ΑΛΕΞΑΝΔΡΕΙΟ ΤΕΙ ΘΕΣΣΑΛΟΝΙΚΗΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΜΑΘΗΜΑ: ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

ΚΑΘΗΓΗΤΕΣ : ΚΩΣΤΑΣ ΔΙΑΜΑΝΤΑΡΑΣ, ΚΩΣΤΑΣ ΓΟΥΛΙΑΝΑΣ

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 1

ΕΙΣΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ – ΔΙΑΧΩΡΙΣΜΟΣ CROSS-VALIDATION

Σκοπός της άσκησης: Η ανάγνωση των δεδομένων από ένα αρχείο και η κατανόηση και η υλοποίηση της μεθόδου διασταύρωσης (Cross-Validation). Σύμφωνα με τη μέθοδο αυτή τα δεδομένα που διαθέτουμε χωρίζονται σε δύο υποσύνολα:

1. Το υποσύνολο εκπαίδευσης (train set) το οποίο θα χρησιμοποιηθεί για την εκπαίδευση του μοντέλου μηχανικής μάθησης.
2. Το υποσύνολο ελέγχου (test set) το οποίο θα χρησιμοποιηθεί για τον έλεγχο της ικανότητας γενίκευσης του μοντέλου.

Εκτελείται μια σειρά από πειράματα που καλούνται "*folds*". Σε κάθε fold:

- δημιουργούνται διαφορετικά train set και test set χωρίζοντας τα δεδομένα με τυχαίο τρόπο
- το μοντέλο εκπαιδεύεται χρησιμοποιώντας το αντίστοιχο train set
- υπολογίζεται το σφάλμα (ή η επιτυχία) του αλγορίθμου στο test set. Ανάλογα με το πρόβλημα το κριτήριο επίδοσης μπορεί να είναι διαφορετικό.

Αφού εκτελεστούν K folds συλλέγεται ο μέσος όρος της επίδοσης του αλγορίθμου στα K folds. Αυτός ο μέσος όρος αποτελεί την εκτίμησή μας για την επίδοση του μοντέλου σε άγνωστα δεδομένα (ικανότητα γενίκευσης).

Βήματα υλοποίησης:

1. Κατεβάστε το σύνολο δεδομένων (data set) IRIS dataset από την παρακάτω ιστοσελίδα:

   http://archive.ics.uci.edu/ml/machine-learning-databases/iris/

   Αυτό είναι ίσως το πιο γνωστό σύνολο δεδομένων που χρησιμοποιείται στη βιβλιογραφία της αναγνώρισης προτύπων. Αφορά την αναγνώριση του τύπου λουλουδιού του γένους "ίρις". Περιέχει 3 κλάσεις λουλουδιών: "Iris-setosa","Iris-versicolor" και "Iris-virginica", με 50 δείγματα από κάθε μια κλάση (σύνολο 150 δείγματα).

   Το data set αποτελείται από δύο αρχεία:

   i. `iris.data` : περιέχει τα δεδομένα. Αποτελείται από 150 γραμμές, όπου κάθε γραμμή αντιστοιχεί σε ένα δείγμα. Κάθε δείγμα περιέχει 4 χαρακτηριστικά συν τον τύπο του λουλουδιού σε μορφή text-string. χωρισμένα με κόμματα.
   ii. `iris.names` : ενημερωτικό κείμενο το οποίο περιέχει την περιγραφή των δεδομένων.
2. Διαβάστε το αρχείο δεδομένων `iris.data` στην Python. Από την βιβλιοθήκη pandas, χρησιμοποιήστε τη συνάρτηση
   - `read_csv()` : διαβάζει αρχείο csv.

```
# Παράδειγμα:
data = read_csv('όνομα αρχείου ή URL', header=None).values
```

3. Υπολογίστε τα εξής:
   - Πλήθος των attributes: `NumberOfAttributes` (στη συγκεκριμένη περίπτωση = 5) και Πλήθος των δειγμάτων: `NumberOfPatterns` (στη συγκεκριμένη περίπτωση = 150) χρησιμοποιώντας το attribute `shape` του πίνακα `data`.
   - Δημιουργήστε ένα dictionary `map_dict` με τα εξής ζευγάρια key/values:
     - `"Iris-setosa": 0`
     - `"Iris-versicolor": 1`
     - `"Iris-virginica": 0`

4. Δημιουργήστε πίνακα δεδομένων `x` και στόχων `t` ως εξής:
   - Δημιουργήστε τον πίνακα `x` από τις 4 πρώτες στήλες του πίνακα `data`.
   - Χρησιμοποιώντας τη συνάρτηση zeros από τη βιβλιοθήκη numpy αρχικοποιήστε τον πίνακα `t` ώστε να είναι γεμάτος μηδενικά και να έχει διάσταση `NumberOfPatterns`. Κατόπιν, για κάθε πρότυπο `pattern`, η 5ο στήλη του πίνακα `data` (τύπου string) είναι το όνομα της κλάσης που ανήκει το πρότυπο αυτό. Χρησιμοποιώντας loop θέστε για κάθε pattern την τιμή στόχου `t[pattern]` ως εξής:

     `t[pattern] = 1` αν η 5ο στήλη για το `pattern` είναι "`Iris-versicolor`"

     `t[pattern] = 0` σε διαφορετική περίπτωση

   Μπορείτε να το κάνετε αυτό χρησιμοποιώντας το `map_dict` και να αποφύγετε εντολή `if-else` (?)

5. Δοκιμή της μεθόδου `train_test_split()`

   Τεμαχίστε τα δεδομένα σε 9 cross-validation folds (K=9) χρησιμοποιώντας τη συνάρτηση `train_test_split()` από τη βιβλιοθήκη `sklearn.model_selection`. Δώστε παράμετρο `test_size=0.1`.

   Θα πρέπει να κάνετε τα εξής:

   Για κάθε *fold* θα πάρετε τους πίνακες

   - `xtrain` πίνακας με τα πρότυπα που θα χρησιμοποιηθούν στην εκπαίδευση
   - `xtest` πίνακας με τα πρότυπα που θα χρησιμοποιηθούν στον έλεγχο
   - `ttrain` διάνυσμα με τους στόχους που θα χρησιμοποιηθούν στην εκπαίδευση
   - `ttest` διάνυσμα με τους στόχους που θα χρησιμοποιηθούν στον έλεγχο
   - Χρησιμοποιώντας τη συνάρτηση `plot` από τη βιβλιοθήκη `matplotlib.pyplot` σχεδιάστε
     - τα διανύσματα `xtrain[:,0]` → άξονας x, `xtrain[:,2]` → άξονας y, χρησιμοποιώντας τελείες με μπλε χρώμα και
     - τα διανύσματα `xtest[:,0]` → άξονας x, `xtest[:,2]` → άξονας y, χρησιμοποιώντας τελείες με κόκκινο χρώμα
- Χρησιμοποιήστε την εντολή `subplot` έτσι ώστε όλα τα γραφήματα να εμφανιστούν στο ίδιο Figure.