ΑΛΕΞΑΝΔΡΕΙΟ ΤΕΙ ΘΕΣΣΑΛΟΝΙΚΗΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΜΑΘΗΜΑ: ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

ΚΑΘΗΓΗΤΕΣ : ΚΩΣΤΑΣ ΔΙΑΜΑΝΤΑΡΑΣ, ΚΩΣΤΑΣ ΓΟΥΛΙΑΝΑΣ

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ

ΕΙΣΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ – ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ ΠΡΟΤΥΠΩΝ – ΕΠΙΛΟΓΗ ΠΡΟΒΛΗΜΑΤΟΣ - ΔΙΑΧΩΡΙΣΜΟΣ CROSS-VALIDATION

Σκοπός της άσκησης: Η ανάγνωση των δεδομένων από ένα αρχείο, η γραφική παράσταση των προτύπων στο επίπεδο, η επιλογή της κλάσης που θέλουμε να διαχωρίσουμε απ' τις υπόλοιπες δύο 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:

- 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.
- 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
```

- Υπολογίστε το Πλήθος των χαρακτηριστικών, `NumberOfAttributes` (= 5) και το Πλήθος των δειγμάτων, `NumberOfPatterns` (= 150) χρησιμοποιώντας το attribute shape του πίνακα `data`.
- Δημιουργήστε τον πίνακα των προτύπων `x` από τις 4 πρώτες στήλες του πίνακα `data`. Για κάθε πρότυπο `pattern`, η 5^{ο} στήλη του πίνακα `data` (τύπου string) είναι το όνομα της κλάσης που ανήκει το πρότυπο αυτό.
- Με τη χρήση της εντολής plot δημιουργήστε τη **γραφική παράσταση** των προτύπων **των 3 κλάσεων** με **διαφορετικό σύμβολο και χρώμα για την κάθε κλάση** χρησιμοποιώντας την 1^{η} και 3^{η} στήλη του πίνακα `x`, ώστε να τα απεικονίσετε στο χώρο των 2 διαστάσεων και εμφανίστε τα στο ίδιο γράφημα, ώστε να πάρετε μια ιδέα για το πώς είναι η διασπορά των προτύπων στο χώρο των 4 διαστάσεων. Μη ξεχνάτε ότι η αρίθμηση ξεκινάει απ' το 0.
- Χρησιμοποιώντας τη συνάρτηση zeros από τη βιβλιοθήκη numpy αρχικοποιήστε τον πίνακα `t` ώστε να είναι γεμάτος μηδενικά και να έχει διάσταση `NumberOfPatterns`.
- **Εκχωρήστε** στη μεταβλητή ans την τιμή "y'.
- Για όσο (ans = "y')
  - **Εμφανίστε** το παρακάτω **menu** επιλογών :

    1 **Διαχωρισμός `Iris-setosa`** από `Iris-versicolor και Iris-virginica`

    2 **Διαχωρισμός `Iris-virginica`** από `Iris-setosa και Iris-versicolor`

    3 **Διαχωρισμός `Iris-versicolor`** από `Iris-setosa και Iris-virginica`

    Διαβάστε την επιλογή (1/2/3)

    Αν επιλογή = 1

    Δημιουργήστε ένα dictionary `map_dict` με τα εξής ζευγάρια key/values:

    - `"Iris-setosa": 1`
    - `"Iris-versicolor": 0`
    - `"Iris-virginica": 0`

    Κατόπιν, χρησιμοποιώντας loop θέστε για κάθε pattern την τιμή στόχου `t[pattern]` ως εξής:

    `t[pattern] = 1` αν η 5^{ο} στήλη για το pattern είναι "`Iris-setosa`"

    `t[pattern] = 0` σε διαφορετική περίπτωση

    Αν επιλογή = 2

    Δημιουργήστε ένα dictionary `map_dict` με τα εξής ζευγάρια key/values:

    - `"Iris-setosa": 0`
    - `"Iris-versicolor": 0`
    - `"Iris-virginica": 1`

    Κατόπιν, χρησιμοποιώντας loop θέστε για κάθε pattern την τιμή στόχου `t[pattern]` ως εξής:

    `t[pattern] = 1` αν η 5^{ο} στήλη για το pattern είναι "`Iris-virginica`"

    `t[pattern] = 0` σε διαφορετική περίπτωση

    Αν επιλογή = 3

    Δημιουργήστε ένα dictionary `map_dict` με τα εξής ζευγάρια key/values:

    - `"Iris-setosa": 0`

- "Iris-versicolor": 1
- "Iris-virginica": 0

Κατόπιν, 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.

- Χωρισμός προτύπων σε πρότυπα εκπαίδευσης και ανάκλησης

  Τεμαχίστε τα δεδομένα των πινάκων x και t σε 4 πίνακες:

  - xtrain πίνακας με τα πρότυπα που θα χρησιμοποιηθούν στην εκπαίδευση, τα 40 πρώτα πρότυπα της κάθε κλάσης.
  - xtest πίνακας με τα πρότυπα που θα χρησιμοποιηθούν στον έλεγχο, τα 10 τελευταία πρότυπα της κάθε κλάσης.
  - ttrain διάνυσμα με τους στόχους που θα χρησιμοποιηθούν στην εκπαίδευση, οι 40 πρώτοι στόχοι της κάθε κλάσης.
  - ttest διάνυσμα με τους στόχους που θα χρησιμοποιηθούν στον έλεγχο, οι 10 τελευταίοι στόχοι της κάθε κλάσης.
  - 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

- 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.

  Διαβάστε την απάντηση ans του χρήστη, αν θέλετε να συνεχίσετε.

Οδηγίες κατάθεσης ασκήσεων

1. Συνδεθείτε στο URL: http://aetos.it.teithe.gr/s
2. Επιλέξτε το μάθημα “Μηχανική Μάθηση – Εργαστήριο Χ” (Όπου Χ ο αριθμός του εργαστηρίου του οποίου τις ασκήσεις πρόκειται να καταθέσετε) και πατήστε επόμενο.
3. Συμπληρώστε τα στοιχεία σας. Πληκτρολογείστε USERNAME 00003 και PASSWORD 30000 (Επώνυμο και Όνομα με ΛΑΤΙΝΙΚΟΥΣ ΧΑΡΑΚΤΗΡΕΣ).
4. Αν θέλετε να καταθέσετε μόνο ένα αρχείο μη το βάζετε σε zip file. Αντίθετα, αν θέλετε να καταθέσετε περισσότερα από ένα αρχεία, τοποθετήστε τα σε ένα zip ή rar file.
5. Επιλέξτε το αρχείο που θέλετε να στείλετε επιλέγοντας “choose file” στο πεδίο FILE1 και πατήστε “Παράδοση”