Cross-Validation
Κ. Διαμαντάρας

Ορολογία

- Παράμετρος μοντέλου

  Παράμετρος που χρησιμοποιείται από το μοντέλο ώστε αυτό να βγάλει απόφαση (έξοδο). Για παράδειγμα, σε ένα νευρωνικό δίκτυο MLP παράμετροι είναι τα συναπτικά βάρη και οι πολώσεις των νευρώνων. Σε ένα δίκτυο SVM με πυρήνα RBF παράμετροι είναι οι δείκτες των support vectors και οι συντελεστές λ_i.

- Υπερ-παράμετρος μοντέλου

  Οποιαδήποτε παράμετρος περιγράφει την αρχιτεκτονική του μοντέλου και δεν σχετίζεται άμεσα με την απόφαση (έξοδο). Για παράδειγμα, σε ένα νευρωνικό δίκτυο MLP με ένα κρυφό στρώμα, υπερ-παράμετρος είναι το πλήθος των νευρώνων στο κρυφό στρώμα. Σε ένα μοντέλο SVM με πυρήνα RBF, υπερ-παράμετροι είναι ο συντελεστής C και η παράμετρος γ του πυρήνα.

- Fold:

  Ένα πείραμα Cross-Validation με συγκεκριμένο διαχωρισμό προτύπων σε train set / validation set ή train set / test set.

Η μέθοδος της **διαστάυρωσης** ή **Cross-Validation** αποτελεί μια στατιστική τεχνική με την οποία προσπαθούμε να επιτύχουμε όσο τον δυνατόν πιο ασφαλή εκτίμηση της δυνατότητας γενίκευσης ενός μοντέλου, δηλαδή να προβλέψουμε την επίδοσή του με βάση κάποιο κριτήριο πάνω σε άγνωστα δεδομένα τα οποία δεν έχει χρησιμοποιήσει κατά την εκπαίδευσή του. Ως κριτήριο μπορεί να χρησιμοποιηθεί οποιοδήποτε είναι κατάλληλο για το συγκεκριμένο πρόβλημα όπως, για παράδειγμα, η ακρίβεια (accuracy) για προβλήματα ταξινόμησης ή το μέσο τετραγωνικό σφάλμα (MSE) για προβλήματα παλινδρόμησης.

Η βασική φιλοσοφία της μεθόδου είναι ο τυχαίος διαχωρισμός των δεδομένων σε δύο τμήματα, το **train set** (σύνολο εκπαίδευσης) και το **test** ή **validation set** (το σύνολο ελέγχου). Το μοντέλο εκπαιδεύεται πάνω στο σύνολο εκπαίδευσης και η επίδοσή του κρίνεται πάνω στο σύνολο ελέγχου. Ένα τέτοιο πείραμα καλείται **fold**. Επειδή ένα μόνο fold μπορεί να βγάλει σχετικά μη ασφαλή αποτελέσματα, συνήθως εκτελούμε K folds και παίρνουμε την μέση τιμή της επίδοσης για να εκτιμήσουμε με ασφαλέστερο τρόπο την αναμενόμενη επίδοση του μοντέλου. Το K μπορεί να πάρει οποιαδήποτε θετική ακέραια τιμή, πχ. 5, 10, 100 κλπ. Όσο μεγαλύτερη είναι η τιμή του K τόσο πιο ασφαλής μπορεί να θεωρηθεί η εκτίμηση που κάνουμε.

Υπάρχουν διάφορες στατιστικές αναλύσεις στις οποίες μπορεί να χρησιμοποιηθεί η μέθοδος Cross-Validation ανάλογα με το ζητούμενο αποτέλεσμα. Αυτές περιγράφονται παρακάτω.

Ανάλυση τύπου 1. Εκτίμηση επίδοσης γενίκευσης (Generalization performance estimation)

Σκοπός

Η εκτίμηση της επίδοσης ενός μοντέλου όσον αφορά άγνωστα πρότυπα (εκτίμηση ποιότητας γενίκευσης).

Μέθοδος

Εκτελούμε K folds. Σε κάθε fold χωρίζουμε το σύνολο των δεδομένων με τυχαίο τρόπο σε train set και validation set (πχ σε ποσοστό 80% / 20%). Εκπαιδεύουμε το μοντέλο πάνω στο train set και ελέγχουμε την επίδοσή του στο validation set. Βγάζουμε το μέσο όρο επίδοσης για όλα τα folds και λέμε ότι αυτή είναι η αναμενόμενη επίδοση του μοντέλου.

Αν θέλουμε μπορούμε να επαναλάβουμε όλη τη διαδικασία επιλέγοντας άλλες τιμές υπερ-παραμέτρων.

Ψευδοκώδικας

```
for param1 in range(PARAMETER1_RANGE) {
    for param2 in range(PARAMETER2_RANGE) {
        ....
        for folds in range(NUM_FOLDS) {
            # Χώρισε τα δεδομένα σε train_set και validation_set
```

```
            # Χρησιμοποίησε το train_set για εκπαίδευση του μοντέλου
            # με υπερ-παραμέτρους param1, param2, ...
            # Χρησιμοποίησε το validation_set για εκτίμηση της επίδοσης
            # Σώσε την επίδοση σε ένα array: perf[fold] = performance
        }
        # Για κάθε συνδυασμό υπερ-παραμέτρων  param1, param2, ...
        # υπολόγισε τη μέση επίδοση για όλα τα folds
        # avg_perf[param1, param2, ...] = mean(perf)
    }
}

# Επέστρεψε το avg_perf για κάθε σετ υπερ-παραμέτρων
```

Ανάλυση τύπου 2. Επιλογή Βέλτιστου Μοντέλου (Model Selection)

Σκοπός

Η επιλογή των καταλληλότερων υπερ-παραμέτρων του μοντέλου με βάση την επίδοσή του στο validation set και κατόπιν η εκπαίδευση του μοντέλου με τις καταλληλότερες υπερ-παραμέτρους χρησιμοποιώντας όλα τα δεδομένα.

Μέθοδος

Η μέθοδος αποτελείται από δύο βήματα

- Επιλογή καταλληλότερων υπερ-παραμέτρων

  Δοκιμάζουμε διάφορα σετ υπερ-παραμέτρων. Για κάθε τέτοιο σετ εκτελούμε K πειράματα που λέγονται folds. Σε κάθε fold χωρίζουμε το σύνολο των δεδομένων σε train set και validation set. Εκπαιδεύουμε το μοντέλο πάνω στο train set και ελέγχουμε την επίδοσή του στο validation set. Βγάζουμε το μέσο όρο επίδοσης για όλα τα folds και λέμε ότι αυτή είναι η αναμενόμενη επίδοση του μοντέλου με αυτές τις υπερ-παραμέτρους. Επαναλαμβάνουμε για όλα τα σετ υπερ-παραμέτρων. Βρίσκουμε το καταλληλότερο σετ υπερ-παραμέτρων, αυτό δηλαδή με την καλύτερη αναμενόμενη επίδοση.
- Εκπαίδευση μοντέλου

  Αφού έχουμε βρεί το καταλληλότερο σετ υπερ-παραμέτρων εκπαιδεύουμε το μοντέλο με αυτές τις υπερ-παραμέτρους χρησιμοποιώντας το σύνολο των δεδομένων, δηλαδή χωρίς να τα χωρίσουμε πλέον σε train set και validation set.

Ψευδοκώδικας

```
# Βήμα 1: Εύρεση βέλτιστων υπερ-παραμέτρων
#----------------------------------------
for param1 in range(PARAMETER1_RANGE) {
    for param2 in range(PARAMETER2_RANGE) {
        ....
        for folds in range(NUM_FOLDS) {
            # Χώρισε τα δεδομένα σε train_set και validation_set
            # Χρησιμοποίησε το train_set για εκπαίδευση του μοντέλου
            # με υπερ-παραμέτρους param1, param2, ...
            # Χρησιμοποίησε το validation_set για εκτίμηση της επίδοσης
            # Σώσε την επίδοση σε ένα array: perf[fold] = performance
        }
        # Για κάθε συνδυασμό υπερ-παραμέτρων  param1, param2, ...
        # υπολόγισε τη μέση επίδοση για όλα τα folds
        # avg_perf[param1, param2, ...] = mean(perf)
    }
}
```

```
# Βρες το σύνολο των βέλτιστων παραμέτρων opt_param1, opt_param2, ...
# Είναι αυτές για τις οποίες επιτυγχάνεται το ελάχιστο avg_perf

# Βήμα 2: Εύρεση βέλτιστων παραμέτρων:
#-----------------------------------
# Εκπαίδευσε ξανά το μοντέλο με τις βέλτιστες υπερ-παραμέτρους
# πάνω σε ΟΛΑ τα δεδομένα
# Επέστρεψε το εκπαιδευμένο μοντέλο
```

Σχόλιο Python: Στην βιβλιοθήκη scikit-learn, η μέθοδος αυτή υλοποιείται από την function **`GridSearchCV()`**

http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html.

Αν χρησιμοποιηθεί η επιλογή `refit=False` τότε η συνάρτηση παραλείπει το βήμα 2, και δεν επιστρέφει βέλτιστο εκπαιδευμένο μοντέλο, αλλά μόνο τις τιμές των βέλτιστων υπερ-παραμέτρων `opt_param1, opt_param2,...` και την βέλτιστη μέση επίδοση `avg_perf[opt_param1, opt_param2, ...]`. Στην περίπτωση αυτή, ουσιαστικά, η συνάρτηση εφαρμόζει ανάλυση τύπου 1 όπως περιγράφτηκε παραπάνω.