοιχείων παράχθηκαν 4 γραφικές παραστάσεις (2 για τα φύλλα και 2 για τους κόμβους-μη φύλλα του δέντρου).



Σχήμα 5.5 Γραφική Παράσταση Φύλλων (Αριθμός Φύλλου vs. Αριθμός Στοιχείων)



Σχήμα 5.5 Γραφική Παράσταση Φύλλων (Αριθμός Φύλλου vs. Ποσοστό λανθασμένης Ταξινόμησης)

Οι δύο γραφικές παραστάσεις που εμφανίστηκαν στις δύο προηγούμενες σελίδες είναι άρρηκτα συνδεδεμένες μεταξύ τους αλλά για σκοπούς καλής και ευανάγνωστης παρουσίασης δεν κατέστη δυνατόν να συγχωνευτούν, εξ' ου και παρουσιάζονται ξεχωριστά.

Στην πρώτη γραφική βλέπουμε τον αριθμό του κόμβου-φύλλου σε σχέση με τον αριθμό των στοιχείων που περνούν απ' αυτό.

Παρατηρούμε ότι δεν περνούν πολλά δεδομένα από τα φύλλα αλλά κατανέμονται ομοιόμορφα σε αυτά ενώ υπάρχουν κάποιες εξαιρέσεις τις οποίες αξίζει να δούμε:

Το φύλλο με αριθμό 55 μπορεί να χαρακτηριστεί ως το καλύτερο φύλλο του δέντρου αφού έχουν περάσει απ΄αυτό 123 στοιχεία με ποσοστό λανθασμένης ταξινόμησης μόλις 0.81%! Επίσης πολύ καλό φύλλο μπορεί να χαρακτηριστεί το φύλλο 204 με 0% λανθασμένη ταξινόμηση.

Οι κανόνες (rules) από τον έλεγχο των οποίων πέρασε το φύλλο 55 όπως παρήχθησαν από το πρόγραμμα είναι οι εξής:

	Balance of current account
IF	**>=3**
	Has been employed by
AND	**current employer for >=4**
	Further running credits >=
AND	**2**
AND	**Type of apartment < 3**
AND	**Amount of credit < 9**
THEN	**Class (Output) = 1**

Ενώ οι κανόνες για το φύλλο 204 το οποίο προφανώς βρίσκεται σε μεγαλύτερο βάθος στο δέντρο και άρα έχει περάσει από περισσότερους κανόνες είναι:

	Balance of current account
IF	**>=3**
	Has been employed by
AND	**current employer for < 4**
AND	**Duration in Months >= 7**
AND	**Purpose of credit < 9**
	Further debtors/Guarantors
AND	**< 2**
	Payment of previous credit
AND	**>= 1**
AND	**Marital Status/Sex < 4**
	Payment of previous credit
AND	**< 3**
AND	**Occupation >= 3**
AND	**Purpose of credit < 5**
	Installment in % of
AND	**available income >= 3**

THEN Class (Output) = 1

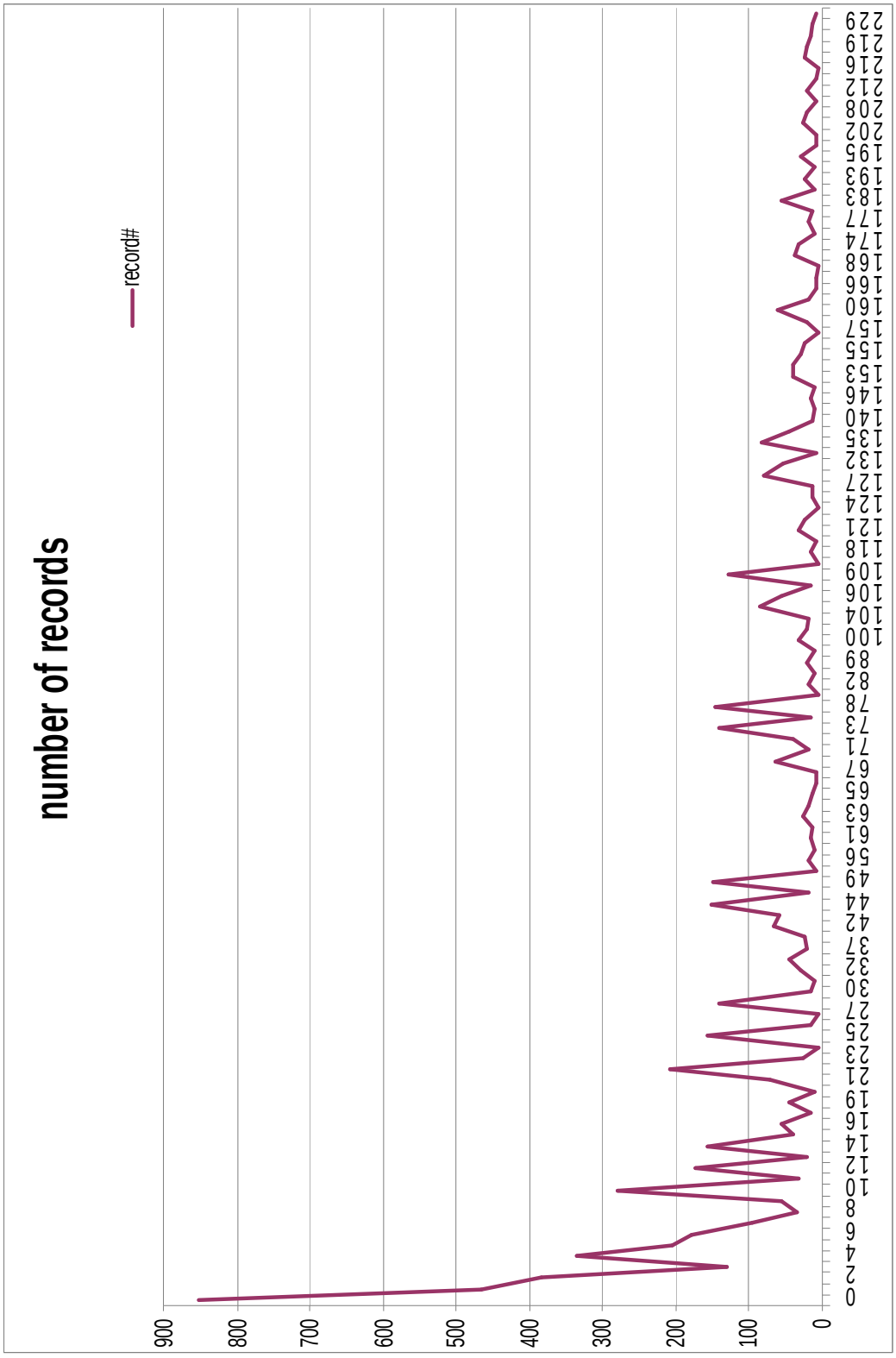
Τέλος, ένα ακόμα παράδειγμα είναι και το φύλλο με αριθμό 161 για το οποίο ελέγχθηκαν οι ακόλουθοι κανόνες:

IF Balance of current account >=3
AND Has been employed by current employer for < 4
AND Duration in Months >= 7
AND Purpose of credit < 3
AND Further debtors/Guarantors < 2
AND Payment of previous credit >= 1
AND Marital Status/Sex < 4
AND Payment of previous credit >= 3
AND Value of savings < 2
THEN Class (Output) = 1

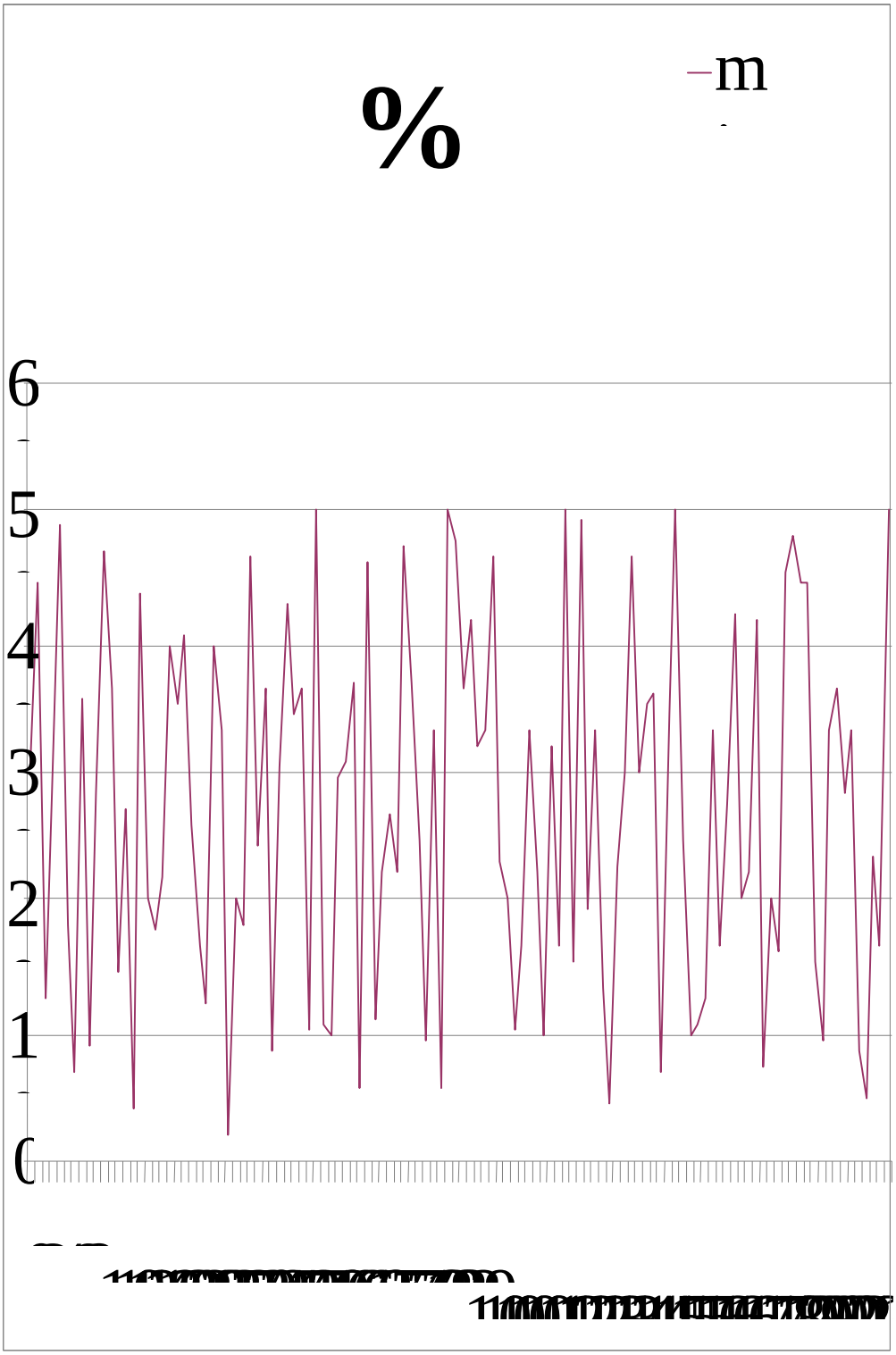
Επίσης, από τις δύο γραφικές παραστάσεις βλέπουμε ότι όσο προχωρά το κτίσιμο του δέντρου, τόσο περισσότερα φύλλα δημιουργούνται με ποσοστό λανθασμένης ταξινόμησης 0%. Αυτό δηλώνει σωστό και όμορφο κτίσιμο του δέντρου, γεγονός που δικαιολογεί και τις προηγούμενες αποφάσεις του προγράμματος για τη σειρά διάσπασης των χαρακτηριστικών.

Ακόμα και οι περιπτώσεις φύλλων που παρουσιάζονται με ψηλά ποσοστά λανθασμένης ταξινόμησης (misclassification) δικαιολογούνται από τον μικρό αριθμό στοιχείων που περνούν από αυτά.

Ακολουθούν τώρα οι 2 γραφικές παραστάσεις που αφορούν τους κόμβους μη φύλλα σε σχέση με τον αριθμό των στοιχείων που περνούν απ'αυτά.



Σχήμα 5.5 Γραφική Παράσταση Κόμβων μη-φύλλων (Αριθμός Κόμβου vs. Αριθμός Στοιχείων)



Σχήμα 5.6 Γραφική Παράσταση Κόμβων μη-φύλλων (Αριθμός Κόμβου vs. Ποσοστό λανθασμένης Ταξινόμησης)

Όπως και πριν, οι δύο γραφικές παραστάσεις που εμφανίστηκαν στις δύο προηγούμενες σελίδες είναι άρρηκτα συνδεδεμένες μεταξύ τους αλλά για σκοπούς καλής και ευανάγνωστης παρουσίασης δεν κατέστη δυνατόν να συγχωνευτούν, εξ' ου και παρουσιάζονται ξεχωριστά.

Στην πρώτη γραφική βλέπουμε τον αριθμό των κόμβων μη-φύλλων σε σχέση με τον αριθμό των στοιχείων που περνούν απ' αυτό. Εύκολα μπορεί να παρατηρήσει κανείς την ομοιομορφία που υπάρχει στην διάσπαση των δεδομένων εσωτερικά του δέντρου (αφού μιλάμε για κόμβους μη φύλλα). Και σε αυτή την περίπτωση υπάρχουν εξαιρέσεις:

Ο κόμβος με τον αριθμό 0 είναι προφανώς η ρίζα (root) του δέντρου γι'αυτό και έχει την πιο ψηλή παρουσία δεδομένων (851 δεδομένα όσο είναι και το σύνολο των δεδομένων του Training Set). Ακολούθως, μετά τη διάσπαση της ρίζας δημιουργούνται οι κόμβοι 1 και 2 από τους οποίους περνούν 467 και 384 δεδομένα αντίστοιχα, ενώ το ποσοστό λανθασμένης ταξινόμησης για τον κόμβο με αριθμό 1 φτάνει μέχρι το 44.54% (για τον 2 είναι 12.76%), ποσοστό όχι τόσο μεγάλο για τον αριθμό δεδομένων που περνούν απ' αυτόν.

Αξίζει να ρίξουμε και μια ματιά και σε κανόνες που ελέγχθηκαν για μερικούς κόμβους από του οποίους πέρασαν αρκετά δεδομένα. Για τον κόμβο με αριθμό 22:

IF	**Balance of current account < 3**
AND	**Most valuable available assets >= 2**
AND	**Payment of previous credit >=2**
AND	**Duration in Months >= 6**

Και για τον κόμβο με αριθμό 49:

IF	**Balance of current account >= 3**
AND	**Has been employed by current employer for < 4**
AND	**Duration in Months >= 7**
AND	**Purpose of credit < 9**
AND	**Further debtors/Guarantors < 2**

5.4. Γενικά Συμπεράσματα

Σε γενικές γραμμές το πρόγραμμα που κατασκευάσαμε παρήγαγε αρκετά καλά αποτελέσματα. Αυτό έγινε αφού η Θεωρία που μελετήθηκε πριν την εκπόνηση ήταν σωστή και ακολουθήθηκε τυφλά και πιστά (ιδιαίτερη βαρύτητα δόθηκε στον αλγόριθμο C4.5 ο οποίος αποτέλεσε και την ραχοκοκαλιά του προγράμματος). Το δέντρο μας κτίστηκε με σωστό τρόπο στην αρχή αλλά όχι τόσο αποδοτικά. Όμως, με τις σωστές τεχνικές που εφαρμόστηκαν, πετύχαμε καλύτερα αποτελέσματα τα οποία μπορούν άνετα να συγκριθούν με αυτά των μελετών της βιβλιογραφίας. Μετρήσεις ακρίβειας του δέντρου με μέσο όρο γύρω στο 70.2% πλησιάζουν τα αποτελέσματα μελετών που ας σημειωθεί ότι η καλύτερη πλησίαζε το 72%.
Άρα, γενικά η τεχνική των Decision/Classification Trees αποτελεί ένα γρήγορο και αρκετά αποδοτικό τρόπο για να προβλεφθεί μια συμπεριφορά.

5.5. Μελλοντική Εργασία

Ως μελλοντική εργασία μπορεί να προταθούν επιπρόσθετες έρευνες και μελέτες γύρω από την μείωση του προβλήματος του overfitting, εκτός από το pruning που είναι διαδεδομένο. Μερικές σκέψεις είναι:

- Να γίνεται έλεγχος βάθους του δέντρου. Να σταματά η διάσπαση ενός κόμβου αν φτάσει σε βάθος το οποίο θα καθοριστεί από τον προγραμματιστή. Η ρίζα θα έχει βάθος 1 ενώ το βάθος κάθε κόμβου είναι το βάθος του πατέρα του +1.
- Να γίνεται έλεγχος στο μέγεθος του κάθε κόμβου. Να σταματά η διάσπαση ενός κόμβου αν ο αριθμός των δεδομένων που περνούν από τον κόμβο είναι μεγαλύτερος από ένα ποσοστό του αρχικού αριθμού των δεδομένων, ο οποίος και πάλι θα καθοριστεί από τον προγραμματιστή.

Από κει και πέρα, το πρόγραμμα μπορεί να δοκιμαστεί σε real-time σύστημα όπως είναι σύστημα Δανειοδότησης Τραπεζών για να δείξει τις πραγματικές του δυνατότητες.

Βιβλιογραφία

[1] *Credit Scoring using neural and evolutionary techniques*, M.B. Yobas, J.N. Crook and P.Ross, Credit Research Centre, Department of Business Studies, University of Ediburgh

[2] *Analyzing Credit Risk Data: A comparison of Logistic Discrimination, Classification Tree Analysis and Feedforward Networks*, Gerhard Arminger, Daniel Enache, Thorsten Bonne, Department of Economics, Bergische Universitat, Germany

[3] *Comparing decision trees with logistic regression for credit risk analysis*, S S Satchidananda Research Director & Professor, *CBIT*, International Institute of Information Technology, Bangalore, India, Jay B.Simha ABIBA Systems
Bangalore, India

[4] *Microsoft Visual C# 2008 Step by Step*, author: John Sharp