1η Εργασία

Επιβλεπόμενη Μάθηση - Ταξινόμηση

Μηχανική Μάθηση και Επεξεργασία Φυσικής Γλώσσας

Χριστακάκης Παναγιώτης

A.M.: aid23004

Πρόγραμμα Μεταπτυχιακών Σπουδών «Τεχνητή Νοημοσύνη & Αναλυτική Δεδομένων» , Πανεπιστήμιο Μακεδονίας

Ημερομηνία: 11/11/2022

Επιβλεπόμενη Μάθηση - Ταξινόμηση

Πίνακας Περιεχομένων

Γραφήματα

1 Εισαγωγή

Σκοπός του προβλήματος του οποίου καλούμαστε να αντιμετωπίσουμε είναι η δημιουργία του καλύτερου δυνατού μοντέλου ταξινόμησης στα δεδομένα που μας δόθηκαν.

Το συγκεκριμένο dataset αφορά ελληνικές εταιρείες όπου για κάθε μια από αυτές μας δίνονται ως δεδομένα κάποιοι χρηματοπιστωτικοί δείκτες απόδοσης, μερικοί δυϊκοί δείκτες δραστηριοτήτων, το έτος στο οποίο αναφέρονται αυτοί καθώς και αν η εκάστοτε εταιρεία έχει κηρύξει χρεωκοπία ή όχι.

Καλούμαστε, λοιπόν, να αναπτύξουμε ένα μοντέλο το οποίο θα μπορεί να ξεχωρίζει αν κάποια εταιρεία θα πτωχεύσει ή όχι, ακολουθώντας όμως του παρακάτω δύο περιορισμούς:

1. Οι πτωχευμένες εταιρείες θα πρέπει να βρίσκονται από το μοντέλο μας με ποσοστό επιτυχίας **τουλάχιστον 62%.**
2. Οι **μη** πτωχευμένες εταιρείες θα πρέπει να βρίσκονται από το μοντέλο μας με ποσοστό επιτυχίας **τουλάχιστον 70%.**

Όπως θα δούμε παρακάτω, τα δεδομένα που μας έχουν δοθεί δεν τηρούν κάποια ισορροπία μεταξύ των πτωχευμένων και μη-πτωχευμένων εταιρειών κάτι το οποίο είναι λογικό να μας δημιουργήσει προβλήματα και δυσκολίες κατά την αναζήτηση, δημιουργία και σύγκριση των μοντέλων. Επομένως, η προσπάθεια μας να παράξουμε ένα μοντέλο που ικανοποιεί και τους δύο περιορισμούς δεν θα μείνει εκεί, αλλά θα επικεντρωθεί και στην βελτίωση του δοσμένου αρχείου δεδομένων, ώστε να έχουμε καλύτερα αποτελέσματα.

Τα μοντέλα τα οποία θα εκπαιδεύσουμε και θα αξιολογήσουμε είναι τα εξής:

Logistic Regression	Decision Trees	Naïve Bayes
k-Nearest Neighbors	Neural Networks	Support Vector Machines
Linear Discriminant Analysis		

Για κάθε ένα από αυτά θα κρατήσουμε μερικές τιμές μετρικών ώστε να μπορούμε να τα συγκρίνουμε μεταξύ τους και να καταλήξουμε στα τελικά μας συμπεράσματα.

2 Μέθοδοι

Για την υλοποίηση των μεθόδων μας χρησιμοποιήσαμε την γλώσσα python και τις κατάλληλες βιβλιοθήκες. Ενδεικτικά αναφέρονται οι pandas, numpy, sklearn, keras και άλλες.

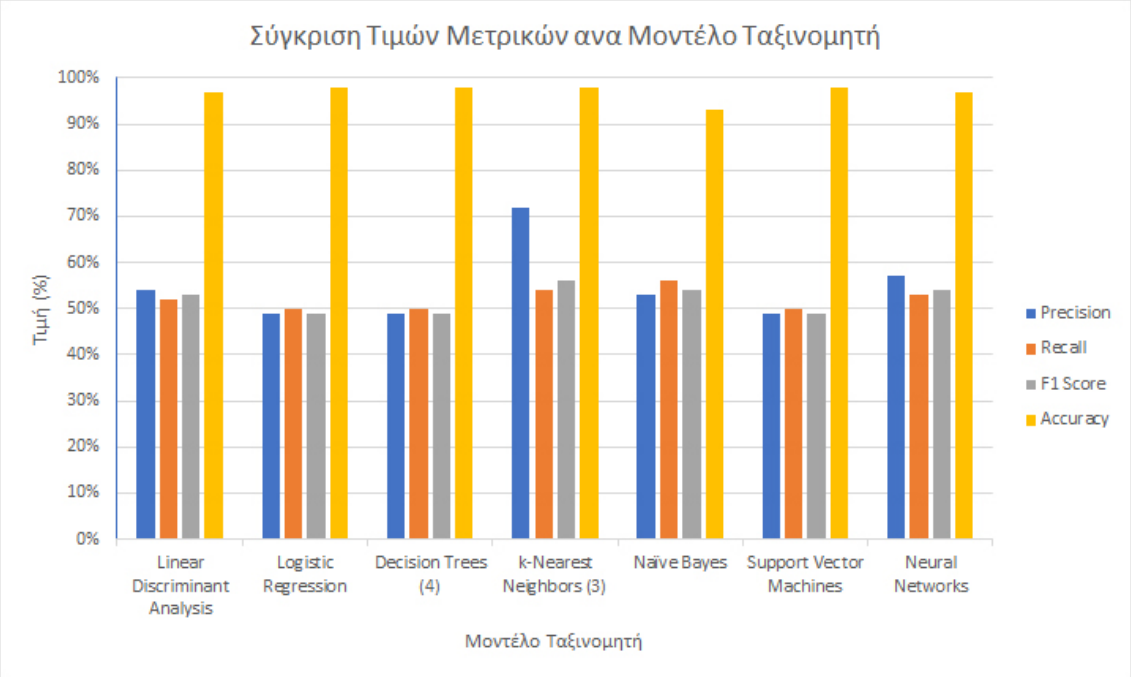
Αρχικά, εισάγονται τα δεδομένα από το .xlsx αρχείο που περιέχει το dataset. Η τελευταία στήλη περιέχει την κατηγοριοποίηση των εταιρειών, επομένως αποθηκεύεται ξεχωριστά ως outputData. Όλες οι υπόλοιπες στήλες, πλην του έτους, αποθηκεύονται ως inputData τα οποία θα αποτελέσουν και τα χαρακτηριστικά του κάθε παραδείγματος – επιχείρησης.

Το μέγεθος του dataset που είναι περί τις 10.000 εγγραφές, μας προσφέρει τη δυνατότητα να χωρίσουμε με άνεση τα δεδομένα μας σε 75% train και 25% test, επιτρέποντας στο testing κομμάτι του κάθε μοντέλου να έχει ένα αρκετά μεγάλο δείγμα για δοκιμή. Η stratify είναι μια αρκετά βοηθητική παράμετρος ,ειδικά σε ένα unbalanced dataset, κατά τη διάρκεια του διαχωρισμού των δεδομένων, καθώς μας προσφέρει ισορροπία στις τιμές κατηγοριοποίησης σε ποσοστιαία κλίμακα που εμείς ορίσαμε νωρίτερα. Ακόμα, η παράμετρος shuffle απαιτείται ώστε η ανάθεση των δεδομένων να γίνει με τυχαίο τρόπο και να μην έχουμε μόνο δεδομένα μίας κλάσης.

Στη συνέχεια, ξεκινάει η εκπαίδευση των μοντέλων ταξινόμησης και η καταγραφή των αποτελεσμάτων. Για κάθε αξιολογούμενο μοντέλο ξεχωριστά μπορούμε να ‘πειράξουμε’ κάποιες παραμέτρους, τα λεγόμενα hyper-parameters. Συγκεκριμένα Logistic Regression και Linear Discriminant Analysis χρειάστηκε μόνο να επιλέξουμε solver ο οποίος έπειτα από δοκιμές αποδείχτηκε ότι είναι ο πλέον πιο αποδοτικός, ενώ για το Naïve Bayes μοντέλο δεν χρειάστηκε καμία παραμετροποίηση. Όσον αφορά το k-Nearest Neighbors το k = 3, φάνηκε να μας προσφέρει τα βέλτιστα αποτελέσματα μετρικών για το συγκεκριμένο dataset. Στα Decision Trees αρχικά δοκιμάστηκαν αυθαίρετα κάποιοι αριθμοί, αλλά τελικά επιλέχθηκε μέγιστο βάθος δέντρου το 4. Με επαναληπτική διαδικασία και δοκιμάζοντας παραμέτρους για C, gamma και kernel, στο Support Vector Machines μοντέλο εφαρμόσαμε 0.1, 1, “rbf” αντίστοιχα για κάθε παράμετρο.

Φτάνοντας στο Νευρωνικό Δίκτυο, για αρχή δοκιμάστηκαν απλές αρχιτεκτονικές με σχετικά λίγους νευρώνες και χωρίς πολλά κρυφά επίπεδα. Συνάρτηση ενεργοποίησης του εκάστοτε επιπέδου επιλέχθηκε η ‘Relu’ εφόσον πρόσφερε καλύτερα αποτελέσματα, ενώ η ‘Softmax’ επιλέχθηκε στο τελευταίο επίπεδο για την κατηγοριοποίηση. Εδώ θα πρέπει να αναφερθεί ότι οι ‘Sigmoid’ και ‘Binary Crossentropy loss’ συναρτήσεις σε συνδυασμό μαζί έδωσαν χειρότερα αποτελέσματα, ενώ πρόκειται για συναρτήσεις που χρησιμοποιούνται για δυαδική κατηγοριοποίηση. Καλύτερα αποτελέσματα δόθηκαν από τον ‘adam’ optimizer και το batch_size επιλέχθηκε έπειτα από δοκιμή διαφορετικών μεγεθών. Εδώ υπήρχε ένα σχετικά καλό – μεγάλο μέγεθος test δεδομένων, οπότε και επιλέχθηκε το 512. Γνωρίζοντας ότι το dataset περιέχει δυσανάλογες τιμές για κάθε κλάση ήταν αναμενόμενο ακόμα και μεγάλα νευρωνικά δίκτυα όπως αυτό του τελικού κώδικά, να δώσουν κακά αποτελέσματα, χωρίς να καταφέρουμε να υπερπροσαρμόσουμε ούτε λίγο τα δεδομένα μας.

Γράφημα 2.1: Ραβδόγραμμα σύγκρισης μετρικών ανά ταξινομητή

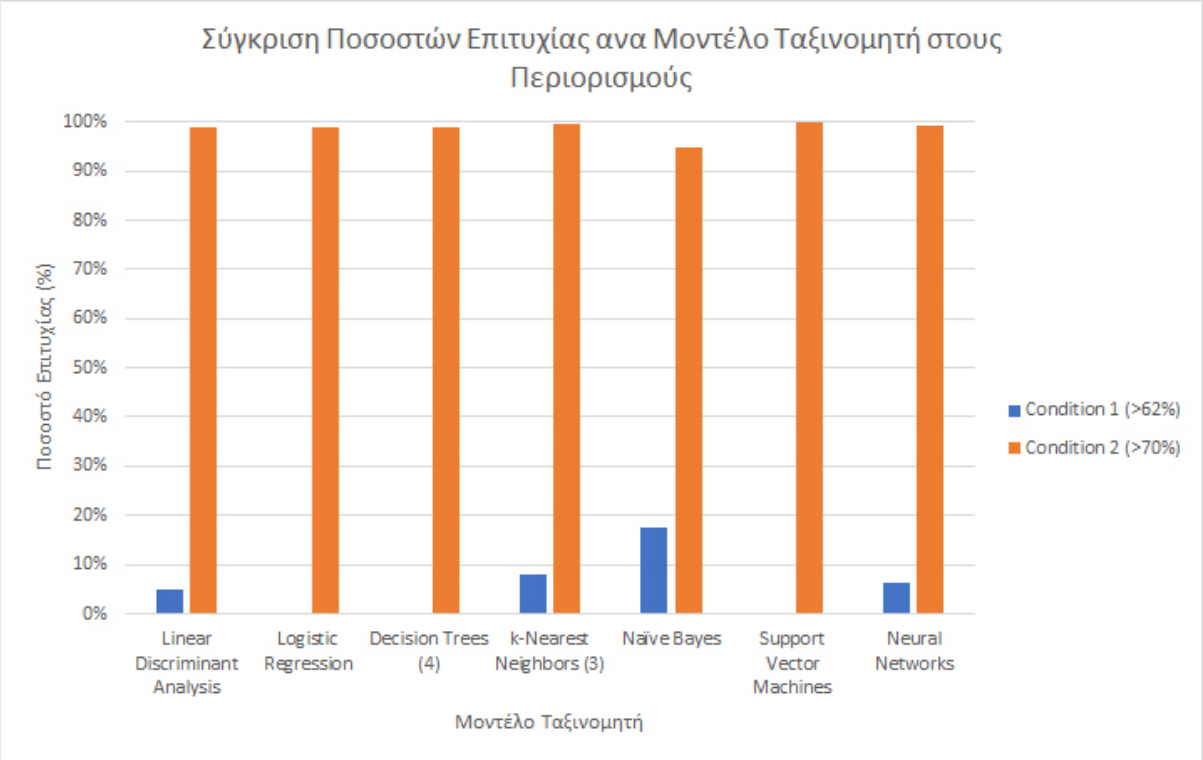


***Σχόλια*:** Αναλύοντας το παραπάνω ραβδόγραμμα, μπορούμε να διακρίνουμε ξεκάθαρα ότι τα μοντέλα που αναπτύξαμε έχουν χαμηλές τιμές στους υπολογισμένους δείκτες. Πιο συγκεκριμένα, Precision και Recall είναι πάντα κοντά στο 50% εφόσον οι ταξινομητές μπορούν να βρουν μόνο την μία κλάση που βρίσκεται σε αφθονία στο dataset, αυτή των μη-χρεοκοπημένων εταιρειών. Αναμενόμενα, το F1 Score, που είναι ο αρμονικός μέσος όρος των Precision και Recall, βρίσκεται και αυτό κοντά στο 50%. Ο μόνος ταξινομητής που εμφανίζει ελάχιστα βελτιωμένη απόδοση στο Precision είναι ο k-NN. Τέλος, το Accuracy είναι σε όλα τα μοντέλα υψηλό, λόγω της σωστής κατηγοριοποίησης της κλάσης που έχει τα περισσότερα παραδείγματα.

Όσον αφορά τους δύο περιορισμούς που πρέπει να καλύπτει το τελικό μας μοντέλο, δημιουργούμε δύο μεταβλητές οι οποίες υπολογίζονται ως εξής:

- Μεταβλητή για Περιορισμό 1: $\frac{(True\ Negative)_{test}}{(Πλήθος\ Χρεοκοπημένων)_{test}}$
- Μεταβλητή για Περιορισμό 2: $\frac{(True\ Positive)_{test}}{(Πλήθος\ Μη-Χρεοκοπημένων)_{test}}$

Γράφημα 2.2: Ραβδόγραμμα σύγκρισης επιτυχίας ως προς τους περιορισμούς

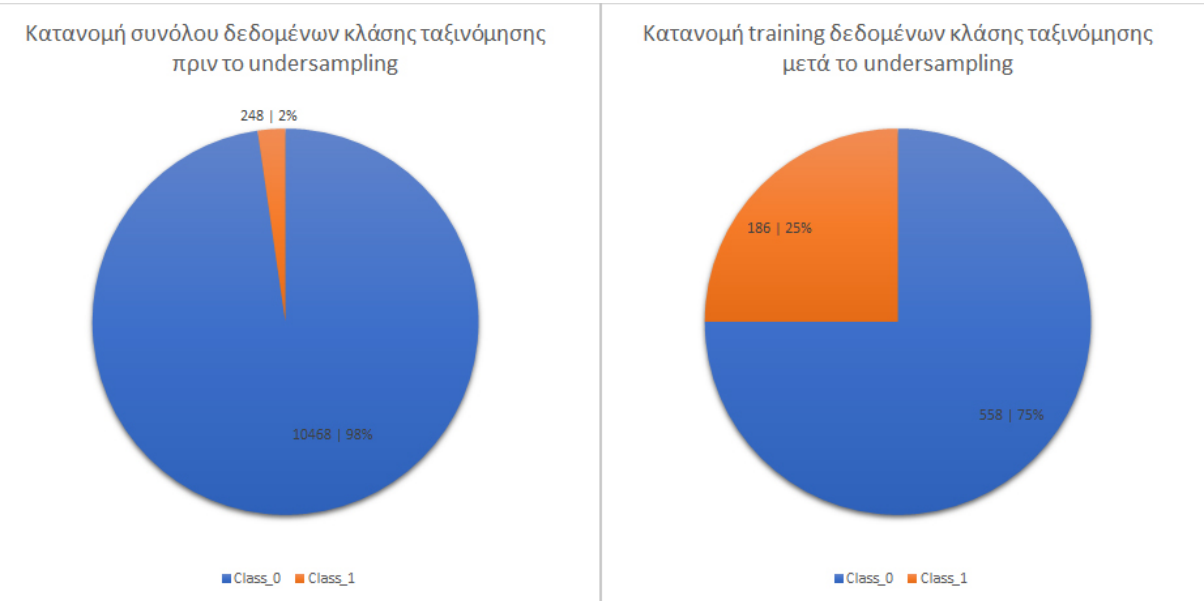


***Σχόλια*:** Στο *Γράφημα 2.2*, βλέπουμε ότι κανένα μοντέλο δεν καλύπτει και τους δύο περιορισμούς του προβλήματος. Αναλυτικότερα, Ο δεύτερος περιορισμός καλύπτεται πάντα απ' όλα τα μοντέλα ταξινόμησης και μάλιστα με άνεση, ενώ ο πρώτος περιορισμός από κανένα, αφού σε όλα το ποσοστό επιτυχίας είναι χαμηλότερο από 20%. Το γεγονός ότι ο Naïve Bayes έχει το μεγαλύτερο ποσοστό στον πρώτο περιορισμό, οφείλεται στην τύχη καθώς έτρεχε η συγκεκριμένη εκτέλεση.

Στο ερώτημα ΙΙ του προβλήματος ζητείται να επαναληφθεί το πείραμα και η καταγραφή των αποτελεσμάτων, αλλά αυτή τη φορά αφού πρώτα έχουμε επεξεργαστεί το training set, έτσι ώστε η αναλογία των κλάσεων που πρέπει να κατηγοριοποιήσουμε να είναι 3 υγιείς προς 1 χρεωκοπημένη. Κάτι τέτοιο θεωρητικά θα προσφέρει τη δυνατότητα στα μοντέλα μας να 'δουν' πιο εύκολα και την κλάση που βρίσκεται σε μειονότητα.

Υπάρχουν διάφοροι τρόποι να γίνει επαναδειγματοληψία ενός dataset. Οι πιο διαδεδομένες τεχνικές είναι το oversampling και το undersampling. Στην περίπτωση μας επιλέχθηκε η υποδειγματοληψία ή αλλιώς undersampling, διότι είναι πιο εύκολο να διαγράψουμε έναν όγκο εγγραφών της πλειοψηφικής κλάσης, παρά να παράξουμε νέα δεδομένα για την άλλη κλάση. Χρησιμοποιώντας την κατάλληλη βιβλιοθήκη 'imblearn.under_samplng', στο training set υπάρχουν πλέον τρείς επιχειρήσεις που δεν πτώχευσαν για κάθε μία χρεωκοπημένη.

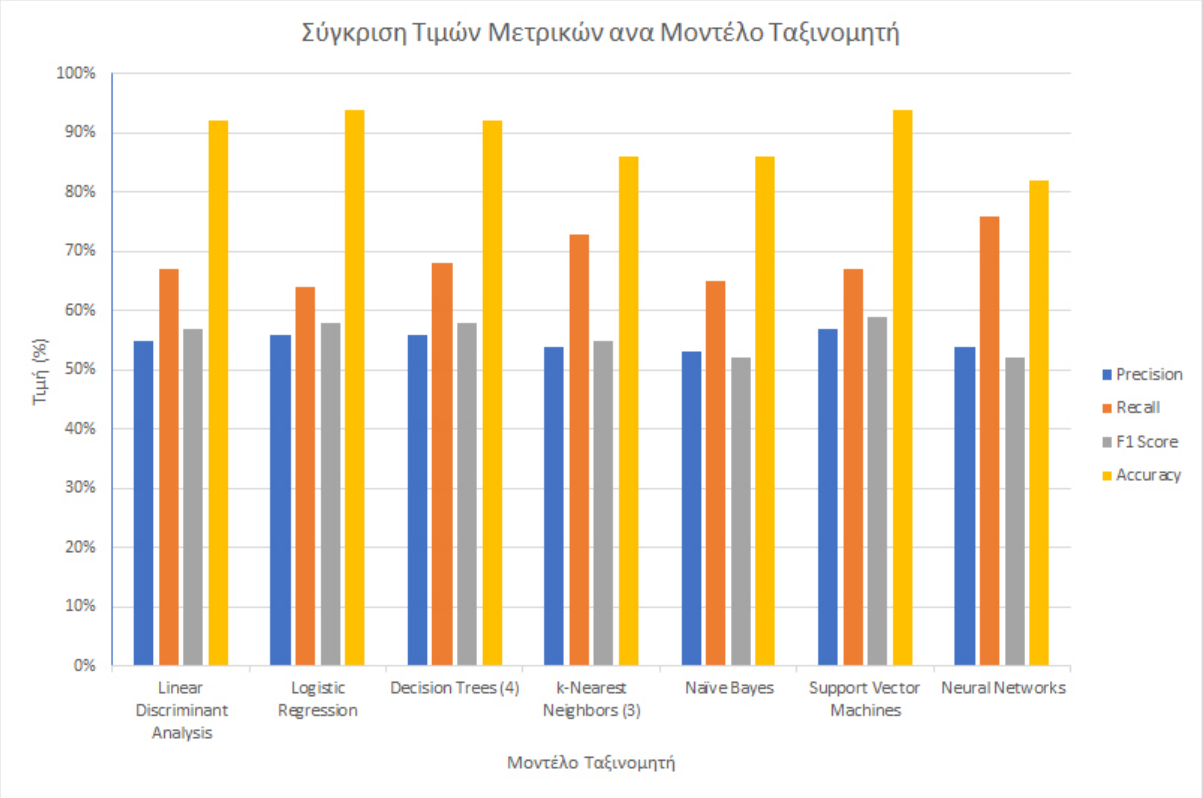
Γράφημα 2.3: Διαγράμματα πίτας κατανομών πριν και μετά το undersampling



***Σχόλια*:** Το *Γράφημα 2.3*, περιέχει δύο διαγράμματα πίτας τα οποία δείχνουν ξεκάθαρα τη δυσαναλογία που υπήρχε στο dataset μεταξύ των δύο κλάσεων. Το 98% των εγγραφών έχουν να κάνουν με τις μη – χρεοκοπημένες επιχειρήσεις, ενώ οι πτωχευμένες αντιστοιχούσαν μόλις στο 2%. Η αναλογία ήταν 1:43. Μετά το undersampling, οι εγγραφές της κλάσης με τις μη – πτωχευμένες εταιρείες μειώθηκαν στις 558 από τις 10468, ενώ η μειοψηφική κλάση εκπροσωπεί πλέον το 25% του συνόλου των παραδειγμάτων εκπαίδευσης.

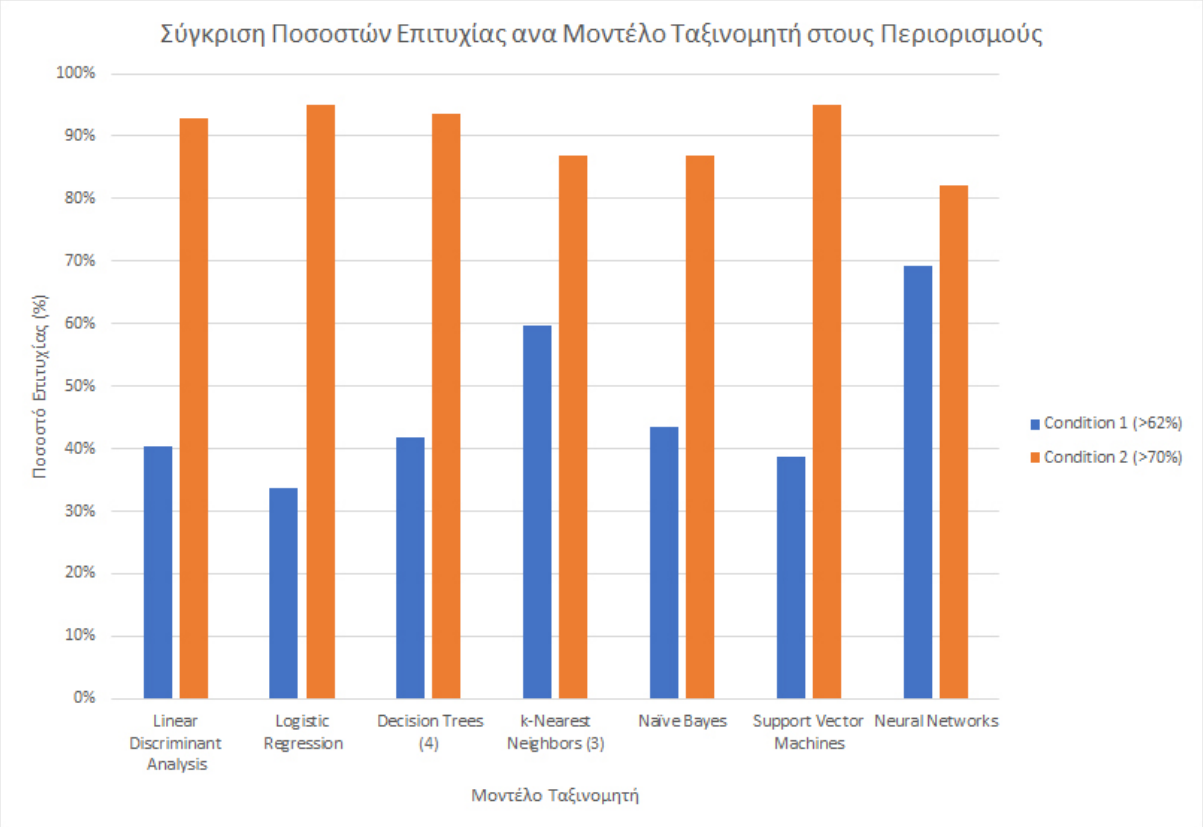
Συνεχίζοντας, εκπαιδεύουμε εκ νέου τα μοντέλα ταξινόμησης και καταγράφουμε τα νέα αποτελέσματα. Οι παράμετροι των μοντέλων παραμένουν ίδιοι, καθώς έπειτα από δοκιμές δίνουν τα βέλτιστα αποτελέσματα. Όσον αφορά το νευρωνικό δίκτυο, ψάχνουμε την κατάλληλη αρχιτεκτονική που θα μας προσφέρει όσο το δυνατόν καλύτερες τιμές, τόσο στις μετρικές όσο και στην ανάγκη μας να καλύψουμε τους δύο περιορισμούς. Πέραν της αρχιτεκτονικής που αλλάζει από την προηγούμενη, αλλαγή υπάρχει επίσης στο μέγεθος των batches, αφού πλέον μιλάμε για ένα πολύ μικρότερο training set και το νούμερο 512 δεν θα επέτρεπε στους νευρώνες να κάνουν τις κατάλληλες αλλαγές στα βάρη κατά την εκπαίδευση. Ακόμα, αυξάνοντας τις εποχές εκπαίδευσης, προσθέτουμε και ‘Early Stopping’ ώστε να προσπαθήσουμε να σταματήσουμε την εκπαίδευση νωρίτερα αν δεν αυξάνεται ως ένα βαθμό η ακρίβεια του μοντέλου. Ως συναρτήσεις ενεργοποίησης κρατήσαμε την ‘Relu’ και στο τέλος την ‘Softmax’. Έπειτα από τις αλλαγές αναμένουμε να δούμε ελαφρώς, έως και αρκετά βελτιωμένα αποτελέσματα.

Γράφημα 2.4: Ραβδόγραμμα σύγκρισης μετρικών ανά ταξινομητή σε νέο training set



***Σχόλια*:** Στο παραπάνω γράφημα, διακρίνουμε τα βελτιωμένα αποτελέσματα που μας προσέφερε η αλλαγή της αναλογίας των κλάσεων στα δεδομένα εκπαίδευσης. Είναι ευκρινές ότι το Recall αυξήθηκε σε σύγκριση με πριν, ενώ το Precision κυμαίνεται στις ίδιες τιμές κρατώντας χαμηλά και το F1 Score. Μπορεί να φαίνεται ότι το Accuracy έχει μειωθεί σε σχέση με τα προηγούμενα αποτελέσματα, αλλά στην πραγματικότητα αυτό είναι κάτι που θέλουμε καθώς φαίνεται ότι τα μοντέλα μας αρχίζουν και κατηγοριοποιούν δεδομένα της μειοψηφικής κλάσης.

Γράφημα 2.5: Ραβδόγραμμα σύγκρισης επιτυχίας των περιορισμών σε νέο training set



***Σχόλια*:** Το *Γράφημα 2.5*, προσφέρει τη δυνατότητα να παρατηρήσουμε ότι πλέον οι δύο περιορισμοί καλύπτονται τουλάχιστον από ένα μοντέλο, αυτό του Νευρωνικού Δικτύου. Ο ταξινομητής με kNN έχει εξίσου βελτιωμένα αποτελέσματα, φτάνοντας μόλις 3% - 4% μακριά από το να ανταπεξέλθει στον πρώτο περιορισμό. Γενικότερα, φαίνεται ότι όλα τα μοντέλα είχαν μια τρομερή βελτίωση στο κομμάτι του πρώτου περιορισμού, ο οποίος ήταν και ο δυσκολότερος διότι αναφερόταν σε δεδομένα της μειοψηφικής κλάσης.

3 Συμπεράσματα

Συμπεραίνοντας, το βέλτιστο μοντέλο βάση αποτελεσμάτων είναι το Νευρωνικό Δίκτυο που δημιουργήσαμε και αυτό γιατί έδειξε βελτίωση σχεδόν κατά 25% όσον αφορά το Recall, ενώ ήταν το μόνο μοντέλο που τήρησε και τους δύο περιορισμούς.

Παρόλα αυτά αν θέλουμε να είμαστε δίκαιοι, τα αποτελέσματα ενδέχεται να ποικίλλουν ανά εκπαίδευση, καθώς τα βάρη των νευρώνων αλλάζουν κάθε φορά και το 69% που πετύχαμε στον πρώτο περιορισμό μπορεί πολύ εύκολα να ανατραπεί αν το μοντέλο δεν αρχικοποιηθεί με τόσο ιδανικά βάρη.

Μια πρόταση που θα μπορούσαμε να κάνουμε για περαιτέρω βελτίωση της επίδοσης, όχι μόνο του Νευρωνικού Δικτύου, αλλά και όλων των μοντέλων ταξινόμησης που δοκιμάσαμε θα ήταν να κάνουμε κι’ άλλη μείωση στην πλειοψηφική κλάση ώστε τα δεδομένα εκπαίδευσης να έχουν πραγματικά τόσες πτωχευμένες όσες μη – πτωχευμένες εταιρείες και η αναλογία να είναι 1:1. Αυτό βέβαια θα οδηγούσε σε μεγάλη μείωση του συνολικού όγκου του training set και θα μας δημιουργούσε προβλήματα λόγω μεγέθους παραδειγμάτων. Ενδεχομένως, η τεχνική του oversampling της μειοψηφικής κλάσης να μας έλυνε κάτι τέτοιο.