Ансамблевые методы

Паточенко Евгений

НИУ ВШЭ

План занятия

- Смещение и разброс ошибки
- Ансамбли моделей
- Бэггинг
- Случайный лес
- Стекинг

Смещение и разброс ошибки

Ошибка модели складывается из трех компонент:

- Смещение (bias) — способность модели приблизить лучшую среди всех возможных моделей
- Разброс (variance) — устойчивость модели к изменениям в обучающей выборке
- Шум (noise) — характеристика сложности и противоречивости данных

$$Q(a) = \mathbb{E}_x bias_X^2 a(x, X) + \mathbb{E}_x \mathbb{V}_x[a(x, X)] + \sigma^2$$

Смещение и разброс ошибки

Смещение

$$bias_X a(x, X) = f(x) - \mathbb{E}_X[a(x, X)]$$

Характеризует среднюю ошибку алгоритма по всем возможным наборам обучающим выборкам

- Показывает насколько хорошо с помощью данного алгоритма $a(x)$ можно приблизить целевую зависимость $f(x)$
- Маленькое смещение – хорошее предсказание целевой переменной в среднем
- Большое смещение – предсказания далеки от истинной переменной

Смещение и разброс ошибки

Разброс

$$\mathbb{V}_X[a(x,X)] = \mathbb{E}_X\left[a(x,X) - \mathbb{E}_X[a(x,X)]\right]^2$$

Характеризует чувствительность алгоритма к изменениям в обучающей выборке

- Показывает дисперсию предсказаний алгоритма в зависимости от обучающей выборки
- Маленький разброс – устойчивая к изменениям в данных модель
- Большой разброс – сильно переобученная чувствительная модель

Смещение и разброс ошибки

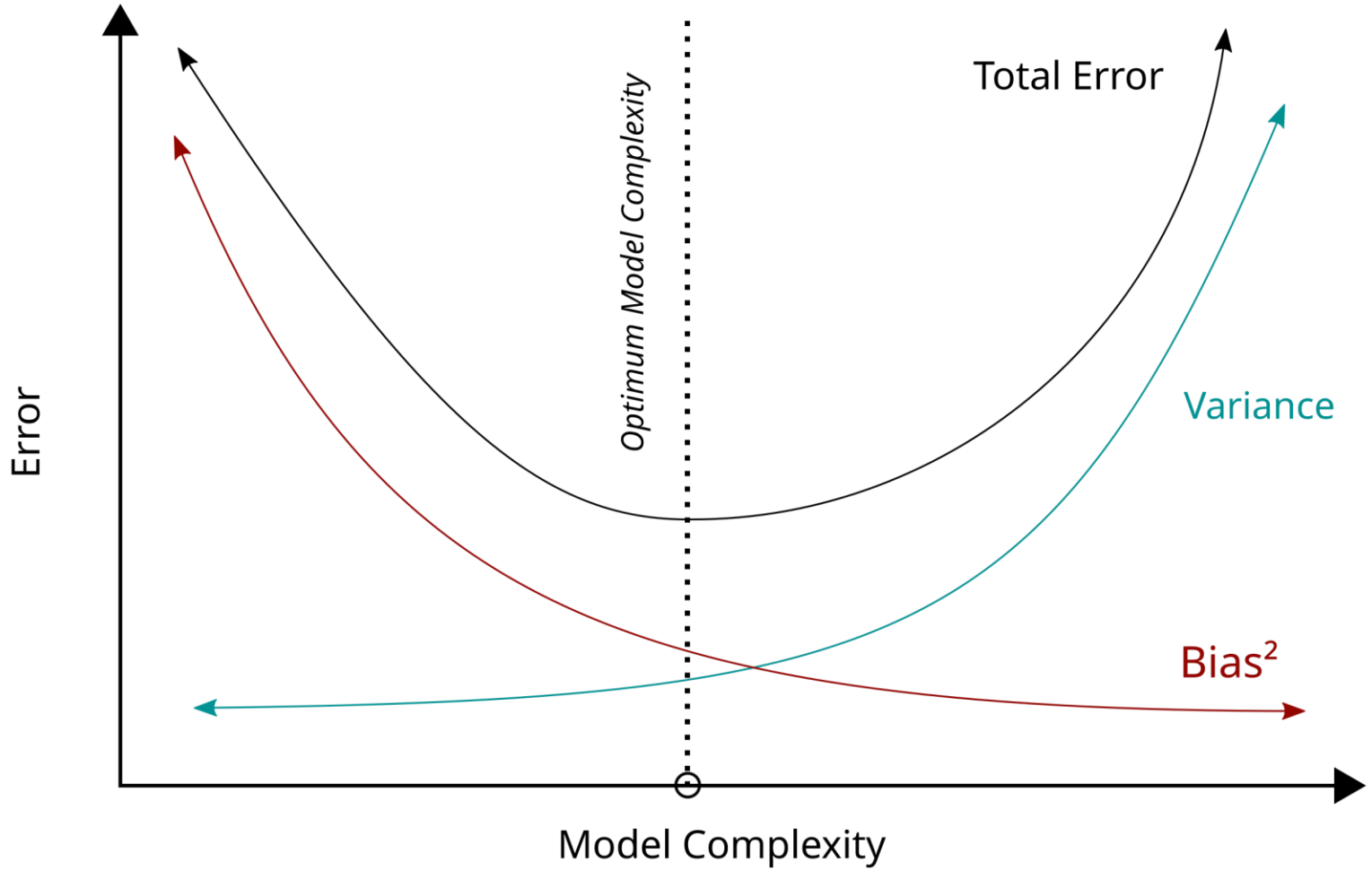
Шум

$$\sigma^2 = \mathbb{E}_X \mathbb{E}_\epsilon [y(x, \epsilon) - f(x)]^2$$

Случайный шум обусловлен природой самих данных. Может возникать потому что:

- Данные на самом деле имеют случайный характер
- Измерительный прибор не может зафиксировать целевую переменную абсолютно точно
- Имеющихся признаков недостаточно, чтобы исчерпывающим образом описать связь между целевой переменной и признаками объекта x

Смещение и разброс ошибки



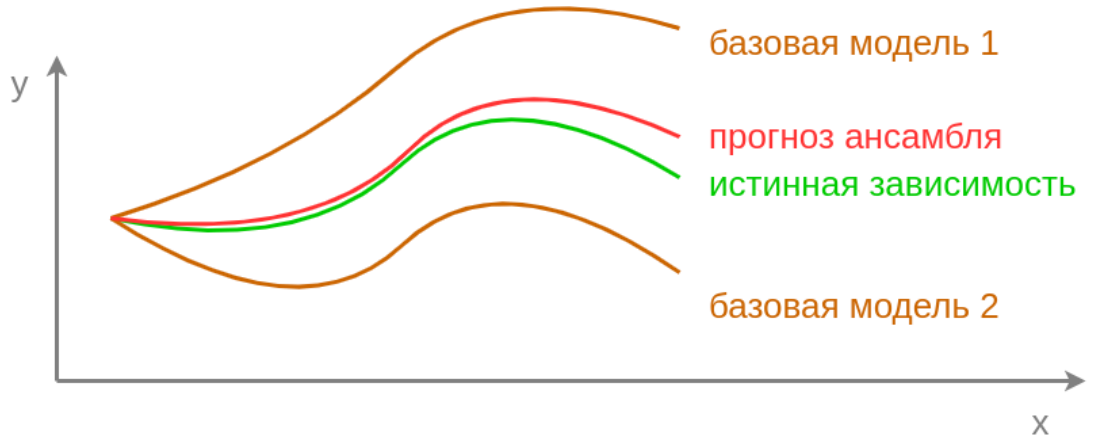
Источник: https://en.wikipedia.org/wiki/Bias%E2%80%93variance_tradeoff

Ансамбли моделей

Существует способ уменьшить ту или иную компоненту ошибки через использование композиции (ансамбля) нескольких моделей

Ансамблем моделей называется подход, который строит свой прогноз, используя не одну модель $f(x)$, а совокупность базовых моделей $f_1(x)$, …, $f_M(x)$ и агрегирующую мета-модель $G(.)$, которая учитывает прогнозы всех базовых моделей:

$$\hat{y}(x) = G\big(f_1(x), \ldots, f_M(x)\big)$$



Источник: https://deepmachinelearning.ru/docs/Machine-learning/Model-ensembles/Model-ensembles

Ансамбли моделей

Классификатор с жестким голосованием: конечный прогноз — класс с наибольшим количеством голосов

Классификатор с мягким голосованием: конечный прогноз — класс с наибольшей вероятностью (усреднение по всем базовым классификаторам). Более точен, так как учитывает вероятности. Работает для базовых классификаторов, у которых есть метод *predict_proba*

Регрессор с голосованием: конечный прогноз — усредненный прогноз базовых алгоритмов

Ансамбли моделей

Способы произвести агрегацию ответов базовых алгоритмов (для регрессии):

- Брать среднее $a(x) = mean(b_1(x), \dots, b_M(x))$
- Брать медиану $a(x) = median(b_1(x), \dots, b_M(x))$
- Брать взвешенное среднее $a(x) = (w_1 b_1(x) + \dots + w_M b_M(x))$

Способы голосования (для классификации):

- Голосование по большинству $a(x) = mode(b_1(x), \dots, b_M(x))$
- Комитет единогласия $a(x) = \min(b_1(x), \dots, b_M(x))$

Ансамбли моделей

В идеале нужно, чтобы модели в ансамбле были независимы

Ансамбли моделей

В идеале нужно, чтобы модели в ансамбле были независимы

Возможно ли такое?

Ансамбли моделей

В идеале нужно, чтобы модели в ансамбле были независимы

Возможно ли такое?

Почему?

Ансамбли моделей

На практике мы можем постараться сделать как можно более независимые модели:

- Использовать модели разных классов
- Использовать различные гиперпараметры
- Использовать разную инициализацию
- Использовать различные подмножества обучающей выборки
- Использовать разные функции потерь

Ансамбли моделей

На практике мы можем постараться сделать как можно более независимые модели:

- Использовать модели разных классов
- Использовать различные гиперпараметры
- Использовать разную инициализацию
- Использовать различные подмножества обучающей выборки
- Использовать разные функции потерь

Больше независимость моделей — больше прирост по качеству даст ансамблирование

Ансамбли моделей

На практике мы можем постараться сделать как можно более независимые модели:

- Использовать модели разных классов
- Использовать различные гиперпараметры
- Использовать разную инициализацию
- **Использовать различные подмножества обучающей выборки**
- Использовать разные функции потерь

Получение псевдовыборок

- Кросс-валидация — K подвыборок с пересечением
- Пейстинг — семплирование объектов без возвращения
- Метод случайных подпространств — все объекты, но множество признаков семплируется без возвращения
- Метод случайных фрагментов — комбинация семплирования объектов и признаков без возвращения
- Бутстреп — семплирование объектов с возвращением

Получение псевдовыборок

- Кросс-валидация — К подвыборок с пересечением
- Пейстинг — семплирование объектов без возвращения
- Метод случайных подпространств — все объекты, но множество признаков семплируется без возвращения
- Метод случайных фрагментов — комбинация семплирования объектов и признаков без возвращения
- **Бутстреп — семплирование объектов с возвращением**

Основа бэггинга и случайного леса

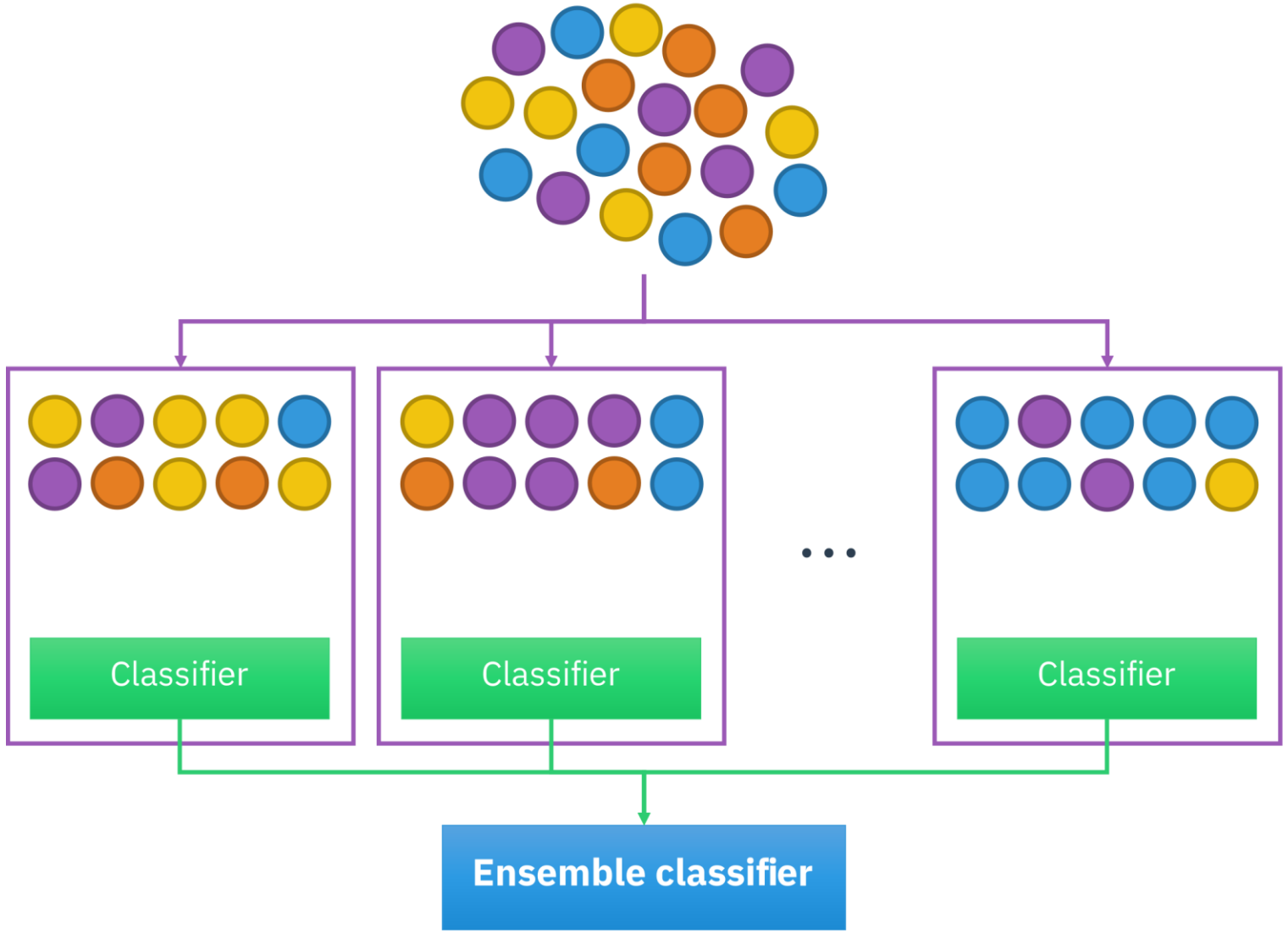
Бэггинг

Бэггинг (bagging, bootstrap aggregating) — метод, при котором несколько моделей обучаются независимо на случайных подвыборках данных:

1. Создаются бутстреп-выборки
2. Обучаются одинаковые модели
3. Предсказания усредняются или голосуются

Бэггинг не ухудшает смещение базовой модели, но при этом способен минимизировать ее разброс

Ссылка на статью Bagging Predictors, Leo Breiman (автор метода)



Источник: https://en.wikipedia.org/wiki/Bootstrap_aggregating

Случайный лес

Случайный лес (Random Forest) — частный случай бэггинга, при котором базовыми моделями выступают деревья решений, а на каждом шаге обучения используется дополнительное случайное подмножество признаков для разделения узлов:

1. Создаются бутстреп-выборки
2. Обучаются несколько деревьев
3. При построении каждого узла дерева выбирается случайное подмножество признаков, из которых выбирается лучший сплит
4. Предсказания усредняются или голосуются

По сути случайный лес — это комбинация бэггинга и метода случайных подпространств (см слайд 18) над решающими деревьями

Статья на scikit-learn про случайный лес

Bootstrap sampling

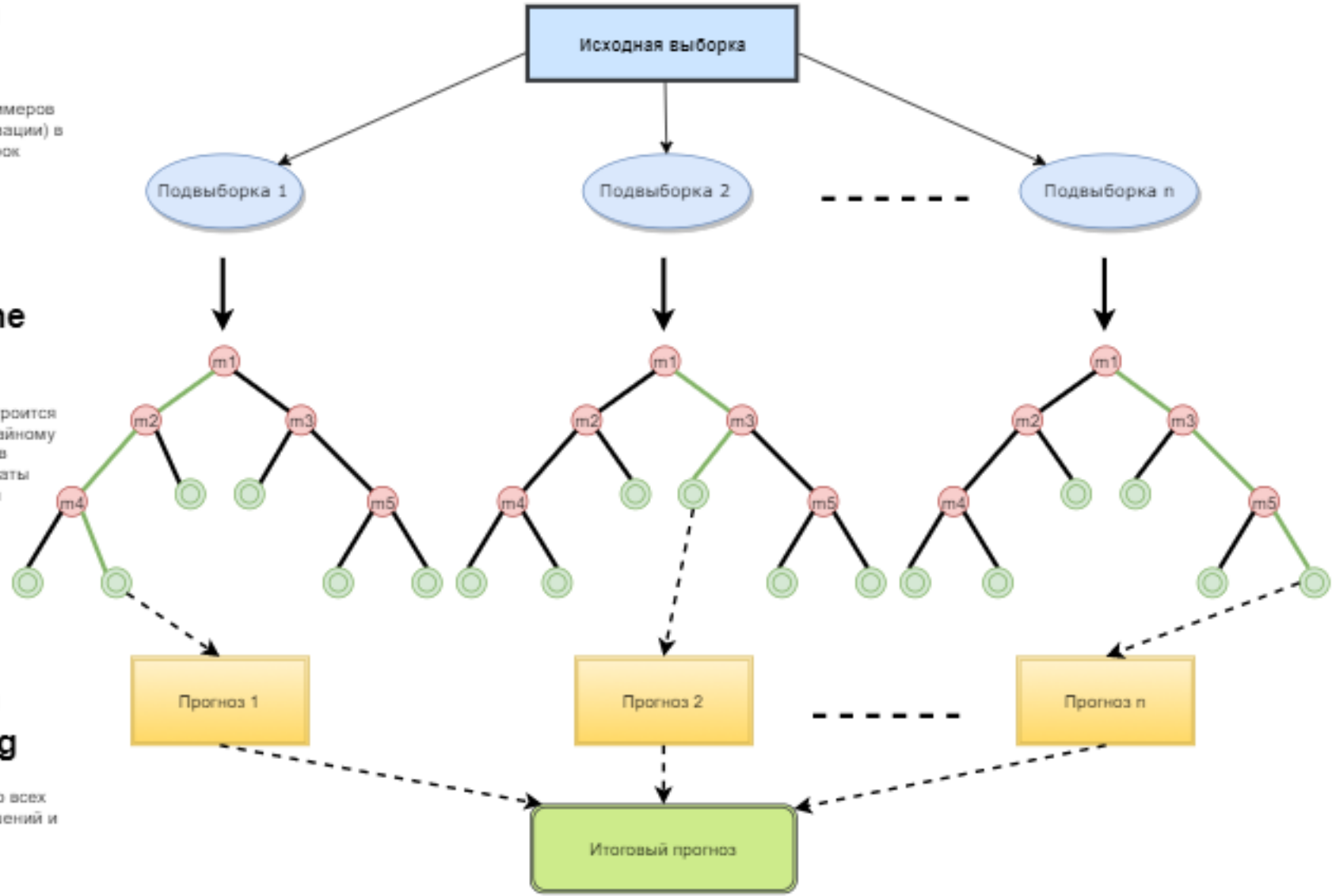
выбирается r (процент) примеров (0.63 в классической реализации) в n случайных подвыборок

Building the models

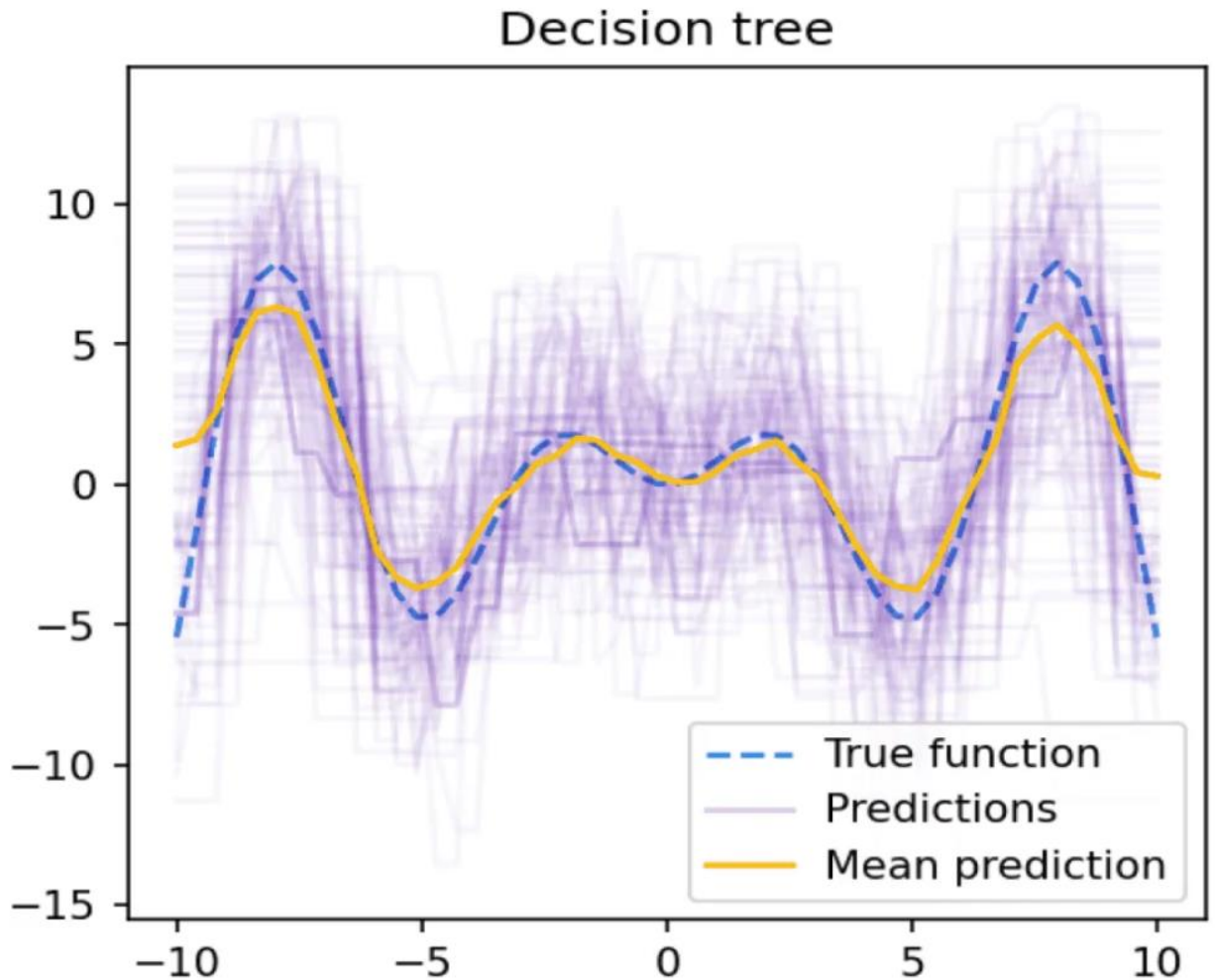
по каждой подвыборке строится дерево решений по случайному набору m признаков (ковариатов)., результаты попадают в листья

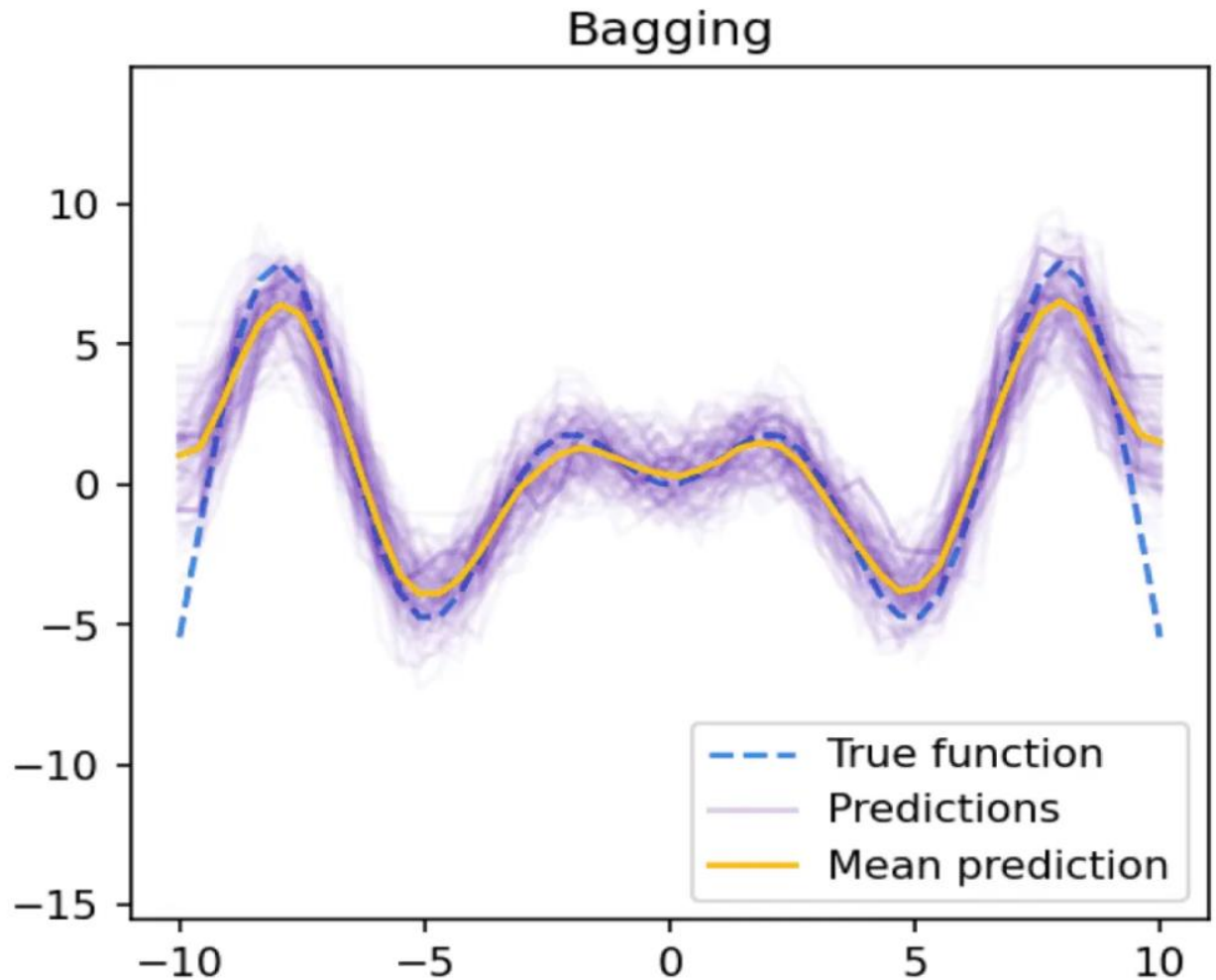
Bootstrap aggregating

собираются результаты со всех построенных деревьев решений и усредняются



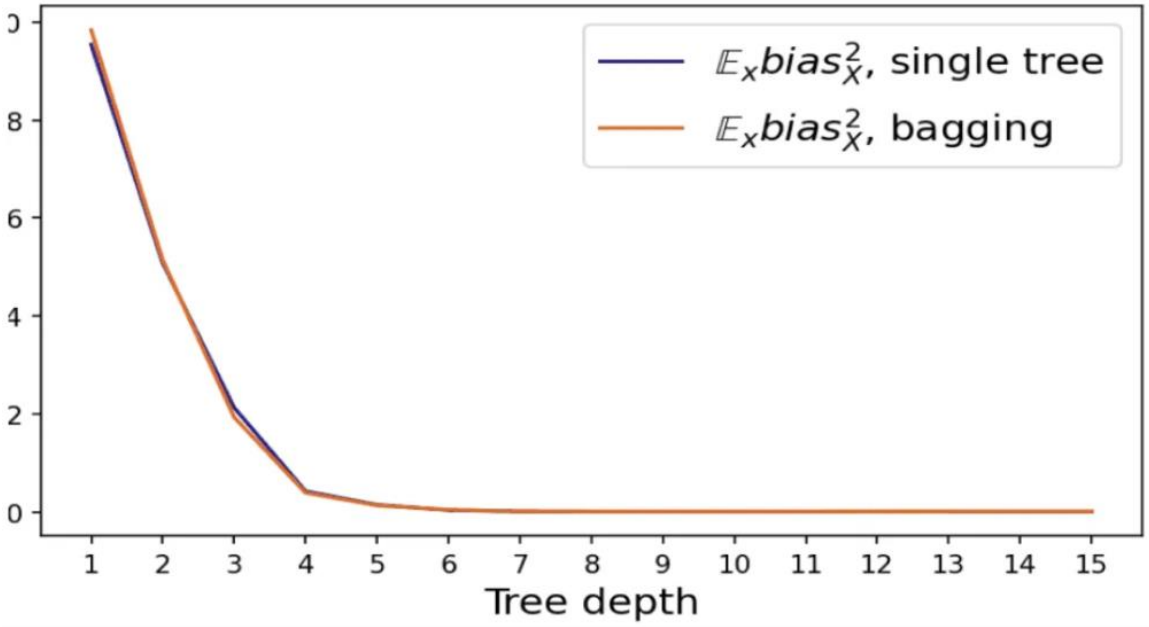
Источник: https://www.mql5.com/ru/articles/3856

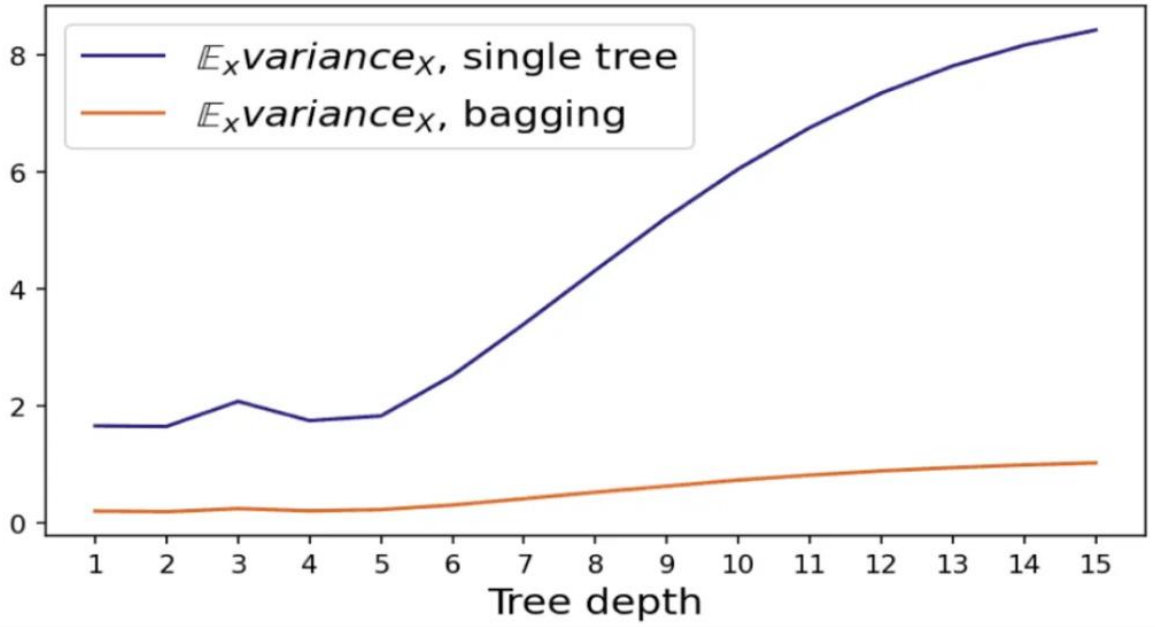




Источник: https://education.yandex.ru/handbook/ml/article/ansambli-v-mashinnom-obuchenii

Случайный лес





Источник: https://education.yandex.ru/handbook/ml/article/ansambli-v-mashinnom-obuchenii

- Смещение не меняется при усреднении
- Значительно снизился разброс

Случайный лес

Глубина деревьев

- Неглубокие деревья имеют малое число параметров, поэтому плохо учитывают закономерности в данных и имеют большое смещение
- Глубокие деревья наоборот слишком сильно запоминают выборку и имеют слишком большой разброс

Так как усреднение деревьев не способно изменить смещение базового алгоритма, но способно уменьшить его разброс, то имеет смысл использовать глубокие деревья

Случайный лес

Количество деревьев

Увеличение числа деревьев уменьшает разброс, но не влияет на смещение. Количество признаков и подвыборок ограничено ⇒ разброс не снижается бесконечно.

Второе ограничение — время работы:

- Random Forest можно параллелить, но количество процессоров ограничено.
- При слишком большом числе деревьев обучение становится медленным, поэтому оптимально сократить их число, немного пожертвовав качеством.

На практике строят график ошибки от числа деревьев и выбирают момент, когда улучшение становится незначительным.

Случайный лес

Количество признаков

Чем больше признаков: тем сильнее похожи деревья ⇒ выше корреляция, меньше эффект ансамблирования.

Чем меньше признаков, тем деревья более разнообразны, но слабее индивидуально.

На практике можно использовать, например:

- Для классификации — корень из количества всех признаков
- Для регрессии — около 1/3 признаков

Случайный лес

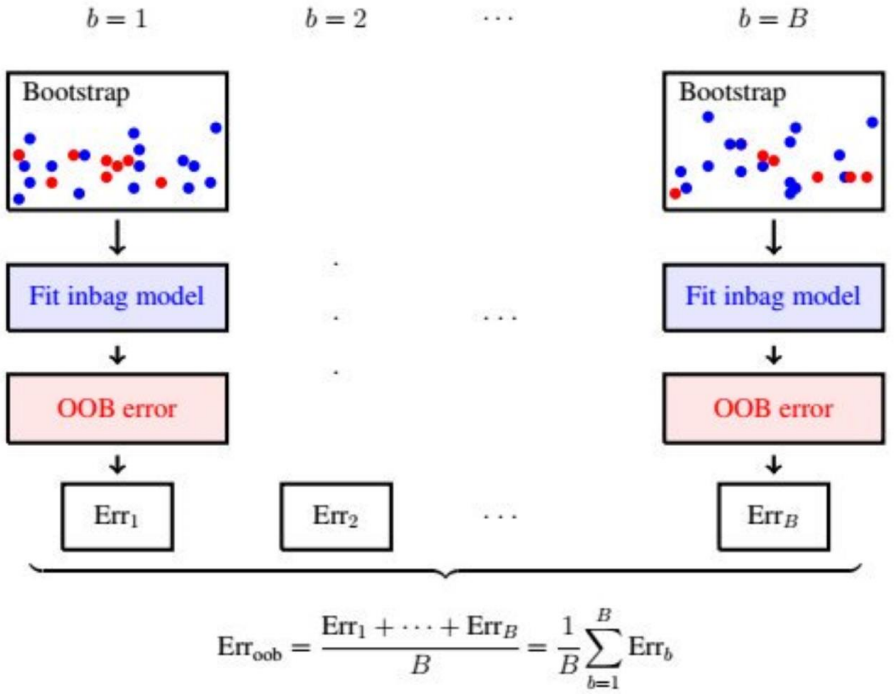
Out-of-bag ошибка — усредненная ошибка на неотобранных образцах по всему случайному лесу

Каждое дерево в случайном лесе обучается по некоторому подмножеству объектов
Для каждого объекта x_n есть деревья, которые на этом объекте не обучались

Пусть $I(n)$ – множество псевдовыборок, куда объект x_n не попал.

Тогда out-of-bag ошибка:

$$OOB = \sum_{n=1}^{N} L(y_n, \frac{1}{|I(x_n)|} \sum_{i \in I(n)} b_i(x_n))$$



Стекинг

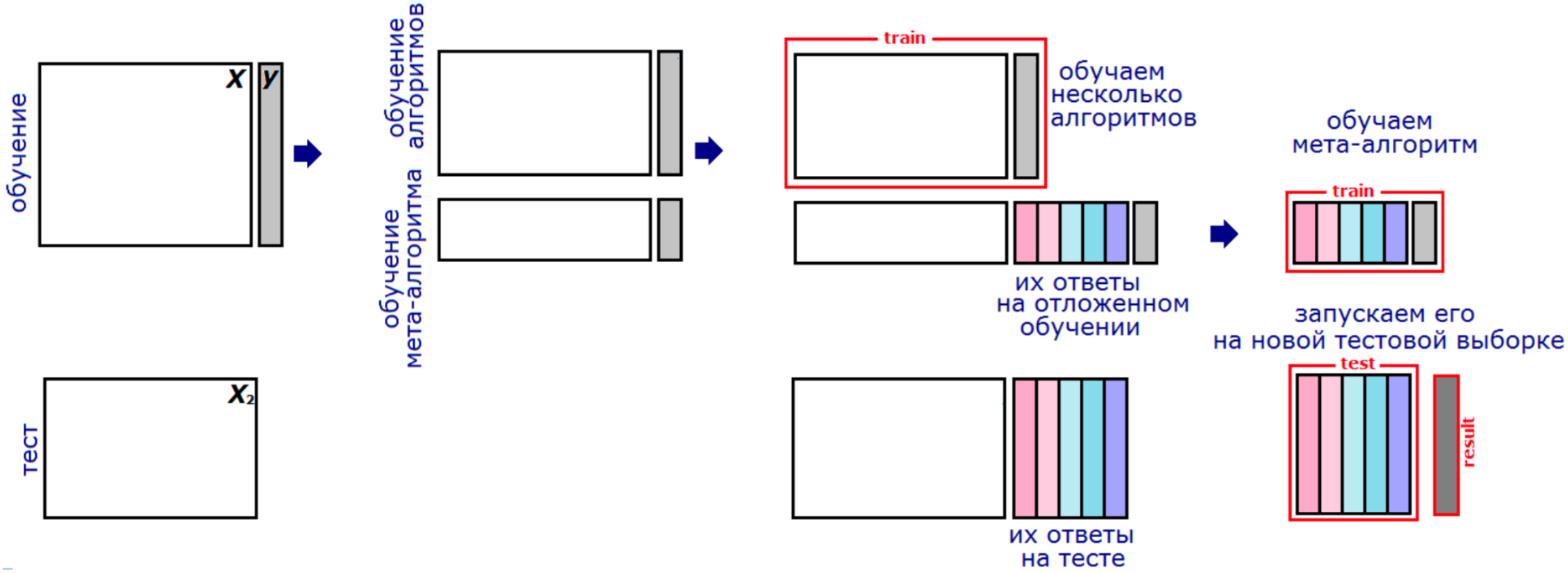
Метод, при котором объединяются разные типы моделей

Вместо простого усреднения предсказания базовых моделей объединяются обучаемой метамоделью:

1. Данные делятся на тренировочную и тестовую выборки
2. Тренировочная часть разбивается на *n* фолдов (как при кросс-валидации)
3. Базовые модели обучаются на *(n−1)* фолдах и делают предсказания на оставшемся — получаются мета-признаки
4. На этих мета-признаках обучается метамодель, которая делает финальное предсказание

Стекинг не нацелен напрямую на уменьшение смещения или разброса, но на практике снижает общую ошибку модели ⇒ снижает и компоненты.

Стекинг



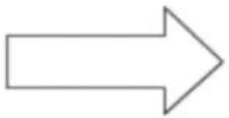
Источник: https://ai-news.ru/2018/07/cteking_stacking_i_blending_blending.html

Стекинг

train_data

	x_0	x_1	x_2	x_3	y
fold_1	0.94	0.27	0.80	0.34	1
	0.02	0.22	0.17	0.84	0
fold_2	0.83	0.11	0.23	0.42	1
	0.74	0.26	0.03	0.41	0
fold_3	0.08	0.29	0.76	0.37	0
	0.71	0.76	0.43	0.95	1
fold_4	0.08	0.71	0.97	0.04	0
	0.84	0.97	0.89	0.05	1

Training
m1, m2, m3



train_data_new_features

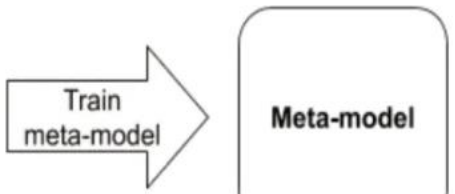
pred_m1	pred_m2	pred_m3
0.96	0.86	0.66
0.03	0.13	0.30
0.90	0.90	0.85
0.12	0.10	0.11
0.03	0.09	0.08
0.77	0.50	0.67
0.19	0.65	0.20
0.91	0.89	0.90

Источник: https://education.yandex.ru/handbook/ml/article/ansambli-v-mashinnom-obuchenii

Стекинг

New train_data for meta-model

pred_m1	pred_m2	pred_m3	y
0.96	0.86	0.66	1
0.03	0.13	0.30	0
0.90	0.90	0.85	1
0.12	0.10	0.11	0
0.03	0.09	0.08	0
0.77	0.50	0.67	1
0.19	0.65	0.20	0
0.91	0.89	0.90	1



Источник: https://education.yandex.ru/handbook/ml/article/ansambli-v-mashinnom-obuchenii