







Στο πρόγραμμα ξεκίνησα διαβάζοντας τα δεδομένα από το text. Τα μετέτρεψα σε numbers, γέμισα τα ερωτηματικά με τη πιο συχνή τιμή από κάθε column και τα αποθήκευσα σε excel. Μετά πέταξα το πρώτο column που είναι τα id, κράτησα την τελευταία στήλη που ήταν τα classes μου και ύστερα το πέταξα από τον πίνακα και αυτό το column. Ύστερα έκανα το normalization ώστε να είναι από [0,1] οι τιμές και πλέον τα data μου ήταν ο υπολειπόμενος πίνακας με τις τιμές από [0,1].

Μετά ξεκίνησαν οι ταξινομητές με πρώτον τον KNN ο οποίος τον δοκίμασα με 5 neighbors και με 10. Έγινε η διαδικασία χωρισμού των data σε train και test. Μετά έγινε το fit του train(fitcknn) και ύστερα το predict του test κομματιού. Τα αποτελέσματα τα πήραμε από την classperf όπου μας βγάζει για το success rate, το sensitivity και το specificity.

Παρόμοια διαδικασία χρειάστηκε και ο Naïve Bayes όπου έθεσα το prior μία ως 0.3 – 0.7 και την άλλη φορά με 0.2 0.8. Σαφώς θα μπορούσαμε να δοκιμάσουμε και άλλους συνδυασμούς. Η διαδικασία είναι αρκετά παρόμοια με του KNN(test-train, fit(fitcnb), predict) με τη μόνη διαφορά ότι χρειάστηκε να κάνουμε και tabulate.

Επίσης παρόμοια και ο Svm όπου η διαδικασία είναι ίδια με του knn και δοκιμάζουμε με kernel function την RBF τη μία φορά και με την polynomial την άλλη φορά για να δούμε διαφορές σε αποτελέσματα. Θα μπορούσαμε να δοκιμάσουμε και με την linear ή να δοκιμάσουμε κάποιους solvers όπως ISDA L1QP ή SMO.

Τέλος δοκίμασα και τον Decision Tree με Max Number of Splits την default τιμή και με τιμή 7 για να δω διαφορές. Η διαδικασία πάλι ίδια με train-test, fitctree και predict.

Το συμπέρασμα από τις δοκιμές αυτές είναι ότι καλύτερο success rate και sensitivity έχουν οι knn και svm με τον knn πρώτο αλλά φαίνεται ότι αυτοί οι 2 με βάση το input ο ένας είναι καλύτερος από τον άλλον ανάλογα την περίπτωση. Και οι 2 όμως επιτυγχάνουν πολύ αξιόπιστα αποτελέσματα Ο knn όμως έχει το ελάττωμα ότι δουλεύει πολύ καλά κυρίως για μικρό πλήθος δεδομένων. Αμέσως μετά είναι ο naïve bayes ο οποίος βγάζει και εκείνος πολύ καλά αποτελέσματα και βλέπουμε ότι βγάζει και τα καλύτερα σε specificity. Άρα σίγουρα βγάζει καλά αποτελέσματα από τις τιμές που βγάζει αφού είναι ιδιαίτερα επιτυχής στο success rate. Επίσης είναι κατάλληλος για πολυδιάστατα δεδομένα . Τέλος είναι ο Decision Tree που βγάζει ικανοποιητικά αποτελέσματα αλλά σε σχέση με τους άλλους 3 αλγόριθμους φαίνεται ότι είναι σε χειρότερη κατάσταση. Φαίνεται όμως ότι είναι καλός με μεγάλο πλήθος δεδομένων και πολυδιάστατα δεδομένα.

Λέκκας Νικόλαος – sdi1600089