Ανάλυση Δεδομένων

Υπεύθυνη καθηγήτρια: κα. Παναγιωτίδου Σοφία

4η εργασία

Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
Τμήμα Μηχανολόγων Μηχανικών
31/05/2024

Δημήτριος Νεντίδης, 6821

Περιεχόμενα

1 Εισαγωγή

Σε αυτήν την εργασία η οποία αποτελεί και την συνέχεια των δύο προηγούμενων ζητήται η ταξινόμηση των στοιχείων σε υψηλό ή χαμηλό. Στην αρχή εξετάζεται η μέθοδος Δένδρων Ταξινόμησης ενώ στην συνέχεια εξετάζονται ensemble μέθοδοι που βασίζονται σε αυτήν, Bagging, Random Forest και Boosting. Τέλος γίνεται μια σύγκριση για διάφορες παραμέτρους.

2 Δένδρα Ταξινόμησης

Για το χτίσιμο των δένδρων ταξινόμησης γίνεται χρήση του κριτηρίου Gini Index για την επιλογή της κάθε προβλεπτικής μεταβλητής.

$$\text{Gini Index} = 1 - \sum_{i=1}^{n} p_i^2$$

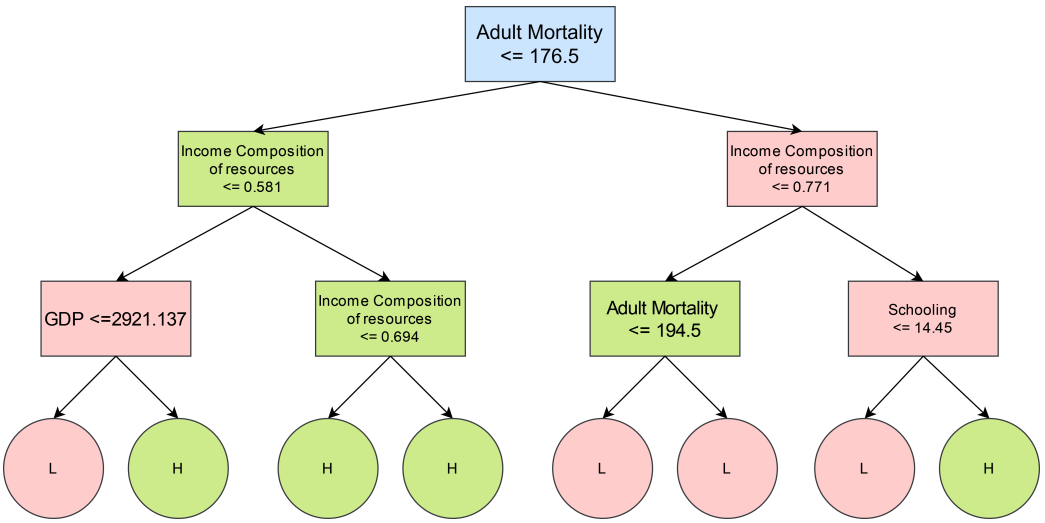
όπου p_i είναι η πιθανότητα μιας συγκεκριμένης κλάσσης i και n είναι ο συνολικός αριθμός κλάσσεων.

2.1 Depth 3

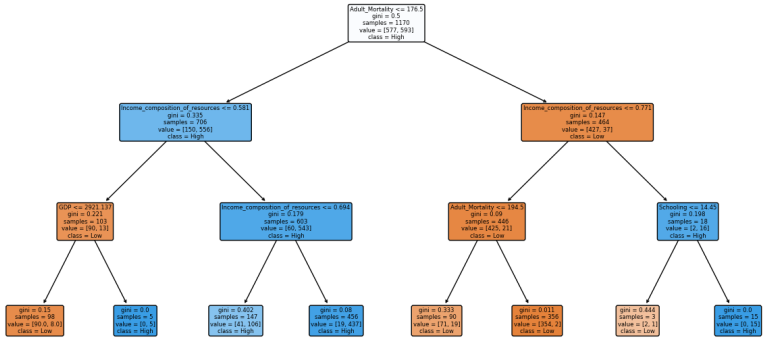
Για βάθος ίσο με 3 προκύπτει το παρακάτω δένδρο. Το root του δένδρου είναι η θνησιμότητα ανάμεσα σε ενήλικες. Στην συνέχεια στο αριστερό branch, εμφανίζονται τα κριτήρια "Income Composition of Resources" τα οποία εξετάζονται σε δύο διαφορετικές τιμές και το ΑΕΠ. Από την άλλη εμφανίζεται το "Income Composition of Resources", το "Adult Mortality" και το "Schooling".

Άξιο παρατήρησης είναι ότι γίνεται η χρήση των ίδιων προβλεπικών μεταβλητών παραπάνω από μια φορά σε διαφορετικές τιμές σύγκρισης. Αυτό ταιριάζει και με τα αποτελέσματα από την απλή γραμμική παλινδρόμηση της πρώτης εργασίας όπου αποδιδόταν 52.7% της μεταβλητότητας σε αυτήν την προβλεπτική μεταβλητή με πολύ μικρό p-value. Αντίστοιχα η απλή γραμμική παλινδρόμηση για την Ενήλικη Θνησιμότητα είχε R^2 0.505 με εξίσου μικρό p-value.

Το σφάλμα εκπαίδευσης, 0.08 και το σφάλμα δοκιμής 0.10, δείχνει μέτριο bias, ενώ η διακύμανση είναι επίσης μέτρια. Το μοντέλο ως έχει



Σχήμα 1: Δένδρο βάθους 3.



Σχήμα 2: Δένδρο βάθους 3.

	Predicted 0	Predicted 1	Support
Training Data			
Actual 0	517	60	577
Actual 1	30	563	593
Testing Data			
Actual 0	134	19	153
Actual 1	11	129	140

Πίνακας 1: Confusion Matrix για ένα Δένδρο βάθους 3.

	Precision	Recall	F1-score
Training Data			
0	0.95	0.90	0.92
1	0.90	0.95	0.93
Testing Data			
0	0.92	0.88	0.90
1	0.87	0.92	0.90

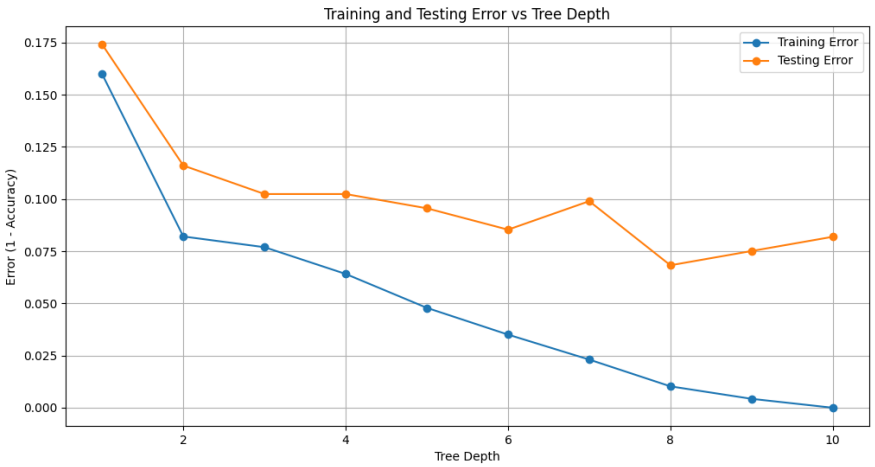
Πίνακας 2: Classification Report για ένα δένδρο βάθους 3.

παρουσιάζει μεγαλύτερο sensitivity το οποίο κυμαίνεται περίπου 0.04 πάνω από το specificity.

Ενδιαφέρον παρουσιάζει το γεγονός ότι κάποια splits οδηγούν σε φύλλα με κοινό output. Αυτό συμβαίνει καθώς αυτά τα splits μειώνουν το impurity των φύλλων, αυξάνοντας έτσι την εμπιστοσύνη στην κάθε πρόβλεψη.

2.2 Σύκγριση βάθους

Βλέποντας τις τιμές για το βάθος του δένδρου μπορεί να παρατηρηθεί ότι το testing error μειώνεται σε γενικές γραμμές με την αύξηση του βάθους. Έχοντας γράψει αυτό ωστόσο, η μείωση δεν είναι σημαντική από βάθος 3 εώς και 7, με ελάχιστο να εμφανίζεται για 8. Από εκεί το σφάλμα αυξάνεται. Το training error μειώνεται συνεχώς. Φαίνεται μάλιστα ότι μετά το βάθος 8 επιπέδων η διαφορά του testing και του training error αυξάνεται.



Σχήμα 3: Σύγκριση Test error σε διαφορετικά βάθη.

3 Μέθοδος Bagging

Κέντρικη ιδέα της μεθόδου Bagging ή Bootstrap Aggregating είναι η εκπαίδευση πολλαπλών μοντέλων βάσης (base estimators) για τυχαία subset των δεδομένων εκπαίδευσης. Στην προκειμένη περίπτωση γίνεται χρήση του μοντέλου ενός δένδρου με 8 επίπεδα βάθος καθώς αυτό είχε την βέλτιστη επίδοση προηγουμένως. Κάθε μοντέλο εκπαιδεύεται ξεχωριστά και στην συνέχεια μέσω μιας λειτουργίας "ψήφου πλειοψηφίας" βγαίνει το τελικό μοντέλο. Το πλήθος των εκτιμητών είναι ουσιαστικά το πλήθος των μοντέλων, ή δένδρων στην προκειμένη περίπτωση που θα χρησιμοποιηθούν. Στόχος είναι η μείωση του overfitting έχοντας κάθε μοντέλο να πιάνει διαφορετικό μέρος του data set.

Το Bagging μηδενίζει το train error ενώ το σφάλμα δοκιμής κυμαίνεται στο 0.07. Το bias είναι πολύ χαμηλό ενώ το μοντέλο γενικεύει καλά και το variance είναι χαμηλότερο σε σχέση με ένα δένδρο. Με μηδενικό σφάλμα τόσο το sensitivity όσο και το specificity είνιαι 1 για το training set ενώ στο testing set είνια 0.92 και 0.94 αντίστοιχα, ομοίως βελτιωμένα σε σχέση με 1 δένδρο.

	Predicted 0	Predicted 1	Support
Training Data			
Actual 0	576	1	577
Actual 1	0	593	593
Testing Data			
Actual 0	144	9	153
Actual 1	11	129	140

Πίνακας 3: Confusion Matrix for Bagging with Decision Trees

	Precision	Recall	F1-score
Training Data			
0	1.00	1.00	1.00
1	1.00	1.00	1.00
Testing Data			
0	0.93	0.94	0.94
1	0.93	0.92	0.93

Πίνακας 4: Classification Report for Bagging with Decision Trees

4 Μέθοδος Random Forest

Η μέθοδος Random Forest πάει την ιδέα του Bagging παραπέρα προσθέτοντας ένα επιπλέον επίπεδο τυχαιότητας. Ομοίως με πριν, κάθε δένδρο εκπαιδεύεται σε διαφορετικό μέρος των δεδομένων, ωστόσο εδώ σε κάθε split του δένδρου μόνο ένα τυχαίο μέρος των υποψηφίων χαρακτηριστικών λαμβάνεται υπόψη. Εδώ ο αριθμός των features που λαμβάνονται υπόψη σε κάθε split δίνεται ως $sqrtp$, όπου p ο συνολικός αριθμός προβλεπτικών μεταβλητών.

Το μοντέλο Random Forest πέτυχε τα καλύτερα αποτελέσματα, με σφάλμα εκπαίδευσης 0.00 και σφάλμα δοκιμής 0.05. Έχει πολύ χαμηλό bias, καθώς ταιριάζει τέλεια στα δεδομένα εκπαίδευσης, και χαμηλή διακύμανση (variance), όπως υποδεικνύεται από το χαμηλότερο σφάλμα δοκιμής μεταξύ όλων των μεθόδων. Η Random Forest μειώνει αποτελεσματικά τη διακύμανση μέσω του μέσου όρου πολλαπλών δέντρων, οδηγώντας σε ισχυρή γενίκευση και ανθεκτικότητα στα νέα δεδομένα. Η ευαισθησία στο σύνολο εκπαίδευσης είναι 1 και στο σύνολο δοκιμής 0.94, ενώ η εξειδίκευση είναι 1 και 0.95.

	Predicted 0	Predicted 1	Support
Training Data			
Actual 0	577	0	577
Actual 1	0	593	593
Testing Data			
Actual 0	146	7	153
Actual 1	8	132	140

Πίνακας 5: Confusion Matrix for Random Forest

	Precision	Recall	F1-score
Training Data			
0	1.00	1.00	1.00
1	1.00	1.00	1.00
Testing Data			
0	0.95	0.95	0.95
1	0.95	0.94	0.95

Πίνακας 6: Classification Report for Random Forest

5 Μέθοδος Boosting

Αντίθετα με τις δύο προηγούμενες μεθόδους όπου τα μοντέλα φτιάχνονται ανεξάρτητα μεταξύ τους, εδώ τα μοντέλα φτιάχνονται με σειριακό τρόπο, με το καθένα να προσπαθεί να βελτιώσει το προηγούμενο. Στην παρούσα εργασία γίνεται η χρήση του αλγορίθμου AdaBoost για την εύρεση του αποτελέσματος. Έχοντας το learning rate ίσο με 1 εξασφαλίζει ότι κάθε φορά μόνο το προηγούμενο δένδρο λαμβάνεται υπ΄όψη.

Το μοντέλο που αναπτύχθηκε με τον αλγόριθμο AdaBoost παρουσίασε σφάλμα εκπαίδευσης 0.03 και σφάλμα δοκιμής 0.08. Το bias είναι χαμηλό, αλλά ελαφρώς υψηλότερο από το και το Random Forest, υποδεικνύοντας ότι δεν ταιριάζει τέλεια στα δεδομένα εκπαίδευσης. Η διακύμανση είναι υψηλότερη από αυτή του Random Forest. Η μικρή αύξηση του σφάλματος δοκιμής σε σύγκριση με το Random Forest υποδεικνύει ότι είναι πιο ευαίσθητο στα δεδομένα εκπαίδευσης και κάνει overfit πιο εύκολα. Το sensitivity στο σύνολο εκπαίδευσης είναι 0.97 και στο σύνολο δοκιμής 0.91, ενώ το specificity είναι 0.97 και 0.94

	Predicted 0	Predicted 1	Support
Training Data			
Actual 0	557	20	577
Actual 1	15	578	593
Testing Data			
Actual 0	144	9	153
Actual 1	13	127	140

Πίνακας 7: Confusion Matrix for AdaBoost

	Precision	Recall	F1-score
Training Data			
0	0.97	0.97	0.97
1	0.97	0.97	0.97
Testing Data			
0	0.92	0.94	0.93
1	0.93	0.91	0.92

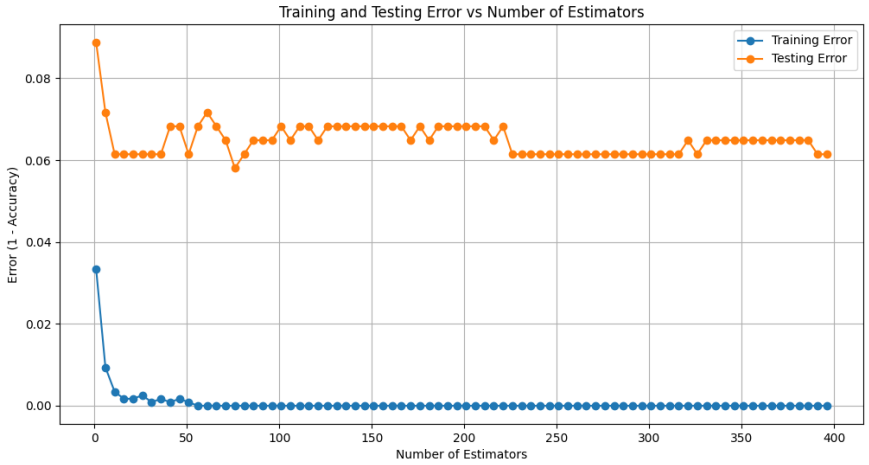
Πίνακας 8: Classification Report for AdaBoost

αντίστοιχα.

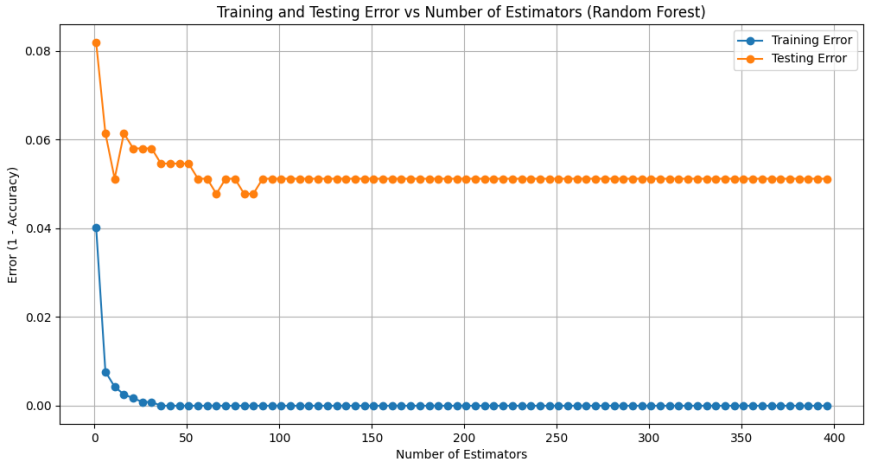
6 Σύγκριση n estimators

Σε όλες τις μεθόδους που δοκιμάστικαν n αλλαγή του πλήθους των estimators είχε παρόμοια συμπεριφορά. Για την μέθοδο Bagging το test error ταλαντώνεται γύρω από μια τιμή γύρω στο 0.06 με το ελάχιστο να παρουσιάζεται για 75 εκτιμητές. Το train error μηδενίζεται για n μεγαλύτερο του 50. Παρόμοια συμπεριφορά εμφανίζεται και για τα μοντέλα που προκύπτουν από την μέθοδο Random Forest, ωστόσο το test error παραμένει σταθερό για n>100. Ενώ το test error ακολουθεί παρόμοια ταλαντωτική συμπεριφορά, το train error μειώνεται διαρκώς για την μέθοδο Boosting, χωρίς όμως να μηδενιστεί ακόμα και για n=400.

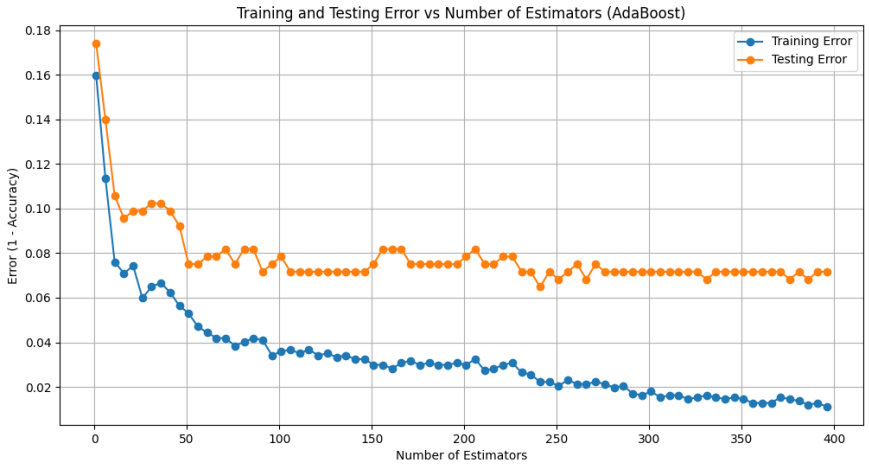
Αξιοσημείωτο είναι τα μοντέλα δεν είναι βέλτιστα για το n που δίνεται στην εκφώνηση. Το βέλτιστο test error για Bagging είναι ελαφρώς κάτω από το 0.06, ενώ για το Random Forest πέφτει κάτω από 0.05 και για Boosting κάτω από 0.07.



Σχήμα 4: Σύγκριση Test error για Bagging με διαφορετικά n.



Σχήμα 5: Σύγκριση Test error για Random Forest με διαφορετικά n.



Σχήμα 6: Σύγκριση Test error για Boosting με διαφορετικά n.

7 Σύγκριση Μεθόδων

Στην αξιολόγηση της απόδοσης των διαφόρων μοντέλων είναι σκόπιμο να συκγριθεί το bias και το variance των διαφόρων μεθόδων.

Το μοντέλο που προκύπτει από την εφαρμογή της μεθόδου των Δένδρων Ταξινόμησης train error 0.08 και test error 0.10, τα μεγαλύτερα από όλες τις μεθόδους που δοκιμάστηκαν, εμφανίζοντας λοιπόν το μεγαλύτερο bias. Το train error δεν είναι πολύ μεγαλύτερο κάτι που δείχνει ότι το variance είναι σχετικά μικρό.

Τόσο η μέθοδος Bagging με Decision Trees όσο και η Random Forest, μηδένισαν το σφάλμα στα δεδομένα εκπαίδευσης με το σφάλμα δοκιμής να βρίσκεται στο 0.07 και 0.05 αντίστοιχα. Το bias είναι ελάχιστο και για τις δύο. Η Random Forest γενικεύεται λίγο καλύτερα αποφεύγοντας το overfitting καλύτερα και έχοντας το ελάχιστο variance από ολες τις μεθόδους που εξετάστηκαν. Το Boosting παρουσίασε σφάλμα εκπαίδευσης 0.03 και σφάλμα δοκιμής 0.08.

Είναι εμφανές ότι την μεγαλύτερη ακρίβεια την εμφανίζει η μέθοδος Random Forest η οποία, όπως είναι λογικό είναι μια από τις ensemble μεθόδους.

Method	**Training Error**	**Testing Error**
Decision Tree	0.08	0.10
Bagging with Decision Trees	0.00	0.07
Random Forest	0.00	0.05
AdaBoost	0.03	0.08

Πίνακας 9: Summary of Training and Testing Errors for Different Methods

Τέλος συκγρίνοντας αυτές τις τιμές με τα αποτελέσματα της προηγούμενης εργασίας γίνεται εμφανές ότι οι μέθοδοι αυτοί είναι σαφώς πιο αποτελεσματικές για την κατάταξη τιμών. Ακόμα και στο βέλτιστο configuration, με threshold = 0.4, η LDA είχε training error ίσο με 0.1145 και testing error 0.088, συγκρίσιμο με τα αποτελέσματα ενός μόνο δένδρου.