Пошук оптимальних гіперпараметрів. Перехресна валідація

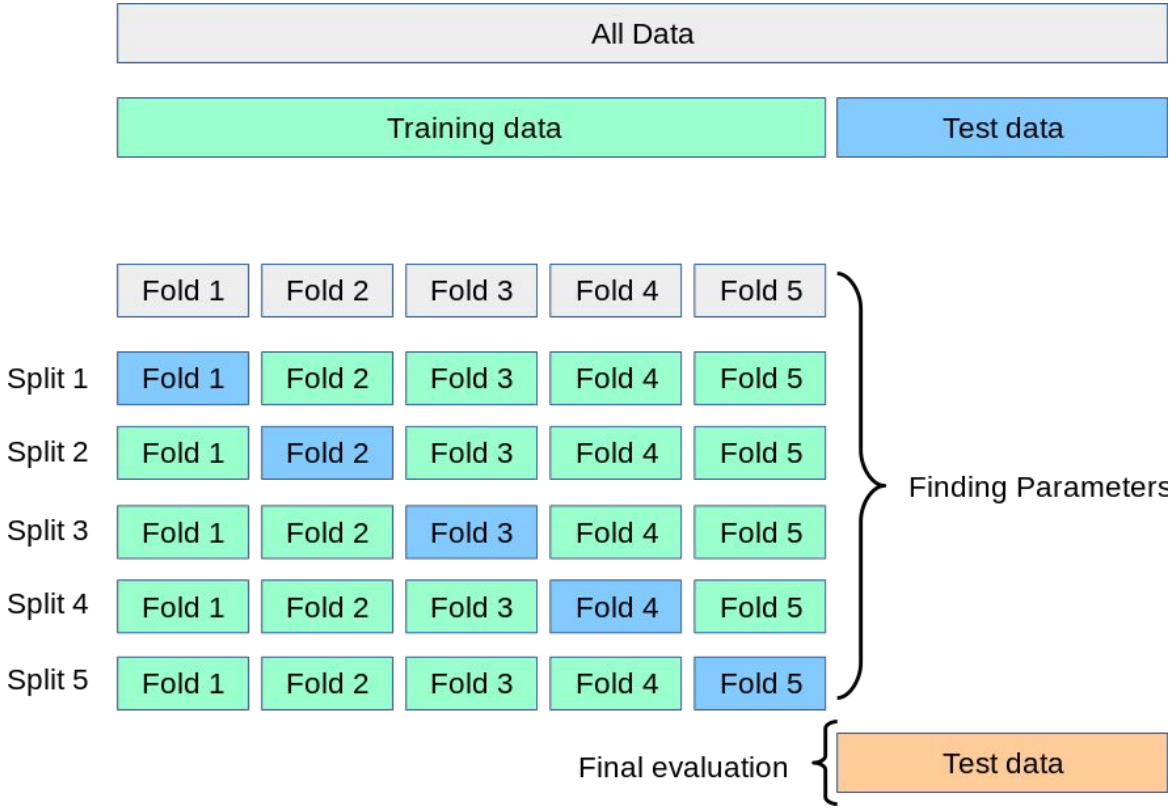
Crossvalidation / Крос-валідація

— метод оцінки точності моделі на незалежних даних. Дозволяє точніше оцінити якість моделі.

k-fold крос-валідація виконується наступним чином

- Розбиваємо дані на k частин.
- Навчаємо модель на k−1 частинах даних, а залишок використовуємо для тестування.
- Повторюємо процедуру k разів.
- Знаходимо середнє та стандартне відхилення метрики якості моделі після k навчань.
- Кожна з k частин даних використовується один раз для тестування.
- Популярні значення k: 3 (коли модель тренується дуже довго), 5, 10

K-fold крос-валідація візуально



Leave-one-out CV

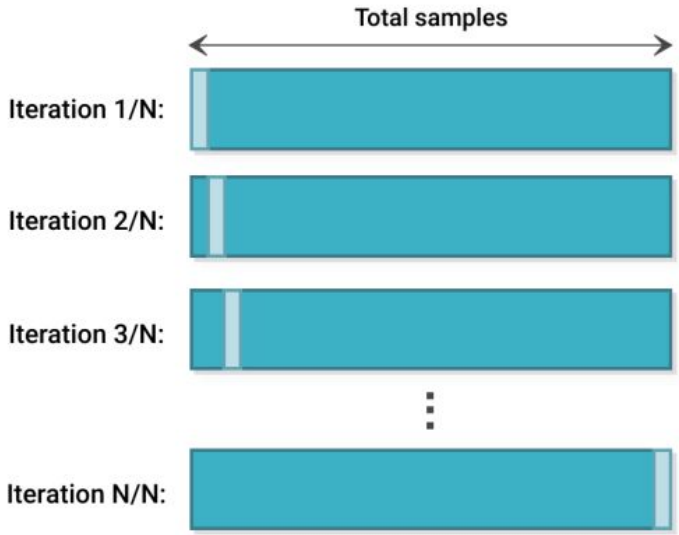
крайній випадок k-Fold CV, коли k рівне n, де n — кількість вибірок в наборі даних. Такий випадок k-Fold еквівалентний методу виключення одного.

Плюси

- ми максимально утилізуємо дані для тренування

Мінуси

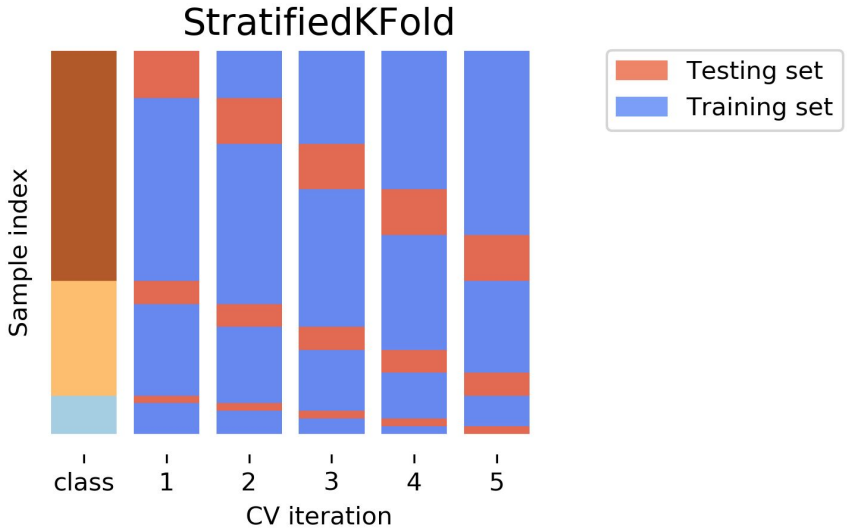
- нам потрібно тренувати кількість моделей, рівну кількості екземплярів у даних



Stratified k-Fold

Використовується в разі незбалансованих з точки зору класів цільової змінної.

Також може використовуватися для рівномірного розбиття з точки зору цільової змінної даних на k фолдів у задачі регресії. Для використання цільову змінну треба перед тим розбити на біни (як для гістограми).



Інші методи перехресної перевірки

- Ще кілька популярних методів перехресної перевірки
  https://neptune.ai/blog/cross-validation-in-machine-learning-how-to-do-it-right

- Solving 9 Common Cross-Validation Mistakes
  https://medium.com/@jan_marcel_kezmann/solving-9-common-cross-validation-mistakes-ac8a6a6944e7

Пошук гіперпараметрів

Для поліпшення якості моделі часто потрібно знайти оптимальні гіперпараметри. Гіперпараметри ми зазвичай шукаємо за допомогою перехресної перевірки.

Як знаходити оптимальні гіперпараметри за допомогою sklearn

Ми можемо оптимізувати будь-які параметри оцінювача в sklearn, які повертає метод estimator.get_params().

Пошук складається з:

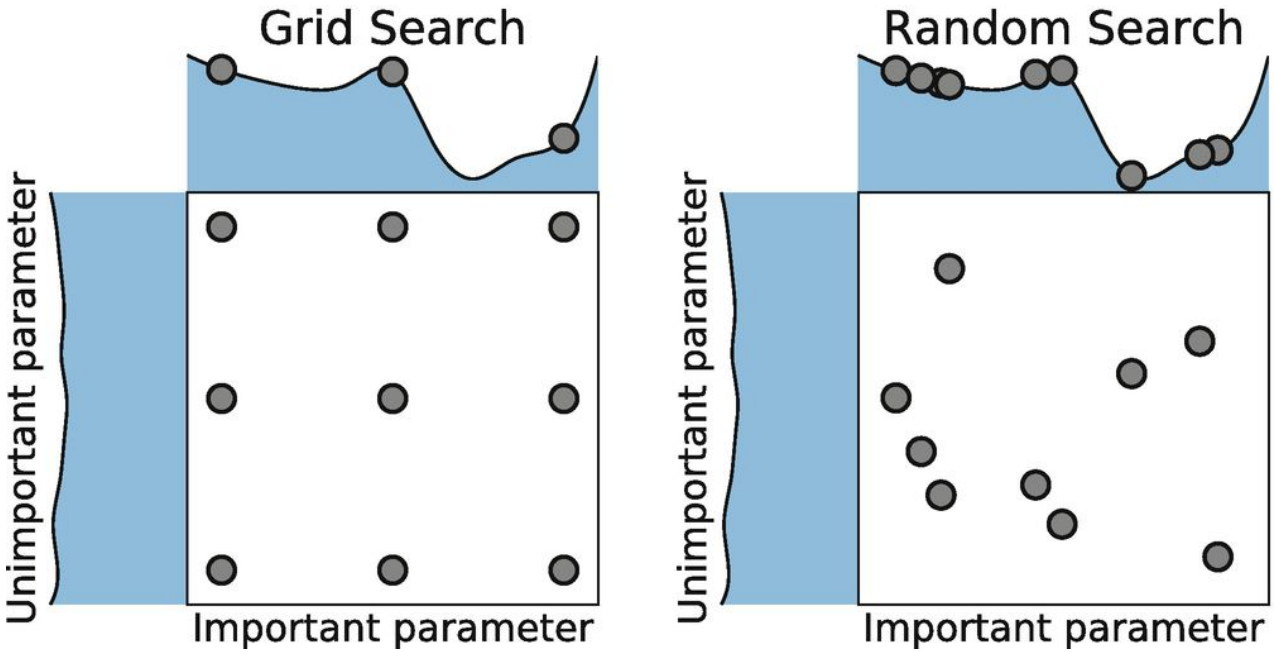
- ➔ оцінювача (регресор або класифікатор, такий як sklearn.linear_model.ElasticNet);
- ➔ простору параметрів;
- ➔ методу пошуку або вибірки кандидатів;
- ➔ схеми перехресної перевірки;
- ➔ функції оцінки якості моделі.

Як шукати оптимальні гіперпараметри за допомогою sklearn

У бібліотеці scikit-learn існують два загальні підходи до пошуку параметрів:

- **GridSearchCV** - для заданих значень вичерпно розглядає всі комбінації параметрів;
- **RandomizedSearchCV** - реалізує випадковий пошук по параметрам, де кожен параметр обирається з розподілу по можливим значенням параметрів.

Grid Search vs Random Search



Grid Search vs Random Search

Random Search має дві основні переваги перед Grid Search:

- "Бюджет" (кількість навчань моделі) може бути вибраний незалежно від кількості параметрів та можливих значень.

- Додавання параметрів, які не впливають на продуктивність, не знижує ефективність пошуку (тому що все одно ми знаходимо метрику якості для важливого параметра).