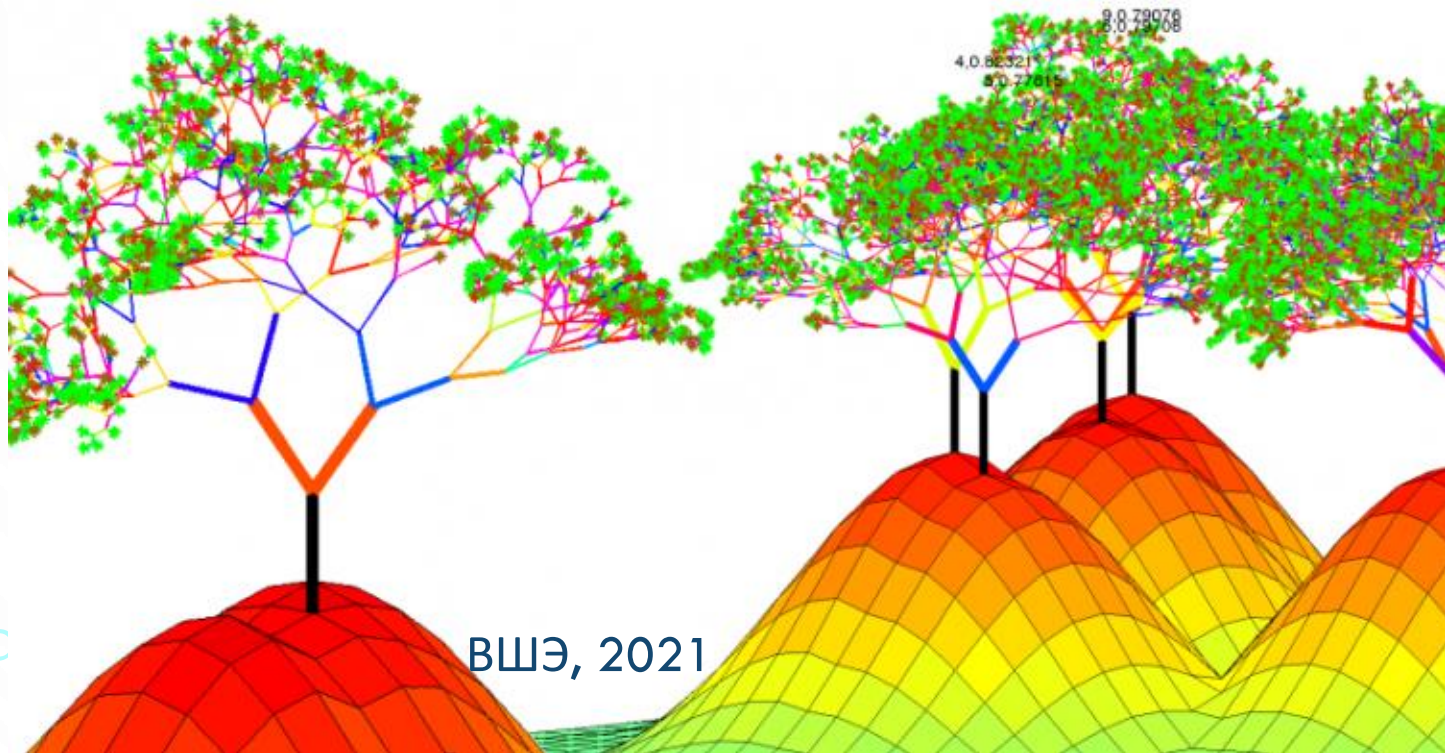


# Занятие 6


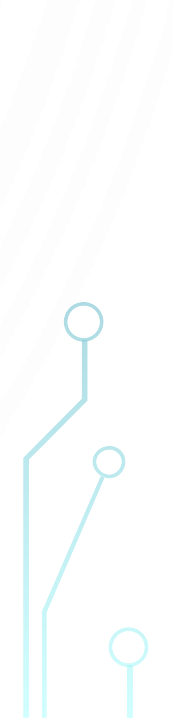
## Композиции алгоритмов

Кантонистова Е.О.





# ПЛАН ЗАНЯТИЯ

- Разложение ошибки
  - Случайный лес
  - Градиентный бустинг
  - Практика
- 
- 



# РАЗЛОЖЕНИЕ ОШИБКИ

# РАЗЛОЖЕНИЕ ОШИБКИ (BIAS-VARIANCE DECOMPOSITION)

Зачастую для улучшения качества модели необходимо понять, из-за чего возникает ошибка в предсказаниях.

- *Модель переобучена?*
- *Модель плохо предсказывает целевую переменную?*
- *В самих данных много неточностей (шумов)*

# РАЗЛОЖЕНИЕ ОШИБКИ (BIAS-VARIANCE DECOMPOSITION)

Зачастую для улучшения качества модели необходимо понять, из-за чего возникает ошибка в предсказаниях.

**Утверждение:** ошибку модели  $a(x)$  можно представить в виде

$$\text{Err}(x) = \text{Bias}^2(a(x)) + \text{Var}(a(x)) + \sigma^2.$$

# РАЗЛОЖЕНИЕ ОШИБКИ (BIAS-VARIANCE DECOMPOSITION)

Зачастую для улучшения качества модели необходимо понять, из-за чего возникает ошибка в предсказаниях.

**Утверждение:** ошибку модели  $a(x)$  можно представить в виде

$$\text{Err}(x) = \text{Bias}^2(a(x)) + \text{Var}(a(x)) + \sigma^2.$$

- $\text{Bias}(a(x))$  - **смещение**.

*Смещение показывает, насколько в среднем модель хорошо предсказывает целевую переменную:*

- ✓ *маленькое смещение - хорошее предсказание*
- ✓ *большое смещение – плохое предсказание*

# РАЗЛОЖЕНИЕ ОШИБКИ (BIAS-VARIANCE DECOMPOSITION)

Зачастую для улучшения качества модели необходимо понять, из-за чего возникает ошибка в предсказаниях.

**Утверждение:** ошибку модели  $a(x)$  можно представить в виде

$$\text{Err}(x) = \text{Bias}^2(a(x)) + \text{Var}(a(x)) + \sigma^2.$$

- $\text{Var}(a(x))$  - **разброс (дисперсия)**.

*Большой разброс означает, что ошибка очень чувствительна к изменению обучающей выборки, т.е.:*

✓ *большой разброс – сильно переобученная модель*

# РАЗЛОЖЕНИЕ ОШИБКИ (BIAS-VARIANCE DECOMPOSITION)

Зачастую для улучшения качества модели необходимо понять, из-за чего возникает ошибка в предсказаниях.

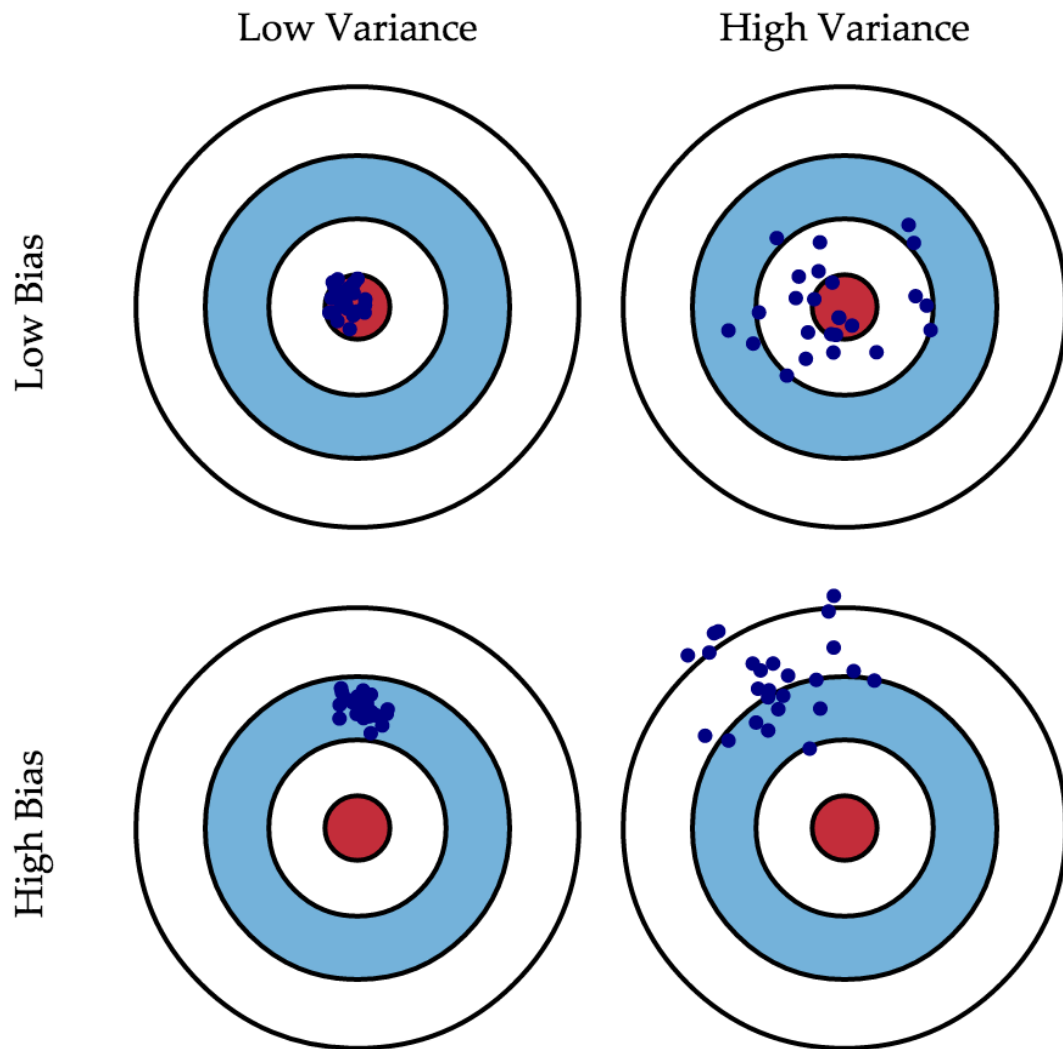
**Утверждение:** ошибку модели  $a(x)$  можно представить в виде

$$\text{Err}(x) = \text{Bias}^2(a(x)) + \text{Var}(a(x)) + \sigma^2.$$

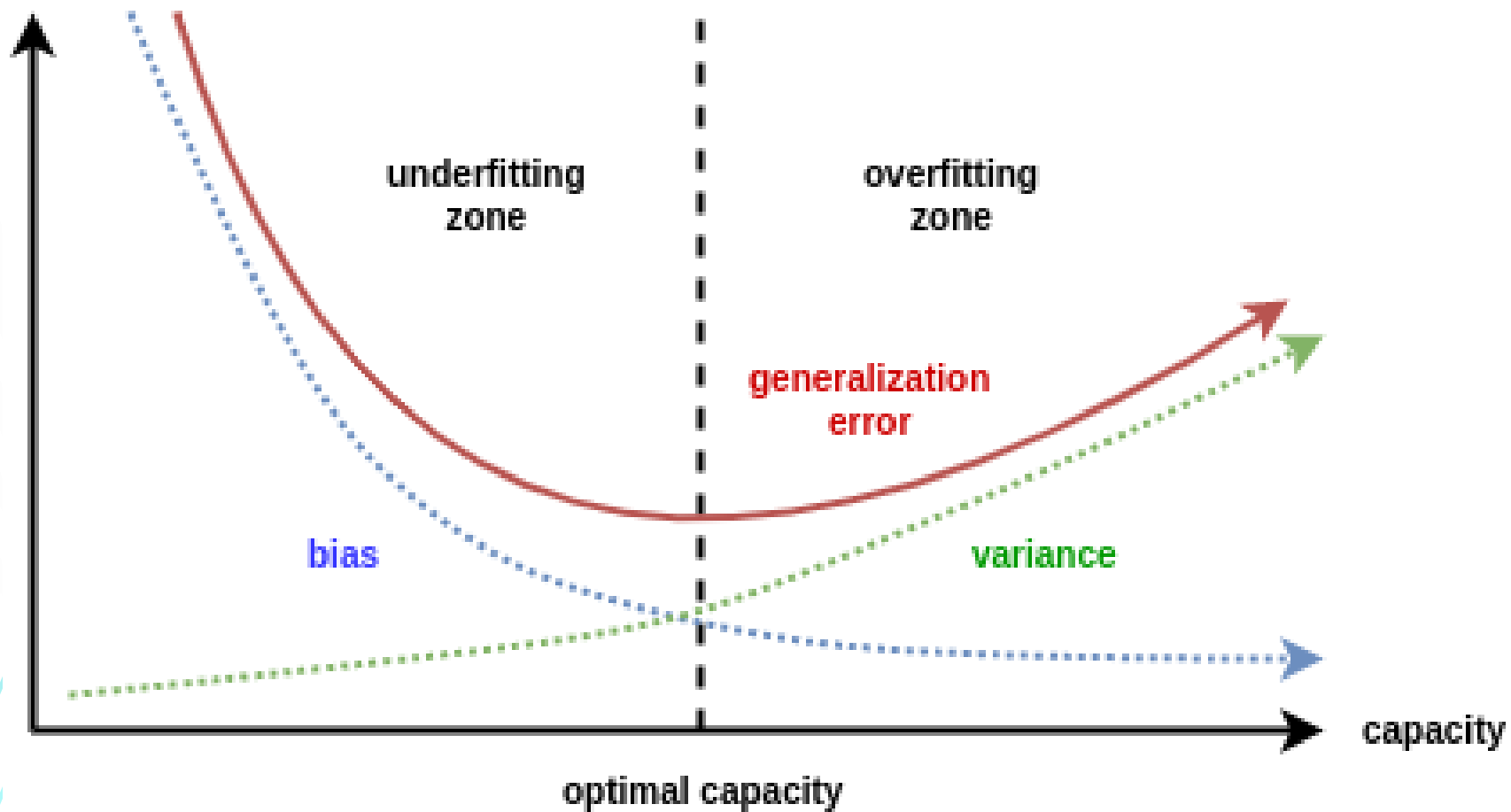
- $\text{Bias}(a(x))$  - **смещение**.
- $\text{Var}(a(x))$  - **разброс (дисперсия)**.
- $\sigma^2$  - неустраняемая ошибка – **шум**.



# СМЕЩЕНИЕ И РАЗБРОС



# BIAS-VARIANCE TRADEOFF





# СЛУЧАЙНЫЙ ЛЕС

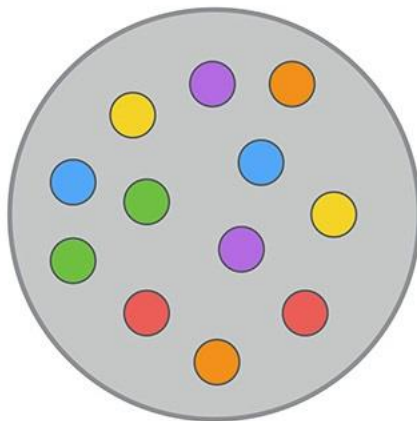
# БУТСТРЭП

Дана выборка  $X$ .

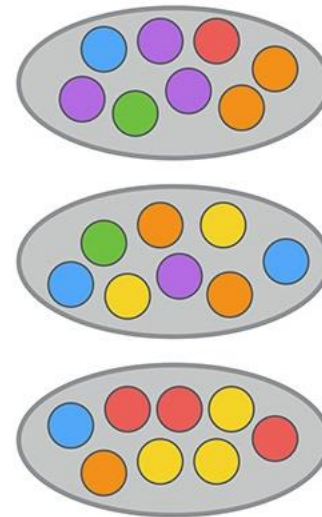
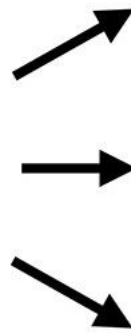
**Бутстрэп:** равномерно возьмем из выборки  $X$   $l$  объектов с возвращением (т.е. в новой выборке будут повторяющиеся объекты). Получим выборку  $X_1$ .

- Повторяем процедуру  $N$  раз, получаем выборки  $X_1, \dots, X_N$ .

Исходная выборка



Бутстрэп выборки

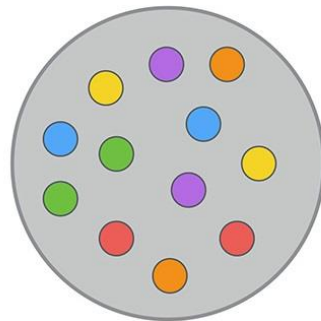


# БЭГГИНГ (BOOTSTRAP AGGREGATION)

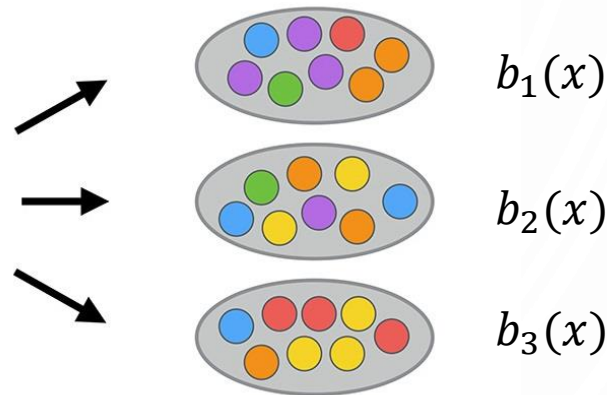
С помощью бутстрэпа мы получили выборки  $X_1, \dots, X_3$ .

- Обучим по каждой из них модель – получим базовые алгоритмы  $b_1(x), \dots, b_3(x)$ .

Исходная выборка



Бутстрэп выборки

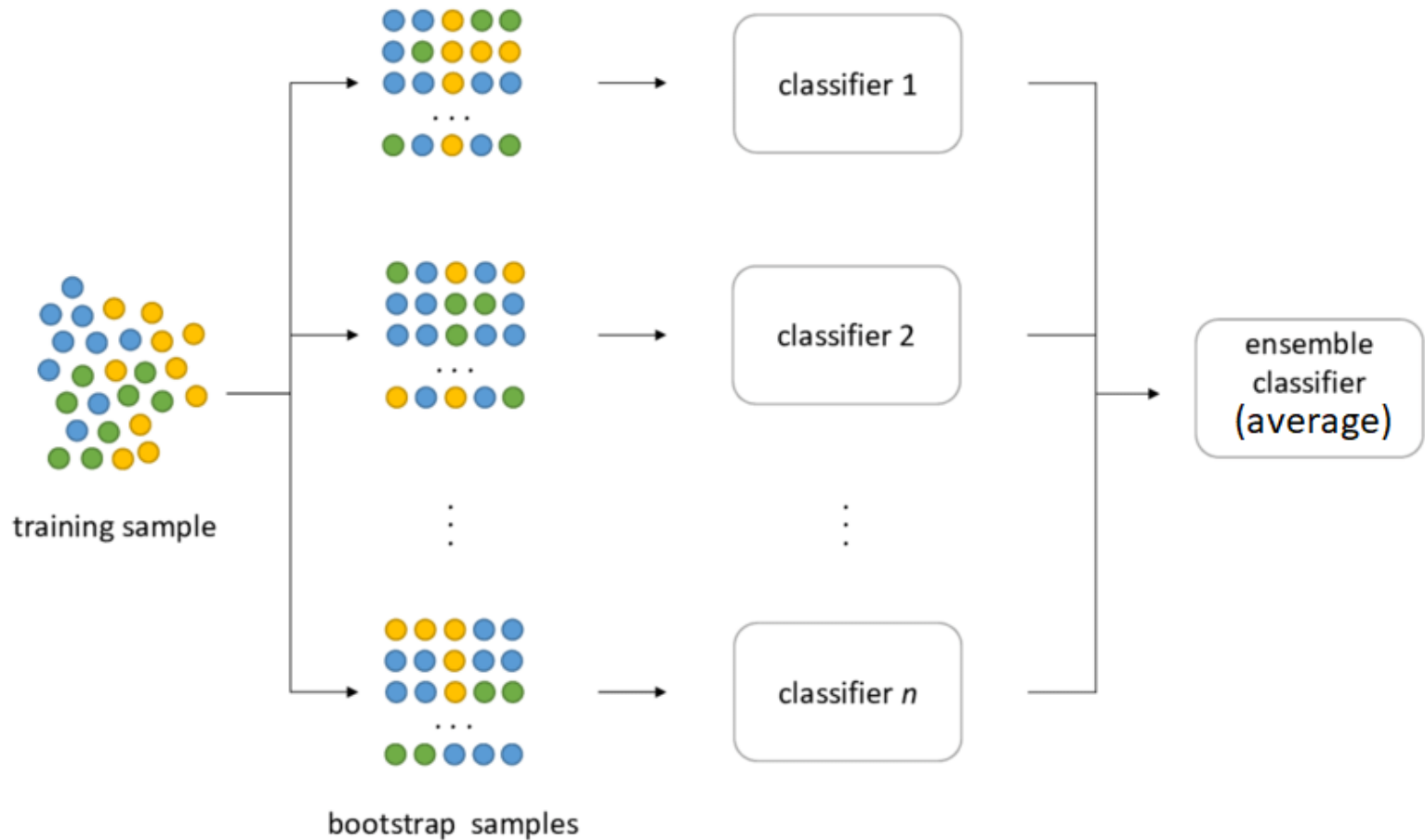


- Построим новую функцию регрессии:

$$a(x) = \frac{1}{3} \sum_{j=1}^3 b_j(x)$$

# БЭГГИНГ (BOOTSTRAP AGGREGATION)

$$a(x) = \frac{1}{N} \sum_{j=1}^N b_j(x)$$



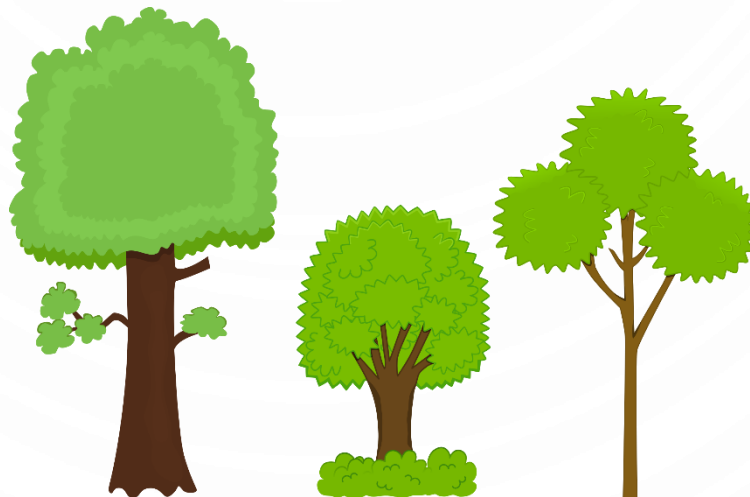
# СМЕЩЕНИЕ И РАЗБРОС У БЭГГИНГА

1) **Бэггинг не ухудшает смещенность модели, т.е. смещение  $a_N(x)$  равно смещению одного базового алгоритма.**

2) **Если базовые алгоритмы достаточно разные, то дисперсия бэггинга  $a_N(x)$  в  $N$  раз меньше дисперсии отдельных базовых алгоритмов.**

# СЛУЧАЙНЫЙ ЛЕС (RANDOM FOREST)

- Возьмем в качестве базовых алгоритмов для бэггинга **решающие деревья**, т.е. каждое случайное дерево  $b_i(x)$  построено по своей подвыборке  $X_i$ .
- В каждой вершине дерева будем искать **разбиение не по всем признакам, а по подмножеству признаков**.







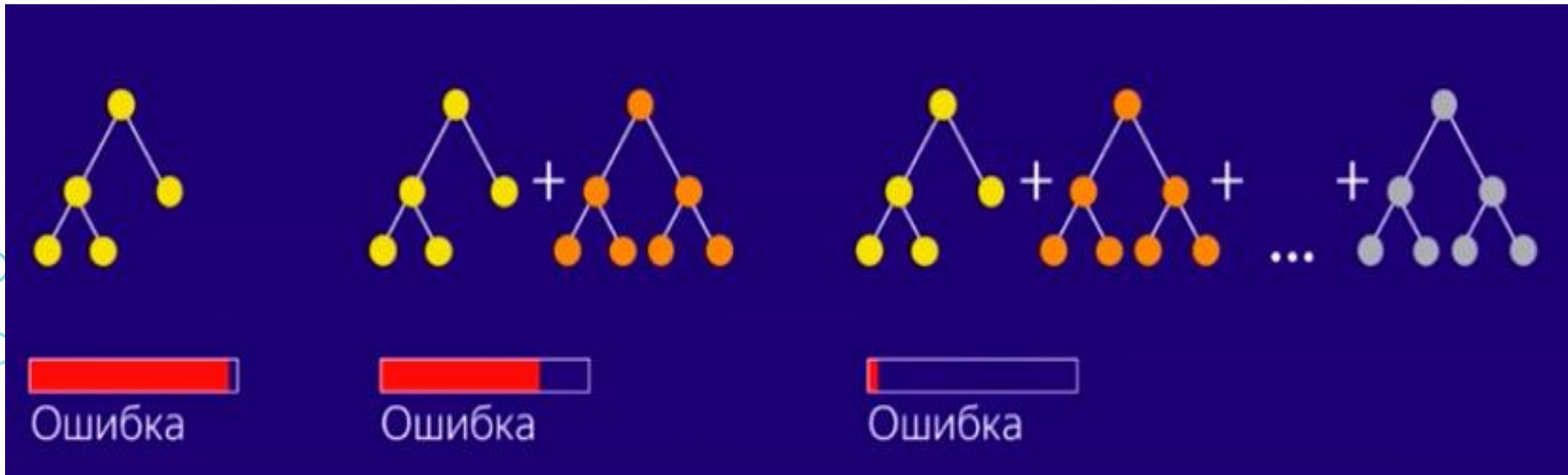
# ГРАДИЕНТНЫЙ БУСТИНГ

# БУСТИНГ

Идея: строим набор алгоритмов, каждый из которых исправляет ошибку предыдущих.

# БУСТИНГ

Идея: строим набор алгоритмов, каждый из которых исправляет ошибку предыдущих.



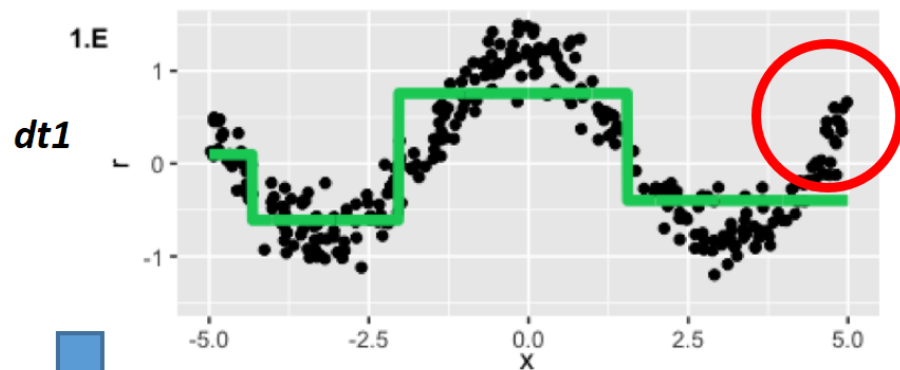
# БУСТИНГ

Пусть  $x$  – объект,  $y$  – правильный ответ на этом объекте.

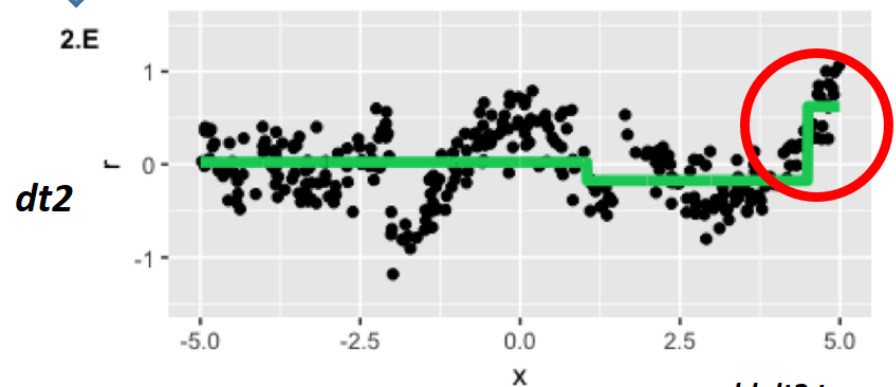
- Бустинг строит деревья *по очереди*, затем предсказания деревьев *суммируются*.
- Например, после двух шагов бустинга композиция имеет вид:
$$a(x) = b_1(x) + b_2(x)$$
- *Следующее решающее дерево  $b_3(x)$  старается предсказать ошибку композиции  $a(x)$* . То есть целевая переменная для этого дерева

$$y_{new} = y - a(x)$$

# БУСТИНГ

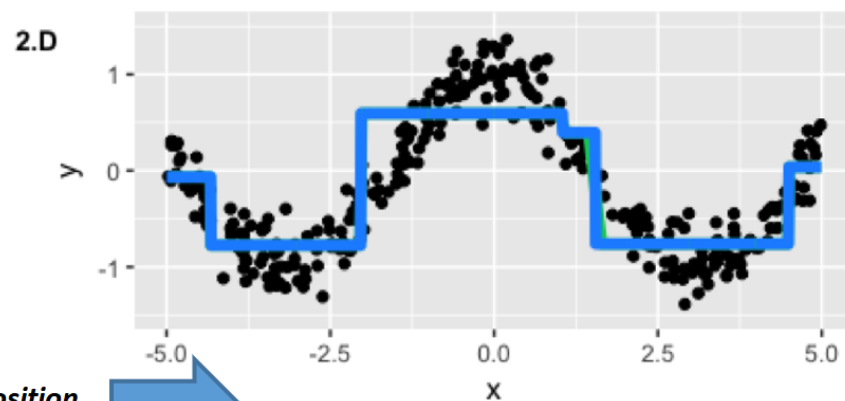


error:  $y - dt1(x)$



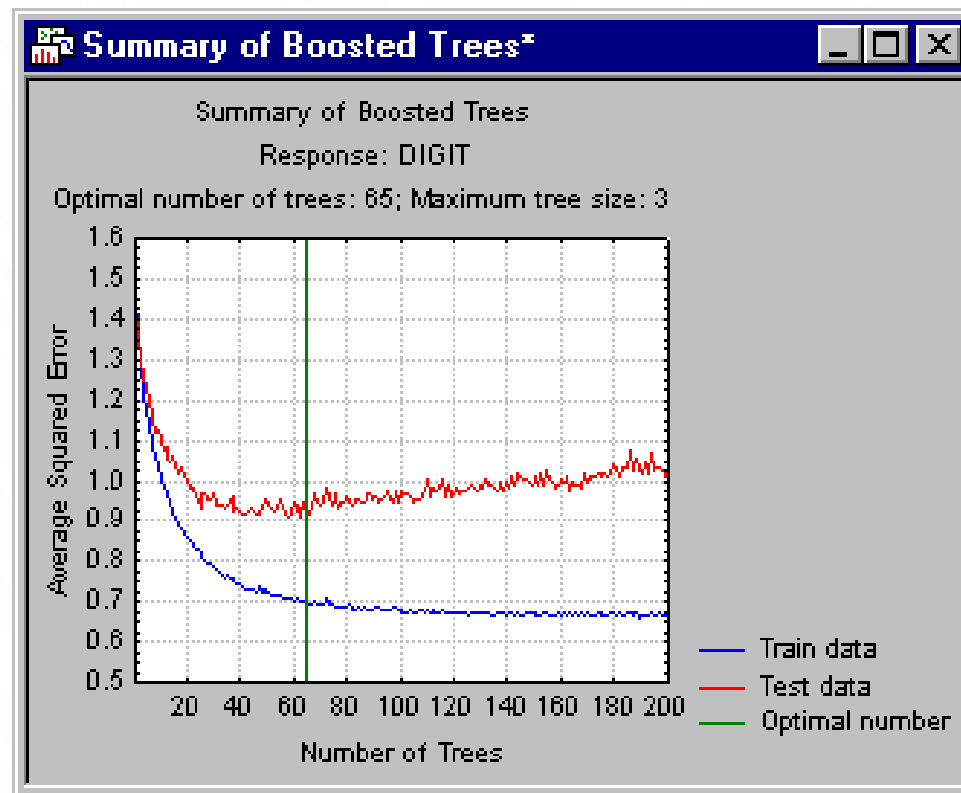
add dt2 to composition

Boosting:  $a(x) = dt1(x) + dt2(x)$



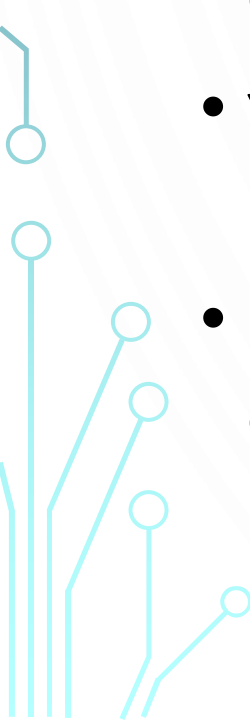
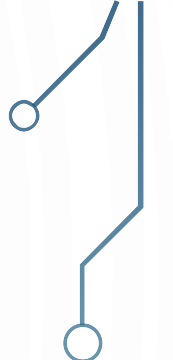
# ПЕРЕОБУЧЕНИЕ БУСТИНГА

- На каждой итерации бустинга мы уменьшаем ошибку на тренировочных данных, то есть, всё больше переобучаемся.
- Поэтому надо подбирать такое число деревьев в бустинге, чтобы ошибка на тесте была наименьшей.





# ИТОГИ

- Оба способа построить композицию алгоритмов – случайный лес и градиентный бустинг – дают отличные сильные модели.
  - У каждой из этих моделей свои особенности, за счёт которых они показывают хорошее качество.
  - На практике нужно пробовать обе модели и смотреть, какая работает лучше.
- 
- 
- 