Μηχανική Μάθηση και Εφαρμογές

1η Άσκηση: Γραμμικά μοντέλα

it22039, ΚΑΡΑΘΑΝΑΣΗΣ ΑΝΔΡΕΑΣ

Τρόπος εκτέλεσης κώδικα:

Για να λειτουργήσει το πρόγραμμα αρκεί απλά να εκτελεστεί το αρχείο test_lr.py και θα εκτυπωθούν όλα τα ζητούμενα της εργασίας.

Λεπτομέρειες υλοποίησης:

Στο αρχείο linear_regression.py υλοποιείται η κλάση LinearRegression και οι τρείς συναρτήσεις της που ζητούνται. Αντικείμενο της κλάσης αυτής θα δημιουργηθεί στο δεύτερο αρχείο test_lr.py όπου υλοποιούνται τα υπόλοιπα ζητούμενα της εργασίας και εφαρμόζεται το μοντέλο σε δεδομένα. Στο αρχείο test_lr.py αρχικά παίρνω τα δεδομένα και τα χωρίζω σε δύο κομμάτια, ένα για εκπαίδευση και ένα για τον έλεγχο του μοντέλου. Το χώρισμα των δεδομένων γίνεται με σπόρο 42 προκειμένου να μπορούν να αναπαραχθούν ξανά ακριβώς τα ίδια split των δεδομένων. Ύστερα δημιουργώ ένα αντικείμενο από την custom κλάση που υλοποίησα στο αρχείο linear_regression.py και ένα από το module της scikit-learn. Κάνω την εκπαίδευση και των δύο μοντέλων και μετά βγάζω τις προβλέψεις των μοντέλων και υπολογίζω το Root Mean Square Error ως μετρική επίσοδης των μοντέλων το οποίο εκτυπώνω. Στη συνέχεια φτιάχνω ένα βρόγχο που τρέχει για 20 επαναλήψεις και φτιάχνει σε κάθε επανάληψη καινούργια τυχαία δεδομένα με τα οπόια κάνει την διαδικασία της εκπαίδευσης, του ελέγχου και του υπολογισμού του RMSE το οπόιο αποθηκύει σε ένα ξεχωριστό πίνακα για κάθε μοντέλο. Τέλος, υπολογίζεται η μέση τιμή και η τυπική απόκλιση του RMSE για τα δύο μοντέλα και εκτυπώνονται. Η τυπική απόκλιση για το custom μοντέλο υπολογίζεται αναλυτικά για να φανεί η διαδικασία και για το μοντέλο της scikit-learn χρησιμοποιείται έτοιμη συνάρτηση από την βιβλιοθήκη numpy. Υπάρχουν επίσης αναλυτικά σχόλια μέσα στον κώδικα που εξηγούν τι κάνω σε κάθε βήμα.

Σύγκριση αποτελεσμάτων;

Όπως φαίνεται από τις τιμές που εκτυπώνονται όταν τρέχουμε το πρόγραμμα, τα δύο μοντέλα (το ένα αυτό που έφτιαξα εγώ και το άλλο η έτοιμη υλοποίηση από τη βιβλιοθήκη scikit-learn) βγάζουν τα ίδια (σχεδόν) αποτελέσματα όταν χρησιμοποιούμε τα ίδια δεδομένα. Βγαίνει έτσι το συμπέρασμα ότι η δική μου υλοποίηση του μοντέλου είναι μάλλον σωστή. Ο λόγος όμως που τα δύο μοντέλα βγάζουν τα ίδια αποτελέσματα είναι ότι για την εκπαίδευση των μοντέλων χρησιμοποιείται η τεχνική της κανονικής παλινδρόμηση ελαχίστων τετραγώνων (Με βάση τις πληροφορίες που κατάφερα να βρω και η κλάση LinearRegression της scikit-learn χρησιμοποιεί την μέθοδο του OLS regression). Δηλαδή τα μοντέλα εκπαιδεύονται / προσαρμόζονται για τη λύση του προβλήματος με αναλυτικό τρόπο και όχι κατά προσέγγιση (analytical vs numerical solution). Αυτό σημαίνει ότι τα βάρη και η μεροληψία (bias) υπολογίζονται σε μία επανάληψη με ακρίβεια, λύνοντας εξισώσεις μερικών παραγώγων κλειστής μορφής με μαθηματικό τρόπο. Σε άλλες τεχνικές εκπαίδευσης που χρησιμοποιούν μεθόδους προσέγγισης της καλύτερης λύσης βελτιώνοντας το μοντέλο επαναληπτικά τα αποτελέσματα μπορεί να διέφεραν σε κάποιο βαθμό από μοντέλο σε μοντέλο. Σε αυτά τα αποτελέσματα παρόλο που αναμένονται ίδιες λύσεις ακριβώς για τα δύο μοντέλα παρατηρείται ότι υπάρχει μία μικρή διαφορά μετά από κάποια δεκαδικά ψηφία. Αυτό

συμβαίνει κατά τη γνώμη μου όχι από λάθος στην υλοποίηση του μοντέλου μου αλλά ως αποτέλεσμα του περιορισμένου precision στη δεκαδική αναπαράσταση των αριθμών και των πράξεων μεταξύ τους.