

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΤΥΠΩΝ
Χειμερινό Εξάμηνο 2019-20

Προπαρασκευή 1ης Εργαστηριακής Άσκησης:
Οπτική Αναγνώριση Ψηφίων

ΠΕΡΙΓΡΑΦΗ

Σκοπός είναι η υλοποίηση ενός συστήματος οπτικής αναγνώρισης ψηφίων. Τα δεδομένα προέρχονται από την US Postal Service (γγραμμένα στο χέρι σε ταχυδρομικούς φακέλους και σκαναρισμένα) και περιέχουν τα ψηφία από το 0 έως το 9 και διακρίνονται σε train και test.

Τα δεδομένα κάθε αρχείου αναπαριστούν τα περιεχόμενα ενός πίνακα (οι τιμές των στοιχείων του πίνακα διαχωρίζονται με κενό). Κάθε γραμμή αφορά ένα ψηφίο (δείγμα). Οι στήλες αντιστοιχούν στα χαρακτηριστικά (features) που περιγράφουν τα ψηφία. Για παράδειγμα, η τιμή του (i,j) στοιχείου αφορά το j -th χαρακτηριστικό του i -th ψηφίου. Κάθε ψηφίο περιγράφεται από 257 τιμές, εκ των οποίων η πρώτη αντιστοιχεί στο ίδιο το ψηφίο (αν είναι το 0, το 1 κτλ.) και οι υπόλοιπες 256 είναι τα χαρακτηριστικά (features) που το περιγράφουν (grayscale values). Ας φανταστούμε το κάθε ψηφίο να απεικονίζεται σε έναν 16×16 πίνακα αποτελούμενο από 256 κουτάκια ("pixels"). Για να εμφανίζεται το κάθε ψηφίο στην οθόνη "φωτίζεται" ένα σύνολο από τέτοια κουτάκια, με τέτοιο τρόπο ώστε η συνολική εικόνα που βλέπουμε να απεικονίζει το θεωρούμενο ψηφίο. Επειδή τα ψηφία εμφανίζονται σε grayscale, κάθε μία από τις 256 τιμές αντιστοιχεί σε μία απόχρωση μαύρου για το αντίστοιχο "pixel". Στόχος είναι η δημιουργία και αποτίμηση (evaluation) ταξινομητών οι οποίοι θα ταξινομούν κάθε ένα από τα ψηφία που περιλαμβάνονται στα test δεδομένα σε μία από τις δέκα κατηγορίες (από το 0 έως το 9).

ΕΚΤΕΛΕΣΗ

Κατεβάστε τα δεδομένα από τις διευκρινήσεις του mycourses.
Χρησιμοποιώντας Python εκτελέστε τα παρακάτω βήματα:

Βήμα 1

Διαβάστε τα δεδομένα από το αρχείο. Τα δεδομένα πρέπει να διαβαστούν σε μορφή συμβατή με το scikit-learn σε 4 πίνακες X_{train} , X_{test} , y_{train} και y_{test} . Ο πίνακας X_{train} περιέχει τα δείγματα εκπαίδευσης (χωρίς τα labels) και είναι διάστασης $(n_{samples_train} \times n_{features})$. Ο y_{train} είναι ένας μονοδιάστατος πίνακας μήκους $n_{samples}$ και περιέχει τα αντίστοιχα labels για τον X_{train} . Αντίστοιχα για τα test δεδομένα.

Βήμα 2

Σχεδιάστε το υπ' αριθμόν 131 ψηφίο (βρίσκεται στη θέση 131) των train δεδομένων. Υπόδειξη: χρησιμοποιήστε τη συνάρτηση `numpy.reshape` για να οργανώσετε τα 256 χαρακτηριστικά σε ένα πίνακα 16×16 , και τη συνάρτηση `matplotlib.pyplot.imshow` για την απεικόνιση του ψηφίου.

Βήμα 3

Διαλέξτε 1 τυχαίο δείγμα από κάθε label (συνολικά 10 δείγματα). Σχεδιάστε τα σε ένα figure με subplots. (Hint `fig = plt.figure(); fig.add_subplot(...)`)

Βήμα 4

Υπολογίστε τη μέση τιμή των χαρακτηριστικών του pixel (10, 10) για το ψηφίο «0» με βάση τα train δεδομένα.

Βήμα 5

Υπολογίστε τη διασπορά των χαρακτηριστικών του pixel (10, 10) για το ψηφίο «0» με βάση τα train δεδομένα.

Βήμα 6

Υπολογίστε τη μέση τιμή και διασπορά των χαρακτηριστικών κάθε pixel για το ψηφίο «0» με βάση τα train δεδομένα.

Βήμα 7

Σχεδιάστε το ψηφίο «0» χρησιμοποιώντας τις τιμές της μέσης τιμής που υπολογίσατε στο Βήμα 6

Βήμα 8

Σχεδιάστε το ψηφίο «0» χρησιμοποιώντας τις τιμές της διασποράς που υπολογίσατε στο Βήμα 6. Συγκρίνετε το αποτέλεσμα με το αποτέλεσμα του Βήματος 7 και εξηγήστε τυχόν διαφορές.

Βήμα 9

(α) Υπολογίστε τη μέση τιμή και διασπορά των χαρακτηριστικών για όλα τα ψηφία (0-9) με βάση τα train δεδομένα.

(β) Σχεδιάστε όλα τα ψηφία χρησιμοποιώντας τις τιμές της μέσης τιμής που υπολογίσατε στο Βήμα 9(α).

Βήμα 10

Ταξινομήστε το υπό αριθμόν 101 ψηφίο των test δεδομένων σε μία από τις 10 κατηγορίες (κάθε ένα από τα 10 ψηφία, 0-9, αντιπροσωπεύει μία κατηγορία) βάσει της Ευκλείδειας απόστασης (υπόδειξη: χρησιμοποιείτε τις τιμές που υπολογίσατε στο Βήμα 9(α)). Ήταν επιτυχής η ταξινόμηση;

Βήμα 11

(α) Ταξινομήστε όλα τα ψηφία των test δεδομένων σε μία από τις 10 κατηγορίες με βάση την Ευκλείδεια απόσταση.

(β) Υπολογίστε το ποσοστό επιτυχίας για το Βήμα 11(α).

Βήμα 12

Υλοποιήστε τον ταξινομητή ευκλείδειας απόστασης σαν ένα scikit-learn estimator. Μπορείτε να χρησιμοποιήσετε αυτό τον κώδικα σαν βάση, υλοποιώντας τις μεθόδους fit, predict και score

<https://gist.github.com/georgepar/b53c01466d6649c8583497d120b9b479>

Βήμα 13

α) Υπολογίστε το score του ευκλείδειου ταξινομητή με χρήση 5-fold cross-validation

β) Σχεδιάστε την περιοχή απόφασης του ευκλείδειου ταξινομητή.

γ) Σχεδιάστε την καμπύλη εκμάθησης του ευκλείδειου ταξινομητή (learning curve)

ΠΑΡΑΔΟΤΕΑ

(1) Σύντομη αναφορά (σε pdf ή jupyter notebook) που θα περιγράφει τη διαδικασία που ακολουθήθηκε σε κάθε βήμα, καθώς και τα σχετικά αποτελέσματα. Τα αποτελέσματα πρέπει να συνοδεύονται και από ερμηνεία – σχολιασμό.

(2) Κώδικας Python (συνοδευόμενος από σύντομα σχόλια). Προσπαθήστε να κάνετε vectorized υλοποιήσεις.

Συγκεντρώστε τα (1) και (2) σε ένα .zip αρχείο το οποίο πρέπει να αποσταλεί μέσω του mycourses.ntua.gr πριν από τη διεξαγωγή του εργαστηρίου.