

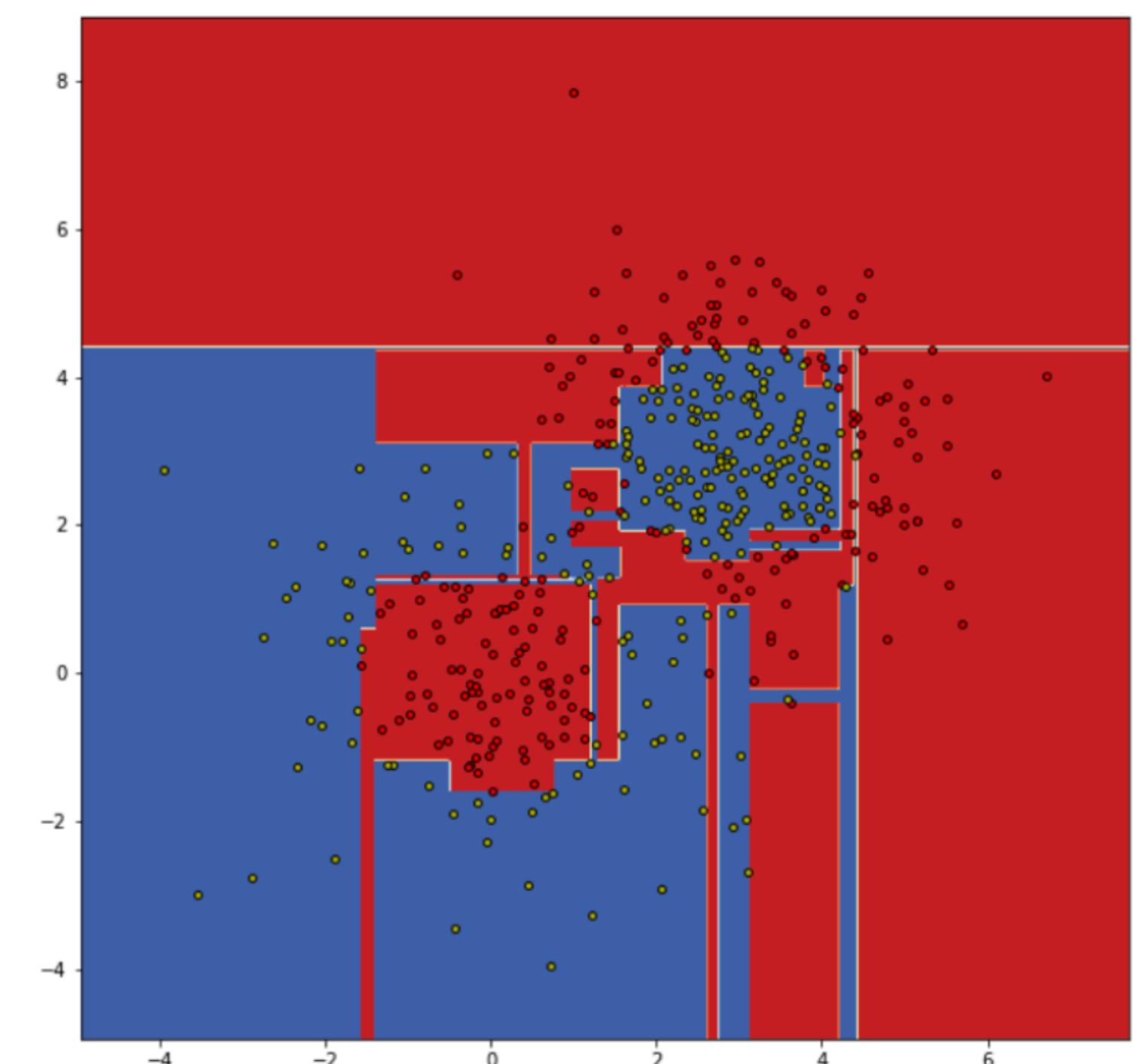
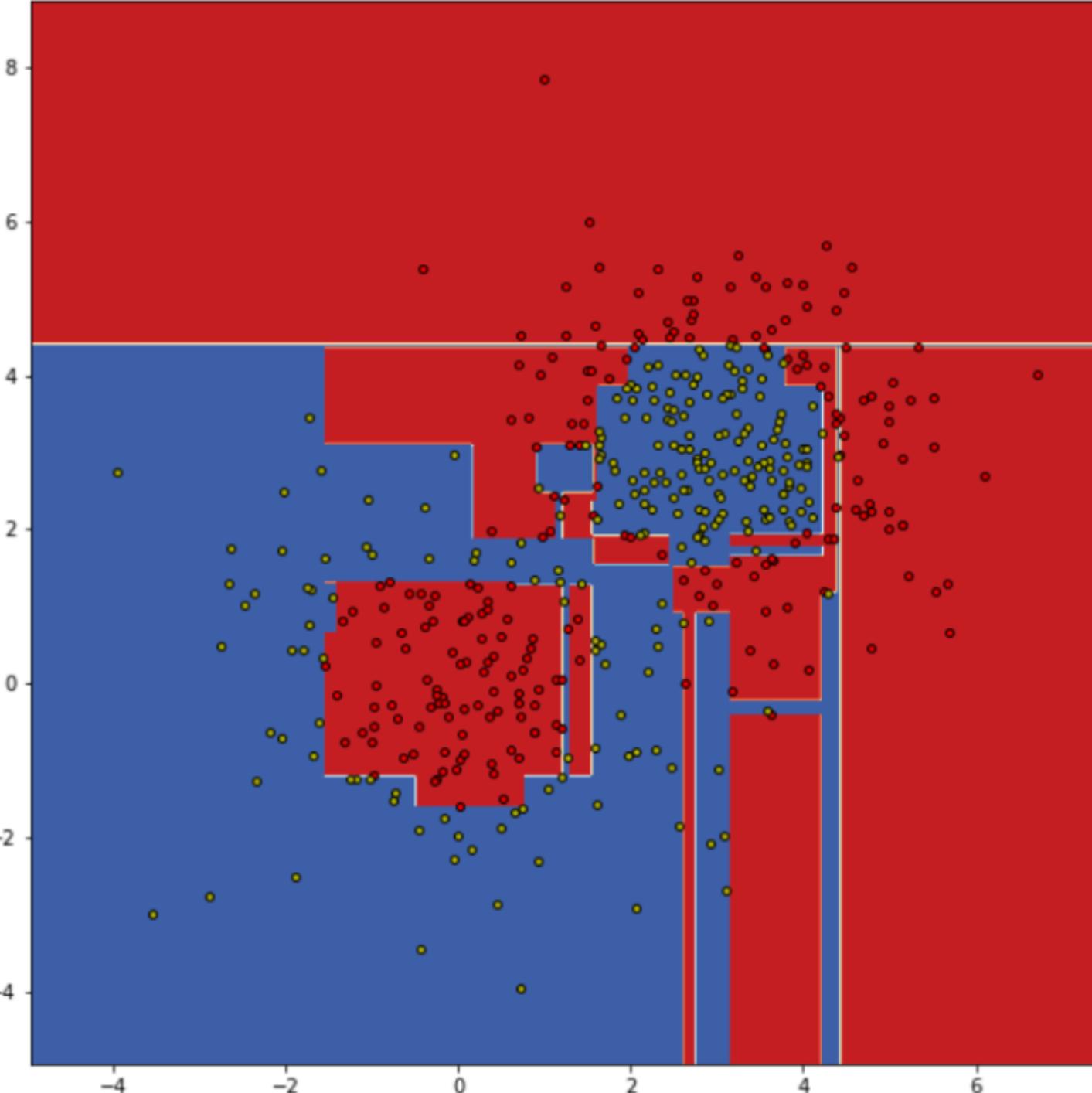
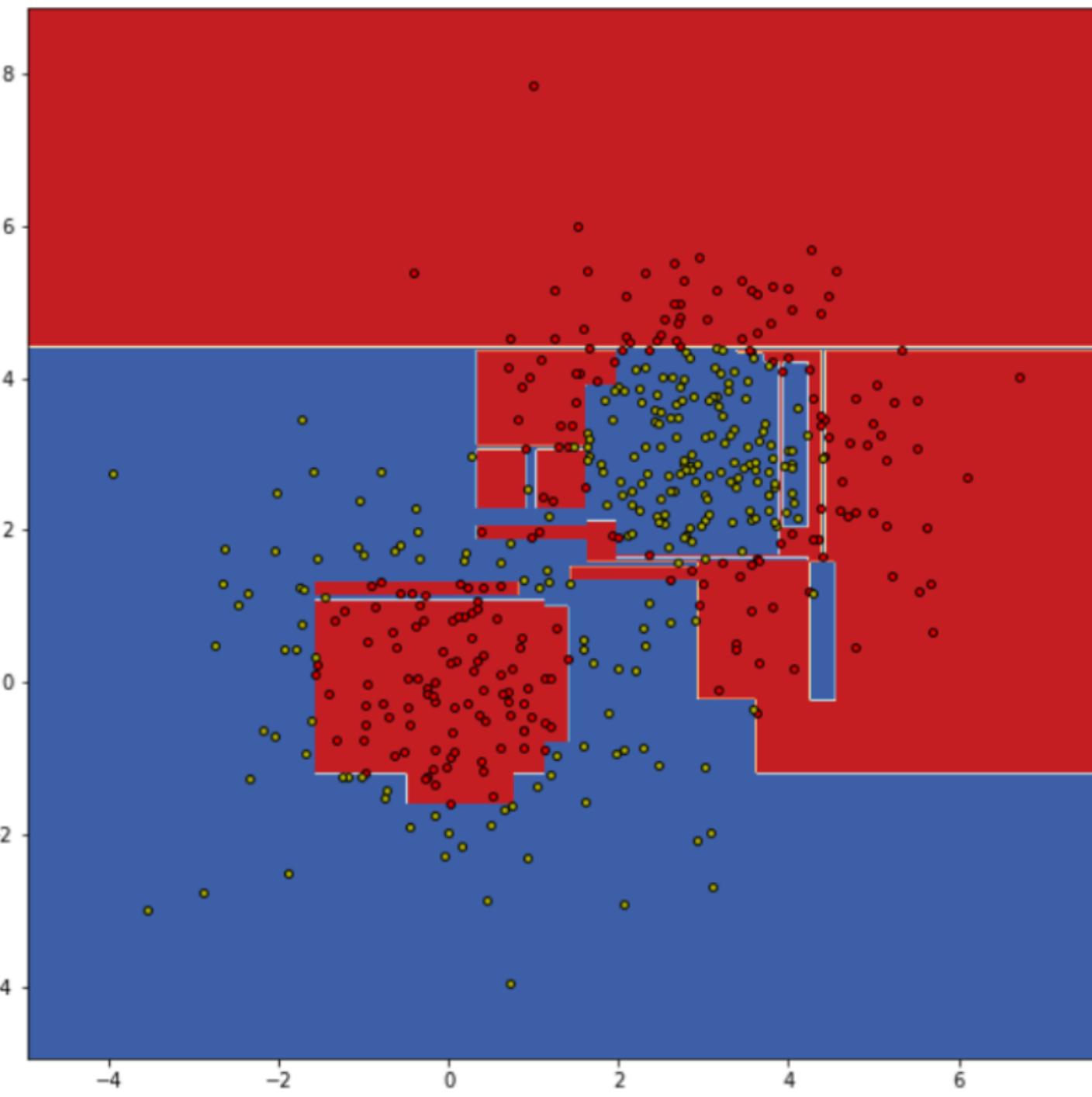
Ensembles

0. Unstable Decision Trees

- $X = (x_i, y_i)_{i=1}^\ell$ — обучающая выборка
- Обучаем модель $a(x)$
- Ожидаем, что модель устойчивая
- То есть не сильно меняется при небольших изменениях в X
- \tilde{X} — случайная подвыборка, примерно 90% исходной
 - *Что будет происходить с деревьями на разных подборках?*

0. Unstable Decision Trees

- *Что будет происходить с деревьями на разных подборках?*



0. Unstable Decision Trees

- Что будет происходить с деревьями на разных подборках?
- А если на всех моделях построить ансамбль и голосовать большинством?

$$a(x) = \arg \max_{y \in \mathbb{Y}} \sum_{n=1}^N [b_n(x) = y]$$

The diagram illustrates the calculation of a weighted sum of binary predictions from multiple models. A red arrow points to the term $a(x)$, which represents the final prediction. A green arrow points to the term $b_n(x) = y$, which represents the prediction of model n for input x . The equation shows that the final prediction $a(x)$ is the class for which more models in the ensemble voted, determined by the arg max of the weighted sum of individual model predictions.

За какой класс больше
моделей проголосовало,
тот и победил

Предсказание класса от
модели под номером n

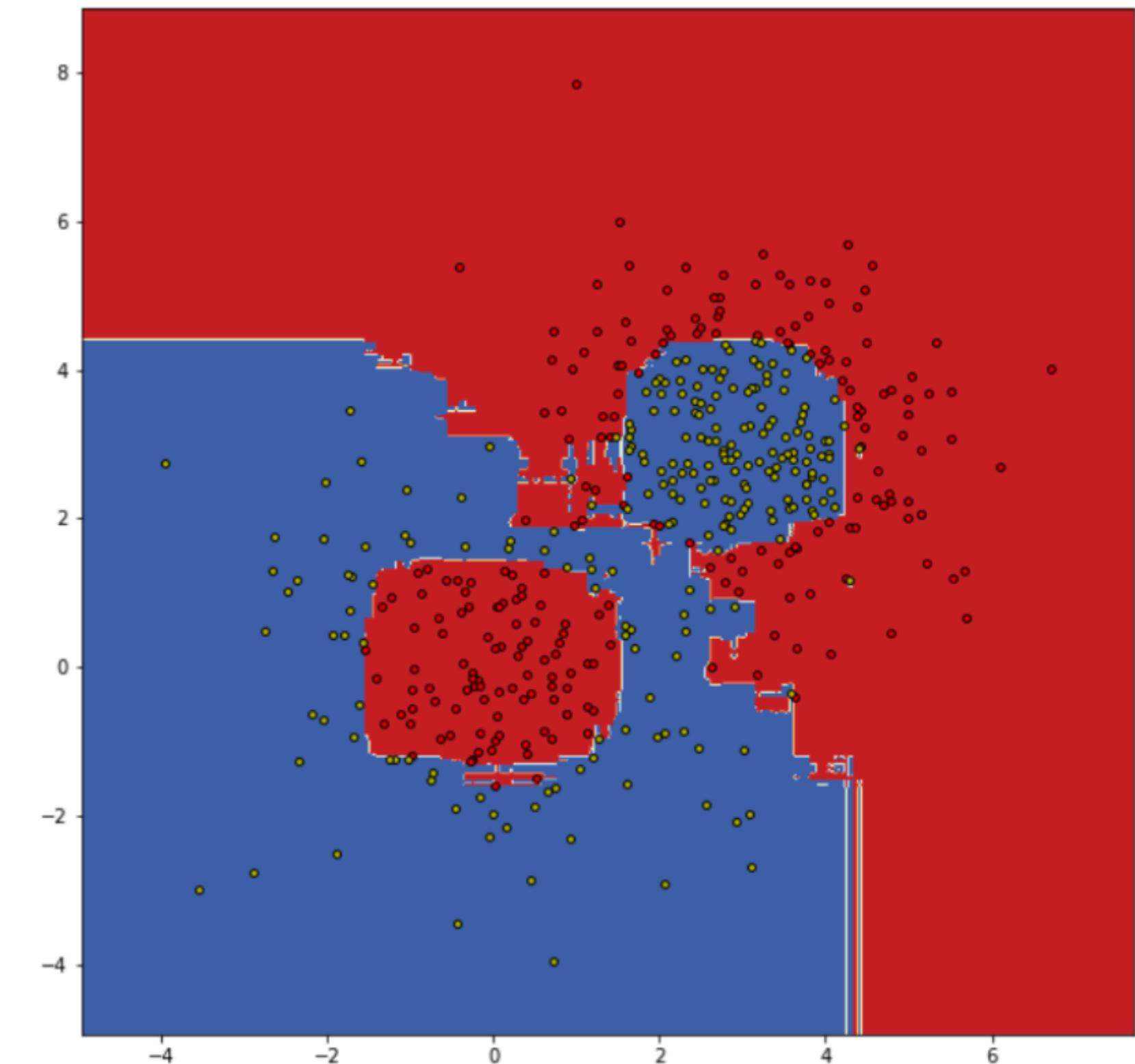
0. Unstable Decision Trees

- Что будет происходить с деревьями на разных подборках?
- А если на всех моделях построить ансамбль и голосовать большинством?

За какой класс больше
моделей проголосовало,
тот и победил

$$a(x) = \arg \max_{y \in \mathbb{Y}} \sum_n [b_n(x) = y]$$

Предсказание класса от
модели под номером n



0. Как сделать ансамбль?

- Классификация

- Базовые модели: $b_1(x), \dots, b_N(x)$
- Каждая - лучше случайного угадывания
- **Majority Voice**

$$a(x) = \arg \max_{y \in \mathbb{Y}} \sum_n [b_n(x) = y]$$

- Регрессия

- Базовые модели: $b_1(x), \dots, b_N(x)$
- Каждая - лучше случайного угадывания
- **Усреднение наблюдений**

$$a(x) = \frac{1}{N} \sum_n b_n(x)$$

0. Как сделать ансамбль?

- Классификация

- Базовые модели: $b_1(x), \dots, b_N(x)$
- Каждая - лучше случайного угадывания
- **Majority Voice**

$$a(x) = \arg \max_{y \in \mathbb{Y}} \sum_n [b_n(x) = y]$$

- Регрессия

- Базовые модели: $b_1(x), \dots, b_N(x)$
- Каждая - лучше случайного угадывания
- **Усреднение наблюдений**

$$a(x) = \frac{1}{N} \sum_n b_n(x)$$

Откуда взять базовые модели?

Вариант Uno:
независимо обучить на разных данных

Вариант Dos:
последовательно обучить чтобы улучшать предшественника

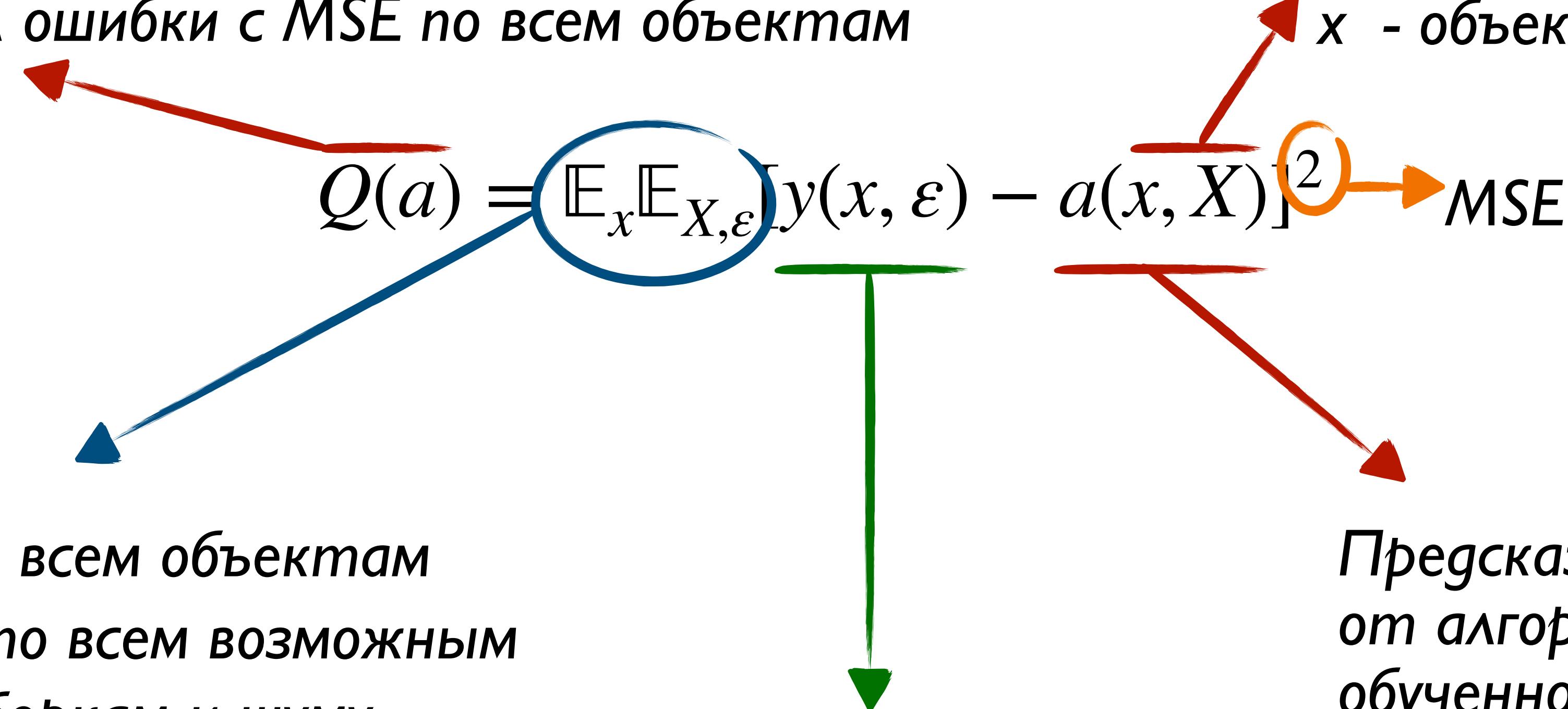
1. Bias-Variance decomposition

- *Функционал ошибки с MSE:*

$$Q(a) = \mathbb{E}_x \mathbb{E}_{X,\varepsilon} [y(x, \varepsilon) - a(x, X)]^2$$

1. Bias-Variance decomposition

- Функционал ошибки с MSE по всем объектам



\mathbb{E}_x - среднее по всем объектам

$\mathbb{E}_{X,\varepsilon}$ - среднее по всем возможным обучающим выборкам и шуму

Эти два матожидания не зависят, их можно брать как первое идущее перед вторым

Целевая зависимость для объекта x , предсказываемая с точностью до шума ε

$$y(x, \varepsilon) = f(x) + \varepsilon$$

X - обучающая выборка

x - объект тестовой выборки

Предсказание для объекта x от алгоритма a обученного на X

1. Bias-Variance decomposition

- Функционал ошибки с MSE:

$$Q(a) = \mathbb{E}_x \mathbb{E}_{X,\varepsilon} [y(x, \varepsilon) - a(x, X)]^2$$

- Аналогичное представление функционала:

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

1. Bias-Variance decomposition

- Функционал ошибки с MSE:

$$Q(a) = \mathbb{E}_x \mathbb{E}_{X,\varepsilon} [y(x, \varepsilon) - a(x, X)]^2$$

- Аналогичное представление функционала:

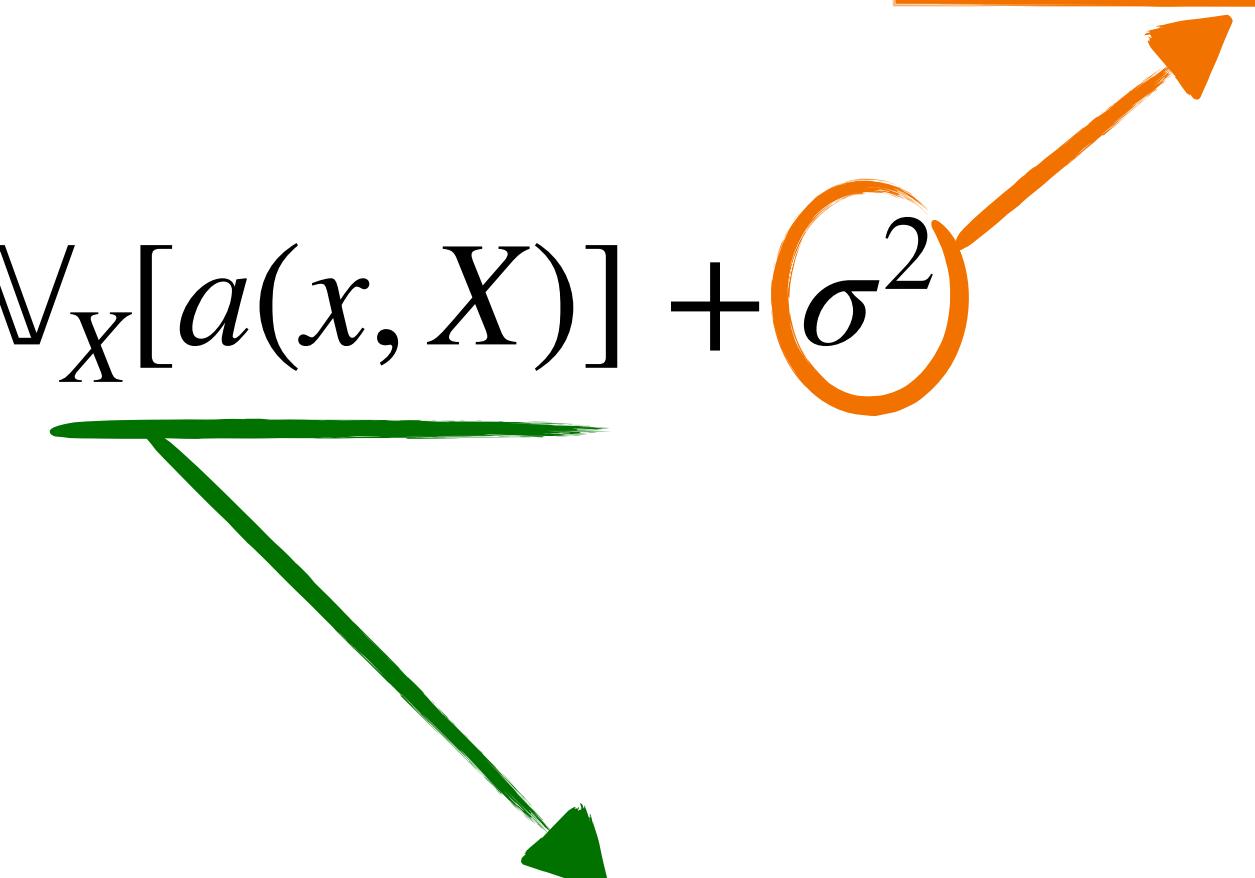
$$Q(a) = \mathbb{E}_x (\text{bias}_X^2 a(x, X)) + \mathbb{E}_x \mathbb{V}_X[a(x, X)] + \sigma^2$$

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

смещение предсказания алгоритма, усреднённого по всем возможным обучающим выборкам, относительно истины

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

неустранимый *шум* в данных



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

разброс предсказаний алгоритма в зависимости от обучающей выборки

1. Bias-Variance decomposition

- Аналогичное представление функционала:

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

- **Bias** (смещение) - насколько хорошо можно с помощью выбранного метода обучения & семейства алгоритмов приблизиться к идеальному алгоритму

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

где:

$$f(x) = \mathbb{E}(y | x) = \int_{\mathbb{Y}} y p(y | x) dy — \text{идеальный алгоритм регрессии для MSE}$$

$\mathbb{E}_X[a(x, X)]$ — усредненное предсказание алгоритмов, обученных на всех возможных обучающих выборках

1. Bias-Variance decomposition

- Аналогичное представление функционала:

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

- **Bias** (смещение) - насколько хорошо можно с помощью выбранного метода обучения & семейства алгоритмов приблизиться к идеальному алгоритму

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

Как правило:

- смещение большое у простых семейств моделей: e.g. Линейные классификаторы
- и маленькое у сложных семейств моделей: e.g. Глубокие Решающие Деревья

1. Bias-Variance decomposition

- Аналогичное представление функционала:

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

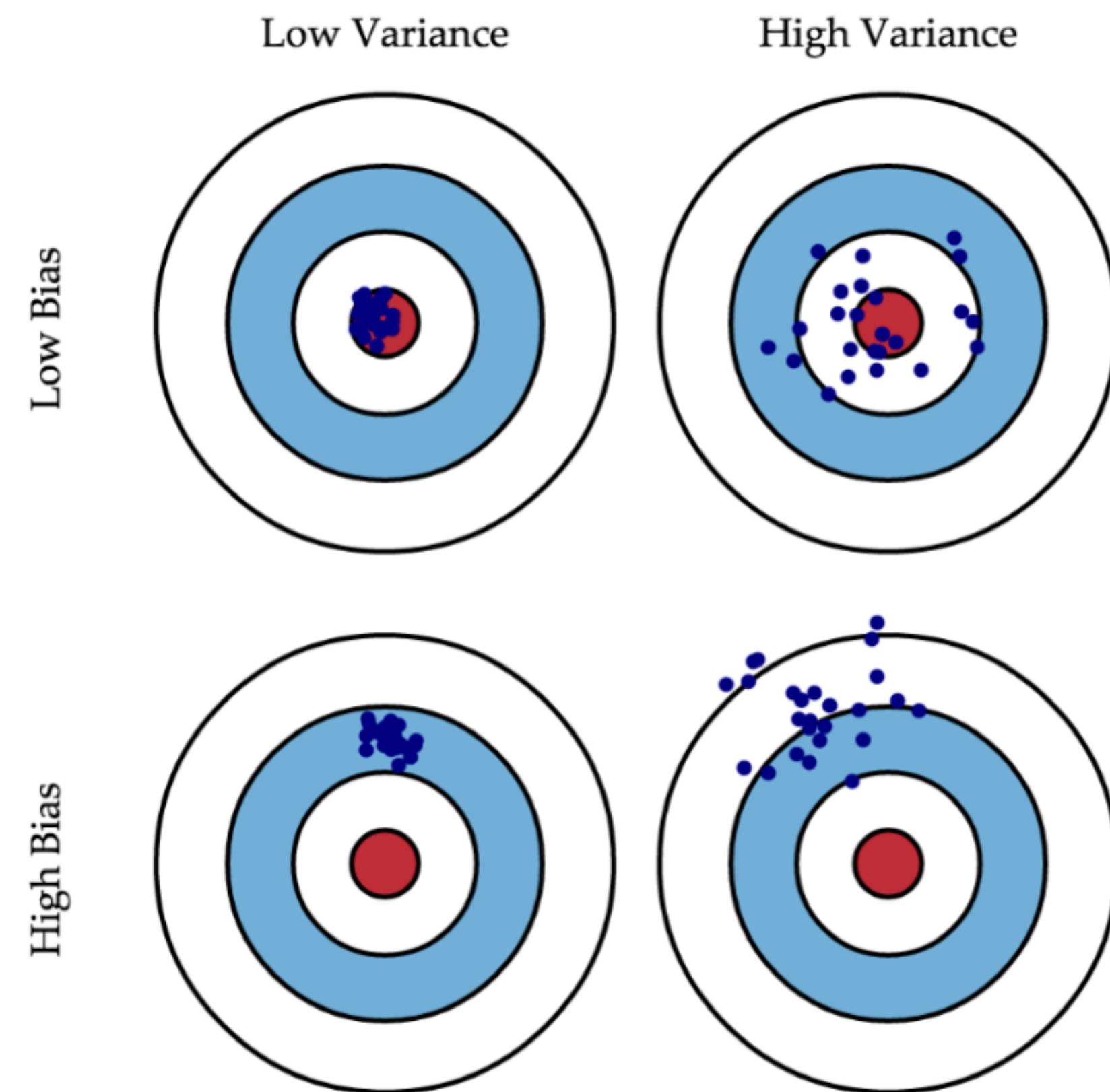
- **Variance** (разброс) - дисперсия предсказаний алгоритмов в зависимости от обучающей выборки, насколько сильно X будет влиять на изменения предсказаний модели

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

Как правило:

- разброс маленький у простых семейств моделей
- и большой у сложных семейств моделей

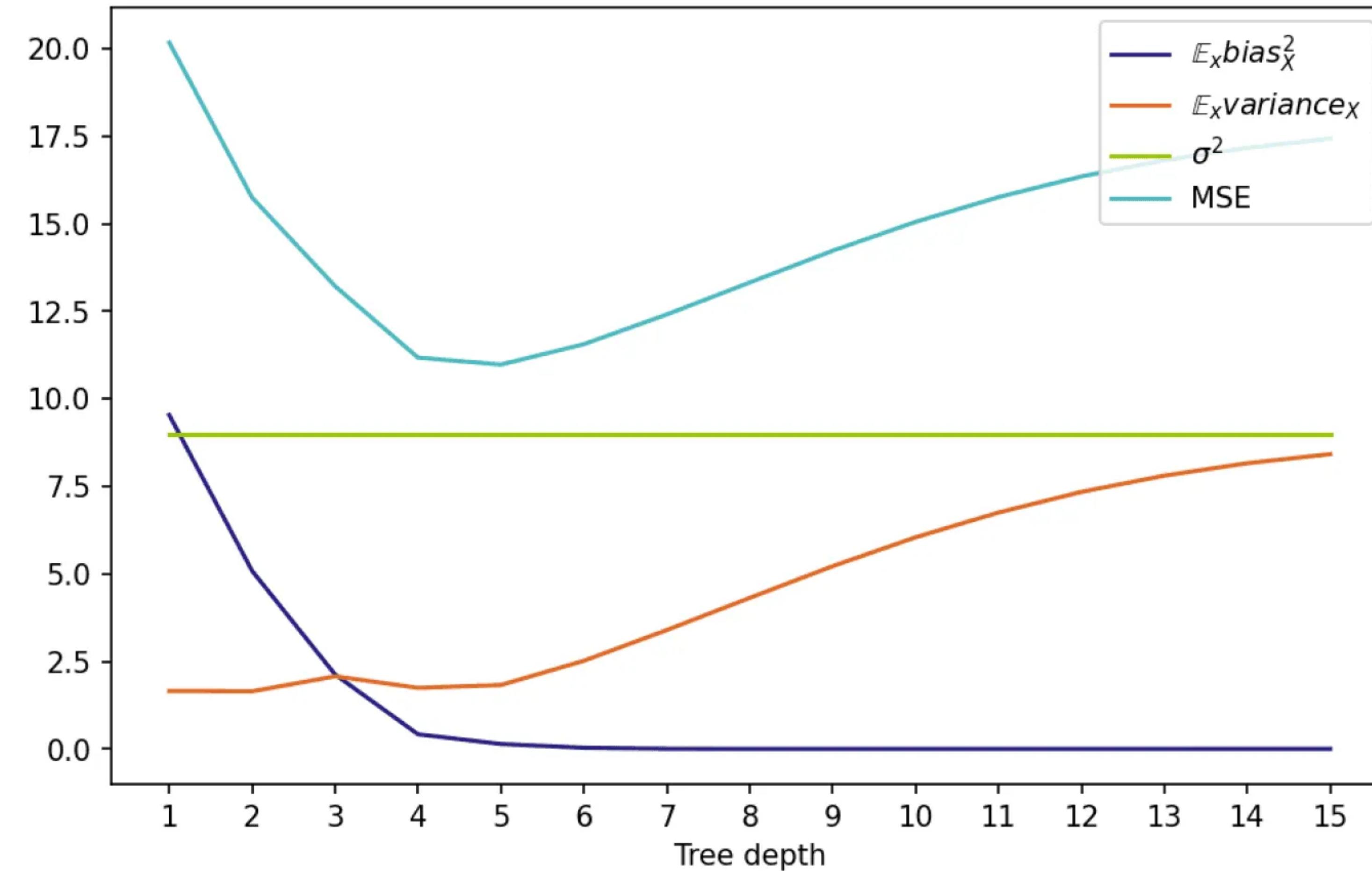
1. Bias-Variance decomposition



- Bias/Смещение - насколько хорошо можно приблизиться к оптимальному алгоритму
- Variance/Дисперсия/Разброс - насколько меняется качество в зависимости от выборки

1. Bias-Variance trade-off

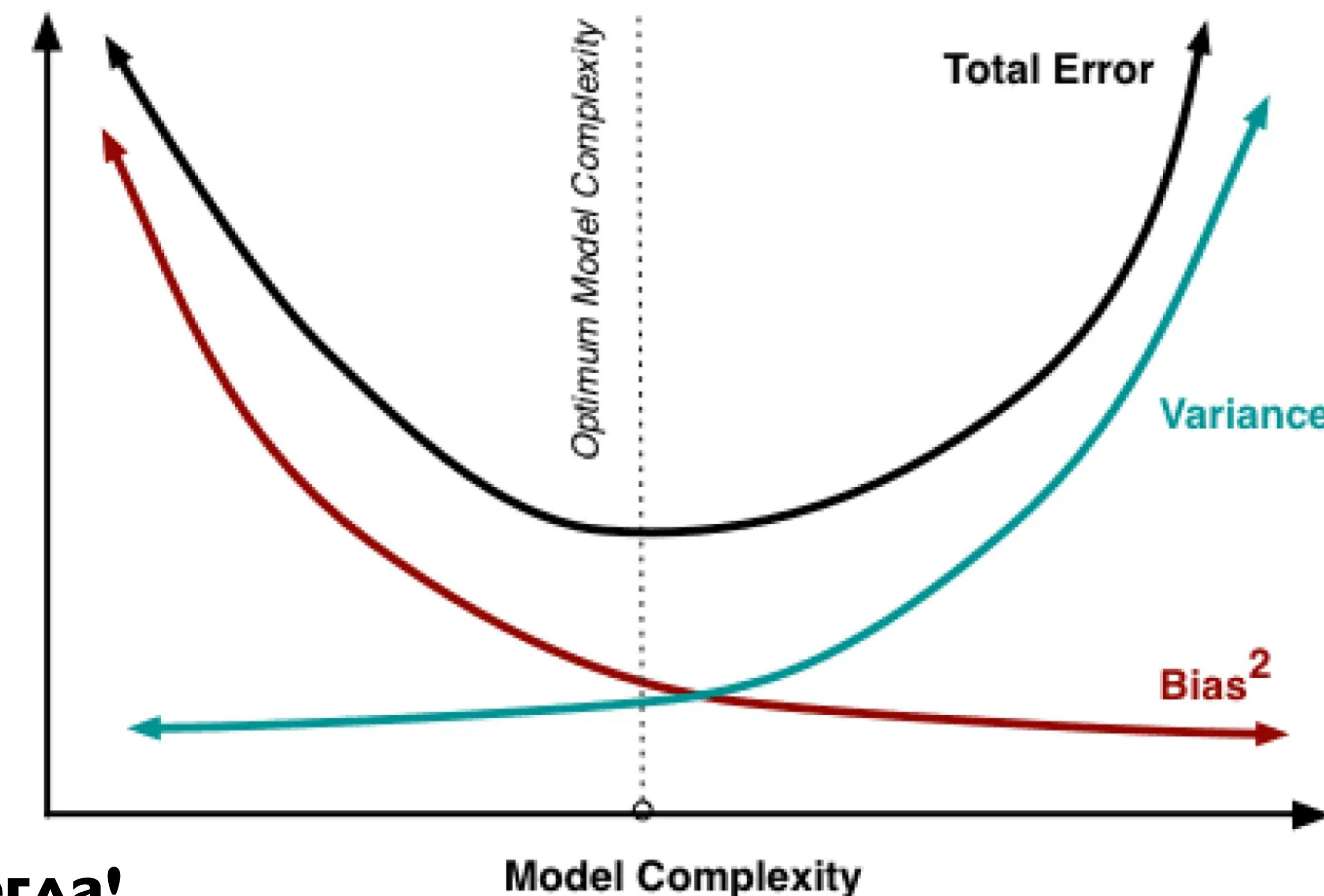
Какую глубину дерева
то выбрать?



1. Bias-Variance trade-off

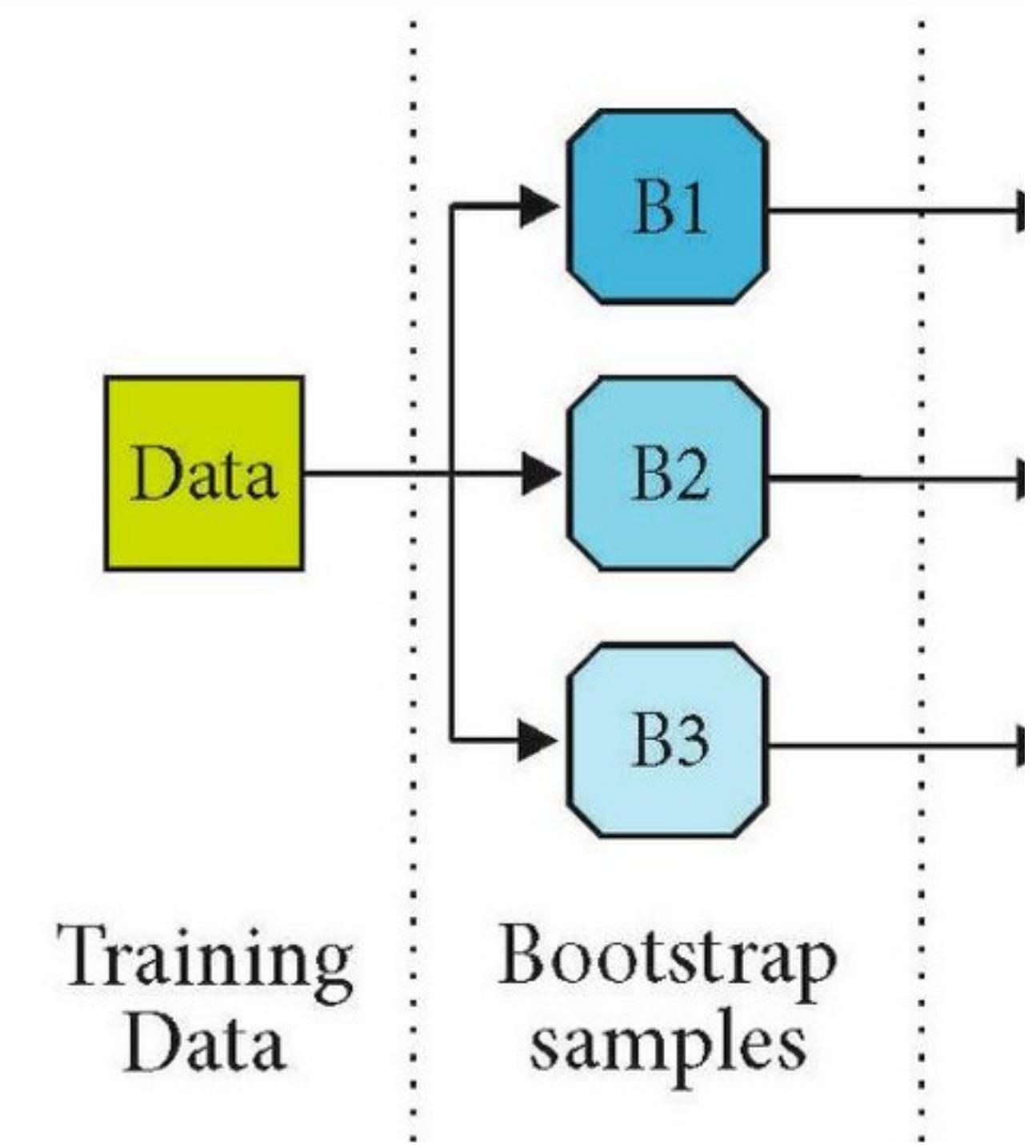
чем выше сложность
обучаемой модели

тем меньше её смещение
тем больше разброс



Так бывает не всегда!

2. Bagging: bootstrap + aggregation



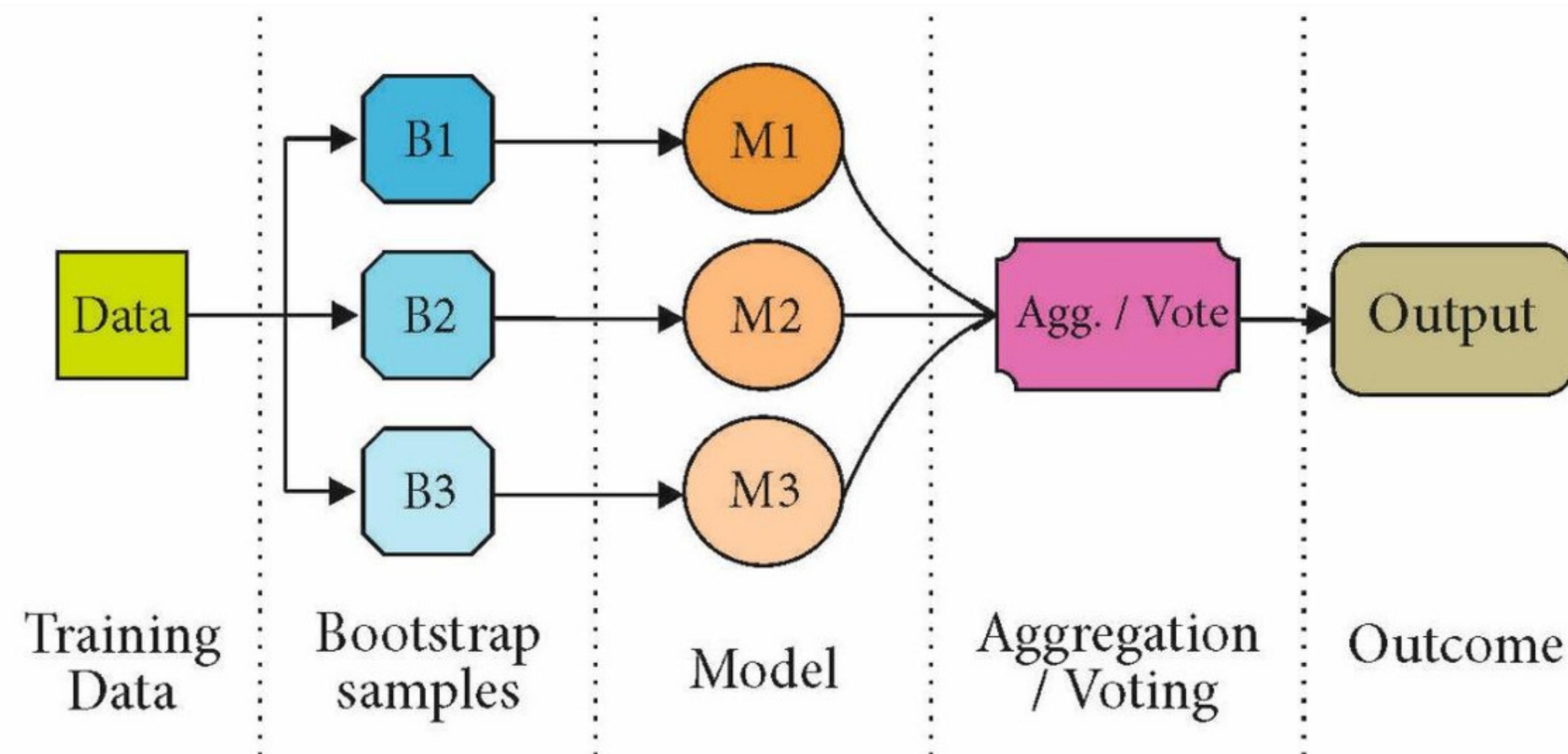
- **Bootstrap**

К новых подвыборок нашего датасета с повторами: $[X^1, X^2, \dots, X^K]$

Каждая позиция в каждом новом наборе данных может быть отдана любому объекту равновероятно

То есть, некоторые объекты встречаются несколько раз, другие ни одного

2. Bagging: bootstrap + aggregation



- **Aggregation**

На каждом датасете обучим выбранную нами базовую архитектуру: $b_i(x) = b(x, X^i)$

Объединим все модели в единый ансамбль: $a(x) = \frac{1}{K}(b_1(x) + \dots + b_k(x))$

когда мы берём
матожидание по всем
обучающим выборкам X ,
то в эти выборки
включены также все
подвыборки, полученные
бутстрепом

2. Bagging: bootstrap + aggregation

- Функционал ошибки через BVD

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

- Рассмотрим Bias (Смещение) ансамбля относительно одной модели

$$\begin{aligned} \text{bias}_X a(x, X) &= f(x) - \mathbb{E}_X[a(x, X)] = f(x) - \mathbb{E}_X \left[\frac{1}{k} \sum_{i=1}^k b(x, X^i) \right] = \\ &= f(x) - \frac{1}{k} \sum_{i=1}^k \mathbb{E}_X [b(x, X^i)] = f(x) - \frac{1}{k} \sum_{i=1}^k \mathbb{E}_X [b(x, X)] = f(x) - \mathbb{E}_X b(x, X) \\ &= f(x) - \mathbb{E}_X b(x, X) = \text{bias}_X b(x, X) \end{aligned}$$

2. Bagging: bootstrap + aggregation

- Функционал ошибки через BVD

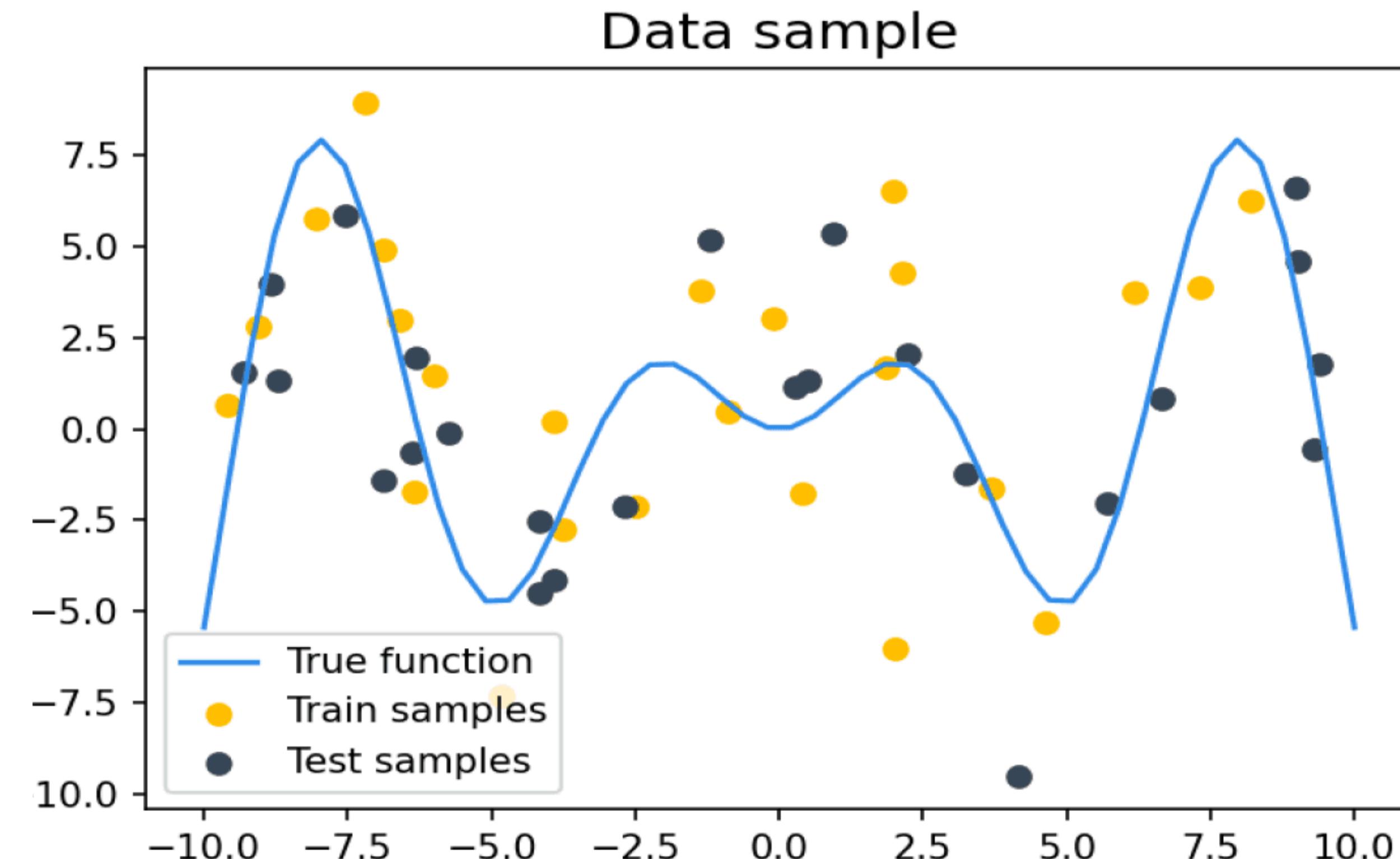
$$Q(a) = \mathbb{E}_x(\text{bias}_X^2 a(x, X)) + \mathbb{E}_x \mathbb{V}_X[a(x, X)] + \sigma^2$$

- Bias (Смещение) ансамбля = Bias одной модели
- Variance (разброс) ансамбля **в К раз меньше** Variance одной модели, при условии некоррелированности базовых алгоритмов !

Бэггинг позволяет объединить несмешенные, но чувствительные к обучающей выборке алгоритмы

—
В несмешенный ансамбль с низкой дисперсией!

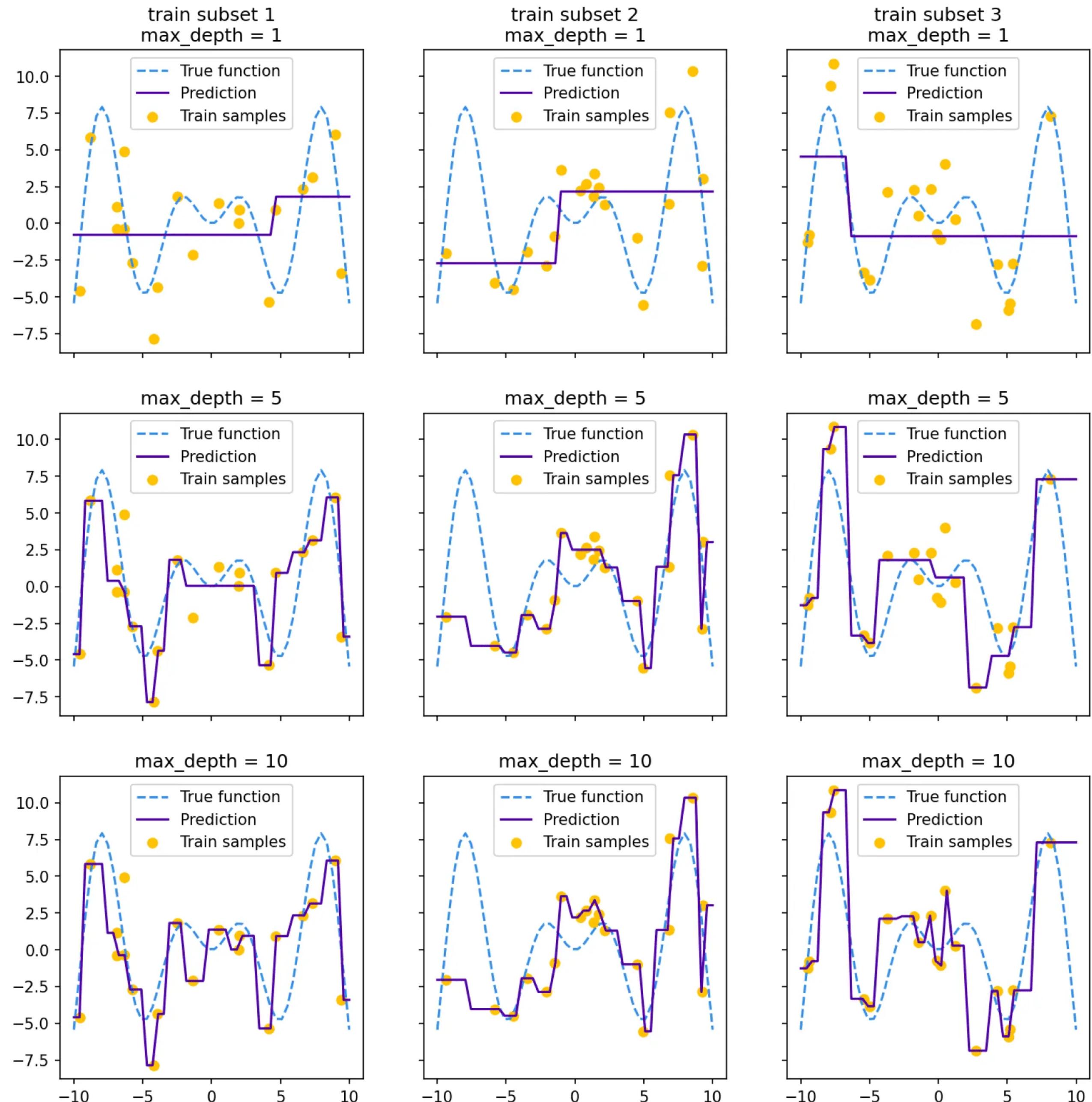
2. Bagging: bootstrap + aggregation



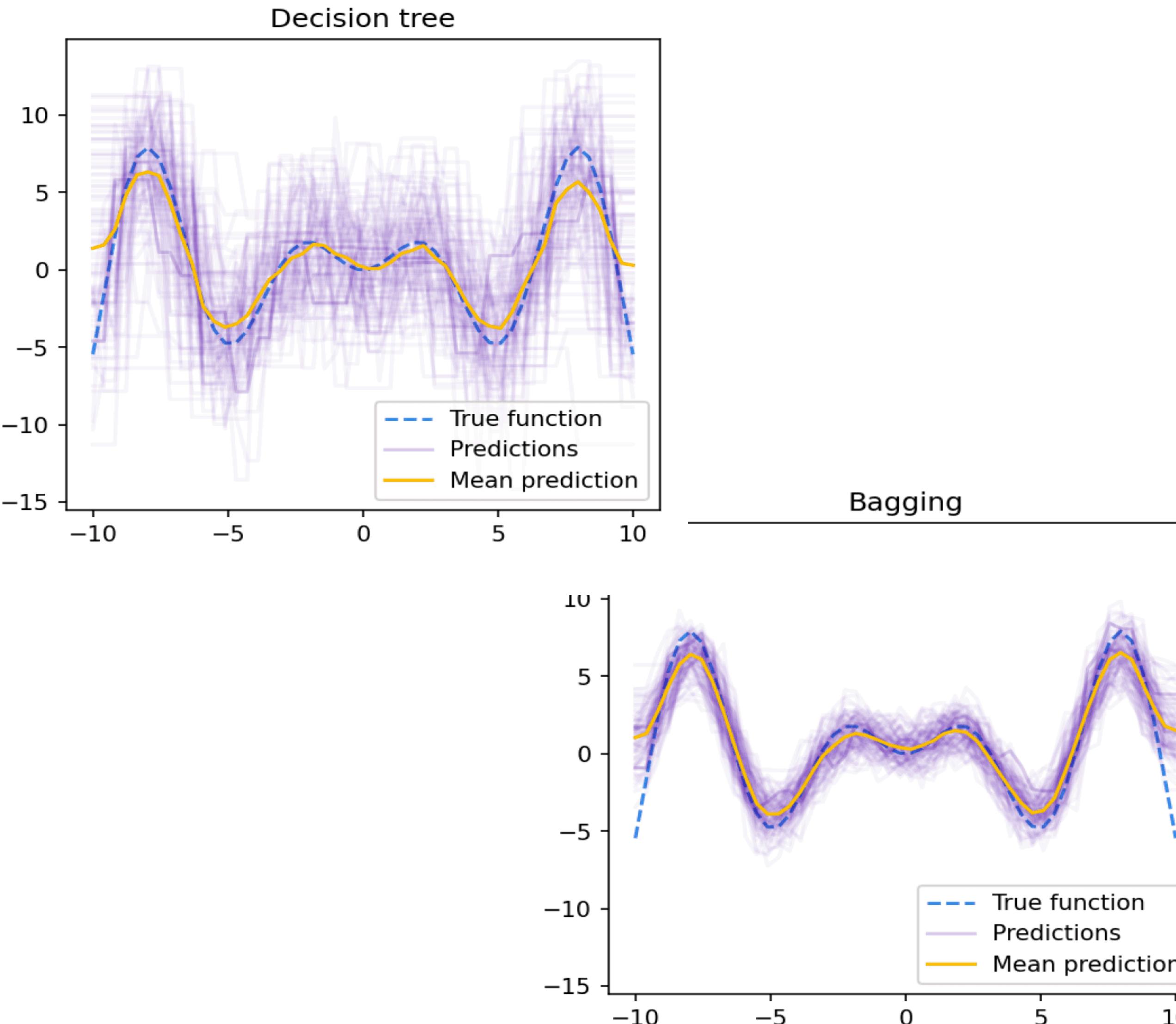
- Функция:
 $y(x, \varepsilon) = x \sin(x) + \mathcal{N}(0, 9)$
- Обучим 100 решающих деревьев глубины 7 на случайных выборках размера 20
- Обучим 100 раз бэггинг над решающими деревьями на случайных выборках размера 20

Decision trees with different maximum depths

- Пример с решающими деревьями разной глубины, обученных на трех разных подвыборках X
- Один столбец — одна подвыборка данных для обучения деревьев



2. Bagging: bootstrap + aggregation



- Функция:
 $y(x, \varepsilon) = x \sin(x) + \mathcal{N}(0, 9)$
- Обучим 100 решающих деревьев глубины 7 на случайных выборках размера 20
- Обучим 100 раз бэггинг над решающими деревьями на случайных выборках размера 20

3. BVD: почему композиции это круто

- Посчитаем ошибку MSE на базовой модели для всех данных \mathbb{X} :

$$\mathbb{E}_x (b_j(x) - y(x))^2 = \underbrace{\mathbb{E}_x \gamma_j^2(x)}_{\gamma_j(x)}$$

- Предположим:

$$\begin{cases} \mathbb{E}_x \gamma_j(x) = 0 \\ \mathbb{E}_x \gamma_j(x) \gamma_i(x) = 0, i \neq j \end{cases}$$

- ошибки равноценна в обе стороны и некоррелированы

3. BVD: почему композиции это круто

- Предположим:

$$\begin{cases} \mathbb{E}_x \gamma_j(x) = 0 \\ \mathbb{E}_x \gamma_j(x)\gamma_i(x) = 0, i \neq j \end{cases}$$

- ошибки равнозначны в обе стороны и некоррелированы

- **Ошибка для модели усреднения: падает в k раз**

$$\mathbb{E}_x(a(x) - y(x))^2 = \mathbb{E}_x\left(\frac{1}{k} \sum \gamma_j(x)\right)^2 = \frac{1}{k^2} \mathbb{E}_x\left(\sum \gamma_j^2(x) + \sum \gamma_i(x)\gamma_j(x)\right) = \frac{1}{k} \mathbb{E}_x \gamma_j^2(x)$$

3. BVD: откуда?

- Ошибка для модели усреднения: падает в k раз
- **Проблема:** много предположений! Как сделать на практике?
- Разложим ошибку на **BVD** - он позволит подобрать подходящие базовые модели!
- Допустим, **среднеквадратичный риск!**

$$\exists \rho(x, y) \rightarrow R(a) = \int_{\mathbb{X}} \int_{\mathbb{Y}} \rho(x, y) \cdot (a(x) - y)^2 dy dx = \mathbb{E}_{x,y} (a(x) - y)^2$$

Распределение пар объект-ответ

Плотность в точке на значение ошибки в ней

3. BVD: откуда?

- Допустим, **среднеквадратичный риск!**

$$\exists \rho(x, y) \longrightarrow R(a) = \int_{\mathbb{X}} \int_{\mathbb{Y}} \rho(x, y)(a(x) - y)^2 dy dx = \mathbb{E}_{x,y}(a(x) - y)^2$$

- Факты про $R(a)$:

- $R(a) = \mathbb{E}_{x,y}(y - \mathbb{E}(y|x))^2 + \mathbb{E}_{x,y}(\mathbb{E}(y|x) - a(x))^2$
- $\hat{a} = \mathbb{E}(y|x)$ - оптимальный алгоритм, где достигается минимум $R(a)$

3. BVD: откуда?

- Факты про **среднеквадратичный риск**:
 - $R(a) = \mathbb{E}_{x,y}(y - \mathbb{E}(y|x))^2 + \mathbb{E}_{x,y}(\mathbb{E}(y|x) - a(x))^2$
 - $\hat{a} = \mathbb{E}(y|x)$ - оптимальный алгоритм
- Но ведь мы **не знаем распределения** $\rho(x, y)!$
- Пусть есть метод обучения модели:
$$\mu : (\mathbb{X}, \mathbb{Y}) \rightarrow \mathcal{A} \text{ — из данных в семейство моделей}$$
$$\mu(\mathbb{X})(x) \Leftrightarrow a(\mathbb{X}, x)$$

3. BVD: откуда?

- Факты про среднеквадратичный риск:

- $R(a) = \mathbb{E}_{x,y}(y - \mathbb{E}(y|x))^2 + \mathbb{E}_{x,y}(\mathbb{E}(y|x) - a(x))^2$

- Есть функция обучения модели:

- $\mu : (\mathbb{X}, \mathbb{Y}) \rightarrow \mathcal{A}$ - из данных в алгоритм
- $\mu(\mathbb{X})(x) \Leftrightarrow a(\mathbb{X}, x)$

- Посчитаем на нем качество метода обучения как среднее $R(a)$ по всем выборкам \mathbb{X} :

$$\text{Loss}(a) = \mathbb{E}_X[\mathbb{E}_{x,y}[y - a(x, \mathbb{X})]^2] = \mathbb{E}_X[R(a)]$$

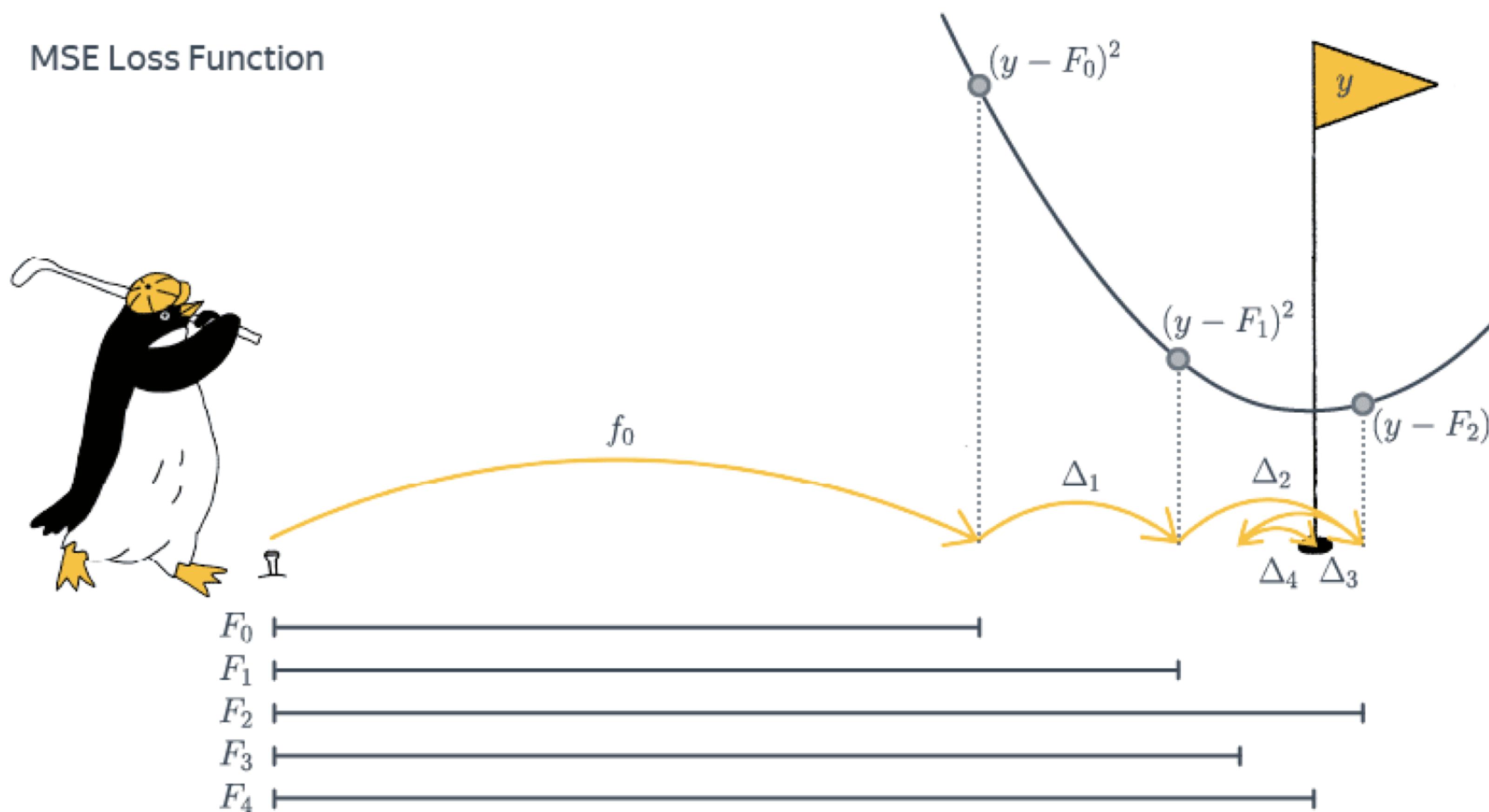
- Подставим формулу $R(a)$ в формулу Loss и получим BVD!

$$L(\mu) = \underbrace{\mathbb{E}_{x,y}\left[(y - \mathbb{E}[y|x])^2\right]}_{\text{шум}} + \\ + \underbrace{\mathbb{E}_x\left[\left(\mathbb{E}_X[\mu(X)] - \mathbb{E}[y|x]\right)^2\right]}_{\text{смещение}} + \underbrace{\mathbb{E}_x\left[\mathbb{E}_X\left[\left(\mu(X) - \mathbb{E}_X[\mu(X)]\right)^2\right]\right]}_{\text{разброс}}.$$

$$e_i = b_1(x_i) - y_i$$

Так не будет: $b_2(x_i) = e_i \rightarrow L = y_i - b_1(x_i) - b_2(x_i) = \underbrace{y_i - b_1(x_i)}_{e_i} - e_i = e_i - e_i = 0$

$$b_2(x_i) \sim e_i \longrightarrow b_3(x_i) = b_2(x_i) - e_i$$



$$b_1 : x_i \rightarrow y_i$$

$$e_1(x_i) = y_i - b_1(x_i)$$

$$b_2 : x_i \rightarrow e_1(x_i)$$

$$e_2(x_i) = e_1(x_i) - b_2(x_i)$$

$$b_3 : x_i \rightarrow e_2(x_i)$$

$$e_3(x_i) = e_2(x_i) - b_3(x_i)$$

$$\text{MSE}(x_i) = \frac{1}{2}(y_i - b_1(x_i))^2 \rightarrow \min$$

$$\frac{\partial \text{MSE}}{\partial b_1} = (\frac{1}{2}(y_i - b_1(x_i))^2)' = -(y_i - b_1(x_i)) = \nabla L$$

$$e_1(x_i) = y_i - b_1(x_i) = -\nabla L$$