Εργασία 4 – Ημι-επιβλεπόμενη Μάθηση & Αξιολόγηση Μοντέλων

Βιολογικό εργαστήριο θέλει να συνδυάσει τη χημική ανάλυση του κρασιού με την επιμέρους αξιολόγηση της ποιότητας τους. Πιο συγκεκριμένα, θα παίρνει δείγματα κρασιών από τα εργοστάσια (είτε λευκά είτε κόκκινα) και θα υπολογίζει με ένα σκορ από το 0 ως το 10 την ποιότητα τους. Διαθέτετε ένα σύνολο δεδομένων με δείγματα κρασιού που περιλαμβάνουν τα παρακάτω χαρακτηριστικά:

1. **Type**: Τύπος του κρασιού (Λευκό/Κόκκινο)
2. **Fixed Acidity:** Σταθερή οξύτητα
3. **Volatile Acidity:** Πτητική οξύτητα
4. **Critic Acid**: Κιτρικό οξύ
5. **Residual Sugar**: Υπολειπόμενα σάκχαρα
6. **Chlorides**: Χλωρίδια
7. **Free Sulfur Dioxide:** Ελεύθερο διοξείδιο του θείου
8. **Total Sulfur Dioxide:** Συνολικό διοξείδιο του θείου
9. **Density:** Πυκνότητα
10. **pH:** Επίπεδα pH
11. **Sulphates:** Θειικά άλατα
12. **Alcohol:** Ποσότητα αλκοόλης
13. **Quality:** Ποιότητα του κρασιού (Σκορ 0-10).

Μέρος 1 – Ημι-επιβλεπόμενη Μάθηση

Δυστυχώς, το εργαστήριο κράτησε τον τύπο του κρασιού μόνο για μερικά δείγματα. Επομένως, σε αυτό το μέρος θα χρησιμοποιήσουμε Ημι-επιβλεπόμενη Μάθηση, ώστε να τα υπολογίσουμε.

1. Φορτώστε το σύνολο δεδομένων *wine-missing.csv* σε ένα DataFrame μέσω της βιβλιοθήκης pandas. Στη συνέχεια, περιγράψτε το κάθε χαρακτηριστικό και δημιουργήστε το αντίστοιχο ιστόγραμμα τους. Για τη μεταβλητή Type, να δημιουργηθεί ραβδόγραμμα για τις τιμές white, red και unknown, όπου unknown είναι άγνωστο αν τα δείγματα κρασιού ήταν λευκά ή κόκκινα.
2. Μετατρέψτε τις τιμές white/red/unknown σε 0/1/2 αντίστοιχα. Έπειτα, δημιουργήστε numpy arrays x (inputs), y (targets), με x να είναι όλες οι μεταβλητές εκτός του type και y το type.
3. Δημιουργήστε 2 διαφορετικά σύνολα δεδομένων: (1) (x_known, y_known) και (2) x_unknown. Το 1ο σύνολο δεδομένων περιέχει όλα τα ζεύγη (inputs, targets) στα οποία είναι γνωστή η μεταβλητή type (white/red), ενώ στο 2ο είναι όλα τα inputs που το target είναι *unknown*.
4. Για το 1ο σύνολο δεδομένων, χρησιμοποιήστε τη συνάρτηση train_test_split(), ώστε να χωρίσετε τα δεδομένα σας σε train-test με ποσοστό 70-30 αντίστοιχα. Θέστε με random_state=42 και την επιλογή stratify=True, ώστε υπάρχει το ίδιο ποσοστό κάθε κλάσης στα train και test αντίστοιχα.

5. Εκπαιδεύστε ταξινομητή Random Forest με random_state=42 στο train set και υπολογίστε την ακρίβεια (accuracy), f1-score, precision και recall, στα train και test sets. Ποιά από τις 2 μετρικές είναι περισσότερο αντιπροσωπευτική στο dataset: Accuracy ή F1? Αιτιολογείστε.
6. Επανεκπαιδεύστε το Random Forest με random_state=42, θέτοντας την παράμετρο *class_weight='balanced'*. Να εξηγήσετε σύμφωνα με το documentation τι κάνει η παράμετρος. Έπειτα, υπολογίστε ξανά τις μετρικές του ερωτήματος 5.
7. Χρησιμοποιήστε τον ταξινομητή του ερωτήματος 6 ώστε να υπολογίσετε τις πιθανότητες που δίνει το μοντέλο για την κλάση κάθε παραδείγματος στο σύνολο x_unknown (y_unknown_proba=model.predict_proba(x_unknown)).
8. Δώστε την ετικέτα (y_unknown_pred) 'red' στα παραδειγμάτων του συνόλου x_unknown για τα οποία ισχύει y_unknown_proba > 0.65. Παρομοίως, δώστε την ετικέτα 'white' στα παραδείγματα του συνόλου για τα οποία ισχύει x_unknown_proba < 0.35.
9. Ενώστε τα σύνολα (x_train, y_train) του 1ου συνόλου με τα δεδομένα που επισημειώθηκαν με ετικέτες στο Ερώτημα 8. Στη συνέχεια, Επανεκπαιδεύστε το Random Forest και υπολογίστε την ακρίβεια στα train/test sets του 1ου συνόλου δεδομένων.
10. Επαναλάβετε τη διαδικασία 6-9 μέχρις ότου να μην υπάρχουν άλλα παραδείγματα στο unknown που να προβλέπονται με υψηλό confidence (δηλαδή έχουν οριακές πιθανότητες).

Μέρος 2 – Αξιολόγηση Μοντέλων

1. Φορτώστε το σύνολο δεδομένων wine-full.csv. Θεωρείτε πως το σκορ (quality) είναι ισορροπημένο (balanced)? Αιτιολογήστε, συμπεριλαμβάνοντας και το αντίστοιχο plot.
2. Συχνά δημιουργείται η ερώτηση: Το κόκκινο η το λευκό κρασί είναι ποιοτικά καλύτερο? Εφαρμόστε κατάλληλη μεθοδολογία και plots, ώστε να αιτιολογήσετε την απάντηση σας.
3. Το γλυκό κρασί έχει μεγαλύτερη ποιότητα από ότι το ξηρό? Αιτιολογήστε την απάντηση σας, λαμβάνοντας υπόψη την ποσότητα υπολειπόμενων σακχάρων.
4. Δημιουργήστε numpy arrays x, y όπου τα inputs είναι όλες οι μεταβλητές εκτός του quality, ενώ y οι μεταβλητή quality. Χωρίστε το σύνολο δεδομένων σε train-test με ποσοστά 90-10 αντίστοιχα. Χρησιμοποιείστε random_state=0.
5. Εκπαιδεύστε DecisionTreeRegressor (random_state=0) στο training set και υπολογίστε το σφάλμα με τη μετρική **MAE** στο test set.
6. Θέλουμε να υπολογίσουμε αν το μοντέλο του ερωτήματος 4 είναι αξιόπιστο. Επαναλάβετε τη διαδικασία 3-4, χρησιμοποιώντας 10 διαφορετικά seed (0-9). Στη συνέχεια, υπολογίσετε το μέσο όρο και την τυπική απόκλιση για τη μετρική **MAE**. Πως μπορούμε να αξιοποιήσουμε αυτές τις τιμές ώστε να είμαστε πιο βέβαιοι για το αναμενόμενο σφάλμα του μοντέλου?
7. Χωρίστε το σύνολο train set σε train-validation με ποσοστά 80-20% αντίστοιχα, χρησιμοποιώντας (random_state=0). Θα πρέπει να έχετε 3 σύνολα δεδομένων (train-validation-test) με ποσοστά 70-20-10 του συνολικού dataset. Θέλουμε να κάνουμε fine-tuning (εύρεση υπερπαραμέτρων) ώστε να βελτιωθεί η ακρίβεια του δέντρου (ερώτημα 3). Δοκιμάστε 15 διαφορετικούς συνδυασμούς με παραμέτρους της επιλογής

σας, διατηρώντας πάντα το random_state=0 και υπολογίστε το MAE στο training και στο validation set για κάθε συνδυασμό.

8. Ποιό από τα δύο σύνολα δεδομένα (training ή validation set) είναι περισσότερο αξιόπιστο για την επιλογή υπερπαραμέτρων? Αιτιολογείστε.
9. Θα δοκιμάσουμε τη μέθοδο cross-validation για την επιλογή παραμέτρων Χρησιμοποιήστε το σύνολο δεδομένων train-test του ερωτήματος 4. Αυτή τη φορά όμως, κάντε fine-tuning με τη βοήθεια cross-validation στο σύνολο train. Για διευκόλυνση, μπορείτε να χρησιμοποιήσετε τη συνάρτηση GridSearchCV της sklearn https://scikit-learn.org/dev/modules/generated/sklearn.model_selection.GridSearchCV.html. Πιο συγκεκριμένα, αντί να υπάρχει σταθερό validation set, η sklearn θα χωρίσει το train set K φορές σε K διαφορετικά σύνολα train-validation και θα υπολογίσει το μέσο σφάλμα. Χρησιμοποιήστε τη μετρική MAE ως scoring, random_state=0 και ορίστε cv=10 ώστε να δημιουργηθούν K=10 folds. Μπορείτε να παραλληλοποιήσετε τη διαδικασία ορίζοντας n_jobs=-1, ώστε να χρησιμοποιήσετε όλους τους πυρήνες του επεξεργαστή.
10. Χρησιμοποιήστε το καλύτερο μοντέλο του ερωτήματος 9 και υπολογίστε τη μετρική MAE στο test set. Ποια από τις μεθοδολογίες που ακολουθήθηκαν (6/7/9) είναι περισσότερο αξιόπιστη? Αιτιολογήστε.
11. Επαναλάβετε την ερώτηση 9, χρησιμοποιώντας πάντα σταθερό max_depth=5. Εμφανίστε το καλύτερο δέντρο (με τη χρήση της plot tree). Επιπλέον, να αναφέρετε τους κανόνες που δημιουργήθηκαν (πχ το κρασί έχει υψηλό σκορ αν περιέχει πολύ ζάχαρη, είναι κόκκινο, κλπ..). Τέλος, να εμφανίσετε ραβδόγραμμα με τη σημαντικότητα κάθε χαρακτηριστικού, ταξινομώντας τα με βάση τη σημαντικότητα τους.

Οδηγίες

- **Χρησιμοποιείστε την πλατφόρμα Google Colab για την υλοποίηση της άσκησης.**
- **Τα plots/πινακάκια να εμφανίζονται επάνω στο Colab. Επίσης, μπορείτε να εισάγετε κελιά για κείμενο και πινακάκια όπως φαίνεται στον παρακάτω σύνδεσμο:** https://colab.research.google.com/notebooks/markdown_guide.ipynb
- **Στο elearning θα υποβάλλετε το link της εργασίας σας στο Github.**