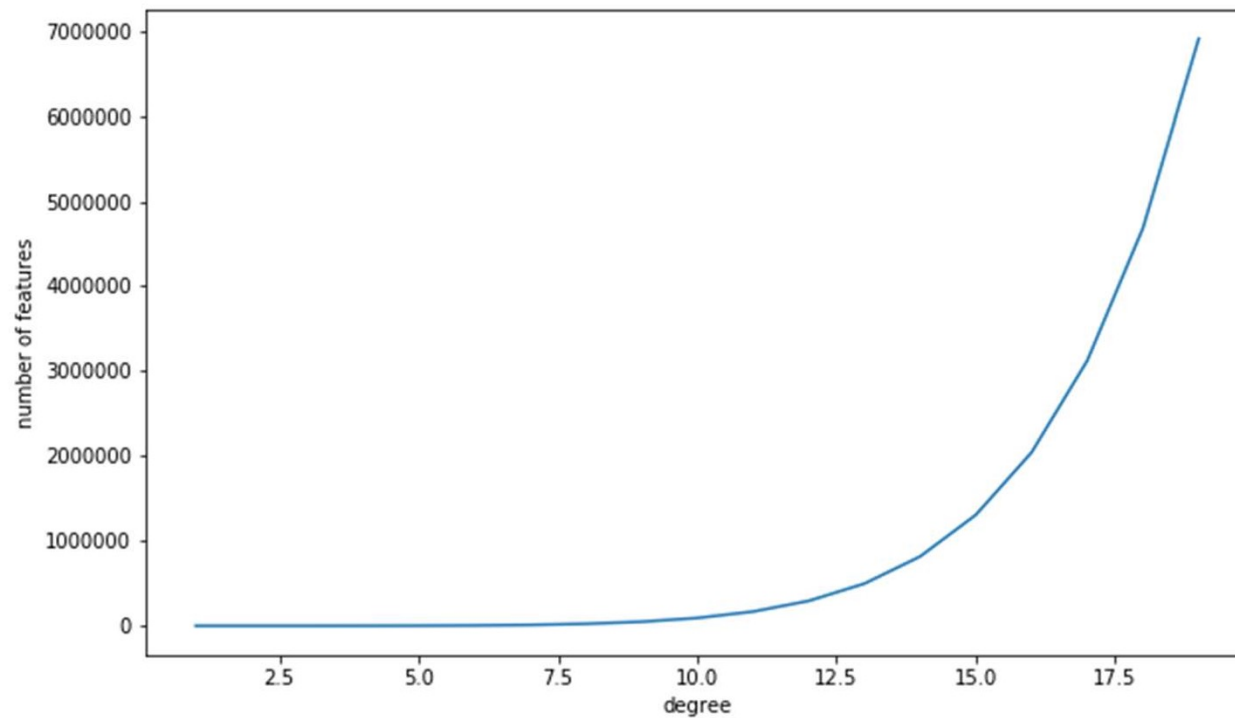
A decorative graphic on the left side of the slide consists of a grid of colored squares. The top row has one teal square. The second row has an orange square followed by a brown square. The third row has an orange-red square, a teal square, and a light brown square. The bottom row has a light brown square, an orange square, an orange-red square, and a brown square.

Бэггинг и случайные леса

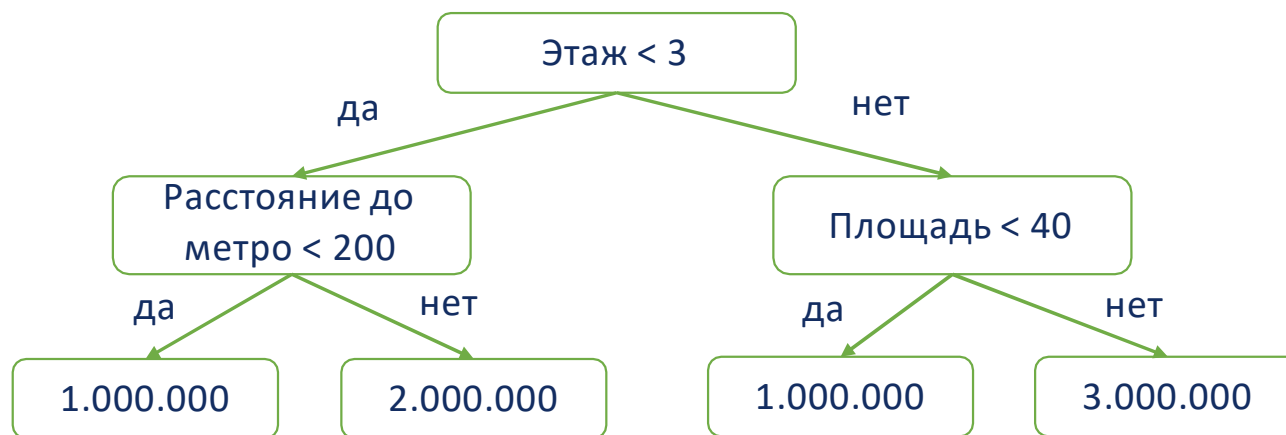
Повторение

Предсказание стоимости квартиры

- Линейная модель с полиномиальными признаками:

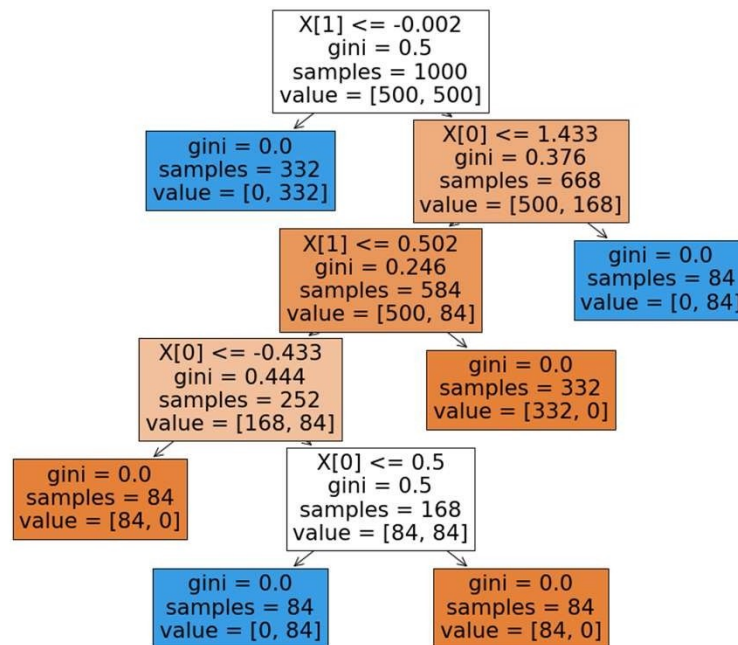
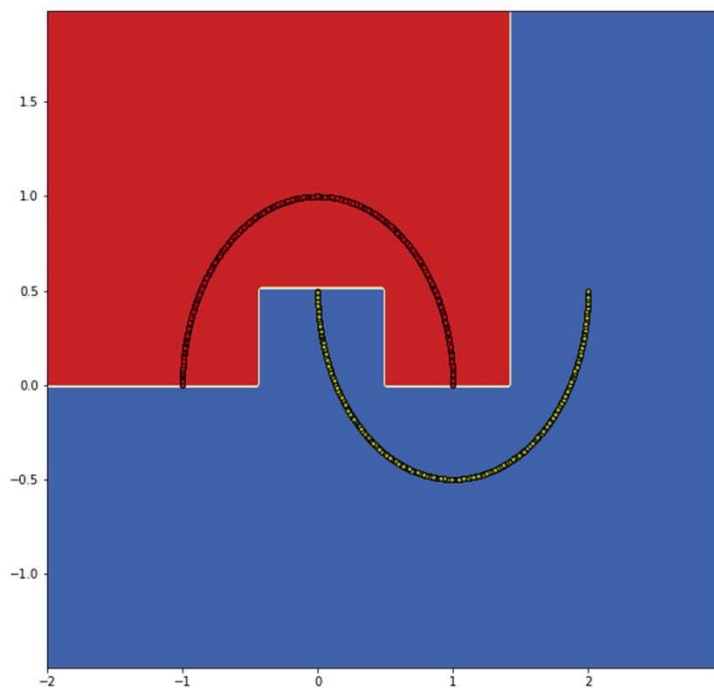


Решающее дерево

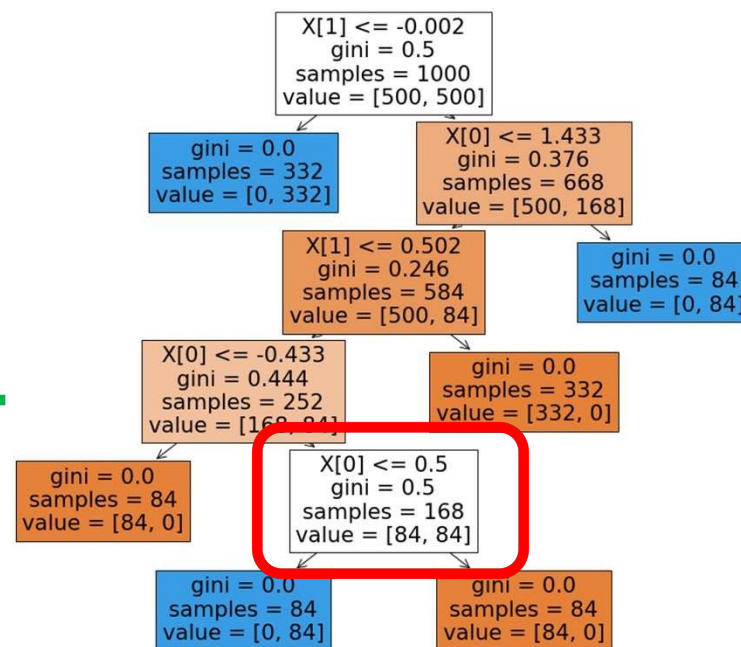
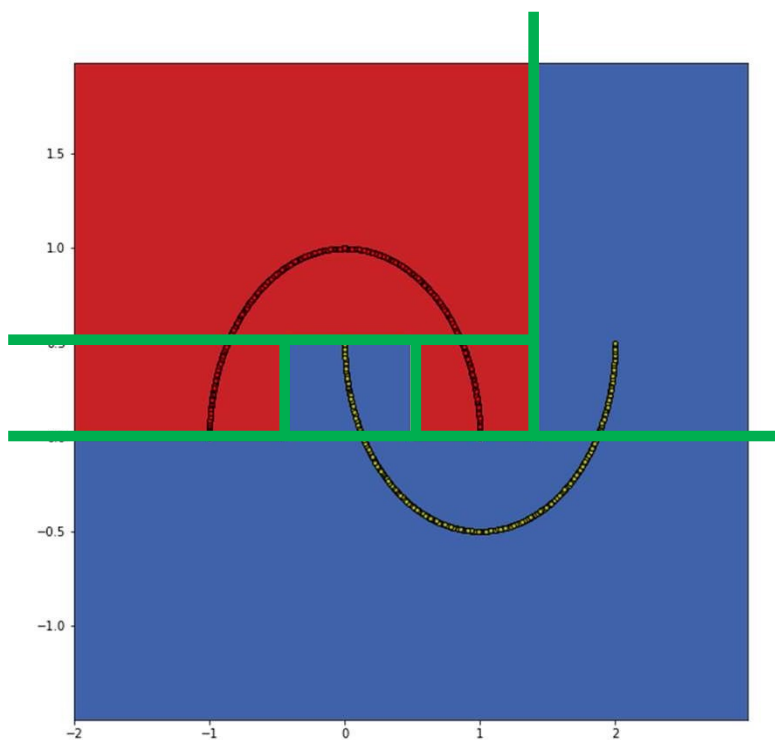


- Внутренние вершины: предикаты $[x_j < t]$
- Листья: прогнозы $c \in \mathbb{Y}$

Решающее дерево



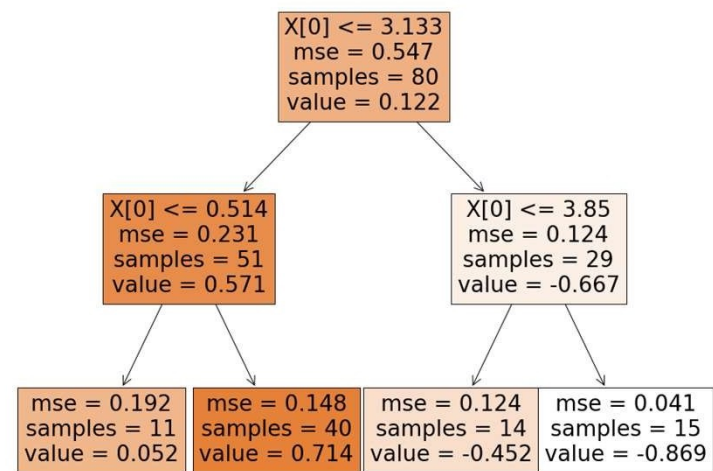
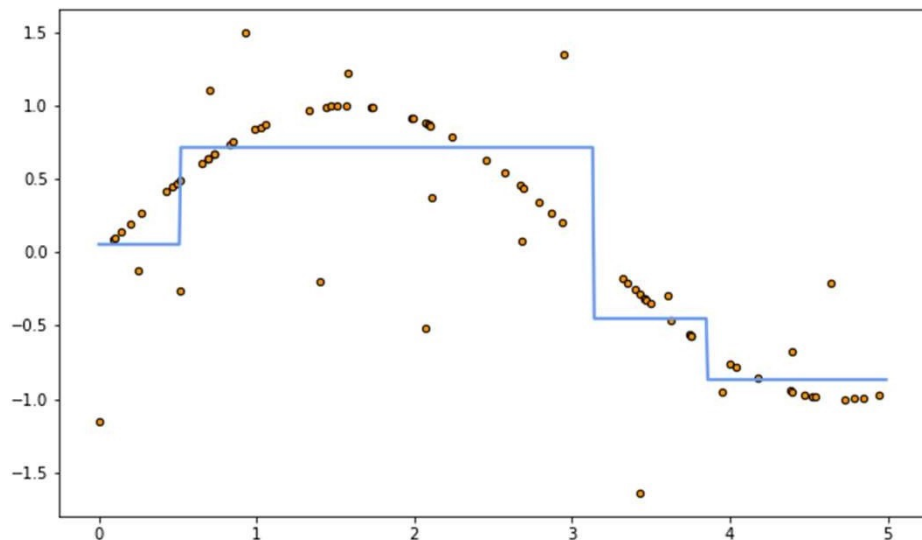
Решающее дерево



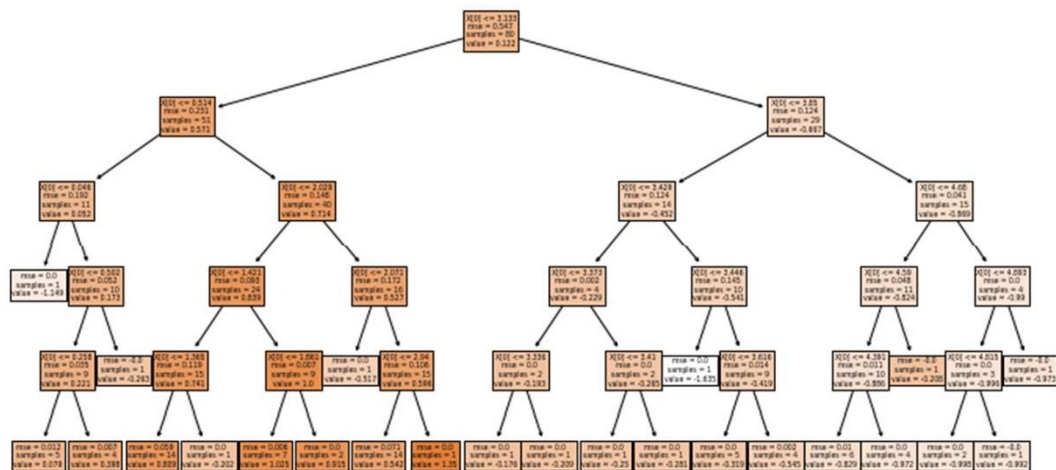
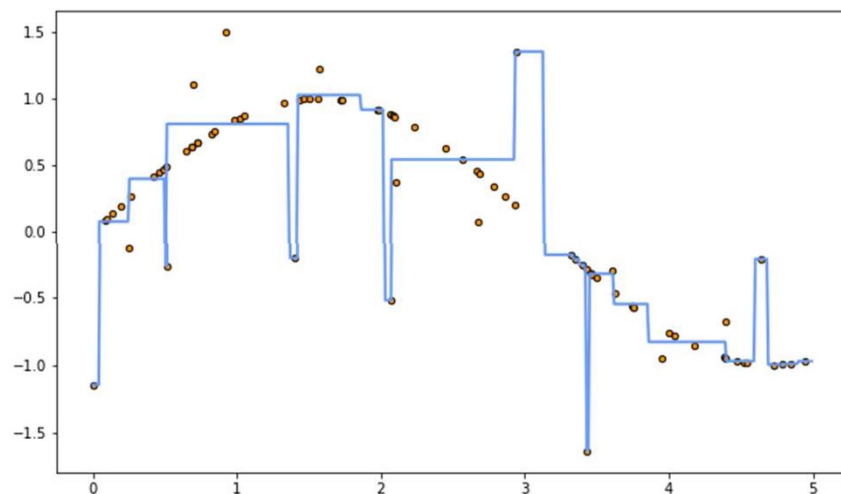
Сложность дерева

- Решающее дерево можно строить до тех пор, пока каждый лист не будет соответствовать ровно одному объекту
- Деревом можно идеально разделить любую выборку!
- Если только нет объектов с одинаковыми признаками, но разными ответами

Решающее дерево для регрессии



Решающее дерево для регрессии



Прогнозы в листьях

- Наш выбор: константные прогнозы $c_v \in \mathbb{Y}$
- Регрессия:

$$c_v = \frac{1}{|R_v|} \sum_{(x_i, y_i) \in R_v} y_i$$

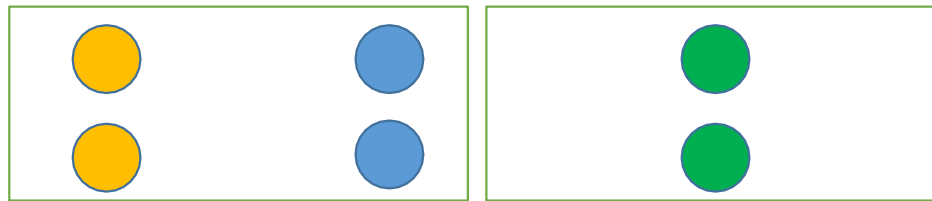
- Классификация:

$$c_v = \arg \max_{k \in \mathbb{Y}} \sum_{(x_i, y_i) \in R_v} [y_i = k]$$

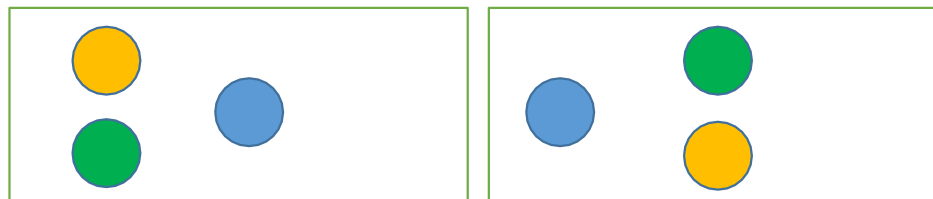
Жадное построение

- Разберёмся на примере
- Начнём с задачи классификации

Как сравнить разбиения?



или



Энтропия

- Дискретное распределение
- Принимает n значений с вероятностями p_1, \dots, p_n
- Энтропия:

$$H(p_1, \dots, p_n) = - \sum_{i=1}^n p_i \log p_i$$

Критерий Джини

$$H(p_1, \dots, p_K) = \sum_{i=1}^K p_i (1 - p_i)$$

- Вероятность ошибки случайного классификатора, который выдаёт класс k с вероятностью p_k

Критерий информативности

$$Q(R, j, t) = H(R) - \frac{|R_\ell|}{|R|} H(R_\ell) - \frac{|R_r|}{|R|} H(R_r) \rightarrow \max_{j,t}$$

- Или так:

$$Q(R, j, t) = \frac{|R_\ell|}{|R|} H(R_\ell) + \frac{|R_r|}{|R|} H(R_r) \rightarrow \min_{j,t}$$

Задача регрессии

$$H(R) = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} (y_i - y_R)^2$$

$$y_R = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} y_i$$

- То есть «хаотичность» вершины можно измерять дисперсией ответов в ней

Жадный алгоритм

- $\text{SplitNode}(m, R_m)$
- Если выполнен критерий останова, то выход
- Ищем лучший предикат: $j, t = \arg \min_{j, t} Q(R_m, j, t)$
- Разбиваем с его помощью объекты: $R_\ell = \{(x, y) \in R_m \mid [x_j < t]\}$,
 $R_r = \{(x, y) \in R_m \mid [x_j \geq t]\}$
- Повторяем для дочерних вершин: $\text{SplitNode}(\ell, R_\ell)$ и $\text{SplitNode}(r, R_r)$

Резюме

- Решающие деревья позволяют строить сложные модели, но есть риск переобучения
- Деревья строятся жадно, на каждом шаге вершина разбивается на две с помощью лучшего из предиктов
- Алгоритм довольно сложный и требует перебора всех предикатов на каждом шаге

Неустойчивость деревьев

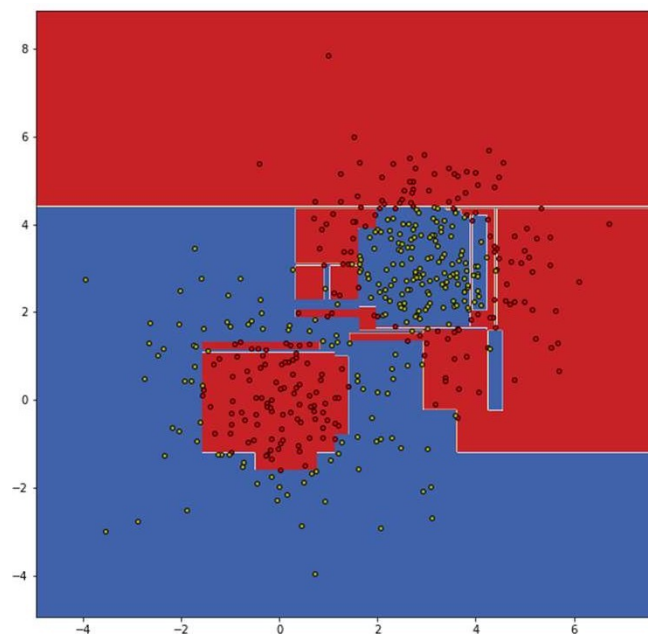
Устойчивость моделей

- $X = (x_i, y_i)_{i=1}^{\ell}$ — 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

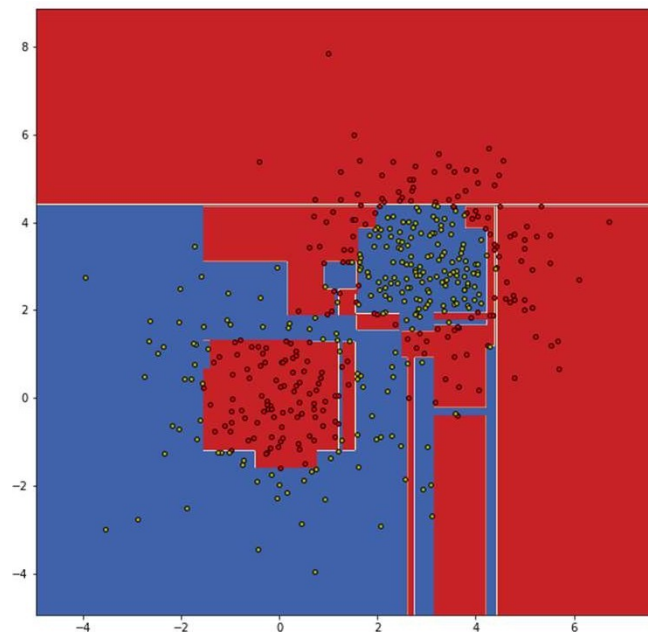
Устойчивость моделей

- \tilde{X} — случайная подвыборка, примерно 90% исходной
- Что будет происходить с деревьями на разных подвыборках?

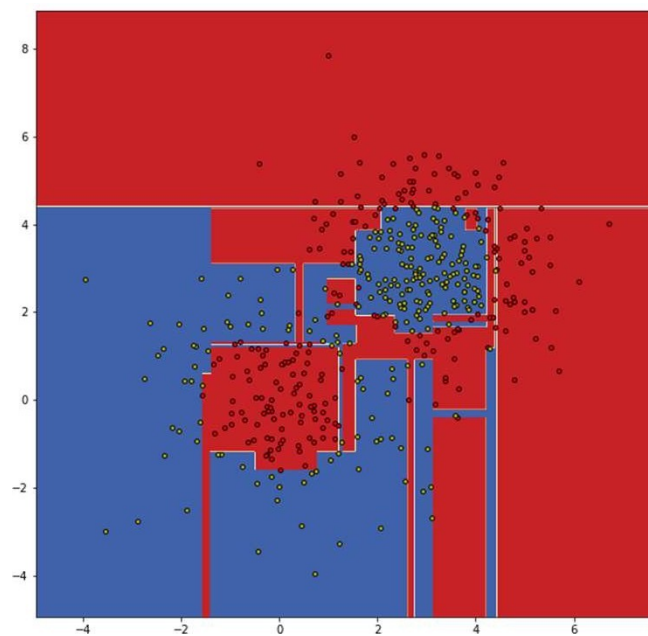
Обучение на подвыборках



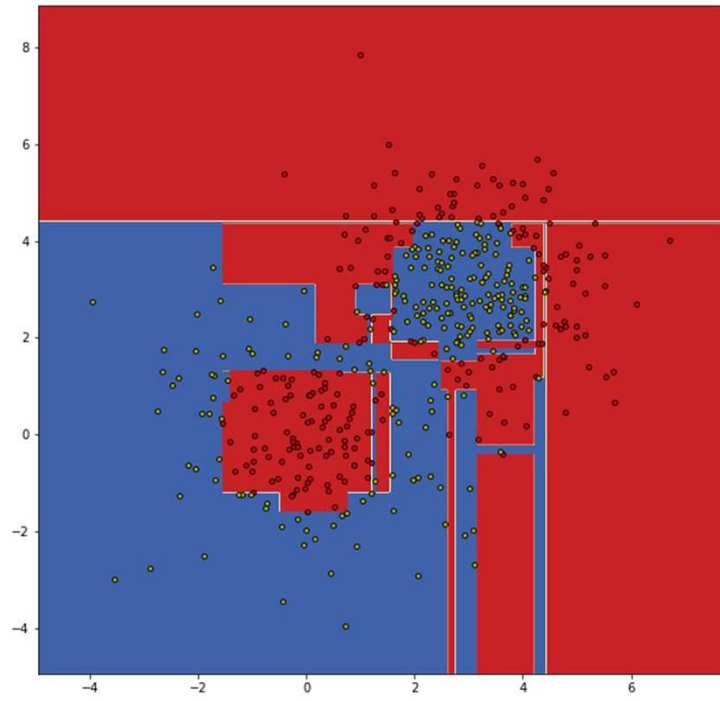
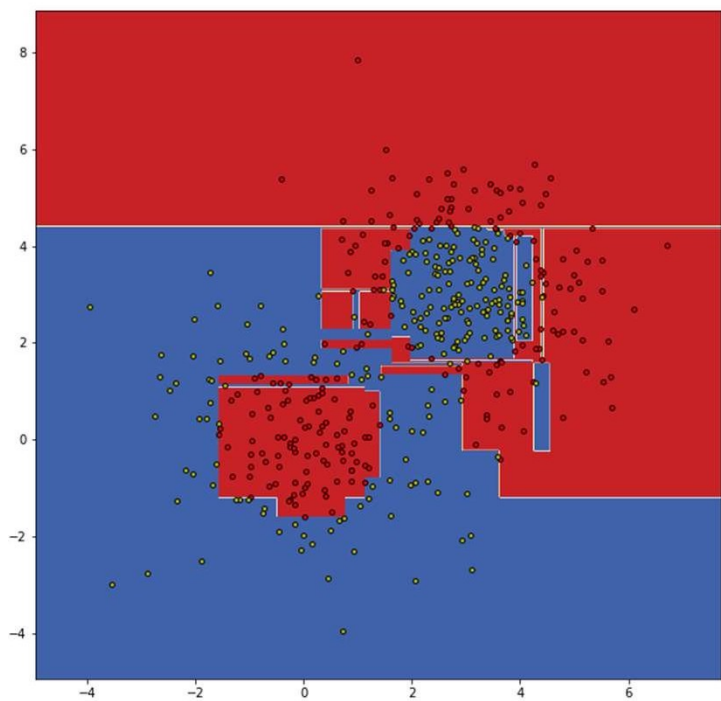
Обучение на подвыборках



Обучение на подвыборках



Обучение на подвыборках



Композиция моделей

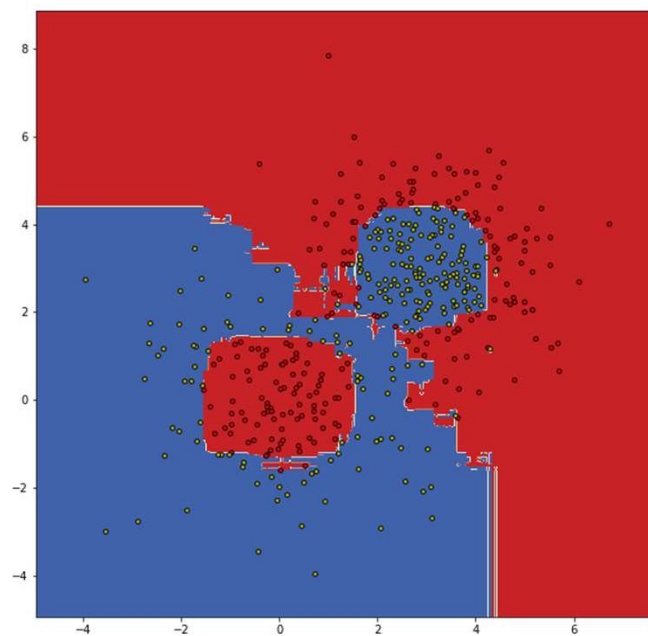
- У нас получилось N деревьев: $b_1(x), \dots, b_N(x)$
- Объединим их через голосование большинством (majority vote):

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

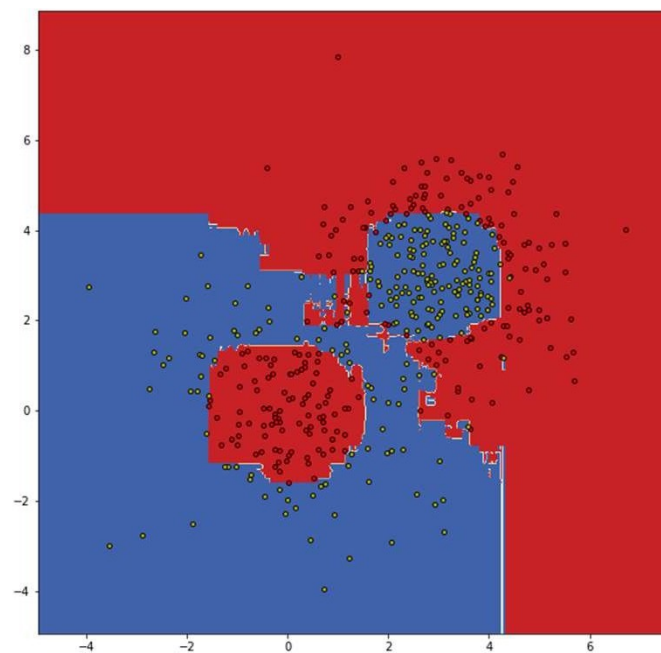
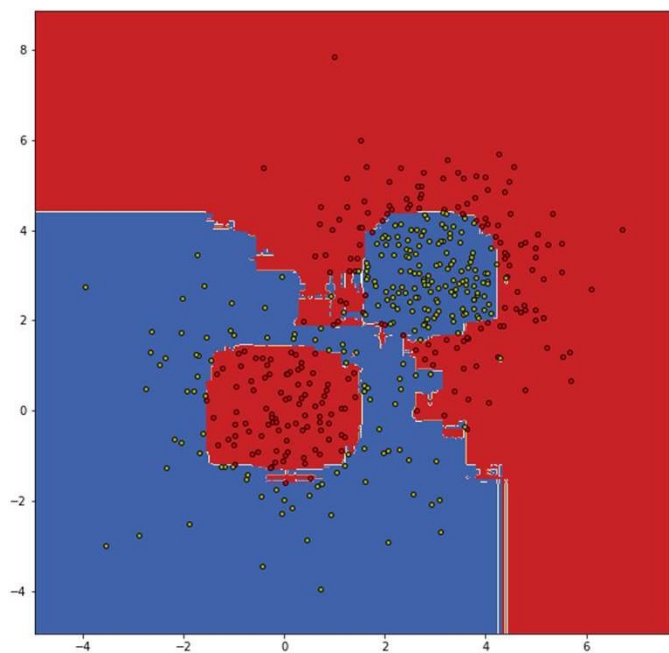
Выбираем класс,
который выбрало
большинство
деревьев

Количество деревьев,
выдавших класс y

Композиция моделей



Композиция моделей



Голосование по большинству и усреднение

Majority vote



Majority vote

- Дано: N базовых алгоритмов $b_1(x), \dots, b_N(x)$
- Композиция: класс, за который проголосовало больше всего базовых алгоритмов

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

Majority vote

- Наблюдение: усреднение результатов повышает их точность
- Измерение артериального давления
- Измерение скорости света
- Усреднение соседних пикселей изображения

Композиции моделей

Общий вид: классификация

- $b_1(x), \dots, b_N(x)$ — базовые модели
- Каждая хотя бы немного лучше случайного угадывания
- Композиция: голосование по большинству (majority vote)

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

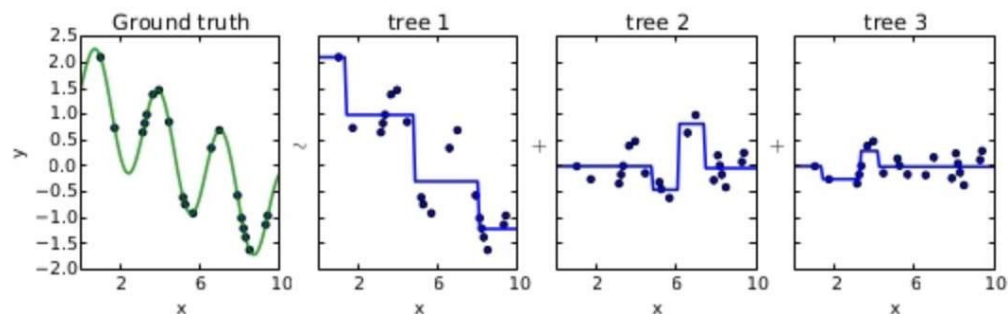
Общий вид: регрессия

- $b_1(x), \dots, b_N(x)$ — базовые модели
- Каждая хотя бы немного лучше случайного угадывания
- Композиция: усреднение

$$a_N(x) = \frac{1}{N} \sum_{n=1}^N b_n(x)$$

Бустинг

- Каждая следующая модель исправляет ошибки предыдущих
- Например, градиентный бустинг



Бэггинг

- Bagging (bootstrap aggregating)
- Базовые модели обучаются независимо
- Каждый обучается на подмножестве обучающей выборки
- Подмножество выбирается с помощью бутстрапа

Бутстрап

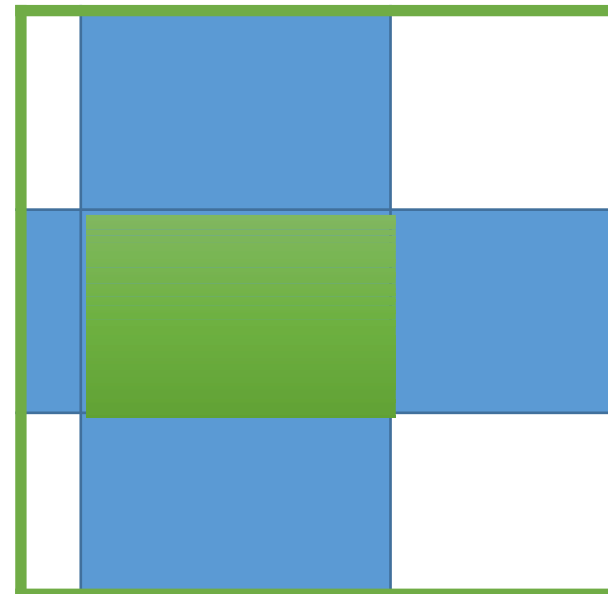
- Выборка с возвращением
- Берём ℓ элементов из X
- Пример: $\{x_1, x_2, x_3, x_4\} \rightarrow \{x_1, x_2, x_2, x_4\}$
- В подвыборке будет ℓ объектов, из них около 63.2% уникальных
- Если объект входит в выборку несколько раз, то мы как бы повышаем его вес

Случайные подпространства

- Выбираем случайное подмножество признаков
- Обучаем модель только на них
- Может быть плохо, если имеются важные признаки, без которых невозможно построить разумную модель

Виды рандомизации

- Бэггинг: случайная подвыборка
- Случайные подпространства:
случайное подмножество признаков



Резюме

- Будем объединять модели в композиции через усреднение или голосование большинством
- Бэггинг — композиция моделей, обученных независимо на случайных подмножествах объектов
- Можно ещё рандомизировать по признакам
- Как лучше всего?

Смещение и разброс моделей

Разложение ошибки на смещение и разброс

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

- Разберём на уровне идеи

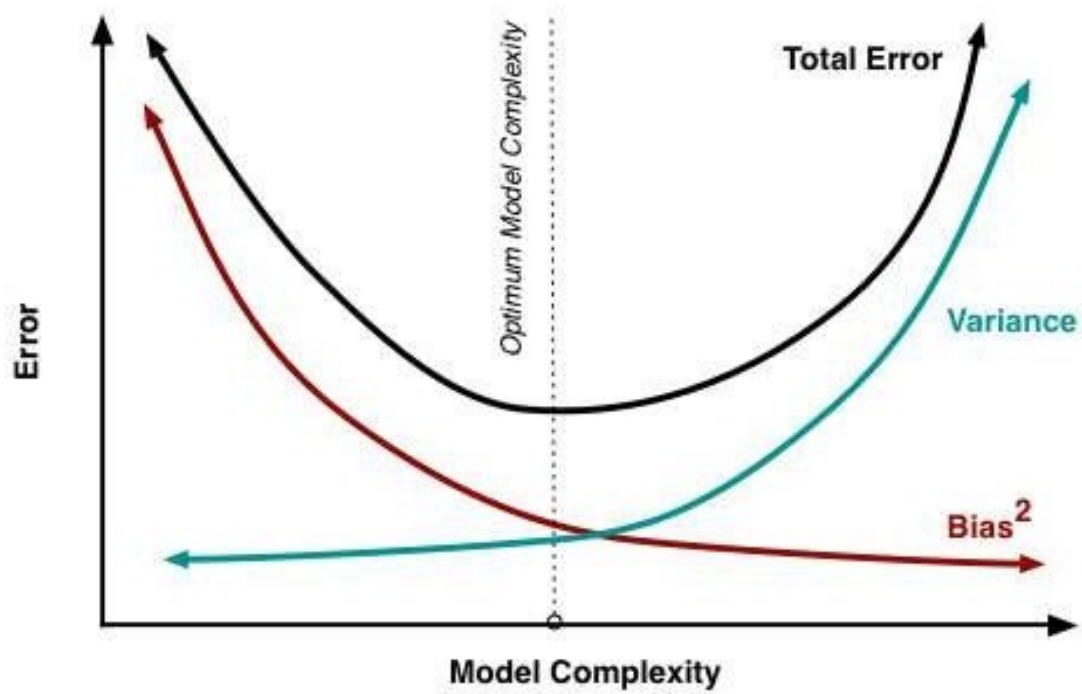
Разложение ошибки на смещение и разброс

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

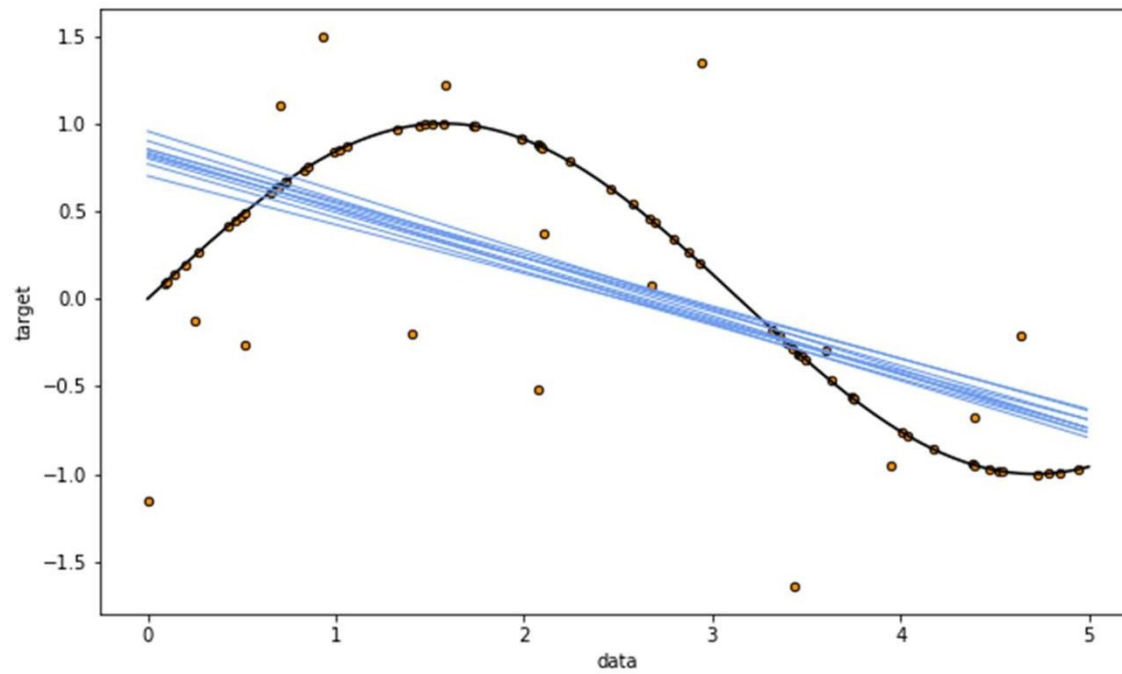
Смещение и разброс

- Высокое смещение может говорить о недообучении (слишком большая ошибка)
- Высокий разброс может говорить о переобучении (слишком сложная модель)

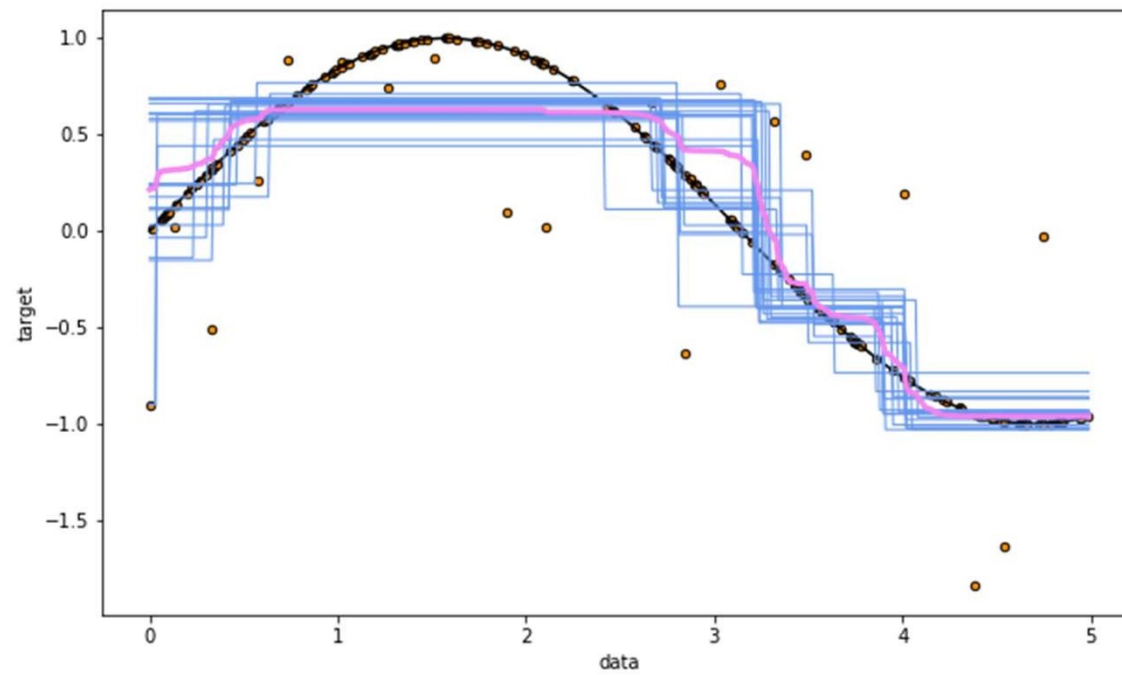
Bias-variance tradeoff



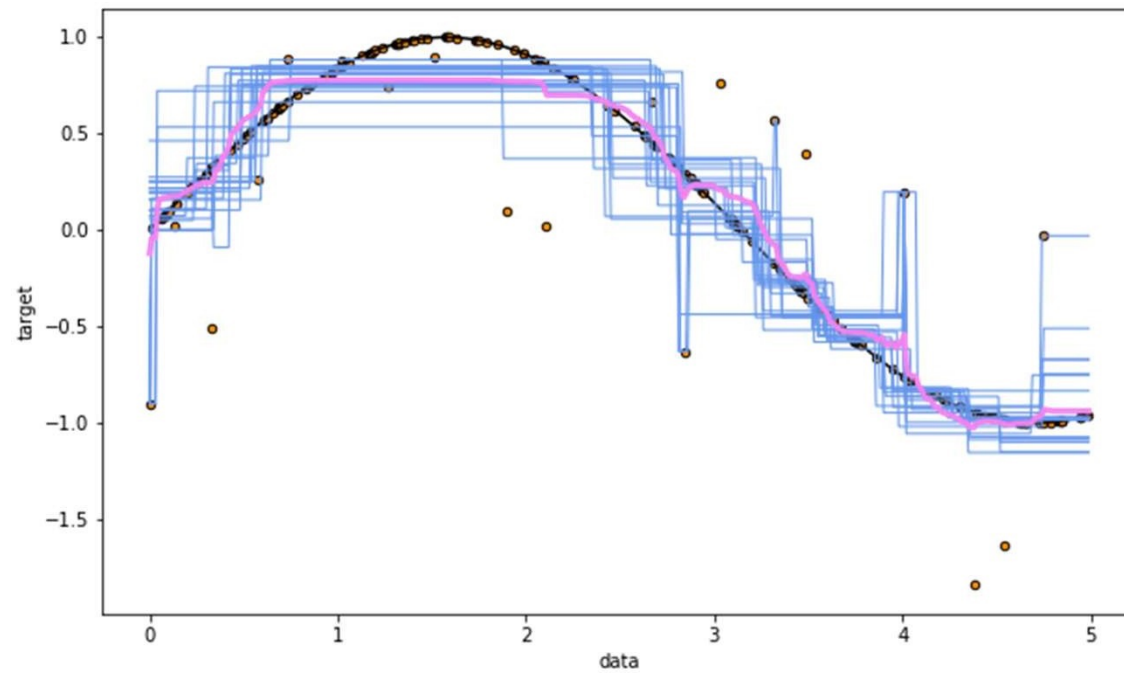
Смещение и разброс: линейная модель



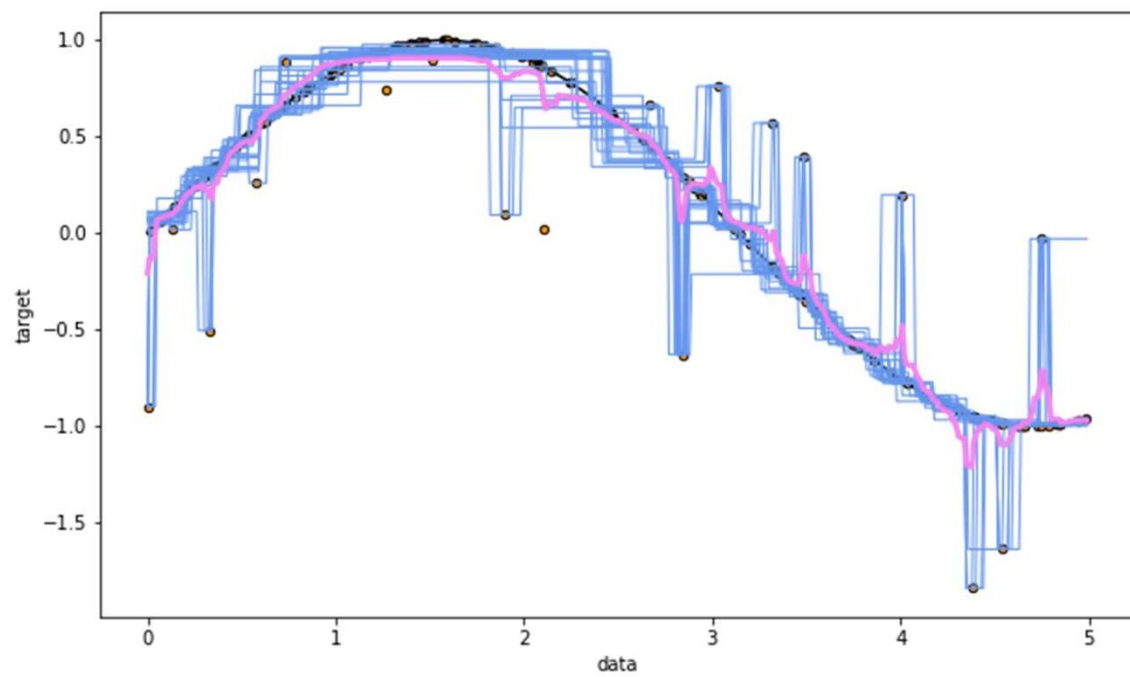
Смещение и разброс: деревья



Смещение и разброс: деревья



Смещение и разброс: деревья



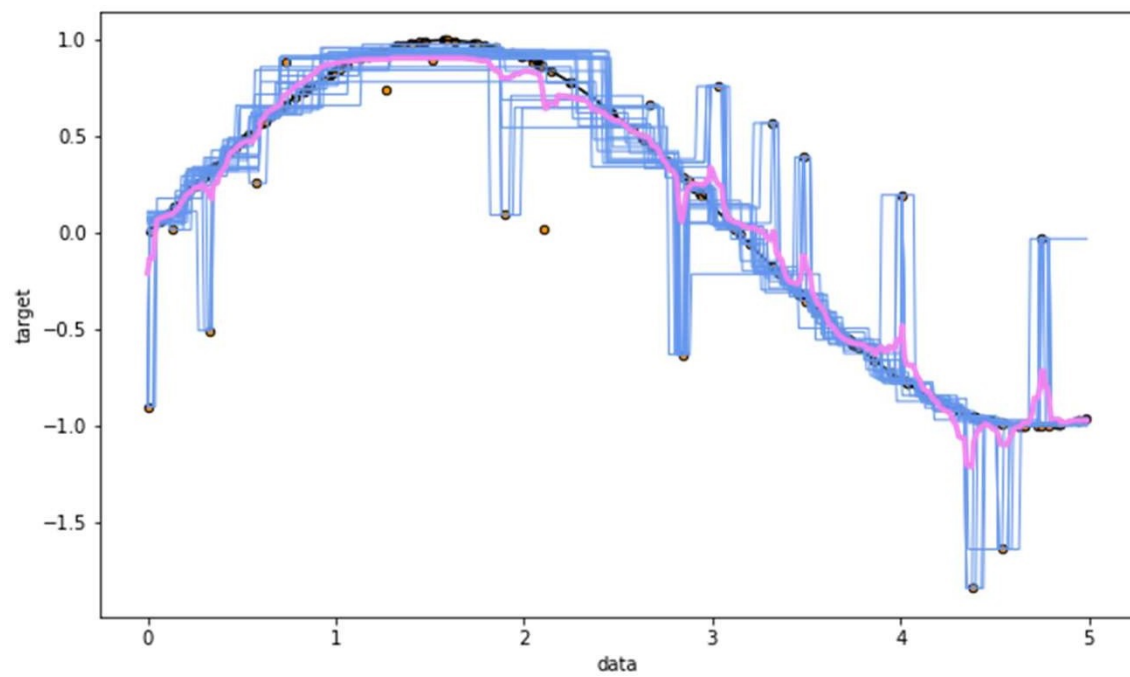
Бэггинг

- Смещение $a_N(x)$ такое же, как у $b_n(x)$
- Разброс $a_N(x)$:

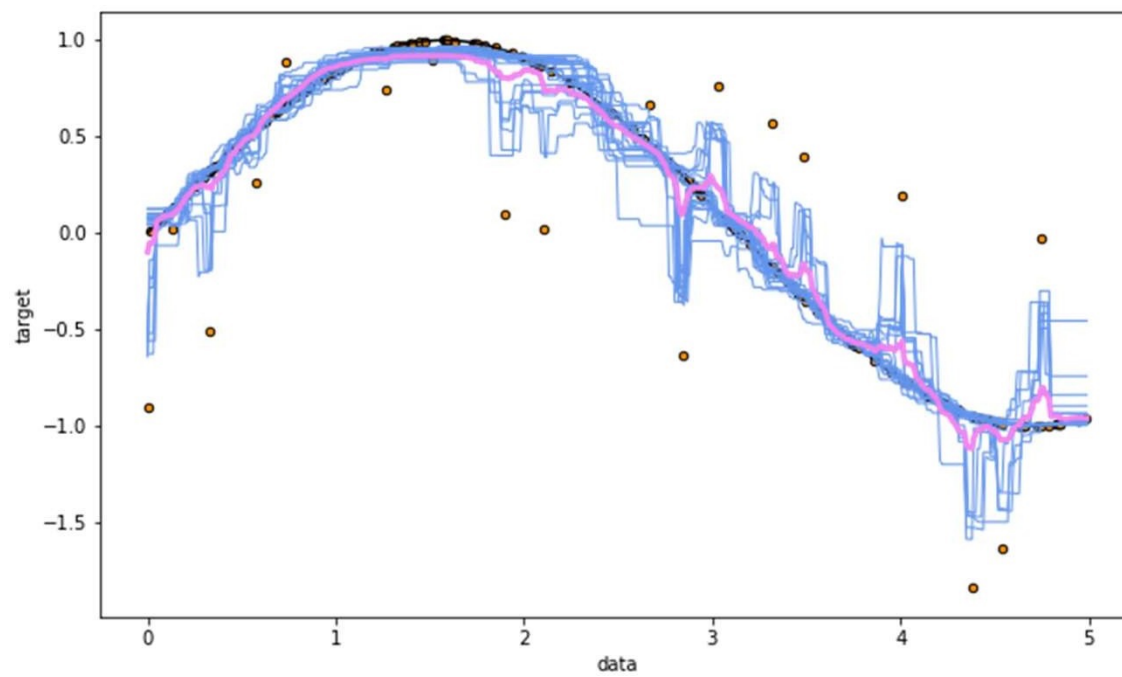
$$\frac{1}{N} (\text{разброс } b_n(x)) + \text{ковариация}(b_n(x), b_m(x))$$

- Если базовые модели независимы, то разброс уменьшается в N раз!
- Чем более похожи выходы базовых моделей, тем меньше эффект от построения композиции

Смещение и разброс: деревья



Смещение и разброс: бэггинг



Случайный лес

Жадный алгоритм

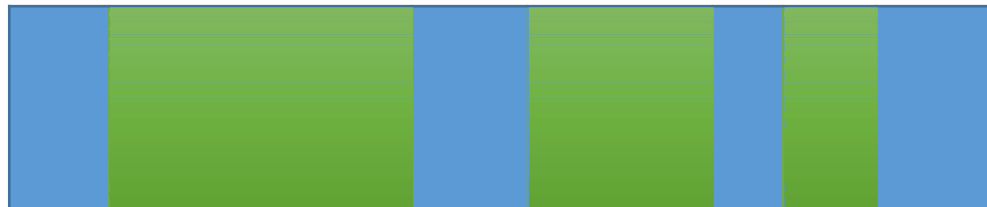
SplitNode(m, R_m)

1. Если выполнен критерий останова, то выход
2. Ищем лучший предикат: $j, t = \arg \min_{j, t} Q(R_m, j, t)$
3. Разбиваем с его помощью объекты: $R_\ell = \{(x, y) \in R_m \mid [x_j < t]\}$,
 $R_r = \{(x, y) \in R_m \mid [x_j \geq t]\}$
4. Повторяем для дочерних вершин: SplitNode(ℓ, R_ℓ) и SplitNode(r, R_r)

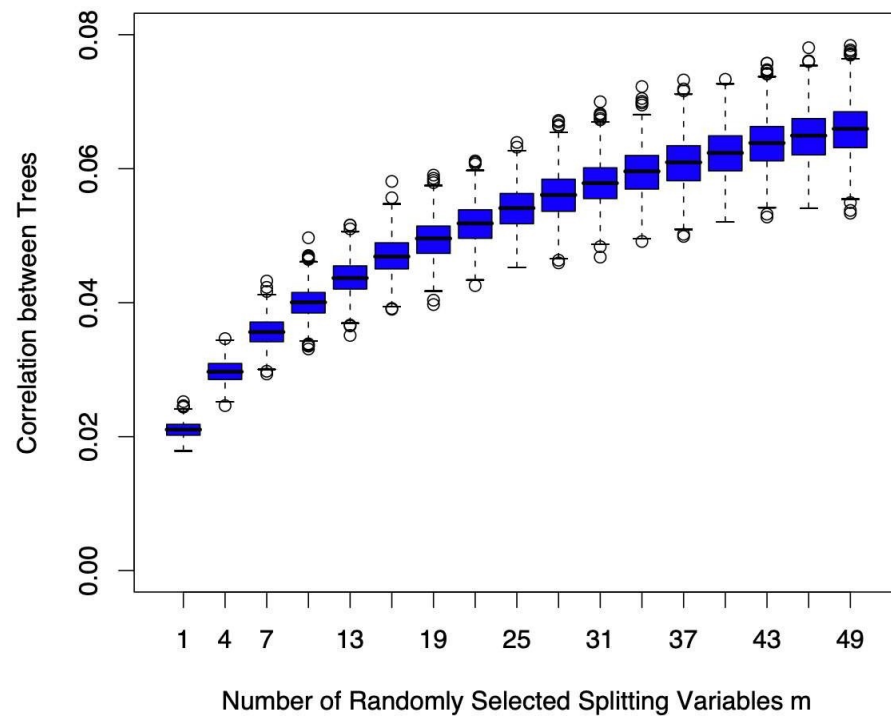
Выбор предиката

$$j, t = \arg \min_{j, t} Q(R_m, j, t)$$

- Будем искать лучший предикат среди случайного подмножества признаков размера q



Корреляция между деревьями



Hastie, Tibshirani, Friedman. The Elements of Statistical Learning.

Корреляция между деревьями

Рекомендации для q :

- Регрессия: $q = \frac{d}{3}$
- Классификация: $q = \sqrt{d}$

Случайный лес (Random Forest)

Для $n = 1, \dots, N$:

1. Сгенерировать выборку \tilde{X} с помощью бустрапа
2. Построить решающее дерево $b_n(x)$ по выборке \tilde{X}
3. Дерево строится, пока в каждом листе не окажется не более n_{min} объектов
4. Оптимальное разбиение ищется среди q случайных признаков

Выбираются заново при каждом разбиении!

Случайный лес (Random Forest)

- Регрессия:

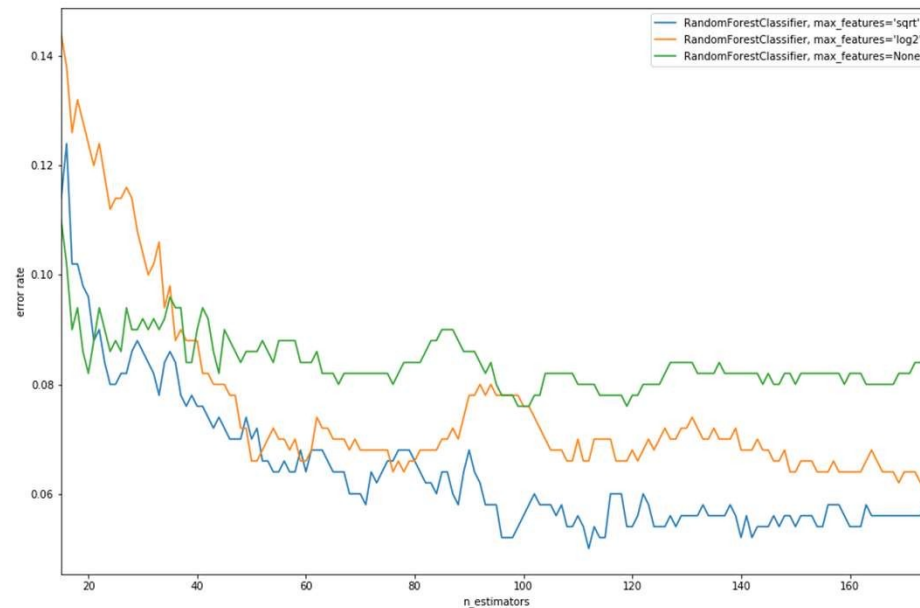
$$a(x) = \frac{1}{N} \sum_{n=1}^N b_n(x)$$

- Классификация:

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

Универсальный метод

- Ошибка сначала убывает, а затем выходит на один уровень
- Случайный лес не переобучается при росте N



Out-of-bag

- Каждое дерево обучается примерно на 63% данных
- Остальные объекты — как бы тестовая выборка для дерева
- X_n — обучающая выборка для $b_n(x)$
- Можно оценить ошибку на новых данных:

$$Q_{\text{test}} = \frac{1}{\ell} \sum_{i=1}^{\ell} L \left(y_i, \frac{1}{\sum_{n=1}^N [x_i \notin X_n]} \sum_{n=1}^N [x_i \notin X_n] b_n(x_i) \right)$$

Для каждого
объекта
выборки

Среднее ответов =
предсказание на
объекте

Суммируем ответы для всех
деревьев, в которые не
попал объект

Out-of-bag (пример)

- 4 объекта: (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , (x_4, y_4)
- 3 дерева:
 - Дерево b_1 : train on (x_1, y_1) , (x_2, y_2)
 - Дерево b_2 : train on (x_3, y_3) , (x_4, y_4)
 - Дерево b_3 : train on (x_1, y_1) , (x_2, y_2) , (x_3, y_3)

$$Q_1 = L(y_1, b_2(x_1))$$

$$Q_2 = L(y_2, b_2(x_2))$$

$$Q_3 = L(y_3, b_1(x_3))$$

$$Q_4 = L\left(y_4, \frac{1}{2}(b_1(x_4) + b_3(x_4))\right)$$

Out-of-bag (пример)

$$Q_1 = L(y_1, b_2(x_1))$$

$$Q_2 = L(y_2, b_2(x_2))$$

$$Q_3 = L(y_3, b_1(x_3))$$

$$Q_4 = L\left(y_4, \frac{1}{2}(b_1(x_4) + b_3(x_4))\right)$$

$$Q_{\text{test}} = \frac{1}{4}(Q_1 + Q_2 + Q_3 + Q_4)$$

Важность признаков

- Перестановочный метод для проверки важности j -го признака
- Перемешиваем соответствующий столбец в матрице «объекты-признаки» для тестовой выборки
- Измеряем качество модели
- Чем сильнее оно упало, тем важнее признак

Резюме

- Случайный лес — метод на основе бэггинга, в котором делается попытка повысить разнообразие деревьев
- Метод практически без гиперпараметров
- Можно оценить обобщающую способность без тестовой выборки

Спасибо за внимание!



Ildar Safilo

@Ildar_Saf

irsafilo@gmail.com

<https://www.linkedin.com/in/isafilo/>