Оценка обобщающей ошибки

Hold-out, k-fold, OOB — lean demo

Уртемеев С.А.

5030102/20101

Цель и контекст

Что измеряем
Обобщающая ошибка — качество модели на новых данных из того же распределения, которых она не видела при обучении.

Почему это не видно напрямую
Истинный риск (ожидание потерь на всей популяции) недоступен: у нас только конечная выборка. Тренировочная ошибка обычно занижена из-за подгонки — по ней нельзя судить о будущем качестве.

Как оцениваем
Используем протоколы, имитирующие появление новых данных: **hold-out**, **k-fold**, **OOB**. Они дают не одно число, а оценку со средним и разбросом при соблюдении правил: без утечек, единый `Pipeline`, стратификация (для классификации), фиксированные сиды.

Протокол без утечек

- **Разделение данных** на обучающую и оценочную части выполняется *до* любых операций.
- Классификация: используется **стратифицированное разбиение**; при зависимостях (пациенты, сессии) — **групповое разбиение**.
- Все операции, параметры которых настраиваются по данным (*масштабирование, отбор признаков, калибровка и пр.*), обучаются *только на обучающей части*; затем *неизменным контуром* применяются к оценочной части.
- **Фиксируем источники случайности** (перестановки, инициализации) для воспроизводимости.
- Сравниваем модели в **одинаковом протоколе**: один и тот же способ разбиения и одни и те же метрики.

Протокол без утечек — схема

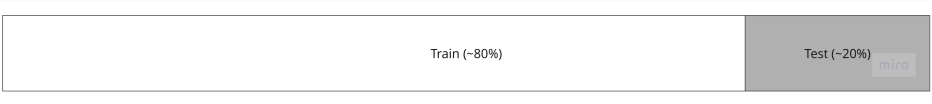
Воспроизводимость: фиксируем источники случайности (перестановки, инициализации).

k-fold: повторяем процесс k раз и усредняем; hold-out: один сплит; OOB: оценка по out-of-bag предсказаниям.

Hold-out (Train/Test)

- Идея: одна разделённая выборка (например, 80/20).
- Плюсы: просто и быстро.
- Минусы: высокая дисперсия оценки на малых данных.
- Опция: повторные сплиты (repeated hold-out) или доверительные интервалы через бутстрап (bootstrap-CI).

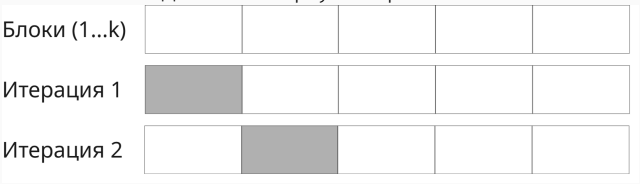
Hold-out: один сплит Train/Test



Отделяем независимую оценочную часть и считаем метрики только на ней

k-fold кросс-валидация

- Идея: усреднение по k разбиениям ($k = 5/10$).
- Для классификации — стратифицированный разбиение (сохраняем доли классов).
- Стабильнее оценки (меньше дисперсия), но дороже по времени.
- При тюнинге гиперпараметров — **nested CV**: внешняя петля для оценки, внутренняя — для подбора.



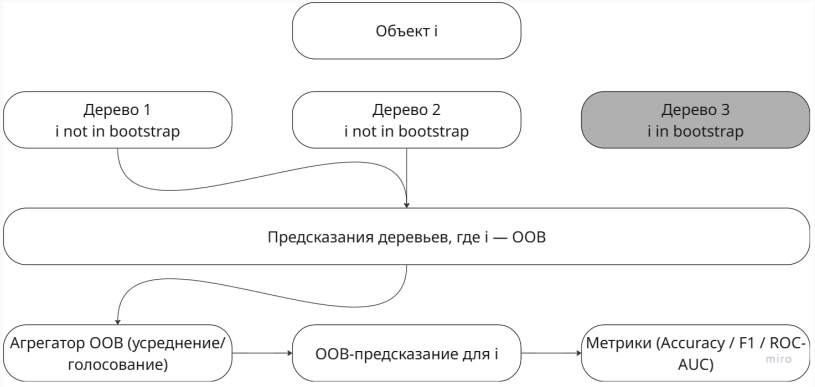
На каждой итерации: 1 блок - валидация, остальные - обучение;
Затем усредняем метрики по всем k

OOB (out-of-bag)

- Для бэггинг-моделей (случайный лес, bagging): обучаем базовые модели на бутстрап-выборках.
- Для каждого объекта формируем OOB-предсказание, *усредняя только те базовые модели, где объект не обучался*.
- По OOB-предсказаниям считаем метрики — на практике близко к k-fold без явного разбиения.
- Ограничения: требуется бутстрап; неприменимо к одиночным моделям без повторной выборки; при тюнинге по обучающей части OOB может быть слегка оптимистичен.

OOB (out-of-bag) - схема

OOB (out-of-bag) для бэггинга: bootstrap = true обязателен



Для каждого объекта i агрегируем предсказания только тех базовых моделей где i не обучался. Доля деревьев, где объект OOB $\approx 37\%$.

Сравнение схем (матрица выбора)

Объем данных \ Бюджет времени	Низкий	Средний	Высокий
Малые	k-fold (5), стратифицированный	k-fold (5–10), ↓ дисперсия	nested CV при тюнинге; иначе k-fold (10)»
Средние	Hold-out (80/20) или repeated hold-out	k-fold (5) или OOB для бэггинга	k-fold (10) или nested CV при тюнинге
Большие	Hold-out (90/10 или 95/5); OOB для бэггинга	OOB (бэггинг) или Hold-out + повторы	Subsample + k-fold (5) или OOB (дёшево)

Метрики для классификации — по назначению

- ROC-AUC — качество ранжирования без порога; обычно стабилен при дисбалансе.
- PR-AUC — фокус на позитивном классе; выбирать при редких позитивных событиях.
- F1 — одно число при фиксированном пороге; балансирует precision/recall.
- Accuracy — просто, но обманчиво при дисбалансе; годится при близких долях классов и равной цене ошибок.
- LogLoss — качество вероятностей (калибровка); штрафует «слишком уверенные» ошибки.

Метрики — дисбаланс и корректный отчёт

- Дисбаланс: **стратифицированные** разбиения; метрики — **PR-AUC**, **ROC-AUC**, **balanced accuracy**.
- **Фиксируем** positive class и **порог** для F1/Accuracy *до* оценки (или подбираем в *внутренней* петле).
- Многокласс: обязательно указывать усреднение (*macro/micro/weighted*).
- Отчитываем **среднее и разброс/CI** по фолдам/повторам, а не одно число.

Пример: задача бинарной классификации

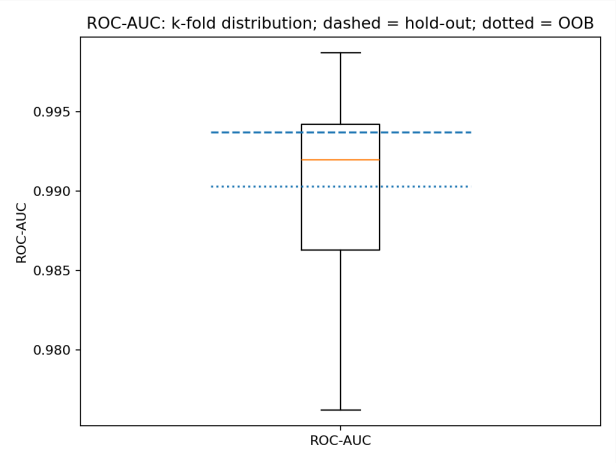
- Объём: 569 наблюдений, 30 числовых признаков.
- Классы: 212 / 357 (умеренный дисбаланс) — используем стратифицированные разбиения.
- Что сравниваем: **Hold-out**, **k-fold (5)**, **OOB**.
- Метрики: **ROC-AUC**, **F1**, **Accuracy** (для F1/Accuracy фиксирован порог и positive class).

Результаты сравнения схем

Схема	Accuracy	F1	ROC-AUC
Hold-out	0.947	0.958	0.994
k-fold (5)	0.953 ± 0.015	0.962 ± 0.011	0.989 ± 0.009
OOB	0.961	0.969	0.990

Значения — средние (для k-fold: среднее $\pm$ стандартное отклонение). Все оценки получены на независимых от обучения предсказаниях.

Выводы



ROC-AUC: коробка — распределение по фолдам, пунктир — Hold-out, точечный — OOB.

- **k-fold** даёт более стабильную оценку (меньше разброс).
- **OOB** близок к k-fold без явного CV.
- **Hold-out** — быстрый ориентир; число может чуть «шуметь».
- Делаем выводы по *среднему и разбросу*, а не по единичному числу.