# مشروع نهاية الفصل 6

مقرر Machine Learning — Decision Trees — مشروع أشجار القرار

**الهدف:** تدرّب شجرة، تشوف الـ overfitting، تضبط العمق، وتقرا الشجرة النهائية.

**الداتا:** sklearn.datasets.load_wine (١٧٨ عينة، ١٣ خاصية، ٣ أنواع). مصدرها UCI. مدمجة في scikit-learn.

## الأسئلة العملية

### 1 جهّز (Prepare)

- حمّل wine وقسّمها ٧٠/٣٠ بـ stratify و random_state=0.

### 2 شجرة حرة (Free tree)

- درّب DecisionTreeClassifier بدون حدود واطبع دقة الـ train والـ test.

### 3 اضبط العمق (Tune the depth)

- جرّب max_depth من 1 لـ 6 بـ cross_val_score (cv=5) على الـ training.

### 4 النهائي (Final)

- درّب بأحسن عمق واطبع دقة الـ test.

### 5 اقرا الشجرة (Read the tree)

- اطبع export_text واكتب أول سؤال بالكلمات.

## بونس

- بونس: جرّب min_samples_leaf بدل max_depth.
- بونس: قارن مع RandomForestClassifier.

## أسئلة تفكير (اكتب إجابة في سطرين)

1. ليه الشجرة الحرة بتعمل overfitting؟
2. ليه الأشجار ما بتحتاجش scaling؟

## ماذا تسلّم؟

- ملف الكود (أو notebook) بيشتغل من الأول للآخر.
- فقرة قصيرة (٥ أسطر) بتلخّص النتيجة وإيه اللي اتعلمته.

# الحل النموذجي

جرّب بنفسك الأول، وبعدين قارن. الأرقام الحقيقية في النتيجة تحت.

## الكود

```
# dataset ships inside scikit-learn (no download needed)
from sklearn.datasets import load_wine
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.tree import DecisionTreeClassifier, export_text

data = load_wine()
X_tr, X_te, y_tr, y_te = train_test_split(data.data, data.target, test_size=0.3,
random_state=0, stratify=data.target)
free = DecisionTreeClassifier(random_state=0).fit(X_tr, y_tr)
print('free tree: train', round(free.score(X_tr, y_tr), 3), 'test',
round(free.score(X_te, y_te), 3))
scores = {d: cross_val_score(DecisionTreeClassifier(max_depth=d, random_state=0), X_tr,
y_tr, cv=5).mean() for d in range(1, 7)}
for d, s in scores.items():
    print('depth', d, round(s, 3))
best = max(scores, key=scores.get)
tree = DecisionTreeClassifier(max_depth=best, random_state=0).fit(X_tr, y_tr)
print('best depth', best, '| test', round(tree.score(X_te, y_te), 3))
print(export_text(tree, feature_names=list(data.feature_names)))
```

## النتيجة

```
free tree: train 1.0 test 0.944
depth 1 0.686
depth 2 0.831
depth 3 0.863
depth 4 0.872
depth 5 0.872
depth 6 0.872
best depth 4 | test 0.944
|--- proline <= 765.00
|   |--- flavanoids <= 1.40
|   |   |--- color_intensity <= 3.72
|   |   |   |--- class: 1
|   |   |--- color_intensity >  3.72
|   |   |   |--- class: 2
|   |--- flavanoids >  1.40
|   |   |--- alcohol <= 13.17
|   |   |   |--- class: 1
|   |   |--- alcohol >  13.17
|   |   |   |--- proline <= 655.00
|   |   |   |   |--- class: 1
```

```
|   |   |   |--- proline >  655.00
|   |   |   |   |--- class: 0
|--- proline >  765.00
|   |--- flavanoids <= 2.17
|   |   |--- nonflavanoid_phenols <= 0.28
|   |   |   |--- class: 1
|   |   |--- nonflavanoid_phenols >  0.28
|   |   |   |--- class: 2
|   |--- flavanoids >  2.17
|   |   |--- alcalinity_of_ash <= 27.50
|   |   |   |--- class: 0
|   |   |--- alcalinity_of_ash >  27.50
|   |   |   |--- class: 1
```

## ملاحظات على الإجابات

- الشجرة الحرة بتفضل تقسّم لحد ما تحفظ الـ training، فدقتها عليه ١٠٠٪ وعلى الـ test أقل.
- اختيار العمق بالـ cross-validation بيحمينا من اختيار بناءً على الـ test.
- الأشجار بتقارن feature واحدة بحد في كل مرة، فمقياسها ما بيأثرش.

Credits: topic follows the matching chapter of Aurélien Géron, Hands-On Machine Learning (3rd ed.). Questions, code and explanations are original.