مشروع نهاية الفصل 9

مقرر Machine Learning — Unsupervised Learning — مشروع الـ Unsupervised Learning

الهدف: تجمّع الداتا بـ KMeans و GMM و DBSCAN من غير labels، وتقيّم المجموعات بـ silhouette وبمقارنتها بالأنواع الحقيقية (ARI).

الداتا: sklearn.datasets.load_wine (١٧٨ عينة، ١٣ خاصية). مصدرها UCI. مدمجة في scikit-learn. بنتجاهل الـ labels وقت التدريب.

الأسئلة العملية

1 جهّز (Prepare)

- حمّل wine واعمل StandardScaler. ما تستخدمش الـ labels في التدريب.

2 KMeans (KMeans)

- جرّب k من 2 لـ 6 واطبع silhouette.

3 قارن بالحقيقة (Compare with the truth)

- خد k=3 واحسب adjusted_rand_score مع الأنواع الحقيقية.

4 GMM (GMM)

- جرّب GaussianMixture بـ 1 لـ 6 مكوّنات واختار بالـ BIC.

5 شذوذ (Anomalies)

- درّب IsolationForest(contamination=0.05) واطبع كام عينة اتعلّمت شاذة.

بونس

- بونس: جرّب DBSCAN بـ eps مختلفة على الداتا بعد PCA لـ ٢ بُعد.
- بونس: قارن KMeans و GMM بالـ ARI.

أسئلة تفكير (اكتب إجابة في سطرين)

1. ليه silhouette مش دايماً بيرجّع عدد الأنواع الحقيقي؟
2. إمتى GMM أحسن من KMeans؟

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.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.mixture import GaussianMixture
from sklearn.ensemble import IsolationForest
from sklearn.metrics import silhouette_score, adjusted_rand_score

X, y = load_wine(return_X_y=True)
Xs = StandardScaler().fit_transform(X)
for k in range(2, 7):
    lab = KMeans(n_clusters=k, n_init=10, random_state=0).fit_predict(Xs)
    print('k =', k, 'silhouette', round(silhouette_score(Xs, lab), 3))
lab3 = KMeans(n_clusters=3, n_init=10, random_state=0).fit_predict(Xs)
print('ARI of k=3 vs real species:', round(adjusted_rand_score(y, lab3), 3))
bic = {k: GaussianMixture(n_components=k, n_init=3, random_state=0).fit(Xs).bic(Xs) for
k in range(1, 7)}
print('BIC:', {k: round(v) for k, v in bic.items()})
print('best by BIC:', min(bic, key=bic.get))
flag = IsolationForest(contamination=0.05, random_state=0).fit(Xs).predict(Xs)
print('flagged as anomalies:', int((flag == -1).sum()))
```

النتيجة

```
k = 2 silhouette 0.268
k = 3 silhouette 0.285
k = 4 silhouette 0.246
k = 5 silhouette 0.203
k = 6 silhouette 0.196
ARI of k=3 vs real species: 0.897
BIC: {1: 5741, 2: 5608, 3: 5842, 4: 5670, 5: 5916, 6: 6019}
best by BIC: 2
flagged as anomalies: 9
```

ملاحظات على الإجابات

- الـ silhouette بيقيس شكل المجموعات، مش الأنواع الحقيقية، فممكن يفضّل عدد مختلف.
- الـ GMM أحسن لما المجموعات مستطيلة أو مختلفة الحجم، وبيدّي احتمالات.

- الـ ARI بيقارن المجموعات بالأنواع الحقيقية؛ ١ يعني تطابق تام وقريب من ٠ يعني عشوائي.

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