Clase 084 — Bagging y pasting

Parte: 1 — Machine Learning Clásico · Fuente: Géron, cap. 7. Duración estimada: 55 min.

Clase 084 — Bagging y pasting

> *Parte: 1 — Machine Learning Clásico · Fuente: Géron, cap. 7. Duración estimada: 55 min.*

Objetivo

Que el alumno entrene ensembles entrenando el mismo algoritmo sobre distintos subsets del training set —bagging (con reemplazo) y pasting (sin reemplazo)— y evalúe sin held-out usando out-of-bag (OOB). Cierre con sampling de features (random patches y random subspaces) como puente conceptual a Random Forests.

Resultados de aprendizaje

Al finalizar la clase, el alumno podrá:

1. Distinguir bagging vs pasting y justificar cuándo elegir uno u otro.
2. Entrenar un BaggingClassifier de scikit-learn con un base estimator y n_estimators razonable.
3. Usar oob_score=True para estimar el error de generalización sin tocar el test set.
4. Aplicar sampling de features (max_features < 1.0, bootstrap_features=True) para random patches / random subspaces.
5. Comparar bias/variance del ensemble vs un único árbol, en accuracy y en frontera de decisión.

Temas

#	Tema	Por qué importa
1	Bagging (bootstrap aggregating)	Reduce varianza promediando predictores en
2	Pasting	Igual idea pero sin reemplazo — útil cuand
3	BaggingClassifier API	estimator, n_estimators, max_samples, boot
4	OOB evaluation	Cada predictor ve ~63% de las instancias;
5	Random patches	Sampling de instancias y features.
6	Random subspaces	Sampling solo de features (bootstrap=False
7	Paralelización con n_jobs=-1	Los predictores son independientes — escal

Definiciones y características

Bagging (bootstrap aggregating)

: Entrenar el mismo algoritmo sobre múltiples muestras con reemplazo del training set y agregar predicciones por voto mayoritario (clasificación) o promedio (regresión). Reduce varianza sin aumentar bias.

Pasting

: Igual que bagging pero el sampling es sin reemplazo. Cada instancia aparece a lo sumo una vez por predictor. Bagging suele ganar porque introduce más diversidad.

Bootstrap

: Sampling con reemplazo del mismo tamaño que el original. En promedio cubre ~63.2% de las instancias

únicas; el resto son repetidas.

Out-of-bag (OOB) score

: Para cada predictor, las instancias no muestreadas (~37%) actúan como validation set. Promediando OOB scores se obtiene una estimación del error de generalización sin separar test set. Activado con oob_score=True.

BaggingClassifier

: Meta-estimador de sklearn. Params clave: estimator (base learner), n_estimators (cuántos), max_samples (tamaño de cada muestra, default 1.0 = igual al training set), bootstrap=True (bagging) / False (pasting), oob_score, n_jobs.

Random patches

: Sampling simultáneo de instancias y features. bootstrap=True, bootstrap_features=True, max_features<1.0. Útil cuando hay muchas features (imágenes, alta dimensionalidad).

Random subspaces

: Sampling solo de features, manteniendo todas las instancias. bootstrap=False, max_samples=1.0, bootstrap_features=True, max_features<1.0.

n_estimators

: Número de predictores del ensemble. Más estimators = menos varianza, más cómputo. Típico: 100–500. La curva de mejora se aplana rápido.

Dataset / recursos

make_moons(n_samples=500, noise=0.30) de sklearn — frontera no lineal, ideal para visualizar el efecto smoothing del ensemble vs un árbol único.

Ejercicios

1. Bagging vs árbol único. Entrená un DecisionTreeClassifier y un BaggingClassifier(DecisionTreeClassifier(), n_estimators=500, max_samples=100, bootstrap=True) sobre make_moons. Compará accuracy en test.

2. Bagging vs pasting. Repetí (1) con bootstrap=False. Reportá diferencia de accuracy y discutí.

3. OOB. Entrená con oob_score=True, bootstrap=True. Imprimí bag.oob_score_ y compará con accuracy en test — deberían ser parecidos.

4. Curva de n_estimators. Variá n_estimators ∈ {1, 10, 50, 100, 500, 1000}. Plotteá accuracy test vs n_estimators.

5. Random subspaces. Sobre load_digits() (64 features), entrená con bootstrap=False, max_samples=1.0, bootstrap_features=True, max_features=0.5. Compará con bagging clásico.

Homework verificable

Notebook que: (a) cargue make_moons(500, noise=0.30), split 80/20; (b) entrene árbol único y BaggingClassifier(n_estimators=500, max_samples=100, oob_score=True, n_jobs=-1); (c) reporte accuracy test de ambos y oob_score_ del bag; (d) grafique las dos fronteras de decisión lado a lado; (e) repita con bootstrap=False (pasting) y compare en 1 párrafo.

Criterio de aceptación: Bagging supera al árbol único en test. oob_score_ ≈ accuracy test (diferencia < 0.05). La frontera del bag es visiblemente más suave.

Errores comunes

Síntoma / mensaje	Causa y cómo arreglar
ValueError: Out of bag estimation only ava	Pediste oob_score=True con bootstrap=False
n_estimators=10 y el ensemble no mejora al	Pocos predictores — la varianza no se prom
BaggingClassifier lento con árboles profun	Cada predictor crece sin podar. Fix: n_job
Confundir max_samples con max_features	max_samples sortea filas, max_features sor
OOB score muy distinto al test score	Dataset chico (OOB tiene alta varianza) o

Preguntas frecuentes

¿Bagging o pasting?

Bagging casi siempre. El reemplazo introduce más diversidad entre predictores, que es exactamente lo que reduce la varianza del ensemble. Pasting puede ganar marginalmente si el dataset es enorme y querés evitar repetir instancias, pero la diferencia es chica. Como bonus, bagging te da OOB gratis.

¿Cuántos n_estimators uso?

Empezá con 100. Subí a 500 si tenés cómputo. Más allá de eso, retornos decrecientes — la mejora marginal por estimator se aplana.

¿OOB reemplaza al test set?

Reemplaza al validation set durante tuning. Igual conviene reservar un test set final para reportar el número honesto al stakeholder.

¿Bagging sirve con cualquier base estimator?

Sí, pero brilla con learners de alta varianza (árboles profundos sin podar). Con learners de bajo varianza (regresión logística, naive Bayes) el bagging casi no mueve la aguja.

¿Esto es lo mismo que Random Forest?

Casi. Random Forest = bagging de árboles + sampling de features en cada split (no por árbol). Lo vemos en la clase siguiente.

Referencias

- Géron, Hands-On ML, cap. 7 § "Bagging and Pasting", "Out-of-Bag Evaluation", "Random Patches and Random Subspaces".
- sklearn BaggingClassifier
- sklearn user guide — Bagging
- Breiman, L. (1996). Bagging predictors. Machine Learning, 24(2).

Siguiente clase

Clase 085 — Random Forests y Extra Trees

Apéndice: notebook (primer bloque)

Primera celda ejecutable del notebook de la clase.

```
# Imports y configuración inicial
```

Archivos complementarios

- notebook.ipynb