Clase 053 — Validación temporal: TimeSeriesSplit, walk-forward, blocking

Parte: 1 — Machine Learning Clásico · Fuente: Bergmeir & Benítez (2012) + sklearn TimeSeriesSplit docs. Duración estimada: 70 min.

Clase 053 — Validación temporal: TimeSeriesSplit, walk-forward, blocking

> *Parte: 1 — Machine Learning Clásico · Fuente: Bergmeir & Benítez (2012) + sklearn TimeSeriesSplit docs. Duración estimada: 70 min.*

Objetivo

Aplicar validación correcta para series temporales — donde KFold y train_test_split aleatorio causan leakage del futuro al pasado y métricas infladas. Cubrir TimeSeriesSplit, walk-forward validation (rolling y expanding), blocking para datos con dependencias intra-cluster, purged + embargoed CV (López de Prado, finanzas).

Resultados de aprendizaje

Al finalizar, el estudiante podrá:

- Aplicar sklearn.model_selection.TimeSeriesSplit(n_splits, max_train_size, test_size, gap).
- Implementar walk-forward rolling (ventana fija) y expanding (acumulativo).
- Detectar leakage cuando KFold se usa sobre datos temporales.
- Aplicar purged + embargoed K-Fold para evitar leakage por feature engineering con lags.
- Reportar métricas multi-fold con dispersión (no solo promedio).

Temas

- ¿Por qué KFold falla en series? Aleatorización mezcla pasado y futuro.
- TimeSeriesSplit: split secuencial, train siempre antes que test.
- Expanding vs rolling window.
- gap (embargo) para target leakage con lags.
- Purged CV (López de Prado): elimina overlap entre train y test.
- CombinatorialPurgedKFold para backtesting.

Definiciones y características

- TimeSeriesSplit(n_splits=5): divide la serie en N+1 chunks secuenciales; itera (train=primer N, test=último).
- Walk-forward expanding: train crece, test va avanzando.
- Walk-forward rolling: train tiene tamaño fijo, ventana se mueve.
- Embargo / gap: número de samples ignoradas entre train y test — evita leakage por features con lags.
- Purged CV: elimina del train cualquier sample cuyo target overlapped con el test.

Dataset / recursos

- Serie temporal sintética o seaborn.load_dataset('flights').
- Librerías: scikit-learn, pandas, mlxtend (alternativa con más options).

Ejercicios

1. TSSplit vs KFold leak: con serie sintética con tendencia, comparar score CV con KFold aleatorio vs TimeSeriesSplit. KFold infla.
2. Walk-forward expanding: tscv = TimeSeriesSplit(n_splits=5). Iterar y reportar score por fold.
3. Rolling window: con max_train_size=100, simular walk-forward de window fijo.
4. gap: con feature y_t-1 (target lag), aplicar gap=1 para evitar que test "vea" su propio target.
5. Score con dispersión: reportar mean ± std de RMSE por fold, no solo mean.

Homework verificable

Forecasting con XGBoost en serie de retail:

1. Feature engineering: lags 1, 7, 30 + rolling mean.
2. CV con TimeSeriesSplit(5, gap=30).
3. Reportar RMSE por fold + global.
4. Comparar contra KFold ingenuo — mostrar inflación.

Criterio de aceptación: KFold subestima RMSE en ≥ 30 %; TimeSeriesSplit da estimación realista que se sostiene en test final.

Errores comunes

Síntoma / mensaje	Causa y cómo arreglar
Métrica CV mucho mejor que producción	KFold sobre temporal. Fix: TimeSeriesSplit
Feature y_t-1 y CV sin gap	Target leak. Fix: gap = max_lag.
n_splits muy alto con serie corta	Folds muy chicos. Fix: 3-5 folds.
Reportar solo mean	Variabilidad oculta. Fix: mean ± std.
Rolling con max_train_size mal calibrado	Train muy chico → ruido. Fix: ≥ 1 ciclo es

Preguntas frecuentes

Expanding o rolling?

Expanding usa toda la historia (más data); rolling refleja "olvido" si crees que la dinámica cambia. Probá ambos.

TimeSeriesSplit con feature engineering sobre toda la serie?

Leak. Fix: features con rolling/lag calculadas con min_periods=window para no usar futuro.

Para crypto / trading?

Purged + embargoed (López de Prado Advances in Financial Machine Learning).

Hyperparameter tuning con TimeSeriesSplit?

GridSearchCV con cv=TimeSeriesSplit(5). Funciona idéntico.

Nested CV?

Recomendado para tuning + evaluation honesta. Outer TSSplit para reportar, inner para tunear.

Referencias

- Bergmeir & Benítez (2012), On the use of cross-validation for time series predictor evaluation.
- López de Prado (2018), Advances in Financial Machine Learning, cap. 7.
- sklearn.model_selection.TimeSeriesSplit.
- Hyndman & Athanasopoulos, Forecasting: Principles and Practice.

Siguiente clase

Clase 054 — Proyecto end-to-end: visión, datos, exploración, preparación

Apéndice: notebook (primer bloque)

Serie sintética (trend + seasonality + noise). Comparamos KFold (leak) vs TimeSeriesSplit vs walk-forward expanding/rolling. Requiere: numpy, pandas, scikit-learn, matplotlib.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import Ridge
from sklearn.model_selection import KFold, TimeSeriesSplit, cross_val_score
from sklearn.metrics import mean_absolute_error

rng = np.random.default_rng(42)
np.random.seed(42)

n = 1000
t = np.arange(n)
trend = 0.02 * t
season = 3  np.sin(2  np.pi * t / 50)
noise = rng.normal(0, 0.5, n)
y = trend + season + noise
print('serie:', y.shape, 'min', y.min().round(2), 'max', y.max().round(2))
```

Archivos complementarios

- notebook.ipynb