Wprowadzenie do analizy danych i uczenia maszynowego

Uczenie maszynowe III — CV, HPO

Franciszek Saliński

Koło Naukowe Data Science, Wydział MiNI PW

24 stycznia 2026

Agenda

1. Zbiór walidacyjny i jakie problemy rozwiązuje
2. Crossvalidation
3. Czym jest optymalizacja hiperparametrów?
4. Najpopularniejsze algorytmy do HPO
5. Random Search
6. Grid Search
7. Dlaczego interpretujemy modele, krótko o SHAP

Zbiory train / val / test (przypomnienie)

- Zbiór treningowy (train) – uczymy parametry modelu.
- Zbiór walidacyjny (val) – wybór modelu / hiperparametrów.
- Zbiór testowy (test) – końcowa, niezależna ocena.
- Test powinien być „niewidzialny” podczas trenowania.

Zbiór walidacyjny – co nam daje?

- Mierzymy jakość modelu na danych, których model nie widział w uczeniu.
- Dobieramy hiperparametry bez leaku ze zbioru testowego.
- Unikamy wtórnemu data leakage (które często prowadzi do overfitingu/dopasowania hiperparametrów pod zbiór testowy).

Typowy workflow

1. Dzielimy dane na train/val/test.
2. Na train uczymy model dla wielu ustawień hiperparametrów.
3. Na val wybieramy najlepsze ustawienie.
4. **Raz** raportujemy wynik na teście.

Kiedy walidacja może nie działać?

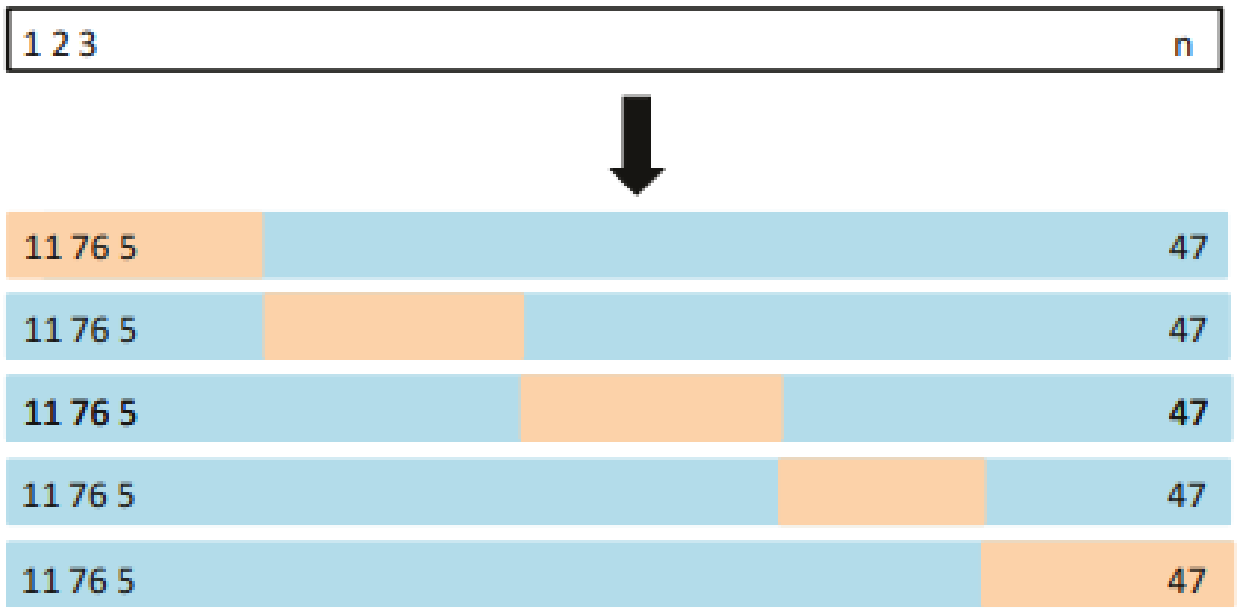
- Gdy mamy mało danych.
- Gdy dane są mocno niestacjonarne (np. czas/przestrzeń) i robimy losowy split.

Cross-Validation – idea

- Zamiast jednego podziału train/val robimy **wiele różnych podziałów**.
- Każdy punkt danych jest raz w zbiorze walidacyjnym.
- Dostajemy stabilniejszą estymację jakości modelu.

K-fold CV (schemat)

- Dzielimy dane na K „foldów” (części).
- Dla $k = 1, \ldots, K$:
  - fold k = walidacja, reszta = trening,
  - uczymy model na treningu i liczymy metrykę na walidacji.
- Wynik końcowy to średnia (często liczymy też odchylenie).



Źródło: Figure 5.5, *An Introduction to Statistical Learning* (str. 207).

Inne warianty CV

- **Stratified K-fold** – zachowuje proporcje klas (klasyfikacja).
- **TimeSeries split** – walidacja uporządkowana w czasie.
- **LOOCV (Leave-One-Out CV)** – $K = n$.

Parametry vs hiperparametry w modelu

- **Parametry** – uczone z danych (np. wagi β w regresji).
- **Hiperparametry** – ustawiamy przed uczeniem (np. λ w Ridge/Lasso, głębokość drzewa, learning rate, liczba drzew).

Czym jest HPO?

- HPO = Hyperparameter optimization.
- Jest to proces doboru hiperparametrów dla modelu w danym zadaniu.
- Hiperparametry decydują o złożoności modelu i pozwalają kontrolować overfitting/underfitting

HPO – przykład

- Regresja Ridge: dobór λ.
- Drzewa decyzyjne: głębokość drzewa.
- Gradient Boosting: learning rate, liczba drzew.

HPO jako problem optymalizacji

- Celem jest znalezienie hiperparametrów, które minimalizują błąd walidacyjny.
- Każda ewaluacja modelu = proces treningu + walidacja (np. Cross-Validation).
- Problem jest nieliniowy, kosztowny obliczeniowo i trudny do rozwiązania analitycznie.

$$\lambda^\star = \arg\min_{\lambda \in \Lambda} \mathcal{L}_{val}\left(A(\lambda; D_{train}), D_{val}\right)$$

- $\lambda^\star$ — optymalne hiperparametry.
- Λ — przestrzeń przeszukiwania hiperparametrów.
- $\mathcal{L}_{val}$ — funkcja straty na zbiorze walidacyjnym.
- $A(\lambda; D_{train})$ — model uczony na zbiorze treningowym z hiperparametrami λ.

Najpopularniejsze algorytmy do HPO

- **Grid Search** – przegląd siatki parametrów.
- **Random Search** – losowanie konfiguracji parametrów.
- **Bayesian Optimization** – inteligentne próbkowanie z modelowaniem rozkładu.

Random Search – idea

- Losujemy hiperparametry z wcześniej zdefiniowanej przestrzeni.
- Każda próba jest niezależna.
- Po określonej liczbie prób wybieramy najlepszy wynik.

Grid Search – idea

- Tworzymy dyskretną siatkę wartości dla każdego hiperparametru.
- Sprawdzamy wszystkie kombinacje.
- Gwarantuje pełne przeszukanie siatki, ale jest czasochłonne.

Grid Search – ograniczenia

- Eksplozja kombinacji – rośnie liczba prób przy wzroście liczby hiperparametrów.
- Skaluje się słabo w przypadku dużej liczby parametrów.
- Może być mniej efektywne od Random Search w dużych przestrzeniach.

Po co interpretowalność modeli?

- Zrozumienie, dlaczego model podejmuje konkretne decyzje.
- Pomaga wykrywać błędy i unikać data leakage.
- Wzrost zaufania do modelu, szczególnie w branżach krytycznych (medycyna, finanse).

Interpretowalność – dwa poziomy

- **Globalna interpretowalność** – jakie cechy mają największy wpływ na model.
- **Lokalna interpretowalność** – dlaczego model podjął taką decyzję dla konkretnej obserwacji.

SHAP – idea

- SHAP to metoda oparta na teorii gier (wartości Shapleya).
- Dzieli wynik predykcji na wkłady cech.
- Każda cecha ma przypisany wpływ na wynik (pozytywny lub negatywny).

SHAP – intuicja

- Punkt odniesienia: predykcja bazowa (np. średnia).
- Każda cecha zmienia predykcję, o określoną wartość.
- Suma wkładów cech = ostateczna predykcja modelu.

Podsumowanie

- Cross-Validation i HPO pozwalają rzetelnie oceniać i optymalizować modele.
- Random Search zwykle najlepszy jako baseline dla HPO.
- Interpretowalność modeli (np. SHAP) zwiększa zaufanie i ułatwia diagnozowanie błędów.