Υπολογιστική Νοημοσύνη
Εργαστηριακές Ασκήσεις ακ. έτους 2020-21

Να κατασκευάσετε σύνολα δεδομένων για τα ακόλουθα προβλήματα:
Σ1) **Πρόβλημα ταξινόμησης τεσσάρων κατηγοριών**: Θα δημιουργήσετε **τυχαία** 6000 παραδείγματα (σημεία (x_1,x_2) στο επίπεδο) μέσα στο τετράγωνο [-1,1]x[-1,1] (3000 για το σύνολο εκπαίδευσης και 3000 για το σύνολο ελέγχου). Στη συνέχεια θα κατατάξετε κάθε παράδειγμα (x_1,x_2) (από τα 6000 παραδείγματα) σε μια κατηγορία από τέσσερεις κατηγορίες ως εξής:
1) εάν $x_1^2+x_2^2<0.25$, τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
2) εάν το (x_1,x_2) ανήκει στο τετράγωνο [-1,-0.4]x[-1,-0.4] κατατάσσεται στην κατηγορία C2,
3) εάν το (x_1,x_2) ανήκει στο τετράγωνο [0.4,1]x[0.4,1] κατατάσσεται στην κατηγορία C2,
4) εάν το (x_1,x_2) ανήκει στο τετράγωνο [-1,-0.4]x[0.4,1] κατατάσσεται στην κατηγορία C3,
5) εάν το (x_1,x_2) ανήκει στο τετράγωνο [0.4,1]x[-1,-0.4] κατατάσσεται στην κατηγορία C3,
6) αλλιώς κατατάσσεται στην κατηγορία C4.

Στη συνέχεια **προσθέτουμε θόρυβο μόνο στο σύνολο εκπαίδευσης** ως εξής: για κάθε παράδειγμα του συνόλου εκπαίδευσης που είναι κατηγορίας C1 ή C2 ή C3, με πιθανότητα 0.1 του αλλάζουμε κατηγορία και το αναθέτουμε στην κατηγορία C4.

Σ2) **Πρόβλημα ομαδοποίησης πέντε ομάδων** (900 παραδείγματα): δημιουργούμε **τυχαία** σημεία (x_1,x_2) στο επίπεδο ως εξής: α) 150 σημεία στο τετράγωνο [0.75,1.25]x[0.75,1.25], β) 150 σημεία στο τετράγωνο [0, 0.5]x[0,0.5], γ) 150 σημεία στο τετράγωνο [0,0.5] x [1.5,2], δ) 150 σημεία στο τετράγωνο [1.5,2]x[0,0.5], ε) 150 σημεία στο τετράγωνο [1.5,2] x [1.5,2], στ) 150 σημεία στο τετράγωνο [0,2]x[0,2].

Στη συνέχεια να κατασκευάσετε:

Π1) **Πρόγραμμα ταξινόμησης** βασισμένο στο **πολυεπίπεδο perceptron** (MLP) με **δύο κρυμμένα επίπεδα** με νευρώνες που έχουν **συνάρτηση ενεργοποίησης:** i) στο πρώτο κρυμμένο επίπεδο την υπερβολική εφαπτομένη (tanh(u)), ii) στο δεύτερο κρυμμένο επίπεδο την λογιστική ή τη γραμμική και iii) για το επίπεδο εξόδου θα ορίσετε εσείς τη συνάρτηση ενεργοποίησης που απαιτείται για το συγκεκριμένο πρόβλημα. Το πρόγραμμα θα πρέπει να αποτελείται από τις ακόλουθες μονάδες:

1) Με χρήση της εντολής define, καθορισμός αριθμού εισόδων (d), αριθμού κατηγοριών (K), αριθμού νευρώνων στο πρώτο κρυμμένο επίπεδο (H1), αριθμού νευρώνων στο δεύτερο κρυμμένο επίπεδο (H2) και είδος συνάρτησης ενεργοποίησης (λογιστική ή γραμμική) στο δεύτερο κρυμμένο επίπεδο.
2) Φόρτωμα των συνόλων εκπαίδευσης και ελέγχου (από αντίστοιχα αρχεία) και κωδικοποίηση των κατηγοριών (ορισμός των επιθυμητών εξόδων για κάθε κατηγορία).
3) Καθορισμός της αρχιτεκτονικής του δικτύου MLP. Ορισμός των απαιτούμενων πινάκων και άλλων δομών ως καθολικών μεταβλητών. Καθορισμός του ρυθμού μάθησης και του κατωφλίου τερματισμού. Τυχαία αρχικοποίηση των βαρών/πολώσεων στο διάστημα (-1,1).
4) Υλοποίηση της συνάρτησης forward-pass (float *x, int d, float *y, int K) η οποία υπολογίζει το διάνυσμα εξόδου y (διάστασης K) του MLP δοθέντος του διανύσματος εισόδου x (διάστασης d).
5) Υλοποίηση της συνάρτησης backprop(float *x, int d, float *t, int K) η οποία λαμβάνει τα διανύσματα x διάστασης d (είσοδος) και t διάστασης K (επιθυμητή έξοδος) και υπολογίζει τις παραγώγους του σφάλματος ως προς οποιαδήποτε παράμετρο (βάρος ή πόλωση) του δικτύου ενημερώνοντας τους αντίστοιχους πίνακες.
6) Χρησιμοποιώντας τα παραπάνω να υλοποιήσετε τον **αλγόριθμο εκπαίδευσης gradient descent και ενημέρωση των βαρών ανά ομάδες των Β παραδειγμάτων (mini-batches) θεωρώντας τα Ν παραδείγματα του συνόλου εκπαίδευσης** (όπου το Β διαιρέτης του Ν και ορίζεται στην αρχή του προγράμματος). Σημειώστε ότι εάν Β=1 έχουμε σειριακή ενημέρωση, ενώ εάν Β=Ν έχουμε ομαδική ενημέρωση. **Στο τέλος κάθε εποχής θα πρέπει υποχρεωτικά να υπολογίζετε και να τυπώνετε την τιμή**

του συνολικού σφάλματος εκπαίδευσης. Τερματίζουμε όταν η διαφορά της τιμής του σφάλματος εκπαίδευσης μεταξύ δύο εποχών γίνει μικρότερη από κάποιο κατώφλι, αφού όμως ο αλγόριθμος έχει τρέξει για τουλάχιστον 500 εποχές.
7) Αφού τερματιστεί η εκπαίδευση του δικτύου να γίνεται υπολογισμός και εκτύπωση της **ικανότητας γενίκευσης** του δικτύου που προκύπτει, υπολογίζοντας το **ποσοστό σωστών αποφάσεων στο σύνολο ελέγχου**.

Χρησιμοποιώντας το πρόγραμμα (Π1) να μελετήσετε το πρόβλημα (Σ1).
Να εξετάσετε και να καταγράψετε πώς μεταβάλλεται η γενικευτική ικανότητα του δικτύου (ποσοστό επιτυχίας στο σύνολο ελέγχου) θεωρώντας τους συνδυασμούς τιμών (H1,H2) π.χ.{(7,4),(8,5),(10,6)}, για συνάρτηση ενεργοποίησης λογιστική ή γραμμική στο δεύτερο κρυμμένο επίπεδο και για τιμές του B={1, N/10, N/100, N} (συνολικά 3x2x4=24 περιπτώσεις). Για το δίκτυο με την καλύτερη γενικευτική ικανότητα που θα βρείτε, να τυπώσετε τα παραδείγματα του συνόλου ελέγχου χρησιμοποιώντας διαφορετικό στυλ (πχ + και -) ανάλογα με το αν το παράδειγμα ταξινομείται από το δίκτυο στη σωστή κατηγορία ή όχι.

Π2) **Πρόγραμμα ομαδοποίησης** με M ομάδες (το M θα ορίζεται με την εντολή #define) βασισμένο στον **αλγόριθμο k-means**. Το πρόγραμμα θα φορτώνει το αρχείο με τα παραδείγματα, θα εκτελεί τον αλγόριθμο k-means με M κέντρα και στο τέλος θα αποθηκεύει τις συντεταγμένες των κέντρων των ομάδων. Η αρχική θέση κάθε κέντρου να γίνεται επιλέγοντας τυχαία κάποιο από τα παραδείγματα. Επίσης θα πρέπει στο τέλος να υπολογίζεται **το σφάλμα ομαδοποίησης ως εξής: για κάθε παράδειγμα x_i υπολογίζουμε την Ευκλείδεια απόσταση $||x_i-\mu_k||^2$ από το κέντρο μ_k της ομάδας στην οποία ανήκει και αθροίζουμε τις αποστάσεις για όλα τα παραδείγματα x_i**.

Π3) **Πρόγραμμα ομαδοποίησης** με M ομάδες (το M θα ορίζεται με την εντολή #define) βασισμένο στον **αλγόριθμο LVQ** για ομαδοποίηση. Το πρόγραμμα θα φορτώνει το αρχείο με τα παραδείγματα, θα εκτελεί τον αλγόριθμο LVQ με M κέντρα και στο τέλος θα αποθηκεύει τις συντεταγμένες των κέντρων των ομάδων. Επίσης θα **πρέπει στο τέλος να υπολογίζεται το σφάλμα ομαδοποίησης.** Η αρχική θέση κάθε κέντρου να γίνεται επιλέγοντας τυχαία κάποιο από τα παραδείγματα. Ο ρυθμός μάθησης n να μειώνεται στο τέλος κάθε **εποχής** (π.χ. n(t+1)=0.95 n(t)) ξεκινώντας από μια κατάλληλη **αρχική τιμή** (π.χ. n=0.1).

Να εφαρμόσετε τα προγράμματα Π2 και Π3 στο σύνολο δεδομένων (Σ2) για M=2,3,4,5,6,7,10 ομάδες.

Για **κάθε** πρόγραμμα Π2 ή Π3 και για **κάθε τιμή του M** να κάνετε τα εξής:
α) Να εκτελέσετε 5 τρεξίματα του προγράμματος και να κρατήσετε τη **λύση με το μικρότερο σφάλμα ομαδοποίησης**.
β) Στη συνέχεια να εμφανίσετε (plot) στο ίδιο σχήμα τόσο τα παραδείγματα (π.χ. με '+') όσο και τις θέσεις των κέντρων που βρήκατε (π.χ. με '*').

Βάσει των αποτελεσμάτων από το πρόγραμμα Π2, να φτιάξετε ένα διάγραμμα που να δείχνει πώς μεταβάλλεται το σφάλμα ομαδοποίησης με τον αριθμό των ομάδων; Μπορεί να χρησιμοποιηθεί το σφάλμα ομαδοποίησης για να εκτιμήσουμε τον πραγματικό αριθμό ομάδων; (στο σύνολο Σ2 ο πραγματικός αριθμός των ομάδων είναι 5).