

ΑΛΕΞΑΝΔΡΕΙΟ ΤΕΙ ΘΕΣΣΑΛΟΝΙΚΗΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΜΑΘΗΜΑ: ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

ΚΑΘΗΓΗΤΕΣ : ΚΩΣΤΑΣ ΔΙΑΜΑΝΤΑΡΑΣ, ΚΩΣΤΑΣ ΓΟΥΛΙΑΝΑΣ

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 4

ΜΟΝΤΕΛΟ MLP – ΕΚΠΑΙΔΕΥΣΗ BACKPROPAGATION

Σκοπός της άσκησης: Η εκτίμηση της επίδοσης ενός ταξινομητή τύπου Multi-Layer Perceptron με ένα κρυφό στρώμα εκπαιδευόμενου με τον αλγόριθμο Back-Propagation. Θα γίνει χρήση της μεθόδου διασταύρωσης (Cross-Validation) και τα κριτήρια επίδοσης:

1. Ακρίβεια (accuracy)
2. Ευστοχία (precision)
3. Ανάκληση (recall)
4. F-Measure
5. Ευαισθησία (Sensitivity)
6. Προσδιοριστικότητα (Specificity)

Βήματα υλοποίησης:

1. Χρησιμοποιήστε το σύνολο δεδομένων IRIS από το προηγούμενο εργαστήριο, καθώς και τον κώδικα από το εργαστήριο αυτό. Θυμίζουμε ότι τα πρότυπα χωρίστηκαν σε δύο κλάσεις ως εξής:
 - Κλάση 0 (στόχος $t=0$): αποτελείται από τα πρότυπα των κατηγοριών "Iris-setosa" + "Iris-virginica",
 - Κλάση 1 (στόχος $t=1$): αποτελείται από τα πρότυπα της κατηγορίας "Iris-versicolor"
2. Θα εφαρμοστεί η μέθοδος `train_test_split()` για $K=9$ folds.
3. Πριν από το Cross-Validation loop ζητήστε από το χρήστη να δώσει πλήθος νευρώνων στο κρυφό στρώμα.
4. Στο Cross-Validation loop θα πρέπει να κάνετε τα εξής:

Για κάθε fold

- Έχετε ήδη δημιουργήσει τους αρχικούς πίνακες προτύπων `xtrain` και `xtest` (χωρίς επαύξηση) καθώς και τα διανύσματα στόχων `ttrain` και `ttest`. Αποφασίστε αν οι τιμές των στόχων θα είναι 0/1, είτε -1/1 και γράψτε τον υπόλοιπο κώδικα ανάλογα με την απόφαση που πήρατε.
- Δημιουργήστε ένα δίκτυο MLP δύο στρωμάτων χρησιμοποιώντας την κλάση
- `MLPClassifier(hidden_layer_sizes, activation, solver, learning_rate, learning_rate_init, max_iter, momentum=0.9, ...)` όπου:
 - `hidden_layer_sizes`=πλήθος νευρώνων στο κρυφό στρώμα
 - `activation` = επιλογή συνάρτησης ενεργοποίησης μεταξύ 'identity', 'logistic', 'tanh', 'relu'. Επιλέξτε 'tanh' (δηλ. σιγμοειδή -1/1) ή 'logistic' (δηλ. σιγμοειδή 0/1)
 - `solver` = επιλογή αλγορίθμου εκπαίδευσης, μεταξύ των
 - Στοχαστική κατάβαση δυναμικού (Stochastic Gradient Descent)
 - LBFGS (Limited memory Broyden–Fletcher–Goldfarb–Shanno)

- *Adam*
 - *max_iter* = πλήθος εποχών
 - *learning_rate*, *learning_rate_init* = βήμα εκπαίδευσης και αλγόριθμος ρύθμισης βήματος εκπαίδευσης (συνήθως 'constant')
 - *momentum* = ορμή (εφόσον αλγόριθμος εκπαίδευσης είναι *sgd*)
- Εκπαιδεύστε το δίκτυο που φτιάξατε χρησιμοποιώντας τη συνάρτηση
 - *fit()* : με εισόδους το μοντέλο, τον πίνακα των προτύπων εκπαίδευσης (*xtrain*), και το διάλυσμα των στόχων εκπαίδευσης (*ttrain*)
- Αφού εκπαιδεύσετε το μοντέλο κάνετε ανάκληση χρησιμοποιώντας τη συνάρτηση
 - *predict()* : με εισόδους το εκπαιδευμένο μοντέλο και τον πίνακα των προτύπων ελέγχου (*xtest*)
- Μετατρέψτε την έξοδο που θα λάβετε σε τιμές 0/1 ή $-1/1$ ανάλογα με τους στόχους που έχετε, χρησιμοποιώντας κάποιο κατάλληλο κατώφλι (0.5 ή 0). Ονομάστε *predict_{test}* το διάλυσμα που πήρατε.
- Καλέστε τη συνάρτηση *evaluate()* από το προηγούμενο εργαστήριο όσες φορές χρειάζεται έτσι ώστε για το συγκεκριμένο fold να υπολογίσετε το Accuracy, Precision, Recall, F-measure, Sensitivity και Specificity.
- Χρησιμοποιώντας κατάλληλο subplot σε grid 3x3 στο figure(1) τυπώστε το εξής γράφημα:
 - δείξτε με μπλε τελείες τους πραγματικούς στόχους $t_{test}(i)$ για όλα τα πρότυπα του test set
 - δείξτε με κόκκινους κύκλους τους εκτιμώμενους στόχους $predict_{test}(i)$ για όλα τα πρότυπα του test set

end *%for*

5. Μετά το τέλος του loop υπολογίστε και τυπώστε στην οθόνη τα εξής:

1. τη μέση τιμή του Accuracy για όλα τα folds
2. τη μέση τιμή του Precision για όλα τα folds
3. τη μέση τιμή του Recall για όλα τα folds
4. τη μέση τιμή του F-Measure για όλα τα folds
5. τη μέση τιμή του Sensitivity για όλα τα folds
6. τη μέση τιμή του Specificity για όλα τα folds

6. Τρέξτε πολλές φορές το πρόγραμμα χρησιμοποιώντας διαφορετικές τιμές

- N1 (πλήθος κρυφών νευρώνων) = 5, 10, 20, 50, 100
- Solver = '*sgd*', '*adam*'

Για τη μέθοδο *sgd* χρησιμοποιήστε

- Σταθερό Βήμα εκπαίδευσης
- Αρχικό βήμα εκπαίδευσης (default)
- Momentum (default)

Παρατηρήστε αν εμφανίζονται διαφορές στην επίδοση του αλγορίθμου.