Data Mining in Action

Проверка качества

- Как измерять качество?

Проверка качества

- Классический подход

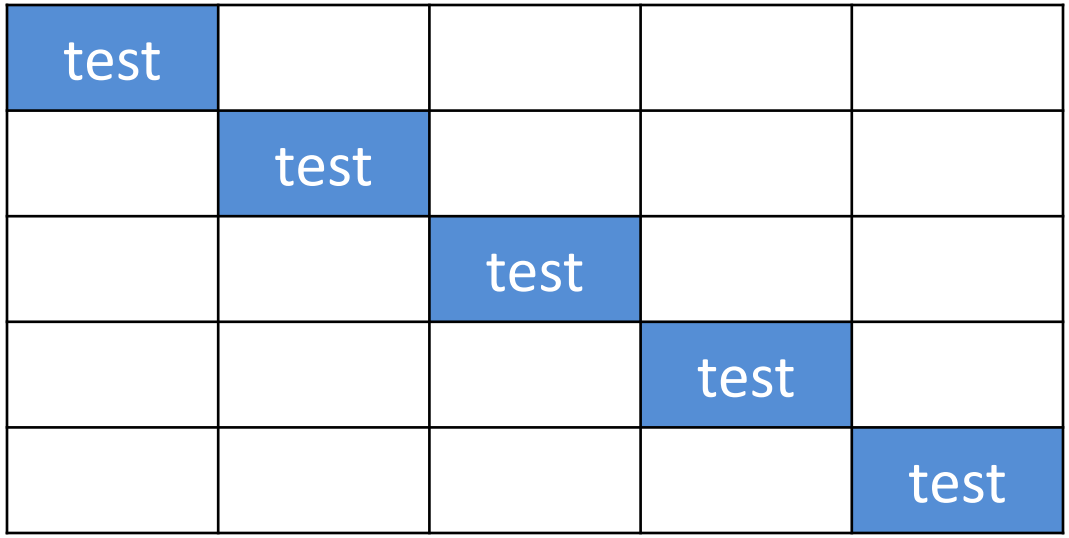
Проверка качества на Kaggle

Стратегия 1: Holdout

Преимущества и недостатки

- + скорость валидации
- - мало данных: можно заточиться под holdout

Стратегия 2: K-fold CV



Преимущества и недостатки

- + уменьшается вероятность «затачивания» под данные
- + в тестировании участвует вся выборка
- - требовательность к ресурсам

Иногда и этого недостаточно!

- Мало данных – Mail.ru ML bootcamp

  210 объектов / 5 фолдов = 42 объекта в тесте

- Выход: уменьшаем количество фолдов, повторяем много раз: 32 x 3-Fold CV

Иногда и этого недостаточно!

- Много шума: Santander
- Выход: повторять 5-fold с различными разбиениями

Рекомендации

- Смотреть на дисперсию оценки качества между фолдами
- Сверять с лидербордом: нужно, чтобы улучшение на CV давало улучшение на LB

Проблема с ROC AUC

- AUC оценивает качество не поточечно, а ответов на всей выборке!

0.1	0
0.2	1

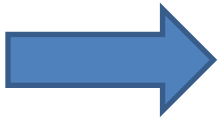
AUC=1

0.3	0
0.4	1

AUC=1

0.5	0
0.6	1

AUC=1



0.1	0
0.2	1
0.3	0
0.4	1
0.5	0
0.6	1

AUC=0.66

Смешивание алгоритмов

- Blending
- Stacking
- Feature-weighted stacking

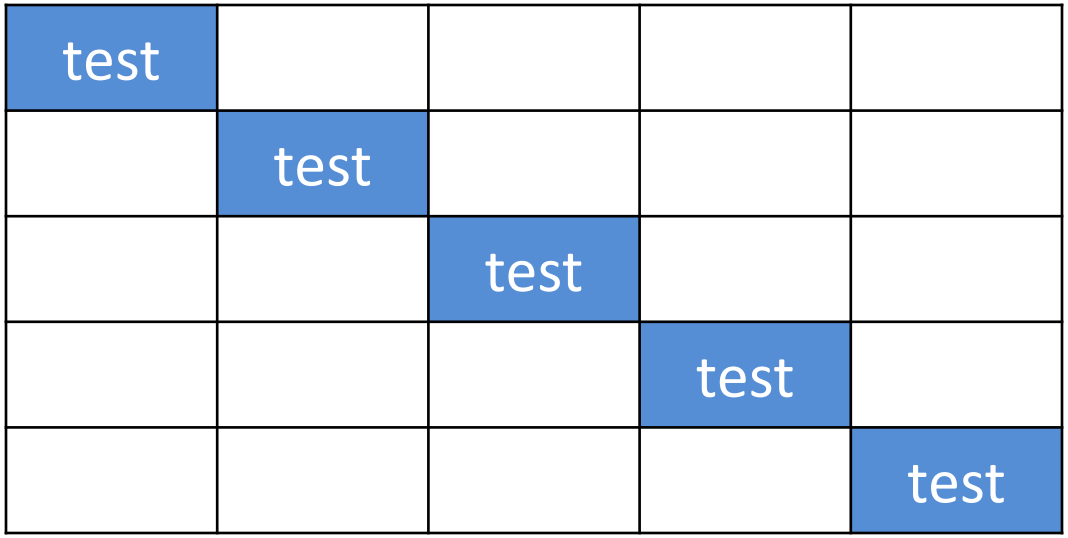
Blending

a(x) = alpha * model1(x) + (1-alpha) * model2(x)

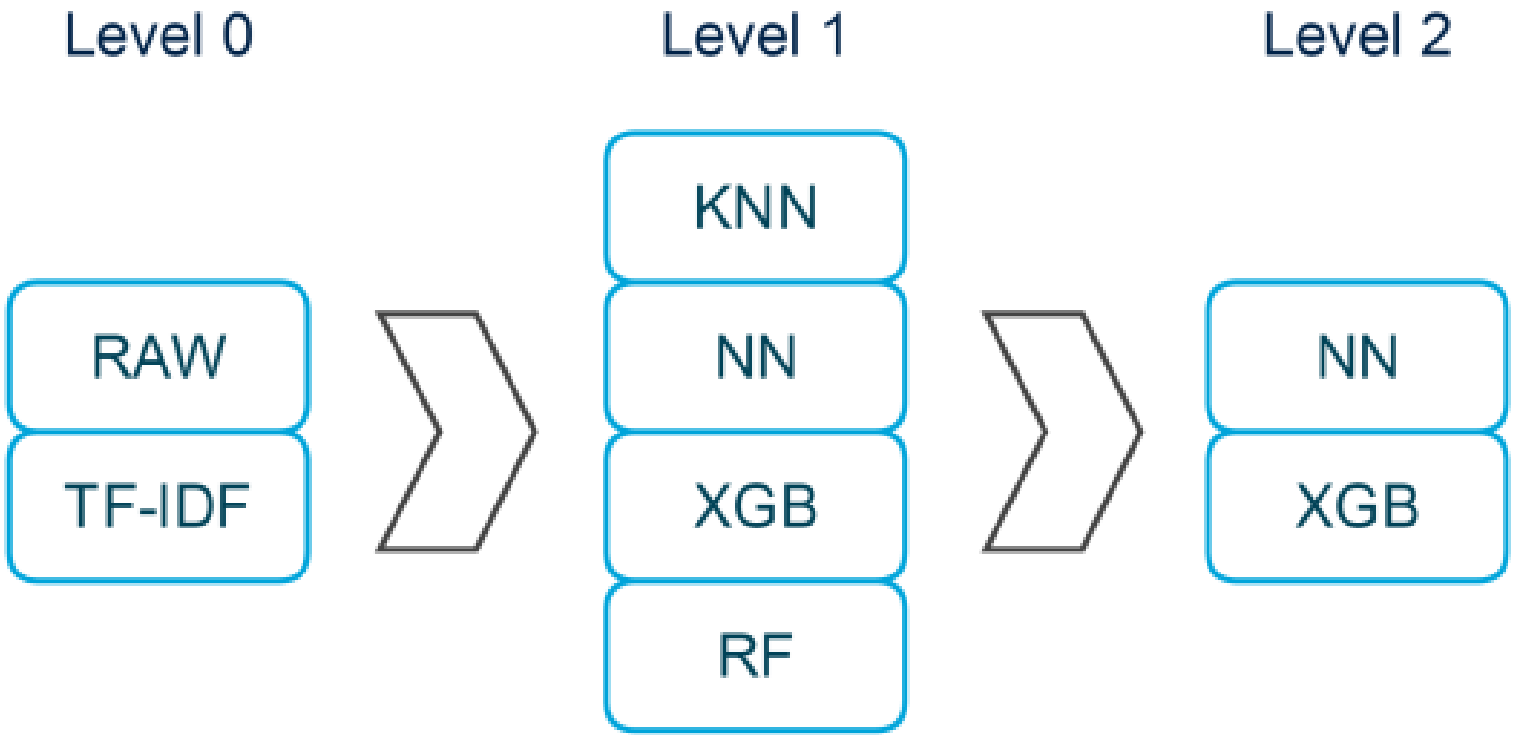
- Нужно смешивать разные алгоритмы!
- Следить за переобучением

Как не переобучиться?

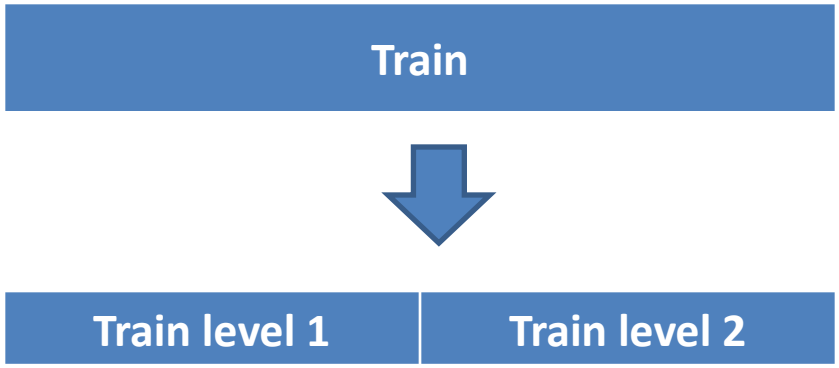
- K-Fold разбиение!

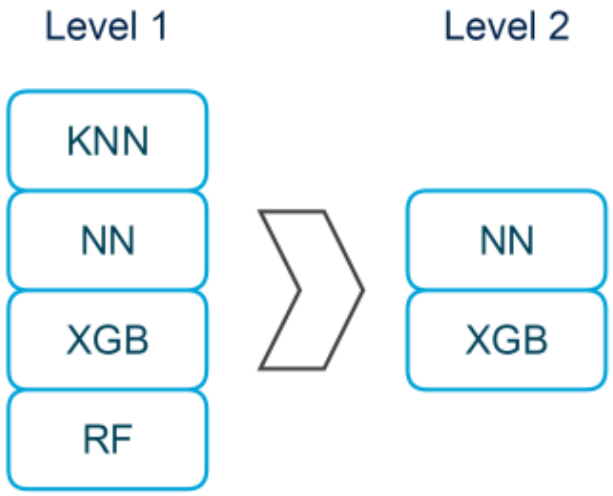


Stacking



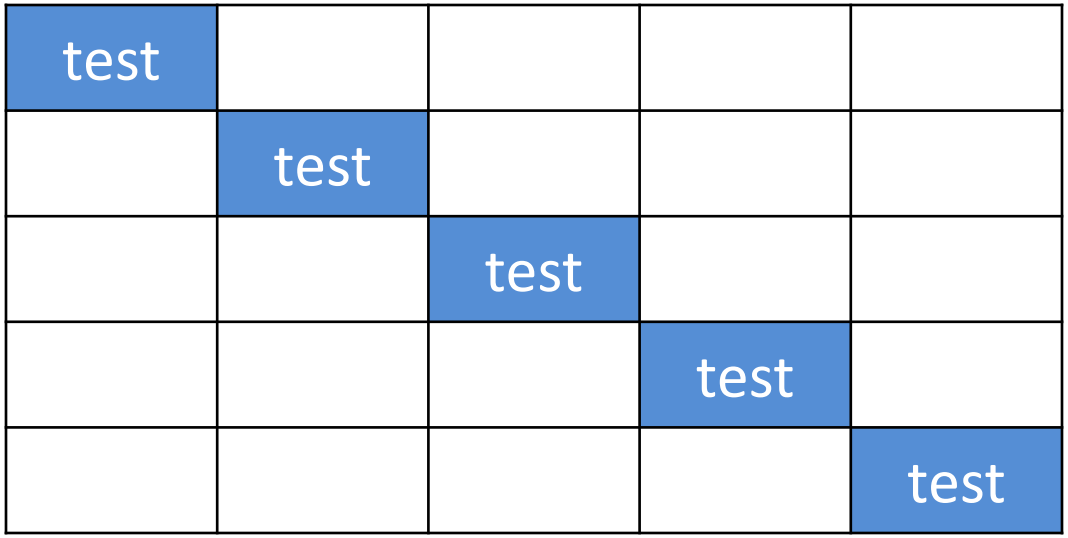
Как делать?


Train
Train level 1
Train level 2


Level 1
Level 2
KNN
NN
XGB
RF
NN
XGB

Как делать?

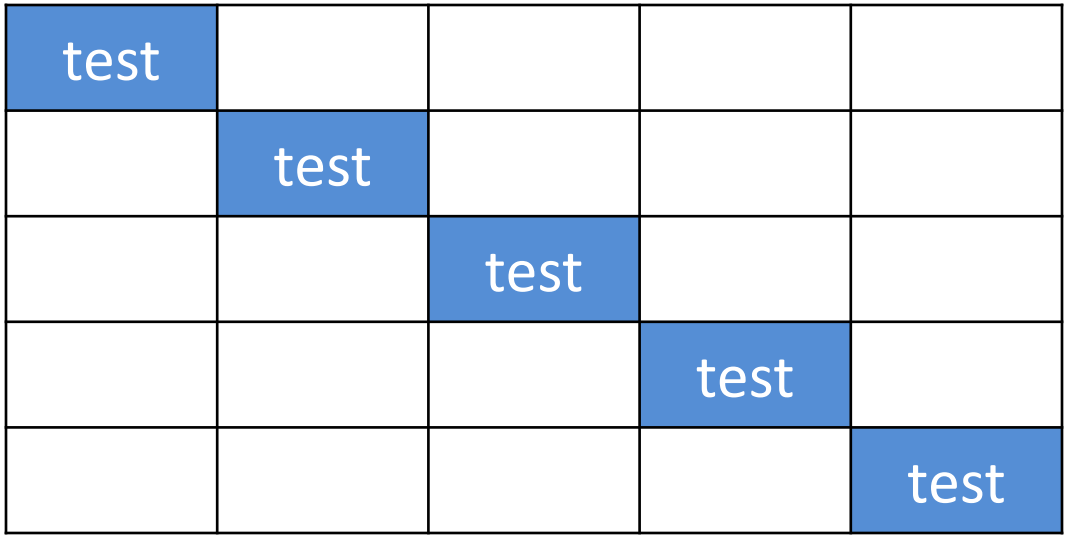
- K-Fold разбиение



Но как быть с тестом?

Предсказание для теста:

- Усреднить по фолдам
- Обучиться на всем трейне, посчитать для теста



Рекомендации

- Фиксированное разбиение!
- Разные выборки: X, scale(X), sparse(X), PCA(X)
- Разные (по природе) алгоритмы
- Не всегда лучшие параметры дают большИй прирост в смеси